



ITS

Institut
Teknologi
Sepuluh Nopember

TUGAS AKHIR - TE145561

**PERANCANGAN KOMUNIKASI ANTAR PLC DAN VISUAL
STUDIO MENGGUNAKAN CX - SERVER LITE UNTUK
AUTOMATION TOOL'S CRIB**

Fredrico Aditrio Pohan
NRP. 1031150000010

Pembimbing
Imam Arifin, S.T., M.T.
Dr. Ir. Ari Santoso, DEA
Ayassy Az Zurqi, S.T

Departemen Teknik Elektro Otomasi
Fakultas Vokasi
Institut Teknologi Sepuluh Nopember
Surabaya 2018

“Halaman ini sengaja dikosongkan”



ITS
Institut
Teknologi
Sepuluh Nopember

TUGAS AKHIR - TE145561

**PERANCANGAN KOMUNIKASI ANTAR PLC DAN
VISUAL STUDIO MENGGUNAKAN CX – SERVER
LITE UNTUK *AUTOMATION TOOL'S CRIB***

Fredrico Aditrio Pohan
NRP. 1031150000010

Pembimbing
Dr. Ir. Ari Santoso, DEA
Imam Arifin, S.T., M.T.
Ayyasy As Zurqi, S.T

Departemen Teknik Elektro Otomasi
Fakultas Vokasi
Institut Teknologi Sepuluh Nopember
Surabaya 2018

“Halaman ini sengaja dikosongkan”



ITS
Institut
Teknologi
Sepuluh Nopember

FINAL PROJECT - TE145561

***DESIGN OF PLC - VISUAL STUDIO
COMMUNICATION USING CX - SERVER LITE FOR
AUTOMATION TOOL'S CRIB***

Fredrico Aditrio Pohan
NRP. 1031150000010

Supervisor

Dr. Ir. Ari Santoso, DEA
Imam Arifin, S.T., M.T.
Ayyasy As Zurqi, S.T

*Department of Electrical Automation Engineering
Faculty of Vocational
Institut Teknologi Sepuluh Nopember
Surabaya 2018*

“Halaman ini sengaja dikosongkan”

PERNYATAAN KEASLIAN TUGAS AKHIR

Dengan ini saya menyatakan bahwa isi sebagian maupun keseluruhan dari Tugas Akhir saya dengan judul:

“PERANCANGAN KOMUNIKASI ANTAR PLC DAN VISUAL STUDIO MENGGUNAKAN CX – SERVER LITE UNTUK *AUTOMATION TOOL’S CRIB*”

adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diizinkan dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri.

Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap di dalam daftar pustaka. Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai dengan peraturan yang berlaku.

Surabaya, 22 Juni
2018

Fredrico Aditrio Pohan
NRP. 1031150000010

“Halaman ini sengaja dikosongkan”

**PERANCANGAN KOMUNIKASI ANTAR PLC DAN
VISUAL STUDIO MENGGUNAKAN CX-SERVER LITE
UNTUK AUTOMATION TOOL'S CRIB**

TUGAS AKHIR

**Diajukan Guna Memenuhi Sebagaimana Persyaratan Untuk
Memperoleh Gelar Ahli Madya Pada**

**Program Studi Komputer Kontrol
Departemen Teknik Elektro Otomasi
Fakultas Vokasi**

**Institut Teknologi Sepuluh Nopember
LEMBAR PENGESAHAN**

Menyetujui,

Pembimbing I,

Pembimbing II,

Pembimbing II

Dr. Ir. Ari Santoso, DEA

NIP. 196602181991021801

Imam Arifin, S.T., M.T.

NIP. 197302222002121001

Ayyasy Az Zurqi, ST

NIP. 045



“Halaman ini sengaja dikosongkan”

PERANCANGAN KOMUNIKASI ANTAR PLC DAN VISUAL STUDIO MENGUNAKAN CX – SERVER LITE UNTUK *AUTOMATION TOOL'S CRIB*

Nama : Fredrico Aditrio Pohan
Pembimbing 1 : Dr. Ir. Ari Santoso, DEA
Pembimbing 2 : Imam Arifin, ST.,MT.
Pembimbing 3 : Ayyasy As Zurqi, S.T

ABSTRAK

Peran *tool's* pada suatu perusahaan sangat vital, karena suatu mesin tidak dapat bekerja jika salah satu *tool's* tidak tersedia. Permasalahan yang terjadi pada saat transaksi pinjam dan kembali *tool's* disebabkan kurangnya informasi mengenai *tool's* yang tersedia sehingga operator menghabiskan waktu untuk mencari *tool's* dan peminjam menghabiskan waktu untuk menunggu operator. Untuk mengatasi permasalahan tersebut diperlukan sistem baru dari *tool's crib* yang merupakan tempat penyimpanan *tool's*. Sistem baru ini menggunakan *web* sebagai penyedia ketersediaan *tool's* dan penggunaan PLC Omron sebagai pengganti operator. PLC dikoneksikan dengan PC menggunakan kabel ethernet dan diprogram dengan *platform* Visual Studio sebagai master dari PLC. Namun dalam komunikasi PLC dengan Visual Studio tidak dapat dilakukan secara langsung sehingga memerlukan aplikasi untuk menjembatani komunikasi tersebut. OPC merupakan solusi dari tidak dapatnya komunikasi antar PLC dengan Visual Studio karena OPC merupakan *software* khusus untuk mengkoneksikan *hardware* dan *software* yang berbeda vendor. OPC yang digunakan yaitu CX – Server Lite dari Omron. Dengan adanya komunikasi antar PLC dan Visual Studio ini, sistem sebelumnya berubah menjadi otomatis dan disebut dengan *Automation Tool's Crib* dan *Automation Tool's Crib* memberikan waktu yang lebih cepat dibandingkan dengan *Tool's Crib* konvensional yaitu kurang lebih 90 detik.

Kata Kunci : OPC, PLC, Visual Studio

“Halaman ini sengaja dikosongkan”

DESIGN OF PLC-VISUAL STUDIO COMMUNICATION USING CX - SERVER LITE FOR AUTOMATION TOOL'S CRIB

Nama : Fredrico Aditrio Pohan
Pembimbing 1 : Dr. Ir. Ari Santoso, DEA
Pembimbing 2 : Imam Arifin, ST.,MT.
Pembimbing 3 : Ayyasy As Zurqi, S.T

ABSTRACT

The role of tool's in enterprise is important, because a machine can't work if one of tool's is not available. The problem that happened during borrow and loan proses of tool's due to lack of information about the availability of tool's so the operator spend some time for searching the tool's and the borrower is waiting to for the operator. To solve this problem, a new system of tool's crib is requied for the tool's storage. This new system uses the web as tool's availability and uses Omron PLC instead of operator. PLC is connected to PC using ethernet cable and programmed with Visual Studio as master of PLC. But in communication between PLC and Visual Studio can't happended because PLC and Visual Studio can't connected directly, so that requies an application to connected them. OPC is solution for of communication between PLC and Visual Studio because OPC is special to connected different hardware and software vendor. This OPC called CX – Server Lite from Omron. With this communication between PLC and Visual Studio, the previous system is changed to automatic and called Automation Tool's Crib and this machine gives the faster time than tools crib more or less 90 second.

Keyword : OPC, PLC, Visual Studio

“Halaman ini sengaja dikosongkan”

KATA PENGANTAR

Puji syukur penulis panjatkan kepada Tuhan Yang Maha Esa yang telah melimpahkan rahmat, taufik serta hidayah-Nya sehingga penulis dapat menyelesaikan Tugas Akhir (TA) serta dapat menyelesaikan laporannya tepat waktu dan tanpa adanya halangan yang berarti.

Laporan ini disusun berdasarkan apa yang telah penulis lakukan pada saat di industri yakni pada PT Astra Otopart Divisi Winteq yang beralamat di Jl. Raya Jakarta - Bogor Km. 47 Nanggewer Mekar Cibinong dengan judul **“Perancangan Komunikasi Antar PLC Dan Visual Studio Menggunakan Cx – Server Lite Untuk Automation Tool’s Crib”** guna memenuhi syarat kelulusan pada Program Studi komputer kontrol, Departemen Elektro Otomasi Fakultas Vokasi Institut Teknologi Sepuluh Nopember (ITS) Surabaya.

Dalam penyusunan laporan ini penulis banyak mendapatkan bantuan dari berbagai pihak, oleh sebab itu penulis ingin mengungkapkan rasa terima kasih kepada Bapak Ayas sebagai penanggung jawab dan pembimbing dari PT Astra Otopart Divisi Winteq dan Bu Lia sebagai penanggung jawab kemahasiswaan di PT Astra Otopart Divisi Winteq. Terimakasih kepada Bapak Abu dan Bapak Helmi sebagai atasan yang mengkoordinir kami dalam melakukan pekerjaan serta Seluruh karyawan dan staf PT Astra Otopart Divisi Winteq yang telah membantu selama kerja praktek dan mengajarkan kami banyak ilmu dari bagaimana cara pembuatan mesin sampai mesin dikirim ke *customer* dan cara memprogram mesin. Bapak Imam Arifin sebagai dosen pembimbing *laboratorium Otomation And Computer System* kami yang telah memberikan arahan dan bimbingan kepada kami dan Bapak Andri selaku dosen pembimbing yang telah membimbing kami selama menjalani kerja praktik Terimakasih juga kepada seluruh staff pengajaran yang telah membantu administrasi kerja praktek kami di Departemen Teknik Elektro Otomasi.

Surabaya, 2 Juni 2018

Penulis

“Halaman ini sengaja dikosongkan”

DAFTAR ISI

PERNYATAAN KEASLIAN TUGAS AKHIR	vii
ABSTRAK	xi
KATA PENGANTAR	xv
DAFTAR ISI	xvii
DAFTAR GAMBAR.....	xxi
DAFTAR TABEL	xxiii
BAB 1	1
1.1. Latar Belakang	1
1.2. Permasalahan	3
1.3. Batasan Masalah	3
1.4. Tujuan	3
1.5. Metodologi Penelitian	4
1.6. Sistematika Laporan.....	5
BAB II	7
2.1. <i>Tool's Crib</i>	7
2.2. <i>Automation Tool's Crib</i>	8
2.3. PLC	9
2.3.1. CPU (Central Processing Unit).....	13
2.3.2. <i>Memory</i>	13
2.3.3. <i>Input/Output Module</i>	15
2.4. Visual Studio.....	16
2.4.1. <i>Project</i>	18
2.4.2. <i>Form</i>	19
2.4.3. <i>Toolbox</i>	19
2.4.4. <i>Properties</i>	19
2.4.5. <i>Event</i>	19
2.4.6. <i>Metode (Method)</i>	20
2.5. Database.....	20
2.5.1. Database MySQL.....	21
2.5.2. Microsoft Access	23
2.6. OPC.....	23
2.7. RFID	25
2.7.1. <i>Interrogator</i>	26
2.7.2. <i>Transponder</i>	27
2.8. HMI (<i>Human Machine Interface</i>)	27
2.9. <i>Solenoid Door Lock</i>	28
2.9.1. Solenoida Linear	29

2.9.2.	Solenoid Rotasi.....	30
2.10.	<i>Limit Switch</i>	30
2.11.	Kabel Ethernet.....	31
BAB III.....		33
3.1.	Perancangan <i>Hardware</i>	33
3.1.1.	Rancang Bangun <i>Pilot Project</i>	34
3.1.2.	Perancangan Sistem <i>Automation Tool's Crib</i>	35
3.1.3.	Rangkaian Daya Pada Mesin <i>Automation Tool's Crib</i>	36
3.1.4.	Rangkaian I/O Pada Mesin <i>Automation Tool's Crib</i>	37
3.1.5.	Rangkaian Relay Untuk Solenoid Door Lock.....	37
3.2.	Perancangan Software	38
3.2.1.	Konfigurasi PLC pada CX Programmer	38
3.2.2.	Konfigurasi File .cdm	39
3.2.3.	Perancangan OPC pada Visual Studio	40
3.2.4.	Program <i>Read</i> Visual Studio.....	42
3.2.5.	<i>Write Value</i> Indikator Lampu Merah	42
3.2.6.	<i>Write Value</i> Indikator Lampu Kuning	43
3.2.7.	<i>Write Value</i> Indikator Lampu Hijau.....	43
3.2.8.	<i>Write Value Door Lock</i>	43
3.2.9.	<i>Write Asynchronous</i>	44
3.3.	Flowchart Proses Pinjam kembali <i>Automation Tool's Crib</i>	45
BAB IV.....		47
4.1.	Pengujian OPC	47
4.1.1.	Pengujian PLC dengan Program Visual Studio ...	48
4.1.2.	Pengujian PLC Pada Form Login	48
4.1.3.	Pengujian PLC pada Form User (Tanpa Transaksi)	49
4.1.4.	Pengujian PLC pada Form User (Dengan Transaksi)	50
4.1.5.	Percobaan PLC Saat kunci diputar	51
4.1.6.	Percobaan kecepatan pinjam dan kembali antara <i>Tool's Crib</i> Konvensional dengan <i>Automation Tool's Crib</i> .	52
4.2.	Manual Book.....	54
BAB V.....		61
DAFTAR PUSTAKA.....		63
LAMPIRAN		65

A-1.	Listing Program.....	65
A-2.	Program <i>write</i> lampu merah.....	79
A-3.	Program <i>write</i> lampu kuning.....	79
A-4.	Program <i>write</i> lampu hijau.....	79
A-5.	Program pembaca <i>Limit Switch</i>	80
A-6.	Flowchart pada form login user dan loginadmin.....	81
A-7.	Flowchart pada <i>form user</i>	82
A-8.	Flowchart RFID	83
A-9.	Flowchart pada form userpinjamkembali dan tools tersedia	84
A-10.	Flowchart komunikasi CX – Server Lite	85
A-11.	Manual book	86
A-12.	Datasheet CPU PLC.....	97
A-13.	Datasheet Input	109
A-14.	Datasheet Output	115
A-15.	Power Supply unit.....	118
A-16.	RFID	125
A-17.	Tag RFID	127
	RIWAYAT PENULIS	129

“Halaman ini sengaja dikosongkan”

DAFTAR GAMBAR

Gambar 2.1	Tool's Crib PT. Astra Otopart Divisi Winteq.....	7
Gambar 2.2	Automation Tool's Crib	8
Gambar 2.3	PLC	9
Gambar 2.4	Database MySQL	21
Gambar 3.1	Ruang Lingkup Kerja	34
Gambar 3.2	Pilot Project	34
Gambar 3.3	Desain Automation Tool's Crib	35
Gambar 3.4	Rangkaian daya mesin Automation Tool's Crib	36
Gambar 3.5	Rangkaian I/O Automation Tool's Crib	37
Gambar 3.6	Rangkaian relay Automation Tool's Crib	38
Gambar 3.7	Konfigurasi PLC pada CX Programmer.....	38
Gambar 3.8	Setting File .cdm	39
Gambar 3.9	Setting File didalam CX Server Lite	40

“Halaman ini sengaja dikosongkan”

DAFTAR TABEL

Table 4.1 Tabel I/O	47
Table 4.2 Tabel Pengujian pada Form Login	48
Table 4.3 Tabel Pengujian pada Form User (Tanpa transaksi)	49
Table 4.4 Tabel Pengujian PLC pada Form User (Dengan transaksi)	50
Table 4.5 Tabel Percobaan PLC saat kunci diputar	51
Table 4.6 Tabel Perbandingan Tool's Crib dengan Automation tool's Crib (1)	52
Table 4.7 Tabel Perbandingan Tool's Crib dengan Automation tool's Crib (2)	53
Table 4.8 Konsumsi Daya Mesin Automation Tool's Crib.....	55
Table 4.9 Komponen yang digunakan	55
Table 4.10 Fitur Mesin Automation Tool's Crib	57
Table 4.11 Kapasitas RFID dan Lemari Automation Tool's Crib .	57
Table 4.12 Dimensi dan Layout mesin Automation Tool's Crib ...	58
Table 4.13 Software yang digunakan.....	58
Table 4.14 Spesifikasi Hardware yang digunakan	59

“Halaman ini sengaja dikosongkan”

BAB 1

PENDAHULUAN

1.1. Latar Belakang

Perancangan Komunikasi antar PLC dan Visual Studio menggunakan CX – Server Lite untuk Mesin *Automation Tool's Crib* ini dimulai dari awal magang yang dilakukan di PT. Astra Otoparts Divisi Winteq. Perusahaan ini merupakan sebuah grup perusahaan komponen otomotif terbesar dan terkemuka di Indonesia yang memproduksi dan mendistribusikan beranekaragam suku cadang bermotor baik roda dua dan roda empat. Grp Astra Otoparts saat ini memiliki 7 unit bisnis, 14 anak perusahaan konsolidasi, 20 entitas asosiasi dan ventura bersama, 1 penyertaan saham perusahaan, serta 11 cucu perusahaan yang di dukung oleh 37.148 orang karyawan. Pada PT. Astra Otoparts ini terbagi menjadi beberapa bagian yaitu bagian manufacturing yang terdiri dari manufacturing divisions dan manufacturing subsidiaries serta bagian trading yang juga terbagi menjadi trading divisions dan trading subsidiaries. PT. Astra Otoparts Divisi Winteq ini termasuk dalam manufacturing divisions.

Dalam proses magang, penempatan tempat magang dipisah menjadi 4 divisi yang pertama RnD (*Research and Development*) yang merupakan pusat riset dari PT. Astra Otoparts Winteq. Kedua yaitu divisi *Machinery* yang merupakan divisi untuk mengerjakan atau meakukan proses *assembly* pada semua mesin yang ada di PT. Astra Otoparts Divisi Winteq. Ketiga ada divisi PMD (*Project Management Development*) yang merupakan divisi sebagai *supervisory* dari semua mesin yang ada di PT. Astra Otoparts Divisi Winteq dan menentukan target selesai dari suatu mesin yang akan dikirim ke *customer*. Keempat yaitu divisi *After Sales* yang merupakan divisi dalam perusahaan yang bertujuan untuk melakukan penjualan terhadap konsumen dan melakukan komunikasi dengan konsumen setelah melakukan transaksi saat terjadi masalah pada mesin yang dipesan. Penempatan divisi yang akan dibahas yaitu divisi RnD dan *Machinery* karena saat awal magan ditempatkan di divisi RnD terlebih dahulu kemudian dipindah tempat ke divisi *Machinery*. Divisi Rnd merupakan divisi riset yang membuat konsep dan desain dari mesin yang akan dipesan. Dalam divisi ini juga bagian yang membuat manual book dari mesin. Riset yang dilakukan yaitu riset AGV (*Automation Guided Vehicle*) yang merupakan kendaraan

pengangkut benda kerja yang mengikuti jalur *magnetic tape* yang dipasang di *line* produksi.

Tiga bulan setelah melakukan magang di divisi RnD, kemudian dipindah ke divisi *Machinery* karena dari divisi tersebut membutuhkan tambahan *man power* untuk mengejar target akhir tahun. Divisi *machinery* ini, mesin yang dikerjakan yaitu mesin *Snap Gauge*. Pengerjaan mesin ini membutuhkan peralatan kerja seperti tang potong untuk memotong kabel sehingga proses *wiring* dapat dilaksanakan. Selain menggunakan tang potong, penggunaan *avometer* juga dibutuhkan untuk proses *troubleshooting* apakah kabel sudah terhubung dengan alamat yang sudah ditentukan di dalam manual book. Alat tersebut tersedia di *tool's crib* yang merupakan operasi manufaktur untuk menyimpan alat, dimana alat tersedia untuk mesin pada suatu pabrik [1] dan dijaga oleh seorang operator. Pada saat proses magang, terdapat tugas mata kuliah *technopreneurship* yang mengharuskan membuat *prototype* dari tugas akhir yang ingin dibuat. Diskusi dilakukan dengan management dari PT. Astra Otopart Divisi Winteq untuk menentukan konsep dari *technopreneurship* dan implementasinya ke tugas akhir. Hasil dari diskusi yang pertama didapati konsep *vending machine* untuk pengambilan baut sesuai dengan ukuran baut yang diinginkan, namun ditolak karena pengambilan baut pada warehouse masih dapat diatasi sehingga perlu dilakukan diskusi lanjutan.

Diskusi selanjutnya, terdapat hasil dari diskusi dilihat dari kebutuhan karyawan saat bekerja, karyawan membutuhkan peralatan kerja namun saat melakukan peminjaman alat pada *tool's crib* terdapat masalah seperti mengantri lama karena banyaknya karyawan yang melakukan peminjaman dan kurang adanya informasi apakah alat yang akan dipinjam tersedia atau tidak. *Tool's Crib* Mesin *automation tool's crib* menjadi solusi dari diskusi tersebut sehingga dibuatlah alat *technopreneurship* yang merupakan miniatur dari *automation tool's crib*. miniatur ini menggunakan laci kecil dengan 3 rak dan kontroler arduino dengan *output*, 3 button untuk membuka 3 rak, *door lock* dan LCD sebagai informasi rak dapat dibuka. Fungsi *door lock* sebagai pengunci saat laci sudah tertutup sehingga tidak dapat dibuka sebelum *button* ditekan. Setelah proses pembuatan miniatur selesai dan berdiskusi mengenai mata kuliah *technopreneurship* ini, disepakati pembuatan mesin *tool's crib* dan diberi nama *Automation Tool's Crib*.

Pada Tugas akhir ini buat *Automation Tool's Crib* dengan system peminjaman barang secara otomatis menggunakan kartu ID

atau login Web. Mesin ini menggunakan RFID sebagai *scanner tool's* yang ada didalam lemari dan sebagai login kartu ID. Setelah login, *door lock* akan terbuka dan karyawan dapat mengambil barang yang akan dipinjam atau akan mengembalikan *Tool's* lalu lemari ditutup dan RFID *scan Tool's* sisa yang ada di lemari. program akan membandingkan jumlah barang saat *scan* sebelum pintu dibuka dan saat *scan* setelah pintu ditutup untuk menentukan apakah karyawan melakukan transaksi pinjam atau kembali. Sinyal RFID di konversikan ke data digital dan dimasukkan ke dalam database sebagai history peminjaman. Pembagian ruang lingkup kerja dibagi dan di diskusikan sehingga terwujud tema buku tugas akhir yang diberi judul Komunikasi antar PLC dan Visual Studio menggunakan CX – Server Lite untuk Mesin *Automation Tool's Crib*

1.2. Permasalahan

Peminjaman alat perkakas pada *Tool's crib* masih menggunakan cara manual sehingga memerlukan koin, dan satu koin hanya bisa ditukar oleh satu alat saja. Koin yang digunakan merupakan koin buatan sendiri oleh PT. Astra Otopart Divisi Winteq sehingga apabila terjadi kehilangan maka karyawan yang koinnya hilang dikenai biaya penggantian. Banyak kasus yang terjadi saat bekerja dan ingin meminjam alat untuk proses Assembly, namun koin yang digunakan untuk meminjam alat hilang karena kelalaian dari karyawan bisa jatuh saat bekerja atau tertinggal. Penggunaan koin ini masuk terbatas sampai sembilan koin saja. Dan setelah konsep dibuat untuk membuat mesin *Automation Tool's Crib* terjadi masalah karena PLC tidak dapat terhubung secara langsung dengan *software* PC karena perbedaan merk.

1.3. Batasan Masalah

Pada proses pembuatan mesin ini menggunakan PLC Omron CJ1M karena PLC ini merupakan modular sehingga bisa ditambahkan I/O sesuai keinginan dan terdapat port Ethernet sebagai perantara komunikasi dengan PC, sedangkan untuk komunikasi antar PLC yang dihubungkan dengan *software* menggunakan CX – Server Lite yang diinstal pada *software* Visual Studio.

1.4. Tujuan

Sistem *Tool's crib* yang semula manual dirubah menjadi otomatis sehingga dapat meningkatkan kualitas kerja pada *line*

produksi dan proses peminjaman alat yang semakin cepat. Dalam penggunaannya, untuk mengatasi terjadinya kesalahan pada sensor dan indikator maka dibuat komunikasi antara visual studio dengan PLC sehingga HMI dapat memonitoring I/O pada PLC.

1.5. Metodologi Penelitian

Dalam pelaksanaan Tugas Akhir yang berupa Perancangan Struktur Database Mesin *Automation Tool's Crib*, ada beberapa kegiatan yang dapat diuraikan mulai dari tahap persiapan yaitu tahap pembelajaran mengenai literature mengenai pembelajaran PHPMyAdmin sebagai database yang digunakan, pembelajaran Visual Studio sebagai *platform* untuk membuat program HMI dan komunikasi RFID, PLC dan PC, kemudian pembelajaran mengenai komunikasi PLC dengan Visual Studio dengan menggunakan OPC. Setelah melalui tahap persiapan, tahap identifikasi dan permodelan sistem merupakan tahap kedua. Pada tahap ini dilakukan identifikasi dari system alat sesuai dengan data yang telah didapatkan dari studi literatur serta dilakukan permodelan dari alat yang akan dikerjakan. Tahap ketiga setelah identifikasi dan permodelan sistem adalah tahap perancangan. Pada tahap ini akan dilakukan sebuah perancangan dan permodelan dari alat *automation Tool's crib*, mulai dari bagian elektronik sampai sistem mekanik. Pada perancangan ini digunakan digunakan RFID untuk mendeteksi keberadaan alat yang ada di dalam lemari karena dapat mencakup banyak alat dalam satu *scan*. RFID melakukan *scan* setelah pintu lemari ditutup rapat sehingga dapat menghitung jumlah *Tool's* yang ada dan diteruskan menuju database MySQL. Data yang keluar dari RFID ini berupa data *hexa* dan harus dikonversikan terlebih dahulu dengan Program Visual Studio. Penguncian lemari menggunakan *doorlock* yang dikendalikan menggunakan PLC. Lemari akan terbuka bila kartu karyawan yang sudah ditempelkan ke RFID untuk *login* dan dibaca oleh database MySQL apakah kartu sudah terdaftar atau belum, setelah terbaca Visual Studio membaca MySQL sudah login dan menggerakkan PLC untuk membuka *doorlock* lemari.

Setelah mendapat teori dasar pada pembuatan Tugas Akhir ini, dilakukan perancangan alat Tugas Akhir ini. Perencanaan dilakukan agar alat yang dibuat sesuai dengan teori dasar yang dimiliki dengan menerapkan ke dalam praktik bertujuan untuk alat yang dibuat memiliki hasil yang sesuai dengan yang diharapkan. Perancangan yang dilakukan terdiri dari perancangan *hardware* yang meliputi perancangan rangkaian elektronika, perancangan mekanik dan

perancangan *software*. Pembuatan mesin atau juga rancang bangun mesin masuk kedalam tahap keempat yaitu tahap pembuatan alat. Mulai dari rancang bangun miniatur sampai rancang bangun mesin *Automation Tool's Crib*. Setelah mesin selesai dibuat, dilakukan pengujian apakah mesin dapat berjalan sesuai dengan *flow* yang ditentukan. Tahap ini masuk kedalam tahap pengujian alat. Tahap terakhir yaitu tahap pembuatan dan penyusunan laporan setelah alat berhasil dibuat dan berkerja dengan baik tanpa adanya *error*, pengambilan data dan analisa data terpenuhi, maka tahap selanjutnya yaitu penyusunan laporan untuk buku Tugas Akhir. Diharapkan buku Tugas Akhir ini bermanfaat bagi semua orang, dan dapat dijadikan pedoman dalam melanjutkan dan mengembangkan ide Tugas Akhir ini.

1.6. Sistematika Laporan

Untuk pembahasan lebih lanjut, laporan Tugas Akhir ini disusun dengan sistematika sebagai berikut:

Bab I PENDAHULUAN

Membahas tentang latar belakang, perumusan masalah, batasan masalah, maksud dan tujuan, sistematika laporan, metodologi, serta relevansi Tugas Akhir yang dibuat.

Bab II TEORI DASAR

Menjelaskan teori yang berisi teori-teori dasar yang dijadikan landasan dan mendukung dalam perencanaan dan pembuatan alat yang dibuat.

Bab III PERANCANGAN ALAT

Membahas perencanaan dan pembuatan tentang perencanaan dan pembuatan *hardware* yang meliputi desain mekanik dan perancangan *software* yang meliputi program yang akan digunakan untuk menjalankan alat tersebut.

Bab IV PENGUKURAN DAN ANALISA

Membahas pengujian alat dan menganalisa data yang didapat dari pengujian tersebut serta membahas tentang pengukuran, pengujian, dan penganalisaan terhadap alat.

Bab V PENUTUP

Berisi penutup yang menjelaskan tentang kesimpulan yang didapat dari Tugas Akhir ini dan saran-saran untuk pengembangan alat ini lebih lanjut.

BAB II TEORI DASAR

Beberapa teori penunjang yang dipaparkan dalam buku Tugas Akhir ini adalah teori dasar mengenai pengolahan database, Pengaturan koneksi PLC Omron CJ1M menggunakan OPC (*Ole for Process Control*) CX Server Lite dengan program Visual Basic.NET yang berada pada *Software* Visual Studio

2.1. *Tool's Crib*



Gambar 2.1 *Tool's Crib* PT. Astra Otopart Divisi Winteq

Tempat penyimpanan aset peralatan dari suatu perusahaan dinamakan Divisi *Tool's Crib*. Selain sebagai tempat penyimpanan, Divisi ini memiliki fungsi sebagai tempat peminjaman alat perkakas untuk proses *assembly* mesin karena dalam proses *assembly* ini memerlukan alat perkakas sebagai media pembuatan mesin seperti merapatkan baut untuk mengencangkan bagian mekanik. Dapat dilihat pada Gambar 2.1 yaitu gambaran dari *Tool's crib* yang berada di PT. Astra Otoparts Divisi Winteq.

Cara kerja dari *Tool's Crib* sendiri dengan menggunakan sistem koin yaitu satu koin digunakan untuk meminjam satu alat perkakas, diawali dengan karyawan datang ke *Tool's Crib* dan meminta admin untuk meminjam alat yang diinginkan. Kemudian admin mengambilkan alat yang akan dipinjam dalam pengambilan alat perkakas admin akan mencari pada lemari dengan alamat dari alat perkakas. Setelah itu karyawan akan menukar koin dengan alat perkakas yang dipinjam.

Setiap bulannya admin akan menulis laporan mengenai peminjaman alat perkakas selama satu bulan, sebagai laporan untuk management agar mengetahui alat yang sering dipakai dan mengetahui berapa lama lagi alat tersebut bisa diapakai dalam jangka

waktu tertentu. Setiap alat memiliki jangka waktu pemakaian sendiri – sendiri, seperti obeng yang batas pemakaian menurut standarnya sampai seribu kali setelah itu ditukar dengan obeng yang baru.

2.2. *Automation Tool's Crib*



Gambar 2.2 *Automation Tool's Crib*

Sistem dari divisi ini yang di anggap kurang cepat dan tepat seiring dengan meningkatnya produksi dalam perusahaan menyebabkan keterlambatan dalam pengerjaan suatu proyek yang dapat merugikan perusahaan. Melalui pemikiran dan diskusi dengan management, dibuatlah konsep dari *Tool's Crib* yang otomatis dan dapat melayani karyawan lebih cepat dan tepat dibandingkan dengan *Tool's crib*. Mesin yang awalnya konsep ini sudah terlaksana sehingga menjadi mesin yang dinamakan *Automation Tool's Crib* yang artinya penyimpanan alat perkakas yang otomatis, artinya proses peminjaman dan pengembalian serta pembukuan sudah menjadi otomatis tanpa adanya admin.

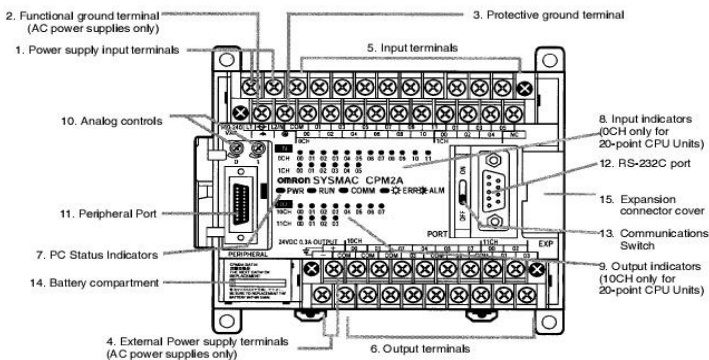
Automation Tool's Crib merupakan pengembangan dari *Tool's Crib* yang awalnya manual menjadi otomatis. Cara kerja dari mesin ini menggunakan RFID sebagai *scanner* dari tag RFID alat dan tag RFID pengguna. Karyawan yang akan meminjam menggunakan kartu tag RFID khusus pengguna sebagai identitas peminjam dengan menempelkan kartu ke kaca dari mesin pada Gambar 2.2. kemudian layar HMI akan memunculkan biodata dari peminjam dan pintu lemari dapat dibuka, sehingga peminjam dapat mengambil alat perkakas yang tersedia. Saat pintu dapat dibuka, terdapat *limit switch* pada bagian bawah dari mesin yang menandakan apabila pintu terbuka maka *limit switch* dalam keadaan *release*, pada keadaan ini RFID berhenti sedangkan saat pintu tertutup keadaan *limit switch*

sedang *press*. Setelah mengambil alat yang diinginkan, pintu akan ditutup dan *limit switch* dalam keadaan *press* kemudian PLC akan membaca bahwa *limit switch press* sehingga *value* dari *limit switch* berubah dan program HMI membaca bahwa pintu tertutup. Setelah pintu tertutup, layar pada HMI akan muncul alat yang dipinjam kemudian kembali ke layar awal login.

Mesin ini bekerja lebih cepat dari pada *Tool's crib* karena prosesnya yang otomatis dan karyawan hanya tinggal mencari alat yang ingin dipinjam tanpa harus menunggu konfirmasi dari admin seperti pada divisi ini. Perbandingan dari *Tool's Crib* dengan *Automation Tool's Crib* dapat dilihat pada tabel 4.6 dan tabel 4.7.

2.3. PLC

CPU Units with 20 or 30 I/O Terminals



Gambar 2.3 PLC

Programmable Logic Controller (PLC) merupakan alat listrik dan elektronik yang dapat diprogram untuk mengontrol berbagai fungsi secara berurutan pada mesin-mesin atau sistem listrik [2]. PLC awal dirancang untuk menggantikan sistem logika relai. PLC ini diprogram dalam "logika *ladder*", yang sangat menyerupai diagram skematik logika relai. Notasi program ini dipilih untuk mengurangi tuntutan pelatihan bagi teknisi sehingga teknisi dapat mempelajari program dengan cepat. PLC dapat di program dan di kontrol oleh operator atau admin melalui komputer. Program yang ada didalam PLC dinamakan program diagram *ladder*. Program ini menggambarkan hubungan yang diperlukan untuk suatu proses. PLC akan mengoperasikan system I/O yang dapat terdiri dari kontak dan

koil yang tersambung dengan sensor dan aktuator. Pada awalnya PLC bekerja untuk kondisi ON dan OFF untuk menggerakkan motor, sensor, dan actuator. PLC juga merupakan suatu *system* elektronika digital yang dirancang agar dapat mengendalikan mesin dengan proses mengimplementasikan fungsi nalar kendali sekuensial, operasi, perwaktuan (*timing*), pencacahan (*counting*), dan aritmatika. Kontroler ini juga memiliki I/O, unit memory, unit kontrol, dan *processor* sendiri sehingga dapat disebut dengan komputer digital. Mereka dapat dirancang untuk pengaturan ganda I/O digital dan analog, rentang suhu yang diperpanjang, ketebalan terhadap gangguan listrik, dan ketahanan terhadap getaran dan benturan. Program untuk mengontrol operasi mesin biasanya disimpan dalam memori yang didukung baterai atau non – volatile (memori yang datanya dapat ditulis atau dihapus, dan data akan hilang ketika tidak mendapat *power* / daya. Memory jenis ini hanya untuk penyimpanan data sementara saja, bukan untuk jangka waktu yang lama).

PLC, kontrol, *sequencing*, dan logika pengaman *interlock* untuk manufaktur terdiri dari *relay*, *timer cam*, *sequencer drum*, dan pengontrol loop tertutup khusus. Karena ini bisa berjumlah ratusan atau bahkan ribuan, proses untuk memperbarui fasilitas-fasilitas tersebut untuk perubahan model tahunan sangat memakan waktu dan mahal, seperti yang diperlukan para profesional listrik untuk secara individu merevisi - *rewire relay* untuk mengubah karakteristik operasional mereka. Ketika komputer digital sudah ada dan menjadi perangkat yang dapat diprogram untuk tujuan umum segera diterapkan untuk mengontrol logika sekuensial dan kombinatorial dalam proses industri. Namun, komputer awal ini membutuhkan pemrogram khusus dan kontrol lingkungan operasi yang ketat untuk suhu, kebersihan, dan kualitas daya. Untuk memenuhi syarat ini, PLC dikembangkan dengan beberapa atribut penting seperti penambahan kipas pada panel agar panas dari panel keluar yang dapat mengakibatkan PLC *overheat*. Ini akan menoleransi lingkungan kerja, itu akan mendukung masukan dan keluaran yang terpisah (*bit - form*) dengan cara yang mudah diperluas, tidak akan membutuhkan bertahun-tahun pelatihan untuk digunakan, dan memungkinkan operasinya untuk di monitor. Karena banyak proses industri memiliki skala waktu yang mudah diatasi oleh waktu respons milidetik, elektronik modern (cepat, kecil, andal) sangat memfasilitasi pembuatan pengendali yang andal, dan kinerja dapat diperdagangkan untuk keandalan.

PLC modern dapat diprogram dalam berbagai cara, mulai dari logika *ladder* yang diturunkan ke Bahasa pemrograman seperti basic dan C. Metode lainnya adalah *logic state*, bahasa pemrograman tingkat tinggi yang dirancang untuk memprogram PLC berdasarkan diagram transisi. Mayoritas sistem PLC saat ini memenuhi standar pemrograman sistem kontrol IEC 61131/3 yang terdiri dari 5 bahasa: *Ladder Diagram (LD)*, *Structured Text (ST)*, *Function Blok Diagram (FBD)*, *Instruction List (IL)* dan *Sequential Flow Chart (SFC)*. Banyak PLC lama yang tidak memiliki terminal pemrograman yang mampu merepresentasikan grafis dari logika, jadi logika diwakili sebagai serangkaian ekspresi logika dalam beberapa versi format Boolean, mirip dengan aljabar Boolean. Ketika terminal pemrograman berevolusi dari analog ke digital, itu menjadi lebih umum untuk logika *ladder* apabila digunakan, untuk alasan yang disebutkan di atas dan karena itu adalah format yang umum digunakan untuk panel kontrol elektromekanik. Format yang lebih baru seperti *structured text* dan *instruction list* (yang mirip dengan cara logika digambarkan ketika menggunakan sirkuit logika terintegrasi digital) masih tidak sepopuler logika *ladder*. Alasan utamanya adalah PiLC memecahkan logika dalam urutan diprediksi dan berulang, dan logika *ladder* memungkinkan programmer (orang yang menulis logika) untuk melihat masalah dengan waktu urutan logika lebih mudah daripada yang mungkin di lain format.

Untuk menjalankan suatu PLC pasti memerlukan program agar membuat PLC tersebut bekerja, program ini disebut *Programming Devices*, yaitu alat untuk memprogram suatu peralatan yang digunakan untuk menuliskan, mengedit, memodifikasi atau memonitoring program yang ada di dalam memori PLC. Sistem pemrograman PLC dapat dilakukan dengan dua cara. Pertama dengan menggunakan *personal computer*, yaitu rancangan kontrol konvensional ditulis dalam bentuk *ladder diagram* dan dapat langsung di-*download* ke PLC. Kedua dengan menggunakan *programming console*, yaitu *ladder diagram* yang telah didapat tadi diubah terlebih dahulu menjadi *mnemonic code* dan selanjutnya dapat dituliskan pada keypad dari *console*. *Mnemonic code* merupakan perintah dasar yang sederhana dan umum digunakan oleh PLC. Dalam penulisan *mnemonic code* mempunyai hubungan erat dengan *ladder diagram* yang dibuatnya. Apabila memasukkan program ke PLC dengan menggunakan *Programming Console*, pemrograman *mnemonic code* haruslah lebih dulu dan benar. Apabila *mnemonic code* salah maka *ladder diagram* pun akan menjadi salah, begitu juga

dengan sebaliknya sehingga PLC tidak dapat dioperasikan. PLC mengenal berbagai macam perangkat lunak, termasuk *State Language*, SFC, dan bahkan C. Yang paling populer digunakan ialah RLL (*Relay Ladder Logic*). Semua bahasa pemrograman tersebut dibuat berdasarkan proses sekuensial yang terjadi dalam *plant* (sistem yang dikendalikan). Semua instruksi dalam program akan dieksekusi oleh modul CPU, dan penulisan program itu bisa dilakukan pada keadaan online maupun offline. Jadi PLC dapat bisa ditulisi program kontrol pada saat ia mengendalikan proses tanpa mengganggu pengendalian yang sedang dilakukan. Eksekusi perangkat lunak tidak akan mempengaruhi operasi I/O yang tengah berlangsung.

PLC yang digunakan pada mesin *automation Tool's crib* yaitu PLC CJ1M keluaran dari Omron dengan seri CJ. PLC ini merupakan PLC tipe modular yang berarti I/O nya bisa ditambah sesuai dengan kebutuhan pengguna dan komponennya terpisah. CJ1M ini terpisah pisah menjadi 4 bagian yaitu CJ1M CPU 13 yang merupakan bagian yang berfungsi sebagai CPU atau *Processor* dari PLC Seri CJ ini. CPU 13 ini memiliki kapasitas memory sebesar 20Ksteps dan mempunyai I/O hingga 640 point. Untuk komunikasi, PLC ini menggunakan serial RS-232 dan Ethernet karena tersedia port *Peripheral Port* (port utama pada PC/Laptop) untuk Ethernet dan RS-232 Port untuk serialnya. Untuk programnya menggunakan *Software* bernama CX Programmer. Bagian kedua merupakan CJ1M PA-202 yang merupakan bagian *Power Supply* untuk unit CPU. CJ1M PA-202 memiliki spesifikasi tegangan sebesar 110 – 240 VAC dan total daya 14 watt. PA-202 memiliki *Output* 5 VDC sebesar 2,8 A dan 24 VDC sebesar 0,4 A. PA-202 memiliki range sebesar 50 Hz – 60 Hz. Bagian ketiga ada CJ1M ID-211 bagian yang merupakan merupakan modul *input* dari PLC. ID-211 ini merupakan *input* digital yang tidak memiliki spesifikasi untuk sinyal digital namun tidak bisa digunakan untuk sinyal analog. PLC ini memiliki *input* sebanyak 16, voltase sebesar 24 VDC dan arus sebesar 7mA. Bagian keempat merupakan CJ1M OC-211 yang merupakan modul *output* dari PLC. OC-211 ini hanya menerima hasil keluaran seperti ON/OFF atau 1/0 karena OC-211 hanya digunakan untuk mengontrol *device external* saja. Modul ini terdiri dari 16 *output* saja serta memiliki voltase 12 -24 VDC dengan maximal arus 0,5 A per point dan 5 A per Unit. PLC memiliki struktur datanya sendiri yaitu:

2.3.1. CPU (Central Processing Unit)

CPU berfungsi untuk mengontrol dan mengawasi semua pengoperasian dalam PLC, melaksanakan program yang sudah tersimpan didalam memory PLC itu sendiri. CPU merupakan Otak dari system PLC. CPU akan bertugas mengambil intruksi memori, mengkodenyanya, dan mengeksekusi intruksi tersebut. Selama proses tersebut, processor akan menghasilkan sinyal kontrol yang memindahkan data I/O port atau sebaliknya, mendeteksi sinyal dari luar CPU. CPU mempunyai elemen kontrol yang disebut *Arithmetic and Logic Unit* (ALU), sehingga mampu mengerjakan operasi logika (penjumlahan, pengurangan, perkalian, pembagian, dan lain – lain) dan aritmatika (operasi OR, AND, NOT, dan lain – lain) [2].

Sebagian besar CPU berisi baterai cadangan yang berfungsi untuk menjaga program operasi yang ada dalam penyimpanan jika terjadi kegagalan *supply daya* pada PLC. Pada PLC tertentu kadang dijumpai pula beberapa prosesor yang sekaligus menjadi satu modul, yang ditujukan untuk mendukung keandalan sistem. Beberapa prosesor tersebut bekerja sama dengan suatu prosedur tertentu untuk meningkatkan kinerja pengendalian.

2.3.2. Memory

Tempat penyimpanan data dalam PLC merupakan memori yang terdapat pada PLC berfungsi untuk menyimpan program dan memberikan lokasi - lokasi dimana hasil – hasil perhitungan dapat disimpan di dalamnya. *Memory* umumnya menjadi satu modul dengan processor/CPU. Jika berbentuk memori eksternal maka itu merupakan *memory* tambahan. Data yang tersimpan di dalam memori berupa *Operating System PLC*, Status *Input – Output*, Data memori, dan Program yang dibuat. Dalam memory PLC ada yang disebut dengan *Operating System Memory*, Data (status) memori, dan program memori. *Operating System Memory* berfungsi menyimpan *Operating System PLC*. Memori ini berupa ROM (*Read Only Memory*) sehingga tidak dapat diubah oleh user. Data (status) memori berfungsi untuk menyimpan status *input-output* tiap saat. Memori ini berupa RAM (*Random Access Memory*) sehingga dapat berubah sesuai kondisi *input-output*. Status akan kembali ke kondisi awal jika PLC mati. Program memori berfungsi untuk menyimpan program pengguna. Jenis memori ini berupa RAM. RAM dapat menggunakan baterai cadangan untuk menyimpan program selama jangka waktu

tertentu. Selain itu dapat berupa EEPROM (*Electrical Erasable Programmable Read Only Memory*).

Untuk kebutuhan program, area memori pada PLC ada *register*, *flag register*, *Auxiliary relays*, *timer*, dan *counter*. *Register* berfungsi untuk menyimpan sekumpulan bit data, baik berupa nibble (4 bit), byte (8 byte), maupun word (16 bit). *Flag register* berfungsi untuk mengindikasikan perubahan kondisi (*state*) *input-output* fisik. *Flag register* berupa satu bit data. CPU pada umumnya memiliki internal flag untuk berbagai keperluan internal PLC. *Auxiliary relays* adalah elemen memori 1 bit dalam RAM yang digunakan untuk memanipulasi data dalam program. *Auxiliary relays* disebut juga *relay* yang imajiner, karena dapat menggantikan fungsi *relay* namun berbentuk program. *Timer* adalah pemberian penundaan waktu dalam suatu proses. *Timer* berasal dari *built in clock oscillator* dalam CPU. *Timer* pada umumnya memiliki alamat khusus. *Counter* adalah komponen penghitung *input* yang diberikan oleh *input device*. CPU memiliki *counter* internal. *Counter* ini umumnya memiliki alamat khusus. Memori juga memiliki berbagai jenis seperti RAM yang merupakan *memory* yang cepat dan bersifat *volatile* (data yang hilang bila arus listrik mati).

RAM digunakan sebagai memori utama dalam PLC, berfungsi sebagai tempat penyimpanan program, program yang telah ditulis tersimpan di dalam RAM yang ada di dalam PLC sehingga dapat diubah/diedit melalui programming unit, namun kerugian penyimpanan di RAM adalah program dan data akan hilang ketika power supply mati, dan untuk mengatasi hal ini RAM dapat di backup dengan baterai, sehingga meskipun power supply mati, program dan data tidak hilang, umumnya bila baterai tidak rusak, program dan data bisa di simpan selama bertahun - tahun. Lalu ada ROM (*Read Only Memory*), merupakan tempat penyimpanan *Operating System* yang dibuat oleh pabrik pembuat PLC. *Operating System* ini hanya dapat dibaca oleh processor dan berfungsi untuk mengeksekusi program yang tersimpan di dalam RAM. Ketiga ada EPROM (*Erasable Programmable Read Only Memory*) adalah jenis memori yang cepat dan juga murah harganya, sama dengan memori RAM hanya saja EPROM bersifat *non - volatile*, artinya isi memori ini tetap ada walaupun supply tegangan hilang.

Untuk keperluan modifikasi program, memori ini harus dikosongkan isinya melalui penyinaran dengan sinar ultraviolet. Karena proses penghapusan untuk memprogram ulang begitu kompleks bahkan meskipun harganya murah, orang cenderung

memilih RAM. Baru bila programnya sudah benar dan lengkap langsung bisa ditransfer ke EPROM secara permanen. Keempat ada EEPROM adalah memori yang mirip dengan memori EPROM, hanya saja untuk proses penghapusannya menggunakan arus listrik. Kapasitas memori tergantung penggunaannya dan seberapa jauh pengoptimalisasikan ruang memori PLC yang dimiliki, yang berarti pula tergantung seberapa banyak lokasi yang diperlukan program kontrol untuk mengendalikan plant tertentu. Program kontrol untuk pengaliran bahan bakar dalam turbin gas tentu membutuhkan lokasi memori yang lebih banyak dibandingkan dengan program kontrol untuk menggerakkan putaran mekanik robot pemasang bodi mobil pada industri otomotif. Suatu modul memori tambahan bisa juga diberikan ke sistem utama apabila kebutuhan memori memang meningkat.

2.3.3. *Input/Output Module*

Input/Output Modul merupakan suatu peralatan atau perangkat elektronik yang berfungsi sebagai perantara atau penghubung (*interface*) antara CPU dengan peralatan *input output devices* [2]. Modul I/O bertugas mengatur hubungan PLC dengan piranti eksternal atau periferal yang bisa berupa suatu komputer host, saklar - saklar, unit penggerak motor, dan berbagai macam sumber sinyal yang terdapat dalam plant. Fungsi dari sebuah modul *input* adalah untuk mengubah sinyal masukan dari sensor ke PLC untuk diproses dibagian CPU. Sedangkan modul *output* adalah kebalikannya, mengubah sinyal PLC kedalam sinyal yang sesuai untuk menggerakkan aktuator. Dari modul *input* dan *output* kita dapat menentukan jenis suatu PLC dari hubungan antara processor dengan modul *input* dan *output* yaitu *compact* PLC. Bila *input* modul processor dan *output* modul dikemas dalam suatu wadah dan Modular PLC yang bila modul *input*, modul *output* dan processor dikemas secara tersendiri.

Modul *input* adalah modul tempat menghubungkan sensor-sensor dengan modul itu sendiri. Sinyal sensor tersebut selanjutnya akan diteruskan ke Processor. Modul *input* berfungsi untuk mengkonversi sinyal diskrit dan analog yang berasal dari *input device* menjadi sinyal digital. Modul *input* bertugas untuk menerima sinyal dari unit pengindra periferal, dan memberikan pengaturan sinyal, terminasi, isolasi, maupun indikator keadaan sinyal masukan. Sinyal-sinyal dari piranti periferal akan di-*scan* dan keadaannya akan dikomunikasikan melalui modul antar muka dalam PLC. Modul *input*

dibagi menjadi dua yaitu *input* analog yang dapat menerima tegangan dan arus dengan level tertentu (misal 0–10V, 4–20mA) dari *input device* analog (misal: sensor analog, potensiometer) dan digital yang berfungsi menghubungkan *input* diskrit fisik (*switch*, sensor) dengan PLC. Sinyal *input* dapat berupa logika 0/1 dan ON/OFF ataupun sinyal analog. Modul ini tersedia dalam tegangan DC dan AC (umumnya: 240VAC, 120VAC, 24VDC, dan 5VDC). Didalamnya terdapat *optoisolator* untuk mencegah lonjakan tegangan tinggi masuk PLC (sebagai pengamanan).

Output Modul berfungsi untuk mengkonversi sinyal digital menjadi sinyal analog sehingga dapat mengaktifkan kembali peralatan luar (*output devices*). Modul *output* mengeluarkan sinyal dari Processor ke kontrol elemen yang diperlukan untuk menggerakkan aktuator sesuai dengan tugas yang telah diberikan. Modul keluaran mengaktifkan berbagai macam piranti seperti aktuator hidrolis, pneumatik, solenoid, starter motor, dan tampilan status titik-titik perifer yang terhubung dalam sistem. Fungsi modul keluaran lainnya mencakup *conditioning*, terminasi dan juga pengisolasi sinyal-sinyal yang ada. Proses aktivasi itu tentu saja dilakukan dengan pengiriman sinyal-sinyal diskrit dan analog yang relevan, berdasarkan watak PLC sendiri yang merupakan piranti digital. Modul *output* juga dibagi menjadi dua yaitu *output* analog yang dapat memberikan tegangan dan arus dengan level tertentu (misal 0–10V, 4–20 mA) pada *output device* analog (misal: motor DC, motor AC, control valve) dan modul *output* digital yang menghubungkan *output* diskrit fisik (lampu, *relay*, solenoid, motor) dengan PLC. Setiap PLC memiliki jumlah I/O yang terbatas berdasarkan tipe PLC. Dalam aplikasi sendiri seringkali I/O dari PLC tidak mencukupi, oleh karena itu diperlukan ekspansi atau penambahan I/O dari suatu PLC sehingga kebutuhan I/O terpenuhi.

2.4. Visual Studio

Software ini dapat digunakan untuk membuat dan melakukan pengembangan banyak aplikasi, bisa aplikasi game, pekerjaan bisnis, aplikasi windows maupun HMI [3]. *Software* ini mencakup yang pertama kompilator yaitu program komputer yang berfungsi untuk menerjemahkan program komputer yang ditulis dalam bahasa program tertentu menjadi program yang ditulis dalam bahasa program lain, bahasa yang diterjemahkan biasanya ditulis dalam bahasa pemrograman tinggi (Seperti bahasa *Pascal*, C++, Basic, FORTRAN,

Visual Basic, Visual C++, Java, xBase, atau Cobol) menjadi bahasa mesin, biasanya dengan bahasa *assembly* sebagai perantara.

Yang kedua yaitu SDK (*software Development Kit*) yang merupakan alat pengembangan perangkat lunak yang memungkinkan pembuatan aplikasi untuk paket *software* tertentu, *software framework*, *hardware platform*, sistem komputer, konsol video game, sistem operasi, atau *platform* pengembang serupa. Beberapa pengembang SDK sangat penting untuk mengembangkan aplikasi platform spesifik, misalnya pengembangan aplikasi android di *platform* java memerlukan *java development kit*, untuk aplikasi iOS, SDK iOS, dan untuk *Platform Windows Universal*, .NET framework SDK.

Ketiga yaitu IDE (*Integrated Development Environment*) adalah program komputer yang memiliki beberapa fasilitas yang diperlukan dalam pembangunan *software* [3]. Tujuan dari IDE sendiri yaitu menyediakan semua utilitas yang diperlukan dalam membangun *software*. IDE memiliki lima fasilitas seperti editor yaitu fasilitas untuk menuliskan kode sumber dari *software*, *Compiler* yaitu fasilitas untuk mengecek sintak dari kode sumber kemudian mengubah dalam bentuk binari yang sesuai dengan bahasa mesin, Kompiler yang dimasukkan ke dalam paket Visual Studio antara lain Visual C++, Visual C#, Visual Basic, Visual Basic .NET, Visual InterDev, Visual J++, Visual J#, Visual FoxPro, dan Visual SourceSafe. *Linker* yaitu fasilitas untuk menyatukan data binari yang dihasilkan oleh *compiler* sehingga data – data binari tersebut menjadi satu kesatuan dan menjadi suatu program komputer yang siap dieksekusi, *debugger* yaitu fasilitas untuk mengetes jalannya program untuk mencari *bug* atau kesalahan yang terdapat pada program.

Keempat yaitu dokumentasi atau MSDN *Library* yaitu bagian dari microsoft yang bertanggung jawab untuk mengelola hubungan perusahaan dengan pengembang dan penguji, seperti pengembang *hardware* yang tertarik pada sistem operasi (OS), dan pengembang *hardware* yang berkembang di *platform* OS menggunakan API atau *scripting* bahasa aplikasi Microsoft. Pada mesin *Automation Tool's Crib* ini menggunakan bahasa pemrograman visual basic.NET yaitu pemrograman yang mengembangkan dan membangun aplikasi yang bergerak di atas sistem .NET Framework, dengan menggunakan bahasa BASIC. Dengan menggunakan alat ini, para *programmer* dapat membangun aplikasi Windows Forms, Aplikasi web berbasis ASP.NET, dan juga aplikasi *command-line*. Alat ini dapat diperoleh secara terpisah dari beberapa produk lainnya

(seperti Microsoft Visual C++, Visual C#, atau Visual J#), atau juga dapat diperoleh secara terpadu dalam Microsoft Visual Studio .NET.

Bahasa Visual Basic .NET menganut paradigma bahasa pemrograman berorientasi objek yang dapat dilihat sebagai evolusi dari Microsoft Visual Basic versi sebelumnya yang diimplementasikan di atas .NET Framework. Untuk membuat obyek-obyek pembantu program, seperti Control Active X, File Help, Aplikasi Internet dan sebagainya serta menguji program (debugging) dan menghasilkan program akhir berakhiran "EXE" yang bersifat executable atau dapat langsung dijalankan. Visual basic.NET memiliki keistimewaan seperti menggunakan platform pembuatan program yang diberi nama developer studio, yang memiliki tampilan seperti C++ dan visual J++, memiliki kompiler handal yang dapat menghasilkan File Executable yang lebih cepat dan tepat, memiliki tambahan saran wizard yang baru. Tambahan kontrol - kontrol baru dan lebih canggih serta peningkatan kaidah struktur bahasa Visual Basic, kemampuan membuat Active X dan fasilitas internet yang lebih banyak, sarana akses yang lebih cepat dan andal untuk membuat aplikasi database yang berkemampuan tinggi, dan Visual Basic.net memiliki beberapa versi baru edisi yang disesuaikan dengan kebutuhan pemakainya.

Dalam visual basic terdapat komponen – komponen sebuah program dibagi menjadi bagian-bagian kecil yang disebut dengan objek. Objek - objek yang terpisah ini dapat diolah sendiri, dan setiap objek memiliki sekumpulan sifat dan metode yang melakukan fungsi tertentu sesuai dengan yang telah kita programkan [4]. Visual Studio ini merupakan *platform* yang menjadi tempat komunikasi dari PLC dilakukan, karena pada visual studio ini sudah diinstal dan ditambahkan *reference* mengenai CX – Server Lite sehingga bisa dapat saling berkomunikasi dan hanya memprogram sesuai *syntax* dari CX – Server Lite. Objek yang dipergunakan dalam program ini yaitu :

2.4.1. Project

Sekumpulan modul dapat disebut sebagai project. Project disimpan dalam file yang berakhiran VBP. Jika kita akan melaksanakan pembuatan program aplikasi, akan terdapat jendela project yang berisi semua file yang dibutuhkan menjalankan program aplikasi Visual Basic.NET pada saat pembuatan program aplikasi baru maka jendela project otomatis akan berisi object form1. Pada jendela project terdapat tiga icon yaitu View Code, View Object, dan

Toggle Folders. Icon View Code dipakai untuk menampilkan jendela editor kode program. Icon View Object dipakai untuk menampilkan bentuk formulir (form) dan icon Toggle Folders digunakan untuk menampilkan folder.

2.4.2. Form

Jendela yang dipakai untuk membuat user *interface* (tampilan) pada HMI disebut *form*. Secara otomatis akan tersedia *form* yang baru jika membuat suatu program aplikasi yang baru, dengan nama Form1. pada umumnya dalam suatu form terdapat garis titik-titik yang disebut dengan Grid. Untuk lebih memahami form ini maka di bawah ini terdapat gambar jendela form.

2.4.3. Toolbox

Kumpulan dari obyek yang digunakan untuk membuat user *interface* serta control bagi program aplikasi berguna untuk menempatkan control pada suatu form dapat dilakukan dengan klik ganda control dalam toolbox, kemudian mengubah besar dan ukurannya serta memindahkannya dengan metode Drag and Drop atau dengan cara mengklik kontrol toolbox, kemudian pindahkan pointer mouse jendela form. Kursor berubah menjadi *Crosshair* lalu tempatkan pada sudut kiri atas dimana kita inginkan kontrol tersebut diletakkan, tekan tombol mouse kiri dan tahan ketika menyeret kursor ke arah sudut kanan bawah.

2.4.4. Properties

Daftar struktur setting properti yang digunakan pada sebuah object terpilih pada Visual Studio. Kotak *drop - down* pada bagian atas jendela berisi daftar semua object pada form yang aktif. Ada tab tampilan, yaitu *alphabetic* (urut abjad) dan *categorized* (urut berdasarkan kelompok).

2.4.5. Event

Event adalah peristiwa atau kejadian yang diterima oleh suatu objek, misalnya click, dblclick, keypress dan sebagainya [3]. Sebuah program yang baik harus mampu mengakomodasi seluruh kemampuan event yang akan dilakukan oleh pemakainya. Tetapi tentu saja untuk menyediakan puluhan kemungkinan event pada

sebuah program tidaklah mudah. Berikut ini beberapa event yang sering digunakan oleh pemakai program, yaitu :

- a. Click : event ini terjadi bila tombol kiri mouse ditekan dan dilepas dengan cepat saat posisi pointer berada di atas objek.
- b. dblclick : event ini terjadi bila tombol kiri mouse ditekan dan dilepas dengan cepat sebanyak dua kali saat posisi pointer berada diatas objek.
- c. Load : event ini terjadi bila sebuah *form* dibuka atau dipanggil
- d. unload : event ini terjadi bila *form* ditutup
- e. Change : event ini terjadi bila isi dari sebuah kontrol diubah
- f. Dragdrop : Event ini terjadi bila tombol kiri mouse ditekan dan ditahan kemudian menyeret/menggeser objek dari satu tempat ke tempat lain, kemudian melepas tombol kiri mouse tersebut.
- g. MouseDown : Event ini terjadi bila tombol kiri mouse ditekan dan ditahan
- h. Resize : Event ini terjadi bila sebuah *form* diubah ukurannya
- i. Deactive : Event ini terjadi ketika anda berpindah dari satu *form* ke *form* yang lain
- j. QueryUnload : Event ini terjadi apabila suatu aplikasi ditutup. Biasanya digunakan untuk memastikan bahwa semua yang berhubungan dengan aplikasi juga telah ditutup, atau memastikan bahwa data telah disimpan sebelum aplikasi ditutup.
- k. Timer : Event ini terjadi ketika event lain dijalankan sebelum event timer ini. event ini hanya berhubungan dengan kontrol timer
- l. Scroll : Event ini terjadi saat anda menyeret kotak kecil pada scroll bar. Event ini hanya berhubungan dengan kontrol scroll bar.
- m. Validate : Event ini digunakan untuk memastikan data telah ditangani dengan *.* diubah menjadi *.jpg.

2.4.6. Metode (Method)

perintah yang sudah tersedia pada suatu obyek yang dapat diminta untuk mengerjakan tugas khusus disebut dengan metode. Fungsi ini tidak dapat didefinisikan di luar *Class*, struktur, maupun modul.

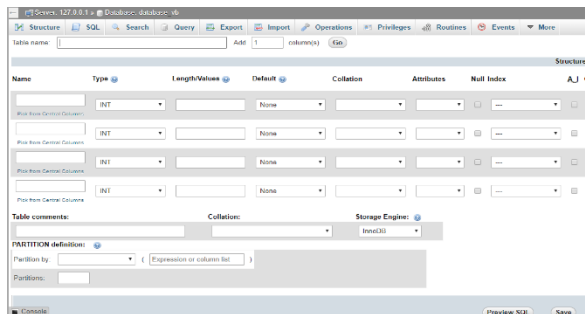
2.5. Database

Basis data (*database*) adalah suatu kumpulan data yang disusun dalam bentuk tabel-tabel yang saling berkaitan maupun berdiri sendiri dan disimpan secara bersama-sama pada suatu media

[5]. Basis data dapat digunakan oleh satu atau lebih program aplikasi secara optimal, data disimpan tanpa mengalami ketergantungan pada program yang akan menggunakannya. Terdapat beberapa aturan yang harus dipatuhi pada file basis data agar dapat memenuhi kriteria sebagai suatu basis data, yaitu kerangkapan data, yang berarti munculnya data-data yang sama secara berulang-ulang pada file basis data. Inkonsistensi data, yaitu munculnya data yang tidak konsisten pada field yang sama untuk beberapa file dengan kunci yang sama. Data terisolasi, disebabkan oleh pemakaian beberapa file basis data.

Program aplikasi tidak dapat mengakses file tertentu dalam sistem basis data, kecuali program aplikasi dirubah atau ditambah sehingga seolah-olah ada file yang terpisah atau terisolasi terhadap file yang lain. Keamanan data, berhubungan dengan masalah keamanan data dalam sistem basis data. Pada prinsipnya file basis data hanya boleh digunakan oleh pemakai tertentu yang mempunyai wewenang untuk mengakses. Integrasi data berhubungan dengan unjuk kerja sistem agar dapat melakukan kendali atau kontrol pada semua bagian sistem sehingga sistem selalu beroperasi dalam pengendalian penuh.

2.5.1. Database MySQL



Gambar 2.4 Database MySQL

MySQL merupakan salah satu perangkat lunak yang menyediakan database SQL (DBMS) yang *multithread*, dan *multi-user*. *Multithread* adalah kemampuan sebuah program untuk melakukan lebih dari satu pekerjaan sekaligus. Keuntungan dari *multithread* adalah sifat respons yang interaktif dan real-time. Multithread ini merupakan teknik pada manipulasi data dimana node-node pada struktur pohon data berisi penunjuk ke node yang

lebih tinggi untuk membuat lintasan struktur menjadi lebih efisien. Sedangkan *Multi-User* adalah suatu sistem dimana lebih dari satu user menggunakan secara bersama satu atau lebih perangkat keras, piranti lunak dan data/ informasi, orang dan prosedur melalui masing-masing komputer atau workstation. MySQL adalah implementasi dari sistem management basis data relasional (RDBMS) [6].

Kehandalan suatu sistem basis data dapat diketahui dari cara kerja pengoptimasinya dalam melakukan proses perintah - perintah SQL yang dibuat oleh pengguna maupun program - program aplikasi yang mememanfaatkannya. Sebagai server basis data, MySQL mendukung operasi basis data transaksional maupun operasi basis data non - transaksional [6]. Pada modus operasi non-transaksional, MySQL dapat dikatakan unggul dalam hal kerja dibandingkan perangkat lunak server basis data kompetitor lainnya. Namun pada modus non - transaksional tidak ada jaminan atas reliabilitas terhadap data yang tersimpan, karenanya modus non-transaksional hanya cocok untuk jenis aplikasi yang tidak membutuhkan reliabilitas data seperti aplikasi *blogging* berbasis *web*, CMS, dan sejenisnya. Untuk kebutuhan sistem yang ditujukan untuk bisnis sangat disarankan untuk menggunakan modus basis data transaksional, hanya saja sebagai konsekuensinya unjuk kerja MySQL pada modus transaksional tidak secepat unjuk kerja pada modus non - transaksional.

My SQL dibuat oleh TeX dan telah dipercaya mengelola system dengan 40 buah database berisi 10.000 tabel dan 500 diantaranya memiliki 7 juta baris. Pada saat ini MySQL merupakan database server yang sangat terkenal didunia, semua itu tak lain karena bahasa dasar yang digunakan untuk mengakses database yaitu SQL (*Structured Query Language*). SQL pertama kali diterapkan pada sebuah proyek riset pada laboratorium riset *san jose*, IBM (*International Business Machines Corporation*) yang bernama system R. Kemudian SQL juga dikembangkan oleh Oracle, Informix dan Sybase. Dengan menggunakan SQL, proses pengaksesan database menjadi lebih mudah dibandingkan dengan yang lain. Dibandingkan dengan database microsoft access, tampilan MySQL sedikit sulit dipahami namun MySQL dapat terhubung dengan internet sehingga *web* yang dibuat sama IP Adressnya dapat mengakses database. Database ini juga dapat diprogram dengan *structur text* dalam membuat tabel, *query*, *relationship* sampai menghapus tabel. Keunggulan dari MySQL sendiri yaitu *software*

yang gratis namun dapat terintegrasi dengan ODBC microsoft windows dan dapat terintegrasi dengan program PHP. Karena program MySQL merupakan program *multiplethread*, sehingga dapat dipasang pada server yang memiliki multi – CPU.

2.5.2. Microsoft Access

Microsoft Access merupakan *software* penyedia database sama seperti MySQL database hanya saja ini berbasis Microsoft. Perbedaannya dengan MySQL, Microsoft Access tidak dapat tersambung dengan internet atau diakses melalui internet sedangkan MySQL dapat. Dari segi grafis, Microsoft Access lebih sederhana sehingga memudahkan pengguna dalam mengelola database. Keunggulan dari Microsoft Access ini dapat dilihat dari segi pemrogramannya yang dapat diprogram juga dengan Bahasa SQL (*Structured Query Leanguage*). Querynya juga dapat diprogram sebagai statement SQL, dan statement tersebut dapat digunakan secara langsung didalam VBA modul pada Microsoft Access untuk secara langsung menghubungkan antar table. Eksistensi yang digunakan Microsoft Access berupa .MDB dikarenakan jenis berkas ini dapat dikoneksikan dengan Microsoft SQL Server. Untuk membuat table, Access menggunakan desain manager yang sudah tersedia table – table dan hanya mengisi data yang di perlukan saja

2.6. OPC

OPC (*Ole for Process Control*) adalah sistem produksi tingkat rendah yang didasarkan pada teknologi akuisisi data tradisional yang dapat diandalkan [7] atau jembatan antara merk yang memungkinkan suatu program aplikasi untuk mengakses sumber data (database) dengan protocol yang sama dan konsisten, OPC bersifat *software bus* dimana program aplikasi (*OPC client*) mengambil data dari OPC data source (*OPC Server*). Dimana OPC server ini dikembangkan oleh vendor masing – masing yang diakses oleh komputer, sehingga pengguna cukup membangun aplikasi *OPC client* sendiri. Spesifikasi OPC didasarkan pada teknologi OLE, COM, dan DCOM yang dikembangkan oleh Microsoft untuk keluarga sistem operasi Microsoft Windows. Spesifikasi ini mendefinisikan satu set standar objek, antarmuka dan metode untuk digunakan dalam kontrol proses dan aplikasi otomatisasi manufaktur untuk memfasilitasi interoperabilitas (kapabilitas dari suatu produk atau sistem yang antar mukanya diungkapkan sepenuhnya untuk berinteraksi dan berfungsi dengan produk atau sistem lain). Pengguna perangkat keras akan

memproses perangkat pemasok perangkat keras dengan mengembangkan antar muka yang sesuai dengan OPC seperti Printer PC, pemasok menyediakan driver printer yang dapat bekerja dengan banyak aplikasi windows. Pengguna dapat memilih terbaik-inclass produk untuk aplikasi tertentu dan dengan mudah mengintegrasikan informasi dari perangkat ini dengan aplikasi perangkat lunak perusahaan.

Spesifikasi OPC yang paling umum adalah Akses Data OPC, yang digunakan untuk membaca dan menulis data real-time [8]. Ketika vendor merujuk ke OPC secara umum, mereka biasanya berarti Akses Data OPC (OPC DA). OPC DA sendiri telah melalui tiga revisi besar sejak awal. Versi kompatibel sebelumnya dalam hal ini Server OPC versi 3 masih dapat diakses oleh klien OPC versi 1, karena spesifikasi menambah fungsionalitas, tetapi masih memerlukan versi yang lebih lama untuk diimplementasikan juga. Namun, Klien dapat ditulis yang tidak mendukung fungsi yang lebih lama karena semuanya dapat dilakukan menggunakan yang lebih baru, jadi Klien yang kompatibel dengan DA-3 tidak akan selalu bekerja dengan Server DA 1.0. Selain spesifikasi OPC DA, OPC Foundation juga mempertahankan spesifikasi OPC *Historical Data Access* (HDA). Berbeda dengan data waktu nyata yang dapat diakses dengan OPC DA, OPC HDA memungkinkan akses dan pengambilan data yang diarsipkan oleh database. OPC memungkinkan berbagai aplikasi yang ditulis Bahasa yang berbeda berjalan pada platform windows yang berbeda untuk di integrasikan.

Pada OPC Server, terdapat beberapa macam OPC *interface* utama, yang pertama yaitu OPC data akses untuk memindahkan data real time dari PLC, DSC, dan perangkat control lainnya ke HMI dan *client display* lainnya. Kedua OPC *Alarm* dan *Event* yang menyediakan alarm dan *event* pemberitahuan *on demand* (berbeda dengan aliran data continue data access) ini termasuk alarm proses, tindakan operator, dan pesan pelacakan. Ketiga OPC *Data Exchange* yaitu OPC yang membawa kita dari *client* atau *server* ke *server* dengan komunikasi melalui jaringan fieldbus Ethernet. Keempat OPC *Historical Data Access* yang menyediakan akses ke real time, terus menerus mengubah data dan menyediakan access ke data yang disimpan dari system SCADA. Kelima OPC *Security* sebagai OPC untuk proses keamanan dan menentukan bagaimana mengontrol akses *client* ke *server* untuk melindungi informasi yang sensitif. Keenam OPC XML – DA yaitu OPC yang Menyediakan fleksibel, aturan, dan format yang konsisten untuk mengungkap data yang

menggunakan XML dan Web Service. Ketujuh OPC *Complex Data* yang merupakan Sebuah pendamping spesifikasi untuk akses data dan XML – DA yang memungkinkan *server* untuk mengekspos dan menjelaskan tipe data yang lebih rumit seperti *structure* biner dan dokumen XML. Yang terakhir atau yang kedelapan yaitu OPC *Command* yang merupakan Sebuah kelompok kerja yang dibentuk untuk mengembangkan suatu set baru antar muka yang memungkinkan OPC *client* dan *server* untuk mengidentifikasi, mengirim dan memantau perintah control yang dijalankan perangkat.

Penggunaan OPC ini menggunakan CX Server Lite yang merupakan bawaan dari Omron. Penggunaan CX Server Lite ini dipilih karena *software* ini tidak perlu floating di layar PC dan memiliki kapasitas yang sedikit dibandingkan OPC lain. CX Server Lite kali ini digunakan dalam Software Visual Studio yang berbasis .NET. Dalam Visual Studio untuk membuat koneksi dengan PLC Omron menggunakan CX Server Lite yang berada pada menu *Toolbox* yang bernama CxsLiteCtrl. Server lite ini menggunakan eksistensi berupa .cdm yang berisi memory data dari PLC yang dibuat melalui CX Programmer. Pada CX Programmer, program ditentukan dengan menggunakan Data Memory yang akan dipakai lalu membuat file .cdm pada newPLC1 dengan nama yang diinginkan. Pada file .cdm dituliskan terlebih dahulu Data Memory yang akan diisi. Pada komponen CxsLiteCtrl terdapat menu properties yang didalamnya terdapat import dari file .cdm tadi sehingga data memory dapat diubah melalui visual studio. Komunikasi ini melibatkan HMI, PLC dan RFID karena ketiga komponen ini saling terintegrasi sesuai dengan cara kerja yang ada pada pembahasan *automation Tool's crib*.

2.7. RFID

Teknologi RFID merupakan teknologi yang menggunakan komunikasi via gelombang elektromagnetic untuk merubah data antara terminal dengan suatu objek seperti barang, hewan maupun manusia [9]. sistem yang secara jarak jauh dan nirkabel digunakan untuk mengidentifikasi peralatan lainnya yang disebut *transponder* atau yang biasa disebut dengan *tag* menggunakan *interrogator* atau *reader*. Tag memiliki identitas unik yang biasa digunakan untuk mengidentifikasi objek yang ditandai. RFID tag dapat bersifat aktif atau pasif [9]. RFID *tag* yang pasif tidak memiliki *power supply* sendiri, sehingga harganya pun lebih murah dibandingkan dengan *tag* yang aktif. Dengan hanya berbekal induksi listrik yang ada pada antena yang disebabkan oleh adanya pemindaian

frekuensi radio yang masuk, sudah cukup untuk memberi kekuatan yang cukup bagi RFID *tag* untuk mengirimkan respon balik.

Tiidak adanya *power supply* pada RFID *tag* yang pasif akan menyebabkan semakin kecilnya ukuran dari RFID *tag* yang mungkin dibuat, bahkan lebih tipis daripada selembar kertas dengan jarak jangkauan yang berbeda mulai dari 10 mm sampai dengan 6 meter. RFID *tag* yang aktif memiliki *power supply* sendiri dan memiliki jarak jangkauan yang lebih jauh. Memori yang dimilikinya juga lebih besar sehingga bisa menampung berbagai macam informasi di dalamnya. Suatu sistem RFID dapat terdiri dari beberapa komponen, seperti *tag*, *tag reader*, *tag programming station*, *circulation reader*, *sorting equipment*, dan tongkat *inventory tag*. Kegunaan dari sistem RFID ini adalah untuk mengirimkan data dari *tag* yang kemudian dibaca oleh RFID *reader* dan kemudian diproses oleh aplikasi computer. Data yang dipancarkan dan dikirimkan tadi bisa berisi beragam informasi, seperti ID, informasi lokasi atau informasi lainnya. Dalam suatu sistem RFID sederhana, suatu objek dilengkapi dengan *tag* yang berisi *microchip* yang ditanamkan di dalamnya yang berisi sebuah kode produk yang sifatnya unik. Pada mesin *automation Tool's crib*, RFID digunakan sebagai alat *scan* dari tag RFID pengguna dan tag RFID alat dan data dari RFID dimasukkan kedalam database sebagai penanda bahwa tag tersebut memiliki nama.

2.7.1. Interrogator

Penggunaan *Interrogator* atau yang biasa disebut *transmitter* RFID merupakan bagian dari sistem RFID yang digunakan untuk mengirim sinyal ke *transponder* atau penerima supaya data bisa diterima. Tugas utama dari alat ini ada dua, yang pertama di fase *downlink* untuk menyediakan daya pada tag pasif dan modulasi sinyal sehingga tag pasif menerima perintah dan data dari alat, dan di fase *uplink*, *interrogator* harus menyediakan sinyal yang tidak dimodulasi agar tag mampu mengembalikan data pada pengirim. *Interrogator* merupakan peralatan yang penting di sistem RFID, oleh karena itu harus dapat digunakan di banyak channel frekuensi radio secara akurat, dan dapat berpindah ke channel lainnya dengan cepat untuk menyesuaikan dengan penggunaan. *Transmitter* yang baik dalam sistem RFID adalah yang memiliki jarak baca yang baik, sehingga untuk memenuhi persyaratan tersebut diperlukan daya yang besar yang mampu diterima oleh *tag*.

2.7.2. Transponder

Sistem komunikasi pada RFID , untuk melengkapi sistem diperlukan pereangkat yang memberikan umpan balik sinyal kepada *interrogator* saat mengirimkan sinyal informasi, *transponder* dapat menerima sinyal dari *interrogator* jika memiliki frekuensi, protocol dan orientasi yang sama dikedua alat komunikasi. Batasan dari tag pasif ada pada modulasi sinyal umpan balik yang memiliki frekuensi yang bervariasi , sehingga pada tag pasif, permasalahan demodulasi dan decoding jarang sekali terjadi, dibandingkan menggunakan komunikasi radio dengan teknologi berdasarkan pada sinyal fasa dan amplitudo dengan kode *error-correcting* . di sisi lain, tag pasif memiliki permasalahan dalam sinyal interferensi yang menghalangi sinyal dari transmitter untuk dapat memberikan umpan balik , atau interferensi pada sinyal pantulan dari tag. Sinyal interferensi yang dipancarkan oleh *interrogator* yang memiliki amplitudo yang tidak beraturan dan diterima oleh tag akan mengalami efek dari perubahan modulasi sehingga saat kembali ke normal, tag dapat mengirimkan sinyal pantulan untuk mengirimkan kembali informasi. Penentuan frekuensi pada tag secara detail yang diatur pada *interrogator* berpengaruh pada pemilahan tag yang berbeda frekuensi dengan transmitter, sehingga komunikasi diantara dua perangkat dengan frekuensi yang sama dapat dikendalikan oleh *interrogator* sebagai *factor control*

2.8. HMI (Human Machine Interface)

Tampilan pada layar mesin yang disebut HMI merupakan sebuah sistem yang menghubungkan antara manusia dan mesin atau bisa disebut dengan tampilan. Pada dunia industri, HMI dapat berupa GUI (*Graphic User Interface*) pada layar monitor yang akan dihadapi oleh operator suatu mesin maupun pengguna yang membutuhkan data kerja mesin. HMI mempunyai kemampuan dalam hal visualisasi untuk monitoring dan data mesin yang terhubung secara online dan real time. HMI akan memberikan suatu gambaran kondisi mesin yang berupa peta mesin produksi di layar monitor dimana dapat dilihat bagian mesin mana yang sedang bekerja. *Monitoring* dapat berupa melihat kerja dari sistem, kerusakan sistem, dan pengambilan keputusan jika terjadi *emergency* [10]. Sistem HMI biasanya bekerja secara online dan real time dengan membaca data yang dikirimkan melalui I/O port yang digunakan oleh sistem controller-nya.

Port yang biasanya digunakan untuk controller dan akan dibaca oleh HMI antara lain adalah port com, port USB, port RS232 dan ada pula yang menggunakan port serial. HMI memiliki fungsi memberikan informasi plant yang up-to-date kepada operator melalui GUI, menerjemahkan instruksi operator ke mesin, memonitor keadaan yang ada di plant, mengatur nilai pada parameter yang ada di plant, mengambil tindakan yang sesuai dengan keadaan yang terjadi, memunculkan tanda peringatan dengan menggunakan alarm jika terjadi sesuatu yang tidak normal, dan menampilkan pola data kejadian yang ada di plant baik secara real time. Pada mesin *automation Tool's crib*, HMI digunakan sebagai tampilan keadaan biodata karyawan, data *Tool's*, *history* peminjaman, *history* pengembalian, dan *history* transaksi.

2.9. Solenoid Door Lock

Komponen elektronika ini digunakan khusus untuk mengunci pintu. Solenoid ini memiliki tuas yang khusus dan kuat untuk menahan beban serta memiliki tuas miring dengan dua sekrup untuk membukanya dengan sudut putar 90, 180. Atau 270 derajat sehingga cocok dengan pintu yang digunakan. Komponen ini akan bergerak apabila diberikan tegangan tergantung dengan spesifikasi solenoid itu sendiri. Pada kondisi normal, solenoid dalam keadaan tuas memanjang yang berarti dalam posisi mengunci. Apabila diberi tegangan, tuas solenoid akan memendek yang berarti sedang dalam keadaan membuka. Komponen ini dipakai sebagai komponen keamanan suatu brangkas atau lemari yang berisi barang berharga. Solenoid ini bisa digabungkan dengan sistem pengunci elektrik berbasis RFID dan password.

Solenoid ini mempunyai dua system kerja, yaitu *Normali Close* (NC) dan *Normaly Open* (NO). Apabila solenoid NC diberi tegangan, maka solenoid akan memanjang sedangkan solenoid NO kebalikan dari solenoid NC. Kebanyakan solenoid *Door Lock* membutuhkan *input* dan tegangan sebesar 12 VDC tetapi ada juga solenoid *Door Lock* yang hanya membutuhkan tegangan sebesar 5 VDC sehingga dapat langsung bekerja dengan tegangan *output* darinpip IC digital. Didalam solenoid terdapat kawat yang melingkar pada inti besi. Ketika arus listrik mengalir melalui kawat ini, maka terjadi medan magnet untuk menghasilkan energi yang akan menarik inti besi ke dalam.

2.9.1. Solenoida Linear



Gambar 2.5 Solenoid Linear

Komponen ini adalah alat elektromagnetik atau elektromekanis yang mengubah energi listrik menjadi sinyal magnetic atau gerakan mekanis. Solenoid jenis ini disebut dengan solenoid linear karena plunger atau aktuatoarnya bergerak secara linear. Solenoid ini biasanya tersedia dalam dua bentuk konfigurasi dasar yaitu solenoid linear tipe Tarik (*Pull Type*) yang dapat menarik beban kearah dirinya apabila diberi arus listrik dan solenoida linear tipe dorong (*Push Type*) yang dapat mendorong beban menjauhi dirinya apabila di berikan arus listrik secukupnya.

Pada umumnya, kontruksi dan struktur dasar solenoid linear tipe Tarik maupun tipe dorong adalah sama, perbedaannya hanya terletak di desain plunger dan arah pegasnya. Cara kerja solenoida linear yaitu ketika arus listrik diberikan ke koil, koil tersebut akan menghasilkan medan magnet. Medan magnet akan menarik plunger yang berada di dalam koil masuk ke pusat koil dan mengkompreskan pegas yang terdapat di satu ujung plunger tersebut. Gaya dan kecepatan plunger tergantung pada kekuatan fluks *magnetic* yang dihasilkan oleh koil. Bila arus listrik dimatikan (OFF), medan electromagnet yang dihasilkan sebelumnya akan hilang sehingga energy yang tersimpan pada pegas yang di kompres tersebut akan mendorong plunger keluar ke posisi semula.

Pada mesin *Automation Tool's Crib* ini *door lock* digunakan untuk mengunci pintu lemari sehingga menjadi komponen *safety* dari mesin tersebut. Solenoid ini dipakai pada mesin *automation Tool's crib* dan dietakan di bagian bawah lemari seperti pada Gambar 2.6. Solenoid ini bekerja saat karyawan melakukan login dan visual studio

mengirim program yang mengaktifkan solenoid selama 10 detik, hal ini dilakukan agar solenoid tidak mengalami *overheat* yang dapat mengakibatkan kerusakan pada solenoid.

2.9.2. Solenoid Rotasi



Gambar 2.6 Solenoid Rotasi

Komponen ini adalah solenoid yang menghasilkan gerakan sudut atau gerakan putar (rotasi) dari posisi netral ke posisi searah jarum jam ataupun posisi berlawanan dengan arah jarum jam dengan sudut tertentu. Solenoid jenis rotasi ini dapat digunakan untuk menggantikan fungsi motor DC kecil ataupun motor stepper yang sudut gerakannya sangat kecil. Berdasarkan sudut gerakannya, solenoid rotasi biasanya tersedia dalam sudut gerakan 25 derajat, 35 derajat, 45 derajat, 60 derajat dan 90 derajat. Ada juga yang tersedia dalam bentuk gerakan yang dapat menuju ke sudut tertentu kemudian kembali lagi ke posisi awal (posisi nol). Cara kerja solenoid rotasi yaitu dapat menghasilkan gerakan rotasi ketika diberikan energy atau arus listrik ataupun pada saat berubah polaritas medan electromagnet. Solenoid rotasi terdiri dari gulungan listrik yang dililitkan disekitar rangka baja dengan disk magnetic yang terhubung ke poros *output* yang berada di atas koil. Pada saat diberikan arus listrik, medan electromagnetic menghasilkan kutub utara dan kutub selatan yang menolak kutub magnet permanen yang berdekatan sehingga menyebabkan putaran pada sudut yang ditentukan oleh konstruksi mekanis solenoid rotasi itu sendiri.

2.10. *Limit Switch*

Saklar ini merupakan jenis saklar yang dilengkapi dengan katub yang berfungsi menggantikan tombol. Prinsip kerja limit switch seperti saklar push ON yang hanya akan terhubung pada saat katubnya ditekan pada batas penekanan tertentu yang telah ditentukan dan akan memutus saat katup apabila tidak ditekan. *Limit switch*

digunakan untuk mengendalikan mesin sebagai bagian dari sistem kontrol, sebagai pengaman interlock, atau untuk menghitung objek yang melewati suatu titik. *Limit Switch* adalah perangkat elektromekanik yang terdiri dari aktuator yang secara mekanis terkait dengan sekumpulan kontak. Ketika suatu benda bersentuhan dengan aktuator, perangkat mengoperasikan kontak untuk membuat atau sambungan memutuskan listrik. Saklar batas digunakan dalam berbagai aplikasi dan lingkungan karena kekasarannya, kemudahan instalasi, dan keandalan operasi. Mereka dapat menentukan ada atau tidaknya, lewat, posisi, dan akhir perjalanan suatu objek. *Limit Switch* digunakan dalam berbagai aplikasi dan lingkungan karena ketahanannya, kemudahan instalasi, dan keandalan operasi. Komponen ini dengan operator tuas rol dipasang di gerbang pada kunci kanal, dan menunjukkan posisi gerbang ke sistem kontrol.

Limit Switch standar adalah komponen kontrol industri yang diproduksi dengan berbagai jenis operator, termasuk tuas, roller plunger, dan jenis kumis. *Limit Switch* dapat langsung dioperasikan secara mekanis oleh gerakan tuas operasi. *Reed Switch* dapat digunakan untuk menunjukkan kedekatan magnet yang dipasang pada beberapa bagian yang bergerak. Proximity beroperasi oleh gangguan medan elektromagnetik, dengan kapasitansi, atau dengan merasakan medan magnet. Perangkat operasi terakhir seperti lampu atau katup solenoid akan langsung dikendalikan oleh kontak saklar batas industri, tetapi lebih biasanya saklar batas akan ditransfer melalui relai kontrol, rangkaian kontrol kontaktor motor, atau sebagai *input* ke pengontrol logika yang dapat diprogram.



Gambar 2.7 Limit Switch

2.11. Kabel Ethernet

Kabel komunikasi LAN (*Local Area Network*) antara PC dengan *Device* lain biasanya menggunakan kabel Ethernet. Kabel

Ethernet menggunakan kabel yang berupa *Transmitter* yang menyalurkan informasi dari *device* ke *device* lain dengan mengubah informasi menjadi sinyal listrik. Kabel ini memiliki kemampuan mengirim banyak data dalam waktu yang cepat dan jangkauan yang jauh. Kecepatan kabel Ethernet dapat mencapai 10 Mb/sec. frekuensi yang di keluarkan juga lebar sehingga dapat mentransmitter kanal frekuensi tinggi seperti televisi. Ethernet dibuat dari bahan tembaga yang dibentuk silinder atau memutar dengan diameter tidak melebihi 0,02 mm dan 1,53 mm. Kabel Ethernet di komunikasikan dengan PLC dan PC yang mempunyai port khusus Ethernet, karena ujung kabel Ethernet berbeda dengan USB pada umumnya dan cara komunikasinya juga berbeda dengan USB. Jika pada USB dikomunikasikan dengan PC maka hanya tinggal setting pada COM dari PC, sedangkan untuk komunikasi Ethernet harus menggunakan setting IP Adress pada PC pengguna. Penggunaan IP Adress ini sebagai alamat atau identitas dari suatu perangkat misalnya antara PC dengan PC atau PC dengan PLC sehingga perangkat tersebut dapat berkomunikasi dengan perangkat lainnya. IP Address sendiri merupakan angka yang berupa numerik yang memiliki 4 blok decimal dengan nilai yang dibatasi antara 0 sampai 255.

Ethernet juga terdapat jenis yang berbeda beda yaitu Ethernet yang umum digunakan dalam sehari hari. Kecepatan pengiriman data mencapai kecepatan 10 Mbit/sec. kedua ada Fast Ethernet yaitu Ethernet yang memiliki kecepatan transfer data mencapai 100 Mbit/sec. yang ketiga yaitu Giga Ethernet adalah Ethernet yang memiliki kecepatan transfer mencapai 1000 Mbit/sec [11]. Pada mesin *automation Tool's crib*, menggunakan kabel Ethernet untuk mengurangi tabrakan komunikasi antara PLC dengan RFID karena RFID menggunakan kabel Serial yang membutuhkan konfigurasi COM pada PC. Saat percobaan menggunakan kabel serial untuk PLC dan RFID, terjadi hilangnya komunikasi dari PLC dan RFID secara bergantian sehingga terjadi *error*.

BAB III

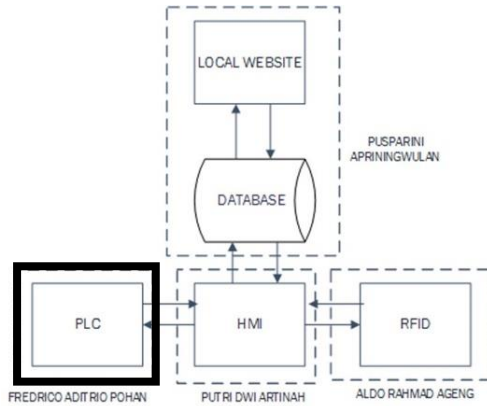
PERANCANGAN ALAT

Pada perancangan alat untuk Komunikasi antar PLC dan Visual Studio menggunakan CX – Server Lite ini memiliki beberapa tahap yaitu tahap prosedur sistem *automation Tool's crib*, rangkaian daya, rangkaian I/O, rangkaian relay untuk *solenoid*, perancangan komunikasi PLC menggunakan *platform* Visual Studio. Penggunaan PC sebagai kontroler dan master untuk RFID dan PLC. Penggunaan RFID pada mesin sebagai pendeteksi identitas dari pengguna dan identitas dari peralatan. Tag RFID yang dipakai berupa kartu sebagai kartu karyawan untuk mengakses mesin dan tag peralatan untuk menentukan identitas dari peralatan yang akan dipinjam ataupun dikembalikan oleh pengguna. RFID menggunakan komunikasi serial yang dihubungkan dengan RS 232 dan dikonversikan dengan USB pada PC.

Data RFID diterima oleh PC menggunakan *platform* Visual Studio. Komunikasi PLC dengan PC menggunakan OPC CX – Server Lite yang diprogram melalui *platform* Visual Studio sehingga I/O dari PLC dapat dimonitoring melalui PC. Sistem akan menunjukkan indikator kuning dan mesin dalam keadaan terkunci menggunakan *solenoid door lock*. PC juga sebagai master dari HMI untuk integrasi dari RFID dan PLC sebagai tampilan kondisi mesin. Database sendiri sebagai penyimpan data dari ID pengguna yang diambil dari *scan* RFID, ID alat, dan waktu transaksi. Dalam pengerjaan mesin ini dibagi menjadi 4 ruang lingkup kerja untuk masing-masingnya agar pengerjaan mesin ini dapat lebih cepat selesai dan lebih efisien. Pembagian ini sebelumnya sudah didiskusikan terlebih dahulu sehingga sesuai dengan yang diinginkan oleh masing – masing orangnya. Ruang lingkup kerja dapat dilihat pada Gambar 3.1.

3.1. Perancangan Hardware

Perancangan ini dibagi menjadi subbab yang akan dibahas mulai dari rancang bangun *pilot project* yang merupakan miniatur dari mesin ini, perancangan sistem *automation Tool's crib*, rangkaian daya, rangkaian, I/O, dan rangkaian *relay* untuk *solenoid door lock*.



Gambar 3.1 Ruang Lingkup Kerja

3.1.1. Rancang Bangun *Pilot Project*



Gambar 3.2 *Pilot Project*

Pembuatan miniatur dari mesin *automation Tool's crib* yang dinamakan *pilot project* untuk menentukan masalah dan potensi yang akan terjadi apabila mesin sudah dibuat sehingga dapat dicari solusi dan pencegahan saat terjadi hal tersebut. Penggunaan komponen dalam *pilot project* ini menggunakan box yang terbuat dari akrilik sebagai pengganti lemari dapat dilihat pada Gambar 3.2. Penggunaan tag RFID juga lebih sedikit hanya sampai 5 tag saja untuk lima alat. Sistem peminjaman dapat dilakukan dengan mendekati kartu RFID ke RFID, lalu layar HMI akan menunjukkan biodata dari pengguna dan dapat melakukan transaksi peminjaman atau pengembalian. Setelah transaksi selesai, peralatan yang dipinjam atau dikembalikan akan disimpan dalam database beserta biodata dan waktu transaksi.

3.1.2. Perancangan Sistem *Automation Tool's Crib*



Gambar 3.3 Desain *Automation Tool's Crib*

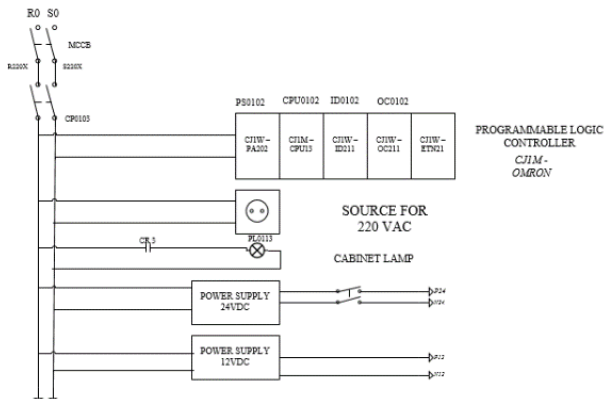
Tool's Crib merupakan tempat penyimpanan aset peralatan dari suatu perusahaan. Selain sebagai tempat penyimpanan, *Tool's crib* memiliki fungsi sebagai tempat peminjaman alat perkakas untuk proses *assembly* mesin. Sistem pelayanan *Tool's Crib* menggunakan koin untuk peminjamannya. Satu koin ditukar untuk satu peralatan saja. Untuk kepemilikan koin, setiap karyawan diberikan kurang lebih 10 koin.

Automation Tools Crib, sama dengan *Tool's Crib* hanya saja proses transaksi pinjam dan kembali dilakukan secara *self service* dan otomatis sehingga tidak perlu menggunakan operator. *Automation Tool's Crib* ini menggunakan tag RFID khusus karyawan sebagai akses untuk melakukan proses transaksi pinjam dan kembali sehingga tidak perlu menggunakan koin. Selain dilakukan secara *self service*, pembukuan peminjaman alat yang awalnya ditulis manual oleh operator sekarang menjadi pembukuan otomatis dengan mengkonversi data peminjam alat beserta alatnya dari database menuju microsoft excel. Mesin ini memiliki durasi transaksi yang lebih cepat dari pada *Tool's Crib* konvensional. Mesin ini juga dapat beroperasi selama 24 jam karena tidak membutuhkan tenaga manusia dan dapat dioperasikan saat perusahaan sedang libur dan ada karyawan yang sedang lembur.

Masalah perawatan dan perbaikan pada mesin ini tidak perlu dilakukan dua bulan sekali seperti yang dilakukan pada divisi tersebut. Perawatan dan perbaikan mesin ini dapat dilakukan secara rutin melalui program *maintenance* yang ada *form* admin. Didalam *form* ini juga dapat digunakan untuk *troubleshooting sensor* dan *indikator* dengan mengirim *value* dari masing – masing *sensor* dan

indikator. Proses perawatan dan perbaikan ini membutuhkan bantuan dari operator sebagai pengawas saat *maintenance*.

3.1.3. Rangkaian Daya Pada Mesin *Automation Tool's Crib*

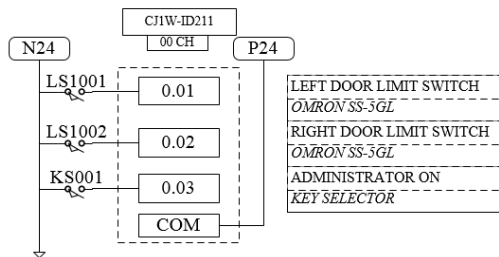


Gambar 3.4 Rangkaian daya mesin *Automation Tool's Crib*

Pembuatan diagram *wiring* dimulai dengan rangkaian daya yang dapat dilihat pada Gambar 3.4. Mesin disuplai dengan tegangan ke seluruh komponen yang ada didalam panel yaitu : PLC Omron CJ1M CPU13, PA202, ID211, OC211, relay, 24 VDC *power supply*, dan 12 VDC *power supply*. Rangkaian tersebut dialiri dengan tegangan sumber sebesar 220 V dan menggunakan pengaman MCB (*Magnetic Circuit Breaking*) agar apabila terjadi peningkatan daya diluar batas dari *breaking capacity* dari MCB, maka MCB akan memutus aliran listrik. Komponen – komponen didalamnya juga dilindungi oleh CP (*Circuit Protector*) yang berfungsi untuk melindungi sirkuit listrik dari kerusakan yang disebabkan oleh kelebihan arus dari kelebihan beban atau korsleting. Fungsi dasarnya adalah untuk mengganggu aliran arus setelah kesalahan terdeteksi. Tidak seperti sekering, yang beroperasi sekali dan kemudian harus diganti, pemutus sirkuit dapat direset (baik secara manual atau otomatis) untuk melanjutkan operasi normal.

3.1.4. Rangkaian I/O Pada Mesin *Automation Tool's Crib*

Rangkaian ini menggunakan PLC. PLC sebagai kontroler yang mengendalikan sensor dan indikator untuk memenuhi kebutuhan dari sistem. Penggunaan PLC dengan tipe modular menjadi pilihan karena menyesuaikan dengan bentuk panel yang telah dibuat. Penggunaan PLC diatur dengan menggunakan *common NPN* yang di koneksikan dengan modul *input* tipe CJ1W – ID211 yang memiliki 16 *point* untuk sensor dan indikator. Sensor dan indikator yang dipakai yaitu *limti switch* dan dipasang pada bagian bawah lemari untuk menandakan pintu dalam kondisi tertutup atau terbuka. *Output* yang digunakan yaitu modul CJ1W – OC211 memiliki 16 *point* dan diatur dengan *common NPN* sama seperti *input*. Komponen dari *output* yaitu *solenoid door lock*, *buzzer* dan relay dan untuk indikatornya ada lampu merah sebagai indikator mesin dalam keadaan urgensi tinggi atau eror, lampu kuning sebagai indikator mesin dalam keadaan *stanby* atau menunggu proses transaksi, dan lampu hijau sebagai indikator mesin dalam keadaan melakukan transaksi

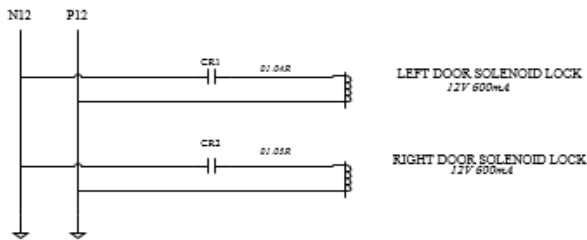


Gambar 3.5 Rangkaian I/O *Automation Tool's Crib*

3.1.5. Rangkaian Relay Untuk Solenoid Door Lock

Rangkaian ini berfungsi sebagai komponen *safety* dari mesin *Automation Tool's Crib*. *Tool's* yang berada didalam lemari tercatat sebagai peralatan tersedia. Sistem keamanan dari *door lock* berfungsi untuk mencegah adanya pencurian atau penyalahgunaan mesin. *Solenoid door lock* akan mengunci secara otomatis setelah pintu dibuka dan saat pintu ditutup, *solenoid* dalam keadaan *retract* sehingga pintu sudah tidak dapat dibuka kembali selain melalui proses login ulang.

Common yang digunakan untuk rangkaian ini menggunakan common NPN dan kontaknya menggunakan *normally open*. Saat output dari PLC dialiri tegangan sebesar 24 V pada relay maka keadaan koil yang awalnya *normally open* menjadi tersambung dengan *solenoid door lock* dan membuat *solenoid* menarik bagian ujungnya agar lemari dapat dibuka.

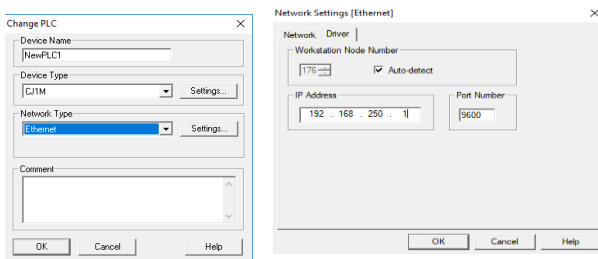


Gambar 3.6 Rangkaian *relay Automation Tool's Crib*

3.2. Perancangan Software

Subab yang dibahas pada perancangan *software* ini meliputi komunikasi antar PLC dan Visual Studio meliputi konfigurasi PLC pada CX – Programmer, konfigurasi file bereksistensi CDM yang merupakan file berisi *data memory* dari PLC, perancangan OPC CX – Server Lite pada Visual Studio, Program *read* dan *write* indikator serta *door lock*.

3.2.1. Konfigurasi PLC pada CX Programmer



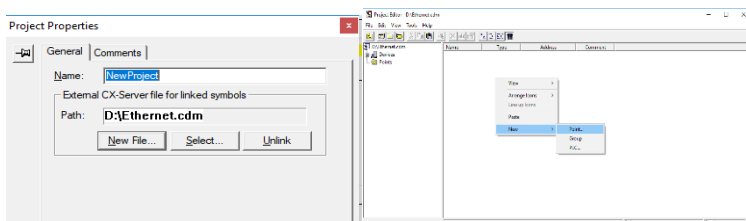
Gambar 3.7 Konfigurasi PLC pada CX Programmer

CX Programmer menggunakan *Device* dengan tipe CJ1M karena *Hardware* yang digunakan merupakan PLC tipe CJ1M. Pada

Device Type dapat dipilih CPU yang sesuai dengan type PLC yang dipakai, karena PLC yang dipakai tipe CJ1M CPU 13 jadi dipilih CPU 13. Pemilihan CPU 13 ini karena memiliki spesifikasi yang besar seperti memiliki 640 point/ 20 unit dan dapat di ekspansi maksimal satu rak dan memiliki 20K steps. CPU 13 juga terdapat ETN (Ethernet) yaitu RJ 45 atau tempat kabel Ethernet di colokan dan disambungkan dari PLC ke PC/Laptop. Pada bagian *Network Type* pilih Ethernet karena komunikasi yang digunakan menggunakan kabel Ethernet.

Gambar 3.2 menunjukkan cara setting *Network Type* dengan memasukan IP yang akan dikirim ke PLC atau IP PLC itu sendiri. Pemilihan IP juga tergantung pemakai, tetapi *default* yang digunakan dalam industri menggunakan 192.168.250.1 untuk memudahkan programmer PLC bagian *maintenance* di industri. Setelah setting IP pada PLC, dapat setting IP Address PC/Laptop dengan konfigurasi pada masing – masing platform PC/Laptop. Platform Windows menggunakan IPv4 untuk mengganti IP Address PC/Laptop pengguna, biasanya menggunakan 192.168.250.2 sehingga komunikasi dengan PLC mudah karena menggunakan IP Address yang mendekati.

3.2.2. Konfigurasi File .cdm



Gambar 3.8 Setting File .cdm

Pembuatan file bereksistensi .cdm ini di buat dengan *software* CX - Programmer seperti pada Gambar 3.3 karena file ini menambahkan point yang isinya berupa Data Memory dari PLC yang digunakan sehingga Data Memory dapat diubah – ubah pada program Visual Studio. Spesifikasi file .cdm ini sesuai dengan CPU yang digunakan, apabila CPU yang digunakan CPU 13 dengan spesifikasi 640 Point/20 Unit pada file .cdm nya juga memiliki spesifikasi yang sama. Point yang digunakan dapat di pilih dan dapat diganti namanya sesuai keinginan pengguna.

Membuat file bereksistensi .cdm pada CX Programmer melalui New Project yang ada pada *Toolbar* sebelah kiri kemudian klik kanan dan pilih *properties*. Muncul tampilan seperti Gambar 3.8 kemudian bisa membuat file baru dengan New File atau menggunakan file yang sudah ada dengan select. File yang telah dibuat akan disimpan dalam *directory* pengguna masing – masing. File ini nantinya akan di import ke CX Server Lite yang berada di form Visual Studio. Setelah membuat filenya, langkah selanjutnya cek Data Memory yang ada pada PLC menggunakan CX – Programmer, dengan memilih pilihan memory pada *Toolbar* yang ada di sisi kiri. Data Memory yang akan digunakan harus kosong atau yang sedang tidak digunakan karena mengurangi error yang ada akibat dari penggunaan Data Memory yang bertabrakan.

Pada file .cdm berisikan device dan point. Device berisi nama PLC yang diberikan dan jenis komunikasi yang digunakan oleh PLC, bila menggunakan serial akan muncul komunikasi toolbus sedangkan untuk Ethernet akan muncul komunikasi Ethernet. Pada menu Points berfungsi membuat point baru dengan cara seperti yang ditunjukkan Gambar 3.4 point ini diisi sesuai dengan nama PLC yang digunakan dan Data Memory yang telah dipilih dan akan dipakai. Point ini dapat diberi nama sesuai dengan keinginan pengguna.

3.2.3. Perancangan OPC pada Visual Studio



Gambar 3.9 Setting File didalam CX Server Lite

Visual Studio merupakan platform yang isinya dapat membuat suatu project berupa HMI (*Human Machine Interface*) atau Web base dengan eksistensi .NET. Pada Visual Studio ini dapat dijadikan HMI

untuk PLC, namun harus menggunakan perantara antar merk dari Visual Studio dengan Platform lain. Perantara yang digunakan yaitu OPC CX – Server Lite yang merupakan pihak ketiga untuk menyambungkan antar platform Visual Studio dengan PLC Omron CJ1M karena CX – Server Lite adalah OPC khusus untuk PLC type Omron saja. Cara penggunaannya, hanya perlu mendownload *Software* ini di Web resmi dari Omron dan diinstall pada PC/Laptop pengguna. Pada Visual Studio setelah di install CX – Server Lite dapat menambahkan pada menu *Tool'sBox* dan memilih komponen yang telah di install tadi. Komponen yang digunakan sebagai komunikasi saja dapat menggunakan CXSLiteCtrl atau CX Lite Control karena CX Lite Control ini berfungsi sebagai perantara antara Visual Studio dengan PLC. Ada banyak komponen omron didalamnya seperti komponen 7 segment apabila memang dibutuhkan. Kelebihan dari CX Server Lite sendiri yaitu dapat digunakan dalam platform yang berdasarkan Microsoft dengan menambah Extension dari CX Server Litenya saja dan penggunaannya juga sangat flexible.

Pada Gambar 3.9 tampak menu *project name* yang nantinya menu tersebut akan meminta untuk import file yang bereksistensi .cdm yang sudah terisi dengan Data Memory tadi. Logo dari CX Lite Control ini nantinya akan *Unvisible* atau tidak terlihat saat program di *Start* sehingga tidak perlu lagi membuat program untuk tidak menampilkan kan logo tersebut. Dalam program, CX Lite Control ini mempunyai deklarasi sendiri namun penggunaan fungsi syaratnya sama dengan program Visual Basic.NET. Sebelum menggunakan CX Lite Control ini, reference yang ada di Visual Studio juga harus ditambahkan karena apabila tidak ditambahkan, program dalam Visual Studio akan error karena tidak dapat membaca fungsi dari CX Lite Control yang digunakan. Penambahan *reference* ini dapat dilakukan di didalam menu project lalu pilih *add reference* lalu pilih CX – Server Lite atau juga dapat menambah *reference* dengan memasukan *library* dengan file berkesistensi dll dengan cara mengcopy file dll nya kemudian pindahkan kedalam dokumen program yang akan dimasukan referencenya pada folder debug. Setelah reference ditambahkan, dalam program dimasukan program untuk *import* dari *reference* tadi sehingga *reference* dapat dipanggil dalam program. Letak program import ini berada diatas *public sub* sehingga tidak memerlukan *end sub* dikarenakan *import* ini akan dipakai secara terus menerus.

3.2.4. Program Read Visual Studio

Pemrograman Visual Studio dapat di buat dengan fungsi – fungsi yang ada. Pemrograman Read PLC merupakan salah satu cara untuk membaca PLC dari Visual Studio. Lampiran A-5 menunjukkan pada baris pertama yaitu *private sub limitpress()* merupakan kondisi dimana program yang dibuat digunakan pada saat limit dalam keadaan *press*. Pada baris kedua menunjukkan deklarasi dari D12X sebagai sebuah string dan point. Untuk program read terdapat pada baris 3 dan 4. Program Read ini digunakan untuk membaca nama PLC dan point yang digunakan serta *value* yang terdapat di Data Memory yang berada pada file berkesistensi *.cdm*. Setelah membaca keadaan *value* PLC terdapat program *if* yang merupakan program untuk mengaktifkan sub *device* lainnya apabila *value* sudah terpenuhi. Program Read ini ditampilkan pada textbox sebagai indikator apakah program sudah bisa menampilkan *value* atau belum.

3.2.5. Write Value Indikator Lampu Merah

Program ini digunakan untuk mengirim *value* yang sudah diprogram dalam Visual studio untuk menggerakkan koil lampu merah yang ada pada CX Programmer. Program lampu merah digunakan sebagai indikator apabila mesin mengalami hal yang tidak diinginkan seperti pembukaan paksa pada pintu lemari dan tidak menutup pintu lemari pada saat sehabis melakukan transaksi. Program ini dapat dilihat pada lampiran A-2 yang menunjukkan *value* 60 merupakan *value* yang dikirim Visual Studio ke PLC sebagai alarm atau peringatan kepada pengguna yang sedang melakukan transaksi apabila melakukan kesalahan. Alarm ini muncul apabila suatu keadaan atau syarat yang keluar muncul dengan menggunakan trigger.

Buzzer digunakan sebagai penanda apabila pengguna melakukan kesalahan atau lupa menutup pintu. Penggunaan *buzzer* sama dengan lampu merah. Pada bagian *design* Visual Studio apabila pengguna menekan tombol menu utama namun tidak menutup pintu maka alarm ini akan berbunyi karena *limit switch* yang berada pada pintu tidak dalam keadaan *press* atau tertekan oleh pintu lemari sehingga *value* yang dikirim tidak valid dan syarat tidak terpenuhi. Syarat ini diprogram dengan program *If* yang berada pada program Read. Program ini dibuat untuk mengatasi adanya kesalahan pemakai dan sebagai program *Safety* untuk standart industry. Warna merah

menunjukkan tingkat urgensi tertinggi. Warna ini membutuhkan tindakan penyelesaian secepatnya.

3.2.6. Write Value Indikator Lampu Kuning

Program *Write value* untuk lampu kuning ini digunakan untuk menunjukkan bahwa mesin dalam keadaan *standby* atau bersiap untuk menunggu proses transaksi. Sama seperti lampu merah, lampu kuning juga diprogram melalui Visual Studio dengan mengganti *Value Data Memory* yang ada untuk menggerakkan koil pada PLC. Program *write* ini dapat dilihat pada lampiran A-3 untuk menyalakan lampu kuning, PLC harus diisi dengan *Value 1*. Indikator lampu ini muncul pada saat keadaan mesin pertama kali dihidupkan, pada saat mesin masuk kedalam menu utama dan saat masuk kedalam menu admin.

Program lampu kuning ini memiliki syarat apabila limit dalam keadaan press atau ditekan. Lampu kuning digunakan karena pada standart mesin yang berada di daerah industri menggunakan lampu kuning sebagai tanda bahwa mesin dalam keadaan siap untuk dijalankan atau dalam mode *Standby*. Warna kuning juga berarti memiliki resiko namun masih dalam keadaan aman. Dalam mode program mesin produksi juga menggunakan lampu kuning sebagai indikator apabila mesin sedang dalam perbaikan program karena terjadi error pada saat produksi.

3.2.7. Write Value Indikator Lampu Hijau

Indikator lampu hijau merupakan program yang digunakan untuk menunjukkan mesin sedang bekerja atau sedang dalam kondisi produksi. Lampu hijau ini juga disimbolkan mesin berfungsi sesuai dengan program yang dibuat atau sesuai dengan cara kerjanya.

Lampiran A-4 menjelaskan untuk menjalankan lampu hijau menggunakan *value* yang berisi 2. Setelah lampu hijau muncul maka msgbox muncul untuk memberikan perintah membuka pintu karena limit sudah diberi timer selama 5 detik oleh timer 3 apabila pintu tidak dibuka, HMI akan kembali ke menu utama.

3.2.8. Write Value Door Lock

Program *write value door lock* digunakan untuk membuka door lock pada lemari sebagai komponen keamanan atau *safety*. Program ini sama dengan lampu merah, kuning, dan hijau karena hanya memberikan *value* dari Visual Studio ke PLC. Perbedaannya

pada Door lock ini menggunakan timer sebagai acuan waktu untuk pengguna membuka pintu lemari tidak terlalu lama. Timer yang dimasukan selama 5 detik sehingga sebelum 5 detik pengguna sudah harus membuka pintu lemari. Pemberian timer ini juga diperuntukkan untuk *door lock* supaya tidak dalam keadaan *retract* saat dialiri listrik karena pada keadaan ini *door lock* akan panas dan dapat merusak door lock itu sendiri.

Value yang di kirm dari Visual Studio 2 untuk menghidupkan *Door lock*. *Door lock* akan berada pada keadaan *retract* saat pengguna login menggunakan kartu RFID dan masuk ke dalam *form user* Visual Studio. Kerja *Door lock* bersamaan dengan lampu hijau. Perbedaannya lampu hijau akan terus menyala sampai saat *form user* berganti ke *form login*, sedangkan *door lock* akan menyala hanya selama 5 detik setelah itu mati.

3.2.9. Write Asynchronous

Program *write asynchronous* adalah program yang merupakan pembacaan yang sangat penting untuk antrian program, dengan kata lain *write asynchronous* memungkinkan program bekerja secara berurutan. Proses *asynchronous* akan mengakses semua untaian UI (*user interface*) karena aktivitas seperti UI biasanya bekerja bersamaan. Jika proses apapun diblokir dalam aplikasi sinkron, maka semua diblokir. Aplikasi yang dibuat akan berhenti merespon atau *floating* di belakang aplikasi utama yang sedang berjalan. Misalnya dalam pembahasan TA ini pada form login yang menerima ID dari kartu RFID yang di *scan*, kemudian masuk ke dalam form user untuk pemanggilan data dengan ID yang dimasukan tersebut akan mengambil laporan biodata, bagian kerja, foto dan kemudian pembaruan database sehingga form tersebut terbuka secara 100%.

Write Async yang dipakai dapat dilihat dalam kurung urutan proses pengiriman data mulai dari sPLCName yang merupakan nama PLC, Point yang merupakan *Data Memory* PLC, dan *Write Value* yang merupakan *value* yang akan dikirimkan ke PLC. Visual Studio akan mengirim data nama PLC apabila sudah cocok akan di lanjutkan menuju *Data memory* yang sudah di buat dalam file bereksistensi *.cdm*, kemudian *value* yang akan dikirim dari Visual Studio ke PLC sehingga proses dilakukan secara berurutan. Fungsi dalam Visual Studio juga terdapat fungsi *Write* saja, fungsi ini melakukan *write* secara langsung sehingga dapat menyebabkan PLC tidak terbaca dan menjadi *error*. *Write* yang dilakukan tidak secara berurutan dibandingkan dengan *write async*. Penggunaan *write asynchronous*

ini terdapat pada setiap program *write value*. *Write async* memanfaatkan dukungan dari Net Framework 4.5 atau versi yang lebih tinggi. Net framework ini berisi class library dan CLR (*Common Language Time*). CLR sendiri menyediakan penampilan dari *application Virtual machine* pada platform.

3.3. Flowchart Proses Pinjam kembali *Automation Tool's Crib*

Proses dari pinjam dan kembali *Tool's* pada mesin memiliki flow yang sama untuk komunikasi CX – Server Lite sehingga dapat dibuat satu flowchart saja, dapat dilihat pada lampiran A-10. Diawali dengan HMI yang menunjukkan layar menu utama untuk login kemudian user ID terdeteksi karena ada karyawan yang menempelkan kartu ID pada leari sehingga masuk kedalam proses login. Setelah proses login selesai, layar HMI akan berubah dan program dari visual basic.NET mengirimkan *value* yang kemudian diproses oleh CX – Server Lite dan di teruskan ke PLC. Proses *read* dari PLC menyalakan lampu hijau yang menunjukkan mesin sedang dalam proses transaksi, *limit switch* menjadi *relase* dan *solenoid doorlock* terbuka sehingga pintu dapat dibuka. Karyawan dapat mengambil atau mengembalikan *Tool's*, setelah itu pintu ditutup dan sensor *limit switch* menjadi *press* untuk menyatakan bahwa pintu telah ditutup kemudian visual basic.NET membaca *value* dari *limit switch* dan halaman HMI kembali menjadi menu utama sehingga proses selesai.

“Halaman ini sengaja dikosongkan”

BAB IV PENGUJIAN ALAT DAN ANALISA DATA

Pengujian dan analisa ini dilakukan untuk mengetahui apakah mesin bekerja sesuai dengan flowchart atau tidak.

4.1. Pengujian OPC

Pengujian ini dilakukan dengan membuat program sederhana read value untuk mengambil nilai Data Memory yang ada dengan menggunakan Visual Studio. Data diambil dengan menggunakan Button dan ditampilkan kedalam Textbox dalam bentuk decimal. Setelah data masuk ke dalam Textbox, dapat dilihat dalam *Software CX-Programmer* apakah Data Memory yang diambil sesuai dengan yang ada pada CX-Programmer.

Kemudian membuat program write value untuk mengubah nilai dari data memory dengan memasukan value yang diinginkan kedalam textbox dan mengirim dengan perintah write asynchronous dalam program button. Setelah menekan button, dapat dilihat pada program CX-Programmer apakah Value dari Data memory yang dituju berubah atau tidak.

Pada Tabel 4.1, value yang diberikan untuk sensor *Buzzer* dan lampu merah sama karena lampu merah tidak memiliki *Buzzer* sendiri sehingga memerlukan tanda bahwa mesin dalam keadaan emergency

Table 4.1 Tabel I/O

No	DM	<i>Input (value)</i>	<i>Output (Sensor)</i>
1	D12	1	Lampu kuning
2	D12	60	Lampu merah
3	D10	30	Lampu hijau
4	D12	2	Solenoid <i>Door Lock</i>
5	D12	60	<i>Buzzer</i>
6	D12	21	Neon

4.1.1. Pengujian PLC dengan Program Visual Studio

Program Read dan Write dalam Visual Studio akan diuji dengan satu cycle transaksi peminjaman atau kembali dan masuk kedalam form admin. Pengujian dilakukan selama 10 kali cycle pinjam, kembali dan masuk kedalam form admin karena didalam form tersebut terdapat program *write asynchronous* yang berisi alamat data memory dan value yang dikirim dari program visual basic.NET yang kemudian diproses oleh CX - Server Lite sehingga PLC dapat menyalakan sensor pada lemari. Dalam proses CX – Server Lite, value akan dikirim bersamaan dengan data memory sesuai dengan program yang telah dibuat. Setelah itu PLC akan mencocokkan data memory yang dikirim oleh CX – Server Lite dan value sesuai dengan sensor yang akan dipakai.

4.1.2. Pengujian PLC Pada Form Login

Tabel 4.2 Tabel Pengujian pada *Form Login*

No	Sensor (Value)				
	Lampu Kuning (1)	Lampu Hijau (30)	Lampu Merah (60)	Solenoid Door Lock (14)	Limit Switch (31)
1	Menyala	Mati	Mati	Release	Press
2	Menyala	Mati	Mati	Release	Press
3	Menyala	Mati	Mati	Release	Press
4	Menyala	Mati	Mati	Release	Press
5	Menyala	Mati	Mati	Release	Press
6	Menyala	Mati	Mati	Release	Press
7	Menyala	Mati	Mati	Release	Press
8	Menyala	Mati	Mati	Release	Press

No	Lampu Kuning (1)	Lampu Hijau (30)	Lampu Merah (60)	Solenoid Door Lock (14)	Limit Switch (31)
9	Menyala	Mati	Mati	Release	Press
10	Menyala	Mati	Mati	Release	Press

Pada Tabel 4.2 merupakan 10 kali percobaan pada Form Login untuk mengirim *value* ke PLC dan terlihat saat masuk ke dalam Form Login, sensor lampu kuning menyala dan *Limit Switch* dalam keadaan *Press*. Keadaan *Door Lock* juga *relase* karena masih dalam form login sehingga pintu masih menutup supaya tidak terjadi transaksi saat pengguna belum login dengan kartu RFID miliknya.

4.1.3. Pengujian PLC pada Form User (Tanpa Transaksi)

Tabel 4.3 Tabel Pengujian pada *Form User* (Tanpa transaksi)

No	Sensor (Value)			
	Lampu Hijau (30)	Lampu Kuning (1)	Door Lock (2)	Limit Switch (31)
1	Menyala	Mati	<i>Retrac</i>	<i>Press</i>
2	Menyala	Mati	<i>Retrac</i>	<i>Press</i>
3	Menyala	Mati	<i>Retrac</i>	<i>Press</i>
4	Menyala	Mati	<i>Retrac</i>	<i>Press</i>
5	Menyala	Mati	<i>Retrac</i>	<i>Press</i>
6	Menyala	Mati	<i>Retrac</i>	<i>Press</i>
7	Menyala	Mati	<i>Retrac</i>	<i>Press</i>
8	Menyala	Mati	<i>Retrac</i>	<i>Press</i>

No	Lampu Hijau (30)	Lampu Kuning (1)	Door Lock (2)	Limit Switch (31)
9	Menyala	Mati	<i>Retrac</i>	<i>Press</i>
10	Menyala	Mati	<i>Retrac</i>	<i>Press</i>

Pada Tabel 4.3 dilakukan percobaan 10 kali masuk kedalam Form User untuk mencoba mengirim *value* dari Visual Studio menuju PLC. Pada table diatas dalaam keadaan pintu masih tertutup dan belum melakukan transaksi sehingga *limit switch* masih dalam keadaan *press* atau menutup dan *solenoid door lock* dalam keadaan *retrac* supaya pintu dapat dibuka tetapi hanya selama 10 detik agar *solenoid* tidak mengalami *overheat*.

4.1.4. Pengujian PLC pada Form User (Dengan Transaksi)

Tabel 4.4 Tabel Pengujian PLC pada *Form User* (Dengan transaksi)

No	Sensor (Value)			
	Lampu Hijau (30)	Lampu Kuning (1)	Door Lock (14)	Limit Switch (35)
1	Menyala	Mati	<i>Relase</i>	<i>Relase</i>
2	Menyala	Mati	<i>Relase</i>	<i>Relase</i>
3	Menyala	Mati	<i>Relase</i>	<i>Relase</i>
4	Menyala	Mati	<i>Relase</i>	<i>Relase</i>
5	Menyala	Mati	<i>Relase</i>	<i>Relase</i>
6	Menyala	Mati	<i>Relase</i>	<i>Relase</i>
7	Menyala	Mati	<i>Relase</i>	<i>Relase</i>

No	Lampu Hijau (30)	Lampu Kuning (1)	Door Lock (14)	Limit Switch (35)
8	Menyala	Mati	<i>Relase</i>	<i>Relase</i>
9	Menyala	Mati	<i>Relase</i>	<i>Relase</i>
10	Menyala	Mati	<i>Relase</i>	<i>Relase</i>

Table 4.4 menjelaskan percobaan dilakukan sebanyak 10 kali untuk proses peinjaman dan kembali. Perbedaannya dengan table 4.3 yaitu pada *solenoid door lock* dan *limit switch*, table 4.3 menunjukkan bahwa *solenoid* dalam keadaan *retract* atau menutup dengan value 2 dan *limit switch* yang masih *press* dengan value 31. Tabel 4.4 *door lock* sudah dalam keadaan tertutup karena pintu telah dibuka dan *Limit switch* dalam keadaan *Relase*. Saat *limit switch relase*, RFID akan mati karena ada syarat dalam program apabila value yang terbaca 35 maka RFID akan berhenti *scanning*, hal ini dibuat agar saat peminjaman atau kembali *listview* yang telah dibuat dapat membandingkan jumlah benda dalam lemari sebelum pintu dibuka dan setelah pintu ditutup. Setelah *limit switch* tertekan karena pintu tertutup maka lampu kuning akan menyala dan kembali seperti pada Tabel 4.1

4.1.5. Percobaan PLC Saat kunci diputar

Tabel 4.5 Tabel Percobaan PLC saat kunci diputar

No	Sensor (Value)			
	Lampu Hijau (30)	Lampu Kuning (1)	Door Lock (14)	Key Switch (35)
1	Menyala	Berkedip	<i>Relase</i>	Diputar
2	Menyala	Berkedip	<i>Relase</i>	Diputar
3	Menyala	Berkedip	<i>Relase</i>	Diputar

No	Lampu Hijau (30)	Lampu Kuning (1)	Door Lock (14)	Key Switch ()
4	Menyala	Berkedip	<i>Relase</i>	Diputar
5	Menyala	Berkedip	<i>Relase</i>	Diputar
6	Menyala	Berkedip	<i>Relase</i>	Diputar
7	Menyala	Berkedip	<i>Relase</i>	Diputar
8	Menyala	Berkedip	<i>Relase</i>	Diputar
9	Menyala	Berkedip	<i>Relase</i>	Diputar
10	Menyala	Berkedip	<i>Relase</i>	Diputar

Pada Table 4.5 di jelaskan bahwa saat kunci diputar, terlihat pada kolom lampu kuning berkedip karena *Key Switch* men *trigger* lampu kuning untuk berkedip. pada program CX – Programmer di setting P_0_1 setiap satu detik kontak akan *Close* dan *Open* lagi selama satu detik terus menerus. Dua indicator ini bisa menyala secara bersamaan karena pada saat *Key Switch* diputar, Visual Studio mengirim dua *value* secara bersamaan.

4.1.6. Percobaan kecepatan pinjam dan kembali antara *Tool's Crib Konvensional* dengan *Automation Tool's Crib*.

Tabel 4.6 Tabel Perbandingan *Tool's Crib* dengan *Automation Tool's Crib* (1)

No	Waktu Pinjam (detik)	Waktu Kembali (detik)
1	61	51
2	90	50
3	95	55

No	Waktu Pinjam (detik)	Waktu Kembali (detik)
4	121	51
5	60	60
6	61	65
7	62	52
8	124	86
9	70	50
10	103	53

Pada tabel 4.6, melakukan percobaan pengambilan data waktu pada *Tool's Crib* konvensional dan didapat data saat waktu pinjam waktu yang didapat kurang lebih 84 detik karena operator harus mencari alat yang ingin dipinjam oleh peminjam. Untuk pengambilan data saat waktu kembali kurang lebih 52 detik karena operator tinggal menaruh alat sesuai rak alat yang dipinjam.

Tabel 4.7 Tabel Perbandingan *Tool's Crib* dengan *Automation Tool's Crib* (2)

No	Waktu Pinjam (detik)	Waktu Kembali (detik)
1	30	19
2	23	20
3	22	20
4	34	17
5	31	22
6	31	23
7	36	16

No	Waktu Pinjam (detik)	Waktu Kembali (detik)
8	31	18
9	27	21
10	31	19

Tabel 4.7 menunjukkan percobaan pengambilan data waktu pada mesin *automation Tool's Crib*. Waktu pinjam yang didapat yaitu kurang lebih 29 detik, karena pengguna harus memilih barang yang ingin dipinjam. Pengambilan data waktu kembali didapatkan kurang lebih 19 detik, peminjam hanya menaruh alat yang telah dipinjam dan menutup pintu lemari.

4.2. Manual Book

Manual book merupakan buku panduan yang digunakan untuk memandu pelaksanaan *troubleshooting* maupun *maintenance* yang mengacu pada standar pabrik. Dengan adanya manual book, dalam pelaksanaan *troubleshooting* dan *maintenance* tidak diperlukan lagi proses coba – coba sehingga dalam pelaksanaannya sudah dapat dilihat pada manual book.

Pada mesin *automation Tool's Crib* dibuat manual book agar saat mesin dalam keadaan eror atau tidak dapat beroperasi, admin dapat melihat manual book dan memperbaiki mesin tersebut. Manual book dapat dilihat dalam lampiran A-11 dimulai dari Bab 1 yaitu mengenai keselamatan kerja sebelum mengoperasikan mesin dan peringatan umum tentang mesin tersebut. Kemudian tindakan *safety* yang merupakan tindakan untuk keselamatan saat mesin dijalankan seperti memastikan tangan atau anggota badan dalam keadaan kering dan tidak basah. *Wiring* label juga terdapat pada manual book ini untuk mengetahui potensi apa yang dapat terjadi dan peringatan agar mesin dapat beroperasi dengan aman. Pada Bab 2 manual book dijelaskan mengenai spesifikasi dari mesin tersebut. Untuk konsumsi daya dari mesin tersebut dapat dilihat dalam

Table 4.8 Konsumsi Daya Mesin *Automation Tool's Crib*

KEBUTUHAN DAYA	
Tegangan	220v
Arus	1A
Konsumsi daya	240W

Penggunaan komponen juga menjadi bagian penting dalam pembuatan mesin ini. Komponen yang digunakan sesuai dengan mesin ini karena spesifikasinya, untuk datasheetnya dapat dilihat pada A-11 Bab 2. Komponen yang digunakan pada mesin ini sebagai berikut :

Table 4.9 Komponen yang digunakan

NO	Nama Barang / Deskripsi Service
1	DR22D0L-E3G_FUJI_PILOT LAMP GREEN
2	DR22D0L-E3R_FUJI_PILOT LAMP RED
3	DR22D0L-E3Y_FUJI_PILOT LAMP YELLOW
4	MORGAN BOOKCASE_INFORMA_WHITE BOOKCASE(216.2X43.4X186)
5	UTICON_POWER Outlet Isi 6
6	Kabel Serial DB9 male-female 10M
7	UC-232_ATEN_USB to Serial Converter
8	Kabel LAN UTP RJ45 10 M
9	BOX MCB 2 POLE(BW32AAG)

NO	Nama Barang / Deskripsi Service
10	Legrand rubber 2P+E POWER PLUG
11	DURA 600_OMNI-ID RFID TEXT
12	Kartu RFID Card UHF ISO18000-6C
13	TL5 T5_PHILIPS_LAMPU LED 4W
14	Kunci Pintu Solenoid Door Lock 12 V Automatic Electronic Arduino
15	KR18268 Long Lever Limit Switch 15 A 250VAC//#24
16	RFID UHF Windshield Sticker Passive Tag EPC Gen2/ISO18000-6C
17	UHF RFID Passive on-metal Tag Screw holes 915MHZ
18	Kartu RFID Card UHF ISO18000-6C
19	UHF Reader & Writer (Middle Range)-CT-I809
20	Selector Key Switch 3 Posisi Stay Put
21	Power Supply 12V 3A Jaring CCTV LED AC 220V to DC 12V 3A Adaptor PSU
22	UHF Windshield 100x45mm with Anti Metal Shield for parking
23	CJ1M – CPU 13
24	CJ1W – ID211
25	CJ1W – OC211
26	CJ1W – PA202
27	CJ1W – ETN 13
28	CARDTEK CT-I809
29	Mifare S50 EPC G2 Card
30	Windshield Sticker Passive tag EPC gen2

Mesin ini memiliki fitur utama yang menjadikan mesin ini lebih unggul dari pada *Tool's Crib* konvensional. Mesin yang sudah dibuat ini memiliki fitur sebagai berikut :

Table 4.10 Fitur Mesin *Automation Tool's Crib*

No	FITUR
1	Transaksi otomatis tanpa operator
2	Pembukuan otomatis dengan database
3	Identifikasi Alat dan pengguna Otomatis dengan RFID
4	Membutuhkan transaksi < 2 menit
5	Maintenance data dan peralatan dapat diawasi
6	Waktu operasi 24 jam

Pembuatan mesin ini menggunakan lemari yang terbuat dari logam dan terdiri dari 3 rak. Dan memiliki kapasitas tersendiri untuk menyimpan alat yang sesuai dengan bentuknya. Untuk kapasitas dari lemari dapat dilihat pada Tabel 4.11 ;

Table 4.11 Kapasitas RFID dan Lemari *Automation Tool's Crib*

KAPASITAS		
lebar rak	47 cm	
panjang rak	87.5 cm	
jumlah rak	3	
kapasitas RFID	255 tools	
Kapasitas Lemari	20	Large size tools
	40	medium Size tools

Lemari yang digunakan memiliki dimensi ukuran tersendiri dan layout tersendiri. Layout ini digunakan untuk menentukan luas area dimana mesin akan diletakan sehingga tidak berhimpitan dengan mesin lainnya dan area mesin tersebut cukup untuk operasi. Berikut dimensi dan layoutnya ;

Table 4.12 Dimensi dan Layout mesin *Automation Tool's Crib*

DIMENSI DAN LAYOUT	
Lemari	
Tinggi	1,4 m
Panjang	0,5
Lebar	0,5
Layout	
Tinggi	2 m
panjang	1,5 m
lebar	1 m

Untuk menjalankan mesin diperlukan *software* pendukung untuk pengoperasiannya. *Software* ini dibagi menjadi *software* HMI, Database, Webserver, Program PLC Omron, dan OPC Omron. Spesifikasi minimum dan rekomendasi dari PC sebagai master juga harus di perhitungkan karena besarnya file dan spesifikasi dari *software* yang digunakan. Berikut Tabel menunjukan spesidikasi *software* dan *hardware* yang dianjurkan untuk mesin ini ;

Table 4.13 *Software* yang digunakan

PC
Software requirement
Visual Studio 2015

PC
CX Serverlite
CX Programmer
XAMPP (webserver)
Microsoft SQL Server
MS.Access 2016

Table 4.14 Spesifikasi *Hardware* yang digunakan

HARDWARE MINIMUM	
Processor	Intel Core i3 and equivalent
RAM	2 GB
HDD	250 GB
VGA	Intel GMA HD Integrated Graphics
Directx	directx9
OS	Windows 7
HARDWARE YANG DISARANKAN	
Processor	Intel Core i5 and equivalent
RAM	4 GB
HDD	500 GB
VGA	NVIDIA GeForce 940M dengan 2GB DDR3 VRAM
Directx	Directx11
OS	Windows 10

Bab 3 dalam manual book lampiran A-11 menjelaskan tata cara merawat dan memperbaiki mesin apabila terjadi masalah dalam pengoperasiannya atau juga untuk *maintenance* rutin. Di bab ini juga menjelaskan cara pengoperasiannya secara manual dan otomatis. Pengoperasian manual dilakukan dengan cara memasukkan nama dan NIP karyawan kedalam *form* login manual yang sudah dibuat dalam *form* Admin. Untuk pengoperasian secara otomatis dapat menggunakan ID card karyawan yang merupakan tag RFID dan menempelkan pada kaca lemari kemudian mengikuti proses yang sudah dijelaskan sebelumnya.

Bab 4 manual book merupakan bab terakhir yang membahas mengenai cara – cara *troubleshooting* dari mesin apabila terjadi eror atau sensor dan indikator yang tidak terbaca. *Trouble shooting* bisa dari *form maintenance* yang berada di *form* admin atau juga melakukan pengecekan langsung *wiring* sensor dan indikator dari mesin.

BAB V

PENUTUP

Dengan adanya Tugas Akhir ini dapat diambil kesimpulan dan saran bahwa PLC sudah dapat berkomunikasi dengan PC menggunakan OPC CX - Server Lite sehingga HMI dapat memonitoring sensor dan perusahaan tidak membutuhkan operator dalam melakukan proses peminjaman atau pengembalian, operator hanya melakukan pembukuan yang dilakukan secara otomatis. Dalam prosesnya, *Automation Tool's Crib* membutuhkan waktu yang lebih cepat dibandingkan dengan *Tool's Crib* konvensional yaitu kurang lebih 90 detik. Untuk saran, penempatan dari *Automation Tool's Crib* ini ditaruh pada suatu ruangan dan RFID diletakan di pintu masuk dan keluar sehingga proses *scanning* dilakukan saat keluar dan masuk ruangan tersebut. Penempatan ini juga bertujuan untuk menempatkan lemari – lemari lainnya yang akan dibuat agar semua *tool's* yang ada pada *tool's Crib* dapat di muat pada lemari.

“Halaman ini sengaja dikosongkan”

DAFTAR PUSTAKA

- [1] P. A. Farrington and J. J. Swain, "DESIGN OF SIMULATION EXPEFUMENTS WITH MANUFACTURING APPLICATIONS," University of Alabama in Huntsville, Huntsville, 1993.
- [2] W. Avilarosa, "Programmable Logic Controller (PLC)," Teknik Elektro Polines, Semarang, 2014.
- [3] J. Enterprise, Pengenalan Visual Studio 2013, Jakarta: Elexmedia Komputindo, 2015.
- [4] A. Kurniadi, Pemrograman Visual Basic, Jakarta: Elex Media Komputindo, 2000.
- [5] J. M. Hellerstein, M. Stonebraker and J. Hamilton, "Architecture of a Database System," University of California ; Massachusetts Institute of Technology , California, 2007.
- [6] E. Tungadi and Irmawati, "Generalisasi Administrasi Basis Data MySql Berbasis Web," Center for Applied ICT Research ; Politeknik Negeri Ujung Pandang, Makasar, 2016.
- [7] M. Hoffmann, C. Büscher, T. Meisen and S. Jeschke, "Continuous integration of field level production data into top-level," Insitute of Information Management in Mechanical Engineering (IMA) of RWTH Aachen University, Aachen, Germany, 2015.
- [8] S.-H. Leitner and W. Mahnke, "OPC UA – Service-oriented Architecture for Industrial Applications," ABB Corporate Research Center, Wallstadter Str. Ladenburg, Germany, 2006.
- [9] D. M. Dobkin, "The RF in RFID Passive UHF RFID in Practice," Elsevier, Burlington, 2008.
- [10] R. Hidayat and S. M. Sumardi, "PERANCANGAN HUMAN MACHINE INTERFACE (HMI) PADA HITCUT MACHINE DENGAN PLC OMRON SYSMAC CP1L," Universitas Diponegoro, Semarang, Indonesia, 2013.
- [11] D. S.Kom, "Jenis Tipe Kabel Ethernet dan Fungsinya," DosenIT.com, 2015.

“Halaman ini sengaja dikosongkan”

LAMPIRAN

A-1. Listing Program

```
Imports MySql.Data.MySqlClient
Imports System.Data.Sql
Imports ReaderB
Imports System.Text
Imports System.Windows.Forms.Keys
Imports System.IO.Ports
Imports System.Reflection
Imports System
Imports System.Collections.Generic
Imports System.ComponentModel
Imports System.Data
Imports System.Drawing
Imports System.Windows.Forms
Imports System.Collections
Imports System.Resources
Imports System.IO
Imports System.Data.OleDb
Imports System.Net
Imports System.Net.Sockets
Imports System.Threading
Imports System.Linq
Imports CXSLite.CXSLiteCtrl
Public Class user
    Public text3 As String
    Private fAppClosed As Boolean
    Private fComAdr As Byte = &HFF
    Private ferrorcode As Integer
    Private fBaud As Byte
    Private fdminfre As Double
    Private fdmaxfre As Double
    Private Maskadr As Byte
    Private MaskLen As Byte
    Private MaskFlag As Byte
    Private fCmdRet As Integer = 30
    Private fOpenComIndex As Integer
    Private fIsInventoryScan As Boolean
    Private fisinventoryscan_6B As Boolean
    Private fOperEPC(36) As Byte
    Private fPassWord(4) As Byte
    Private fOperID_6B(8) As Byte
```

```

Private CardNum1 As Integer = 0
Private list As ArrayList
Private fTimer_6B_ReadWrite As Boolean
Private flInventory_EPC_List As String
Private frmcomportindex As Integer
Private ComOpen As Boolean = False
Private breakflag As Boolean = False
Private x_z As Double
Private y_f As Double
Public IP As String = ""
Public username As String = ""
Public dsname As String = ""
Public mac As String = ""
Public portnum As String = ""
Public tup As String = ""
Public rm As String = ""
Public cm As String = ""
Public ct As String = ""
Public fc As String = ""
Public dt As String = ""
Public br As String = ""
Public pr As String = ""
Public bb As String = ""
Public rc As String = ""
Public ml As String = ""
Public md As String = ""
Public di As String = ""
Public dp As String = ""
Public gi As String = ""
Public nm As String = ""
Public serial As String
Public combo As String
Private siapref As Boolean
Public switch As Boolean = False
Public refreshget As Boolean = False
Public clrflow As Boolean = False
Public compare As Boolean = False
Public rfidclosed As Boolean = False
Public itung As Integer

Dim set1 As Boolean = True
Dim set2 As Boolean = False
Dim sPLCName As String = "NewPLC1"

```

Dim sPointName As String
Dim sDeviceName As String
Dim sPoint As String
Dim bBadQuality As Boolean
Dim sMode As String
Dim iMode As Integer
Dim iSetMode As Integer
Dim bReadOk As Boolean
Dim bWrittenOk As Boolean
Dim iEventCounter As Integer
Dim iSequenceCompleteCounter As Integer
Dim sClockWrite As String
Dim dTime As Date
Dim dTimeNow As Date = Date.Now
Dim sType As String
Dim bIsPoint As Boolean
Dim bIsDevice As Boolean
Dim DeviceNames()
Dim PointNames()
Dim sError As String
Dim bResultFlag As Boolean

Dim Value

Dim ReadValue As Object
Dim WriteValue As Object
Dim ReadArray(1200) As Short
Dim WriteArray(1200) As Short
Dim WriteAsynchValue As Object
Dim WriteAsynchArray(1200) As Short

Dim ReadSeqValue As String
Dim WriteSeqValue As String
Dim ReadSeqArray(1020) As Short
Dim WriteSeqArray(1020) As Short
Dim WriteSeqAsynchValue As String
Public WriteSeqAsynchArray(400) As Short

Dim CompareVal
Dim ReadVal
Dim limitmasihpress As Boolean
Dim limitmasihpress1 As Boolean
Dim limitmasihrelease1 As Boolean

```

Public lockalreadyopen As Boolean
Dim koneksi As Boolean
Public habisbuka As Boolean
Dim dr As OleDbDataReader
Dim tables As DataTableCollection
Dim connString As String
Dim Dap As OleDbDataAdapter
Dim conn2 As OleDbConnection = New OleDbConnection
Dim provider As String
Dim dataFile As String
Dim connStringuser As String

Public Sub subopenDevice() ' UNTUK KONEKSI PLC
    bResultFlag = CxsLiteCtrl1.OpenDevice(sPLCName) ' UNTUK
    MEMBUKA NEWPLC1 PADA CX SERVERLITE1,
    CXSERVERLITE1 BERADA DI MASING2 FORM
    If bResultFlag = True Then ' JIKA BRESULTFLAG SUDAH
    DITEMUKAN, MAKA
        koneksi = True ' KONEKSI BERHASIL
    Else ' JIKA TIDAK
        koneksi = False ' KONEKSI GAGAL
    End If
End Sub

Private Sub autoplcuser() ' UNTUK MENYALAKAN LAMPU
    HIJAU DAN MEMBUKA DOOR LOCK
    Dim poin1 As String = "D12X" ' UNTUK MEMANGGIL DM 12
    PADA CX SERVERLITE
    WriteValue = 2 ' VISUAL STUDIO MENYURUH DM12
    UNTUK DIISI ANGKA 2
    bWrittenOk = CxsLiteCtrl1.WriteAsync(sPLCName, poin1,
    WriteValue) ' BWRITTENOK SAAT CXSERVERLITE1
    MENULISKAN (NEWPLC1, D12X, 2) PADA PLC SEHINGGA
    DAPAT MENAKTIFKAN DM DAN MENJALANKAN
    AKTUATOR
    MsgBox("silahkan buka pintu") ' MENAMPILKAN MSG BOX
End Sub

Private Sub menuutama() ' UNTUK MENYALAKAN LAMPU
    KUNING
    Dim poin1 As String = "D12X" ' UNTUK MEMANGGIL DM 12
    PADA CX SERVERLITE

```

```
WriteValue = 1 ' VISUAL STUDIO MENYURUH DM12
UNTUK DIISI ANGKA 1
bWrittenOk = CxsLiteCtrl1.WriteAsync(sPLCName, poin1,
WriteValue) ' BWRITTENOK SAAT CXSERVERLITE1
MENULISKAN (NEWPLC1, D12X, 1) PADA PLC SEHINGGA
DAPAT MENGAKTIFKAN DM DAN MENJALANKAN
AKTUATOR
```

End Sub

```
Private Sub user_Load(sender As Object, e As EventArgs) Handles
 MyBase.Load ' SAAT HALAMAN USER LOAD TERBUKA
loginuser.bolehmaling = False ' BOLEH MALING MERUPAKAN
KONDISI PINTU SAAT DI LOGIN USER UNTUK ALARM. SAAT
DIUSER ALARM TERSEBUT DINONAKTIFKAN
Call subopenDevice() ' mengaktifkan koneksi plc
tid3.Text = adminloginmanual.text2 'TEXTBOX BANTUAN
UNTUK MENYALIN NIP KARYAWAN DARI HALAMAN
ADMINLOGINMANUAL UNTUK MENAMPILKAN DATA DIRI
KARYAWAN
Timer4.Enabled = True 'UNTUK MENGAKTIFKAN TIMER
YANG DIGUNAKAN UNTUK MEREFRESH KEADAAN LIMIT
TextBox15.Text = "read" ' UNTUK MENULIS KEADAAN
LIMIT
TextBox16.Text = "read" ' UNTUK MENYALIN KEADAAN
LIMIT YANG PERTAMA SAAT PINTU TERTUTUP
TextBox17.Text = "read" ' UNTUK MENYALIN KEADAAN
LIMIT YANG KEDUA SAAT PINTU TERBUKA
TextBox18.Text = "read" ' UNTUK MENYALIN KEADAAN
LIMIT YANG KETIGA SAAT PNTU TERTUTUP LAGI
autoclose.BackColor = Color.Red '
tid2.Text = loginuser.text1 ' TEXTBOX BANTUAN UNTUK
MENYALIN NIP KARYAWAN DARI HALAMAN LOGINUSER
UNTUK MENAMPILKAN DATA DIRI KARYAWAN
Timer7.Enabled = False ' TIMER UNTUK KEMBALI KE
LOGINUSER SECARA OTOMATIS
Call isiGrid() 'UNTUK MEMANGGIL FUNGSI MENGISI
DATAGRIDVIEW
Call persiapanrfid() 'UNTUK MENGAKTIFKAN FUNGSI RFID
```

End Sub

```

Private Sub Button33_Click(sender As Object, e As EventArgs)
Handles Button33.Click ' BUTTON UNTUK KEMBALU KE
HALAMN LOGIN USER/ HALAMN UTAMA
    loginuser.bolehmalng = True 'MENGAKTIFKAN KONDISI
UNTUK ALARM LIMIT AGAR ALARM BUNYI SAAT PINTU
TIDAK DI TUTUP DENGAN BENAR
    button2.PerformClick() ' UNTUK AUTOCLICK BUTTON2
UNTUK MULAI SCANNING RFID
    Call tutuprfid() ' MEMANGGIL FUNGSI UNTUK MEMUTUS
COM ATAU KONEKSI PADA rfid
    If StatusBar1.Panels(1).Text = "COM Closed" Then
        loginuser.Show() ' LOGINUSER TAMPIL
        loginuser.TextBox13.BackColor = Color.Green
        Call menuutama() ' UNTUK MENGAKTIFKAN LAMPU
WARNA KUNING PADA PLC
        Me.Close() ' UNTUK MENUTUP HALAMAN USER
    End If

```

```

End Sub

```

```

Dim limit1 = TextBox15.Text
relaypress()
relayrelease()
ReadValue = Nothing
bReadOk = CxsLiteCtrl1.Read(sPLCName, sPointName &
"D10X", ReadValue, bBadQuality)
TextBox15.Text = ReadValue
If limit1 = "31" Then
    limitmasihpress1 = True
End If

```

```

Dim limit6 = TextBox15.Text
ReadValue = Nothing
bReadOk = CxsLiteCtrl1.Read(sPLCName, sPointName &
"D10X", ReadValue, bBadQuality)
TextBox15.Text = ReadValue
If limit6 = "35" Then
    limitmasihrelease1 = True
End If
'sementara
End Sub

```

```

Private Sub relaypress()

    Dim limit1 = ReadValue

    ReadValue = Nothing
    bReadOk = CxsLiteCtrl1.Read(sPLCName, sPointName &
"D10X", ReadValue, bBadQuality)
    TextBox15.Text = ReadValue
    If TextBox15.Text = "31" Then
        TextBox16.Text = TextBox15.Text
    End If
    If TextBox16.Text = "31" And TextBox17.Text = "35" And
TextBox15.Text = "31" Then
        TextBox18.Text = TextBox15.Text
    End If
End Sub

Private Sub relayrelease()

    Dim limit1 = ReadValue
    ReadValue = Nothing
    bReadOk = CxsLiteCtrl1.Read(sPLCName, sPointName &
"D10X", ReadValue, bBadQuality)
    TextBox15.Text = ReadValue
    If TextBox15.Text = "35" Then
        TextBox17.Text = TextBox15.Text

    End If
End Sub

Public Sub adminbutton()
    ADMINToolStripMenuItem.Enabled = False
End Sub

Private Sub accesdb()
    conn1.Close()
    provider =
"Provider=Microsoft.ACE.OLEDB.12.0;Data Source ="
    dataFile =
"C:\Users\rico\Documents\Database_Final.accdb"
    connString = provider & dataFile
    conn1.ConnectionString = connString
End Sub

Private Sub tutuprfid() '

```

```

Dim b2, b3 As Boolean
button2.PerformClick()
b2 = True
If b2 = True Then
    tabControl1.SelectedIndex = 0
    b3 = True
    If b3 = True Then
        ClosePort.PerformClick()
    End If
End If
End Sub
Public Sub persiapanrfid1()
    list = New ArrayList()
    progressBar1.Visible = False
    fOpenComIndex = -1
    fComAdr = 0
    ferrorcode = -1
    fBaud = 5
    InitComList()
    InitReaderList()
    NoAlarm_G2.Checked = True

    Byone_6B.Checked = True
    Different_6B.Checked = True

    P_EPC.Checked = True
    C_EPC.Checked = True
    DestroyCode.Checked = True
    NoProect.Checked = True
    NoProect2.Checked = True
    fAppClosed = False
    fIsInventoryScan = False
    fisinventoryscan_6B = False
    fTimer_6B_ReadWrite = False
    Label1_Alarm.Visible = False
    Timer_Test_.Enabled = False
    Timer_G2_Read.Enabled = False
    Timer_G2_Alarm.Enabled = False
    timer1.Enabled = False

    Button1.Enabled = False
    button20.Enabled = False
    Button5.Enabled = False

```



```
Button1.Enabled = False
button2.Enabled = False
Button_DestroyCard.Enabled = False
Button_WriteEPC_G2.Enabled = False
Button_SetReadProtect_G2.Enabled = False
Button_SetMultiReadProtect_G2.Enabled =
False
Button_RemoveReadProtect_G2.Enabled = False
Button_CheckReadProtected_G2.Enabled = False
Button_SetEASAlarm_G2.Enabled = False
button4.Enabled = False
Button_LockUserBlock_G2.Enabled = False
SpeedButton_Read_G2.Enabled = False
Button_DataWrite.Enabled = False
Button_BlockErase.Enabled = False
Button_BlockWrite.Enabled = False
Button_SetProtectState.Enabled = False
SpeedButton_Query_6B.Enabled = False
SpeedButton_Read_6B.Enabled = False
SpeedButton_Write_6B.Enabled = False
Button14.Enabled = False
Button15.Enabled = False

DestroyCode.Enabled = False
AccessCode.Enabled = False
NoProect.Enabled = False
Proect.Enabled = False
Always.Enabled = False
AlwaysNot.Enabled = False
NoProect2.Enabled = False
Proect2.Enabled = False
Always2.Enabled = False
AlwaysNot2.Enabled = False
P_Reserve.Enabled = False
P_EPC.Enabled = False
P_TID.Enabled = False
P_User.Enabled = False
Same_6B.Enabled = False
Different_6B.Enabled = False
Less_6B.Enabled = False
Greater_6B.Enabled = False

radioButton1.Checked = True
```

```

radioButton4.Checked = True
radioButton5.Checked = True
radioButton7.Checked = True
radioButton10.Checked = True
radioButton14.Checked = True
button6.Enabled = False
button8.Enabled = False
button9.Enabled = False
button10.Enabled = False
button11.Enabled = False
comboBox5.Enabled = False
radioButton5.Enabled = False
radioButton6.Enabled = False
radioButton7.Enabled = False
radioButton8.Enabled = False
radioButton9.Enabled = False
radioButton10.Enabled = False
radioButton11.Enabled = False
radioButton12.Enabled = False
radioButton13.Enabled = False
radioButton14.Enabled = False
radioButton15.Enabled = False
RadioButton16.Enabled = False
RadioButton17.Enabled = False
RadioButton19.Enabled = False
textBox3.Enabled = False
radioButton_band1.Checked = True
RadioButton16.Checked = True
ComboBox_baud2.SelectedIndex = 3
comboBox9.SelectedIndex = 0
comboBox10.SelectedIndex = 0
radioButton22.Checked = True

If Me IsNot Nothing Then
    automatic()
End If
End Sub

```

```

Private Sub tidbantuan_TextChanged(sender As
Object, e As EventArgs) Handles
tidbantuan.TextChanged
    text1 = tidbantuan.Text

```

```

End Sub

Private Sub loginuser_Load(sender As Object, e
As EventArgs) Handles MyBase.Load

Me.UserdataTableAdapter.Fill(Me.Database_FinalDataSe
t.userdata)
    ADMINToolStripMenuItem.Enabled = False
    Call autoplcloginuser() 'plc
    Timer3.Enabled = True
    Timer2.Enabled = True
    TextBox14.Text = "read"
    bantuanemergency.Text = "read"
    Call persiapanrfid1()
    TextBox13.BackColor = Color.Green

    bolehmaling = True
    clearrfid.BackColor = Color.Pink
End Sub

Private Sub
ComboBox1_SelectedIndexChanged(sender As Object, e
As EventArgs)

End Sub

Private Sub tid1_TextChanged(sender As Object, e
As EventArgs)

End Sub

Private Sub autoplcloginuser()
    Dim poin As String = "D12X"
    Dim alreadyopen As Boolean
    user.subopenDevice()
    alreadyopen = True
    If alreadyopen = True Then
        WriteValue = 1
        bwrittenOk =
CxlsLiteCtrl11.WriteAsync(sPLCName, poin, WriteValue)
    End If
End Sub

```

```

    Private Sub TextBox12_TextChanged(sender As
Object, e As EventArgs) Handles
TextBox12.TextChanged
    Dim str As String
    accesdb()
    conn1.Open()

    str = "SELECT * FROM [userdata] WHERE (id=
'" & TextBox12.Text & "'"")
    Using cmd As OleDbCommand = New
OleDbCommand(str, conn1)
        dr = cmd.ExecuteReader
        dr.Read()
        Try
            TextBox13_2.Text =
dr("nipuser").ToString
        Catch ex As Exception

        End Try

    End Using
    TextBox11.Text = TextBox13_2.Text
    conn1.Close()

End Sub
Public Sub automatic()
    Dim b8 As Boolean
    tabControl1.SelectedIndex = 0
    OpenPort.PerformClick()
    If button8.Enabled Then
        comboBox4.SelectedIndex = 1
        ComboBox_scantime.SelectedIndex = 0
        If comboBox4.SelectedIndex = 1 Then
            radioButton8.PerformClick()
            If radioButton8.Checked Then
                comboBox4.SelectedIndex = 0
                If comboBox4.SelectedIndex = 0
Then
                    button8.PerformClick()
                    tabControl1.SelectedIndex = 1
                    b8 = True
                End If
            End If
        End If
    End If
End Sub

```

```

        If b8 = True Then
            Call tangkap()
        End If
    End If
End If

End Sub
Public Sub tangkap()
    button2.PerformClick()
End Sub
Private Sub TextBox6_TextChanged(sender As Object, e As EventArgs) Handles
    TextBox6.TextChanged, TextBox13_2.TextChanged
    TextBox10.Text = TextBox6.Text
    TextBox12.Text = TextBox10.Text

End Sub

Private Sub RefreshStatus()
    If (ComboBox_AlreadyOpenCOM.Items.Count = 0)
Then
        StatusBar1.Panels(1).Text = "COM Closed"
    Else
        StatusBar1.Panels(1).Text = " COM" +
CStr(frmcomportindex)
    End If
    StatusBar1.Panels(0).Text = ""
    StatusBar1.Panels(2).Text = ""
End Sub
Private Sub accessdbuser()
    conn2.Close()
    provider =
"Provider=Microsoft.ACE.OLEDB.12.0;Data Source ="
dataFile =
"C:\Users\rico\Documents\Database_Final.accd
    connStringuser = provider & dataFile
    conn2.ConnectionString = connStringuser

```

```

End Sub
Public Sub subopenDevice()
    bResultFlag =
CxsliteCtrl1.OpenDevice(sPLCName)
    If bResultFlag = True Then
        koneksi = True
    Else
        koneksi = False
    End If
End Sub

Sub isiGrid()
    accessdbuser()
    DS = New DataSet
    tables = DS.Tables
    Dap = New OleDbDataAdapter("Select * FROM
[tabel_pinjam] where nip like '%" & nrporang.Text &
"%'", conn2)
    conn2.Open()
    Dap.Fill(DS, "tabel_pinjam")
    DataGridView1.DataSource =
DS.Tables("tabel_pinjam")
End Sub
Sub hapus()
    tools1.Text = ""
    namatools.Text = ""
    merkalat.Text = ""
    tanggalkembali.Text = ""

    Private Sub autoplcuser()           Dim poin1 As
String = "D12X"
        WriteValue = 2
        bWrittenOk =
CxsliteCtrl1.WriteAsync(sPLCName, poin1, WriteValue)
        MsgBox("silahkan buka pintu")
    End Sub
    Private Sub menuutama()
        Dim poin1 As String = "D12X"
        WriteValue = 1
        bWrittenOk =
CxsliteCtrl1.WriteAsync(sPLCName, poin1, WriteValue)

End Sub

```

```

Private Sub tutuprfid()
    Dim b2, b3 As Boolean

    b2 = True
    If b2 = True Then
        tabControl1.SelectedIndex = 0
        b3 = True
        If b3 = True Then
            ClosePort.PerformClick()
        End If
    End If
End Sub

```

A-2. Program write lampu merah

```

Sub alarmlogin()
    If TextBox11.Text = "" And TextBox14.Text =
"35" And bolehmaling = True Then
        alarmpintu.Show()
        Dim poin10 As String = "D12X"
        WriteValue = 60
        bWrittenOk =
CxsliteCtrl1.WriteAsync(sPLCName, poin10,
WriteValue)
        End If
End Sub

```

A-3. Program write lampu kuning

```

Private Sub menuutama()
    Dim poin1 As String = "D12X"
    WriteValue = 1
    bWrittenOk =
CxsliteCtrl1.WriteAsync(sPLCName, poin1,
WriteValue)
End Sub

```

A-4. Program write lampu hijau

```

Private Sub autoplcuser()
    Dim poin1 As String = "D12X"
    WriteValue = 2
    bWrittenOk =
CxsliteCtrl1.WriteAsync(sPLCName, poin1,
WriteValue)

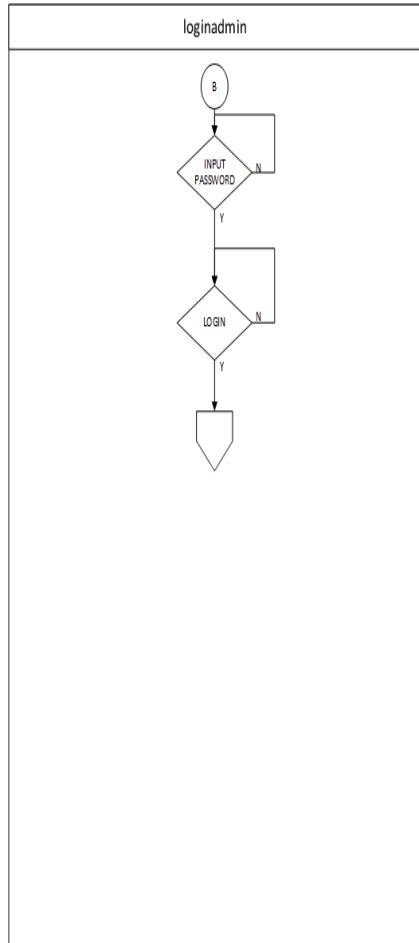
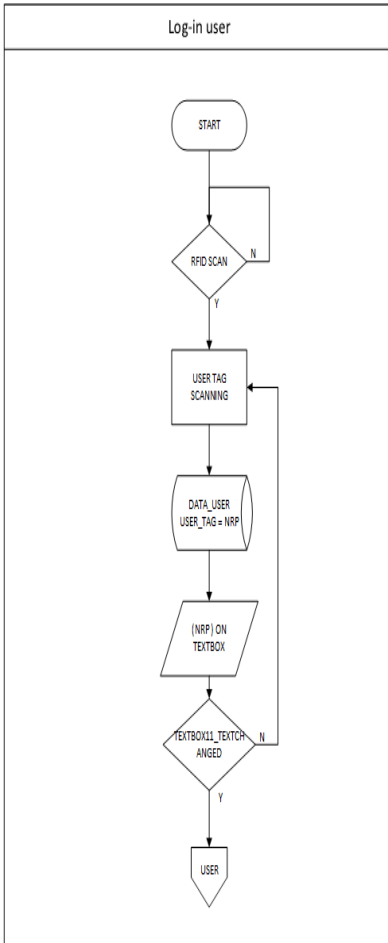
```

```
        MsgBox("silahkan buka pintu")
End Sub
```

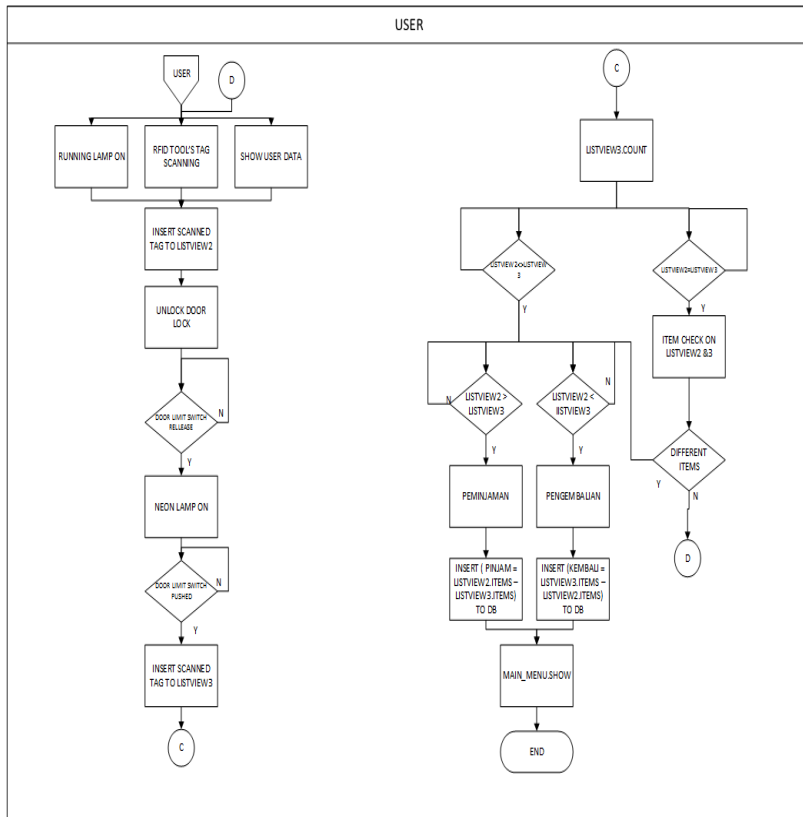
A-5. Program pembaca *Limit Switch*

```
Private Sub Limitpress()  
    Dim poin2 As String = "D12X"  
    ReadValue = Nothing  
    bReadOk = CxsLiteCtrl11.Read(sPLCName, poin2,  
    ReadValue, bBadQuality)  
    textBox1.Text = ReadValue  
    If ReadValue = 7 Then  
        Limitmasihpress = True  
    End If  
End sub
```

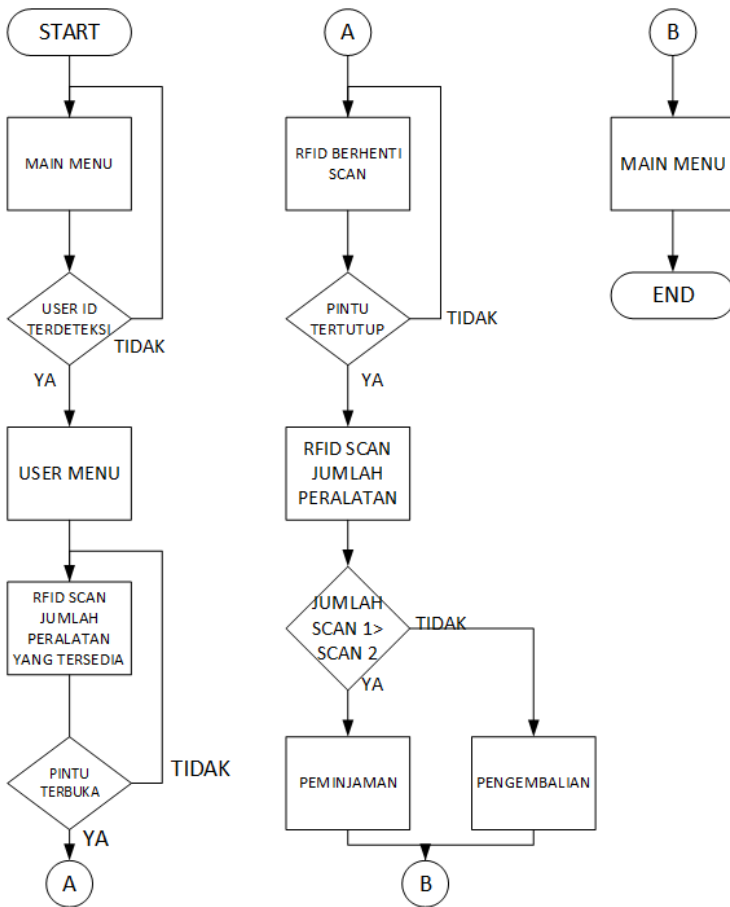

A-6. Flowchart pada form login user dan loginadmin



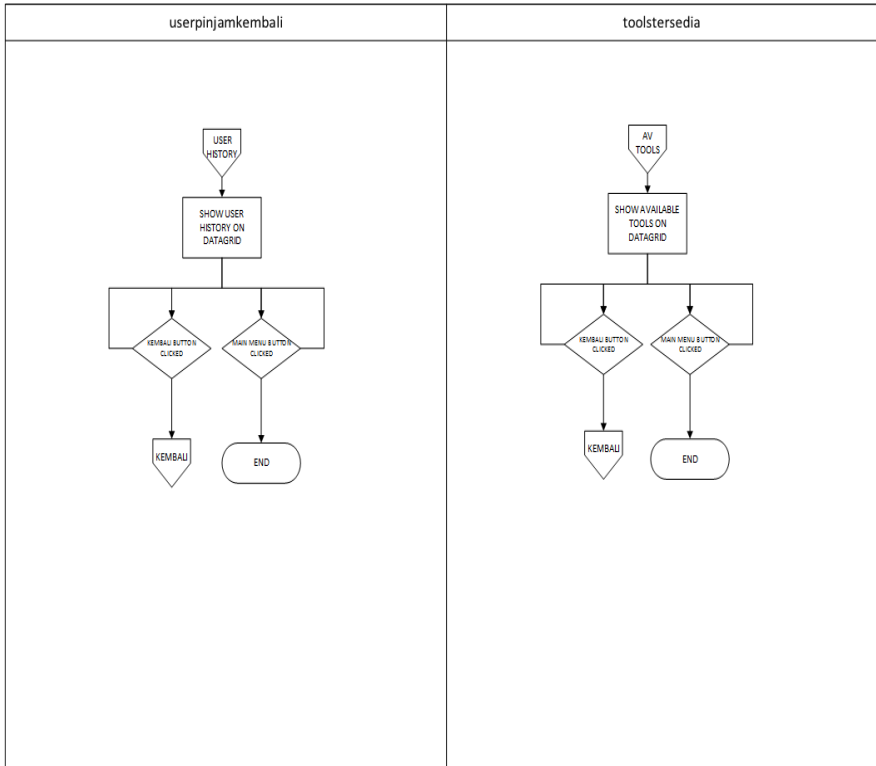
A-7. Flowchart pada *form user*



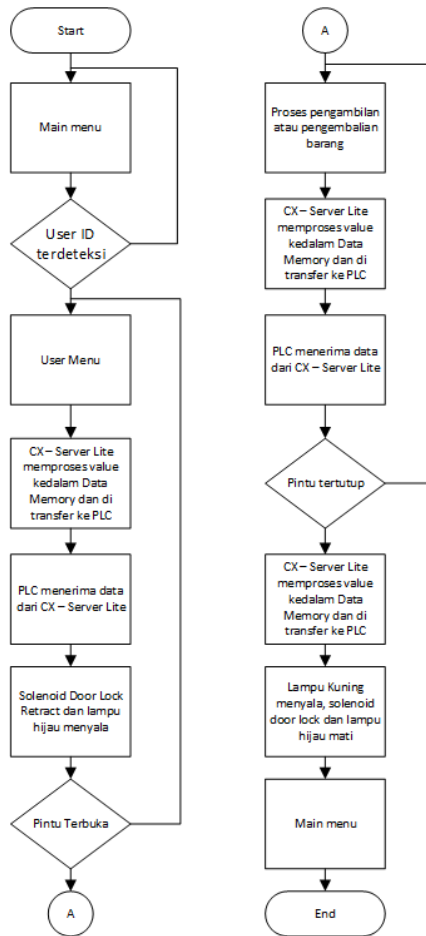
A-8. Flowchart RFID



A-9. Flowchart pada form userpinjamkembali dan tools tersedia



A-10. Flowchart komunikasi CX – Server Lite



A-11. Manual book

1 KESELAMATAN

A.PERINGATAN KESELAMATAN

1. PERINGATAN UMUM

Mesin ini dilengkapi dengan alat keselamatan untuk melindungi pengguna dan mesin dari bahaya kecelakaan. Operasi yang tidak sesuai prosedur dapat mengakibatkan kecelakaan. Oleh sebab itu, operator harus membaca buku petunjuk penggunaan mesin sebelum mengoperasikan dan melakukan perawatan mesin. Apabila ada fungsi yang tidak dijelaskan dalam buku ini, silahkan menghubungi Departemen Technical Service.

Sebelum mengoperasikan mesin, disarankan untuk membaca peringatan umum untuk mencegah kecelakaan.

1. Mesin ini harus dioperasikan oleh personil **terlatih**.
2. Disarankan untuk tidak mengambil *Tools* lebih dari 3
3. Operasikanlah mesin ini dalam keadaan **sehat** dan **fit** untuk bekerja.
4. Area disekitar bagian operasi mesin harus dilengkapi dengan **pencahaya**an yang **sesuai** dan perlengkapan pengaman.
5. **Lantai** pabrik harus dalam keadaan **bersih** dan **kering**.
6. **Rambut** yang **panjang** harus **diikat** dengan rapi.

2. TINDAKAN SAFETY

Gunakan Alat Pelindung Diri (APD) seperti gambar di bawah ini :



I. Safety action ketika mesin dalam keadaan running dan beroperasi

1. Pastikan tangan/anggota badan anda dalam keadaan kering dan tidak basah.
2. Pastikan **Admin mengecek** keteraturan kinerja mesin , proses, dan hasilnya secara **berkala**.

II. Safety Action ketika mesin dalam keadaan bermasalah dan macet

1. Ketika terjadi masalah, tekan EMERGENCY.
2. Pastikan mesin berada dalam keadaan **MATI/OFF** sebelum tindakan / penanganan selanjutnya.

III. Buka Bab 4 buku bagian Maintenance/Troubleshooting

B. **WARNING LABEL**

1. Karena *warning label* menempel pada bagian-bagian mesin agar dapat beroperasi dengan aman, pahami isi *warning label* sebelum mengoperasikan mesin.
2. Jaga kebersihan *warning label* sehingga dapat dibaca dengan jelas.
3. Jangan pindahkan *warning label*. Jika *warning label* dipindahkan, tempelkan kembali ke posisi semula.



Mengindikasikan adanya **tegangan listrik tinggi (berbahaya)**, kegagalan dalam mengikuti instruksi akan menyebabkan kematian atau luka serius.



Instruksi untuk **menutup pintu** agar terhindar dari kemungkinan bahaya.

***CATATAN WARNING LABEL**

Peringatan *Warning Label*

1. Pahami seluruh *warning label* mesin ini.
2. Jaga kebersihan *warning label* agar terbaca jelas.
3. Biarkan posisi *warning label* sesuai tempatnya.

2 *DESKRIPSI MESIN*

A. MACHINE



Mesin *Automation Tool's Crib*

Integrasi Mesin *Tools Crib* merupakan integrasi antara tiga device yang berbeda yaitu Visual Studio, PLC dan RFID.

B. SPESIFIKASI & PRINSIP KERJA MESIN



Desain Mesin

Urutan proses :

1. Login menggunakan kartu RFID dengan menempelkan kartu pada kaca lemari.
2. Setelah mendengar bunyi *Door Lock*, tunggu sebentar lalu buka pintu.
3. Ambil *tools* sesuai dengan yang dibutuhkan (maks 3 *tools*).
4. Tutup pintu setelah mengambil *tools*.
5. Lihat pada layar HMI sudah muncul notifikasi *Transaction Succesfull*

KEBUTUHAN DAYA	
Tegangan	220v
Arus	1A
Konsumsi daya	240W

KAPASITAS		
lebar rak	47 cm	
panjang rak	87.5 cm	
jumlah rak	3	
kapasitas RFID	255 tools	
Kapasitas Lemari	20	Large size tools
	40	medium Size tools

DIMENSI DAN LAYOUT	
Lemari	
Tinggi	1,4 m
Panjang	0,5
Lebar	0,5
Layout	
Tinggi	2 m
panjang	1,5 m
lebar	1 m

PC	
Software requirement	
Visual Studio 2015	
CX Serverlite	
CX Programmer	
XAMPP (webserver)	
Microsoft SQL Server	
MS.Access 2016	

HARDWARE MINIMUM	
Processor	Intel Core i3 and equivalent
RAM	2 GB
HDD	250 GB
VGA	Intel GMA HD Integrated Graphics
Directx	directx9
OS	Windows 7

HARDWARE YANG DISARANKAN	
Processor	Intel Core i5 and equivalent
RAM	4 GB
HDD	500 GB
VGA	NVIDIA GeForce 940M dengan 2GB DDR3 VRAM
Directx	Directx11
OS	Windows 10

Communication	
Ethernet Port	2 port
Wireless Network adapter	Wifi 802.11 b/g/n
USB 2.0	3 Port

SISTEM IDENTIFIKASI	
RFID	
Reader	CARDTEK CT-I809
Tag	Mifare S50 EPC G2 Card
	Windshield Sticker Passive tag EPC gen2

KONTROLER	
PLC	OMRON CJ1M MODULAR
Power Supply	CJ1M - PA202
CPU	CJ1M - CPU13
Input	CJ1M - ID211
Output	CJ1M - OC211
Communication	CJ1M - ETN13

NO	Nama Barang / Deskripsi Service
1	DR22D0L-E3G_FUJI_PILOT LAMP GREEN
2	DR22D0L-E3R_FUJI_PILOT LAMP RED
3	DR22D0L-E3Y_FUJI_PILOT LAMP YELLOW
4	MORGAN BOOKCASE_INFORMA_WHITE BOOKCASE(216.2X43.4X186)
5	UTICON_POWER Outlet Isi 6
6	Kabel Serial DB9 male-female 10M
7	UC-232_ATEN_USB to Serial Converter
8	Kabel LAN UTP RJ45 10 M
9	BOX MCB 2 POLE(BW32AAG)
10	Legrand rubber 2P+E POWER PLUG
11	DURA 600_OMNI-ID RFID TEXT
12	Kartu RFID Card UHF ISO18000-6C
13	TL5 T5_PHILIPS_LAMPU LED 4W

NO	Nama Barang / Deskripsi Service
14	Kunci Pintu Solenoid Door Lock 12 V Automatic Electronic Arduino
15	KR18268 Long Lever Limit Switch 15 A 250VAC//#24
16	RFID UHF Windshield Sticker Passive Tag EPC Gen2/ISO18000-6C
17	UHF RFID Passive on-metal Tag Screw holes 915MHZ
18	Kartu RFID Card UHF ISO18000-6C
19	UHF Reader & Writer (Middle Range)-CT-I809
20	Selector Key Switch 3 Posisi Stay Put
21	Power Supply 12V 3A Jaring CCTV LED AC 220V to DC 12V 3A Adaptor PSU
22	UHF Windshield 100x45mm with Anti Metal Shield for parking

C. FITUR UTAMA MESIN

Automation Tools Crib ini membuat peminjaman dan pengembalian secara otomatis.

No	FITUR
1	Transaksi otomatis tanpa operator
2	Pembukuan otomatis dengan database
3	Identifikasi Alat dan pengguna Otomatis dengan RFID
4	Membutuhkan transaksi < 2 menit
5	Maintenance data dan peralatan dapat diawasi

3 OPERASI & PERAWATAN

A. PERSIAPAN SEBELUM BEROPERASI

1. Pastikan tidak ada bahaya di sekitar mesin.
2. Pastikan alat keselamatan tersedia.
3. Periksa dan pastikan kabel sudah tersambung dengan power source.

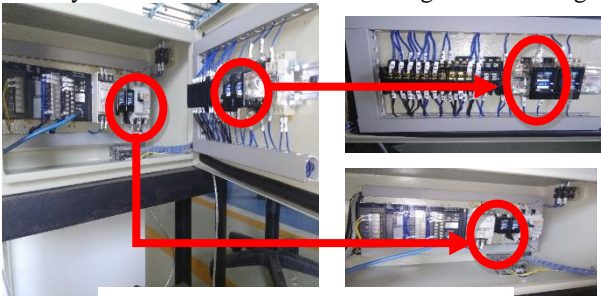
B. MENYALAKAN MESIN

1. Nyalakan mesin dengan mendorong *switch* NFB dari posisi OFF ke posisi ON



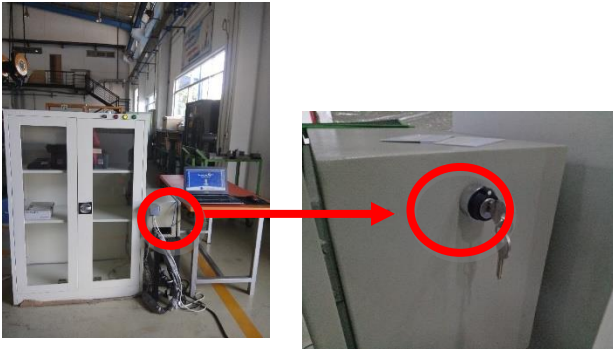
NFB Master Panel

2. Nyalakan PLC pada NFB Box dengan mendorong *switch*



Layout Master Panel

C. DESKRIPSI PANEL OPERASI



NO	ITEM	DESKRIPSI
1	Key Switch	Akses untuk admin

D. OPERASI MESIN

PENGOPERASIAN MANUAL

1. Periksa apakah kabel *power* terhubung dengan benar.
2. Aktifkan NFB utama untuk menyalakan mesin.
3. Pastikan lampu *indicator power* menyala.
4. Pastikan tidak muncul alarm ditandai dengan *indicator tower lamp* berwarna merah. Apabila *indicator tower lamp* berwarna merah, buka software *Automation Tools Crib*.
5. Pada *operation panel* pilih *Key Switch* dengan seizin admin.
6. Pada Form Admin pilih login manual
7. Masukkan NIK dan Password

PENGOPERASIAN OTOMATIS

1. Pastikan tidak ada bahaya di sekitar peralatan.
2. Pastikan peralatan keselamatan tersedia.
3. Periksa apakah kabel terhubung dengan benar.
4. Pastikan lampu *indicator* menyala.

5. Pastikan tidak muncul alarm ditandai dengan *indicator tower lamp* berwarna merah. Apabila *indicator tower lamp* berwarna merah, buka *Software Automation Tools Crib*.
6. Tempelkan kartu pada kaca lemari untuk login.
7. Buka pintu setelah terdengar bunyi *Door Lock*.
8. Ambil *Tools* sesuai kebutuhan atau kembalikan *Tools* sesuai dengan tempatnya.
9. Tutup pintu dengan rapat.
10. Proses selesai.

4 TROUBLESHOOTING

Apabila terjadi masalah (error) selama mesin beroperasi :

1. Tekan tombol emergency.
2. Nonaktifkan tombol dengan memutar tombol ke kiri.
3. Ikuti petunjuk troubleshooting di bawah ini :

1) HMI

Problem	Kemungkinan Penyebab	Tindakan Perbaikan
1. Program berhenti tiba – tiba	- Program Not responding	- Buka kembali program .Exe
A-1. <i>Object out of String</i>	- Kabel Ethernet terlepas	- Pasang kembali kabel Ethernet - Lepaskan dan masukan kebalik kabel Ethernet ke USB
A-2. <i>Serial Comm Error</i>	- USB RFID terlepas	- Pasang kembali kabel RFID - Lepaskan dan masukan kebalik kabel RFID ke USB

A-12. Datasheet CPU PLC

1-1 Introduction

1-1-1 Overview

The CJ1M CPU Units with Ethernet Functions combine the high-speed, large-capacity, multifunctional capabilities of the SYSMAC CJ-series CPU Unit in a micro PLC with the functionality of an Ethernet Unit.

Note The Ethernet functional element is part of the CPU Unit and cannot be removed.

The CJ1M-CPU1□-ETN incorporates a CPU functional element combined with an Ethernet functional element. The CPU functional element has the same functionality as a CJ1M-CPU1□ CJ1M CPU Unit (without built-in I/O) with unit version 3.0 or later. The Ethernet functional element provides the main functions of the CJ1W-ETN21 Ethernet Unit.

Product name	Product model number	Configuration	
		CPU functional element	Ethernet functional element
		CPU Unit with same functionality	Ethernet Unit with similar functionality
CPU Unit with Ethernet Functions	CJ1M-CPU11-ETN	CJ1M-CPU11	CJ1W-ETN21
	CJ1M-CPU12-ETN	CJ1M-CPU12	
	CJ1M-CPU13-ETN	CJ1M-CPU13	

CPU Functional Element

Model	Program capacity	I/O points	Data Memory	Expansion Racks
CJ1M-CPU11-ETN	5 Ksteps	160 points	32 Kwords (No Extended Data Memory)	None
CJ1M-CPU12-ETN	10 Ksteps	320 points		
CJ1M-CPU13-ETN	20 Ksteps	640 points		1 max.

Note Details on the CPU Functional Element are not provided in this manual. Refer to the *CJ1M CPU Unit Operation Manual*.

Ethernet Functional Element

Model	Physical layer	Maximum number of nodes in FINS network	Communications service
CJ1M-CPU11-ETN	100BASE-TX, 10BASE-T	254	<ul style="list-style-type: none"> • FINS communications service • FTP server • Automatically adjusted clock information. • Web functions
CJ1M-CPU12-ETN			
CJ1M-CPU13-ETN			

1-1-2 Features

- The CPU functional element has the functionality of a CJ1M CPU Unit without built-in I/O.
- The CPU Unit supports FINS communications services, FTP services, and automatic clock adjustment, without requiring installation of a separate Ethernet Unit.

Note Socket services and sending/receiving mail are not supported.

- The unit settings and status monitoring of the Ethernet functional element can be performed from the Web browser.

1-2 Specifications

1-2-1 Performance Specifications

CPU Functional Element

Item		Specification		
Model		CJ1M-CPU13-ETN	CJ1M-CPU12-ETN	CJ1M-CPU11-ETN
Functional element version		Ver. 3.0 or later		
I/O points		640	320	160
User program memory		20 Ksteps	10 Ksteps	5 Ksteps
Maximum number of Expansion Racks		1 max.	Not supported.	
CJ-series Basic I/O Units		Total: 9 + 10 Units max. (CPU Rack: 9 Units Expansion Rack: 10 Units)		Total: 9 Units max.
CJ-series Special I/O Units				
CJ-series CPU Bus Units				
Data Memory		32 Kwords		
Extended Data Memory		Not supported.		
Pulse functions		Not supported.		
Interrupt inputs		2	1	
PWM output points		None		
Maximum subroutine number		1,024	256	
Maximum jump number for JMP Instruction		1,024	256	
Function blocks (FB)	Maximum number of definitions	128		
	Maximum number of instances	256		
Flash memory	Function block program memory	256 Kbytes		
	Comment file	64 Kbytes		
	Program Index file	64 Kbytes		
	Symbol tables	64 Kbytes		

Ethernet Functional Element Transfer Specifications

Item		Specification	
Functional element version		Ver. 1.0 or later	
Media access method		CSMA/CD	
Modulation method		Baseband	
Transmission paths		Star form	
Baud rate		100 Mbit/s (100Base-TX)	10 Mbit/s (10Base-TX)
Transmission media		Unshielded twisted-pair (UDP) cable Categories: 5, 5e Shielded twisted-pair (STP) cable Categories: 100 Ω at 5, 5e	Unshielded twisted-pair (UDP) cable Categories: 3, 4, 5, 5e Shielded twisted-pair (STP) cable Categories: 100 Ω at 3, 4, 5, 5e
Transmission distance		100 m (distance between hub and node)	
Number of cascade connections		No restrictions if switching hubs are used.	
CPU Bus Unit System Setup Area capacity		994 bytes (See note 2.)	

Common Specifications

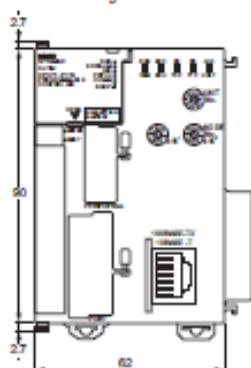
Item	Specification
Current consumption (supplied by Power Supply Units)	0.95 A at 5 VDC
Noise immunity	2 kV on power supply line (conforming to IEC61000-4-4)
Vibration resistance	10 to 57 Hz, 0.075-mm amplitude, 57 to 150 Hz, acceleration: 9.8 m/s ² in X, Y, and Z directions for 80 minutes (Time coefficient: 8 minutes × coefficient factor 10 = total time 80 min.) (according to JIS C0040)
Shock resistance	147 m/s ² 3 times each in X, Y, and Z directions (Relay Output Unit: 100 m/s ²) (according to JIS C0041)
Ambient operating temperature	0 to 55°C
Ambient operating humidity	10% to 90% (with no condensation)
Atmosphere	Must be free from corrosive gases.
Ambient storage temperature	-20 to 75°C (excluding battery)
Grounding	Less than 100 Ω
Enclosure	Mounted in a panel.
Weight (Unit only)	210 g max.
Dimensions	62 × 90 × 65 mm (W × H × D)
Safety measures	Conforms to cULus and EC Directives.

- Note**
1. The number of steps in a program is not the same as the number of instructions. For example, LD and OUT require 1 step each, but MOV(021) requires 3 steps. The program capacity indicates the total number of steps for all instructions in the program. For details, refer to the *SYSMAC CJ-series Programmable Controllers Operation Manual (W393)* for the number of steps required for each instruction.
 2. The Ethernet functional element in the CJ1M CPU Unit with Ethernet Functions is also allocated in the CPU Bus Unit System Setup Area.

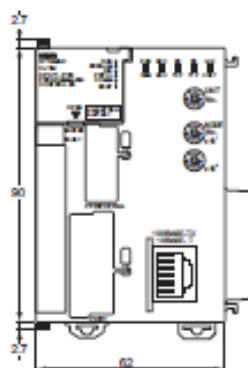
Dimensions (Same for All Units)

Front

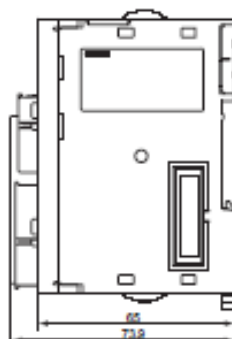
Version 1.5 or Higher



Version 1.4 or Lower



Side



1-2-2 Comparison between Ethernet Functional Elements and Ethernet Units

The following table shows the differences between CJ11M CPU Units with Ethernet Functions and CJ-series Ethernet Units.

Item	CJ-series Ethernet Unit	CJ11M CPU Units with Ethernet Functions
Model number	CJ1W-ETN21	CJ11M-CPU11-ETN CJ11M-CPU12-ETN CJ11M-CPU13-ETN
Physical layer	100BASE-TX, 10BASE-T	Same
Number of nodes on FINS network	254	Same
Removing Ethernet functional element	Possible	Not possible
Server specification	Specification by IP address or host name specifications (DNS client function)	Same

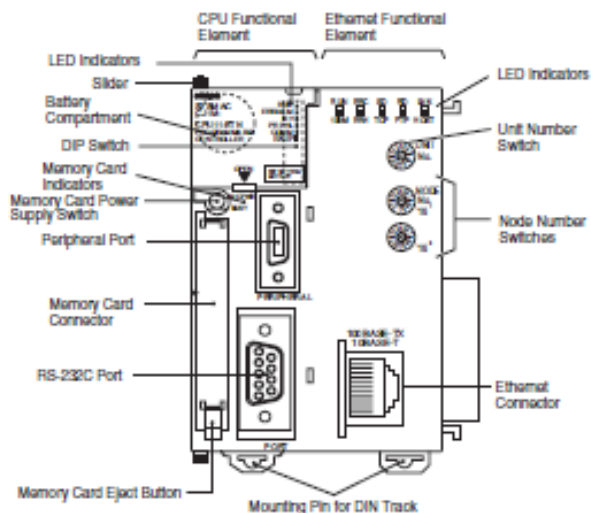
Item	CJ-series Ethernet Unit	CJ1M CPU Units with Ethernet Functions		
Communications service	FINS communications service	FINS/UDP FINS/TCP	Same	
	FTP server function	The CPU Unit's file memory (Memory Card or EM file memory) can be read/written.	The CPU functional element's file memory (Memory Card only) can be read/written.	
	Automatic clock information adjustment	The CPU Unit's internal clock data can be automatically adjusted to the clock data received from the SNTP server	Same	
	Web functions	The Unit settings can be made and status can be read from a Web browser using the Web server.	Same	
	Mail functions	Mail send functions Mail receive functions	Not possible	
	Socket service function	TCP socket services UDP socket services	Not possible	
	FINS commands	RESET		Same
		CONTROLLER DATA READ		Same Responds to CJ1W-ETN21
		CONTROLLER STATUS READ		Same
		ECHOBACK TEST		Same
		BROADCAST TEST (READ RESULTS)		Same
		BROADCAST TEST (SEND TEST DATA)		Same
		ERROR LOG READ		Same
		ERROR LOG CLEAR		Same
		REQUEST TO OPEN UDP SOCKET		Not possible
		REQUEST TO RECEIVE UDP SOCKET		Not possible
		REQUEST TO SEND UDP SOCKET		Not possible
		REQUEST TO CLOSE UDP SOCKET		Not possible
		REQUEST TO OPEN TCP SOCKET (PASSIVE)		Not possible
		REQUEST TO OPEN TCP SOCKET (ACTIVE)		Not possible
		REQUEST TO RECEIVE TCP SOCKET		Not possible
		REQUEST TO SEND TCP SOCKET		Not possible
		REQUEST TO CLOSE TCP SOCKET		Not possible
		EXECUTE PING COMMAND		Same
		REQUEST TO CHANGE REMOTE NODE FOR FINS/TCP CONNECTION		Same
		REQUEST TO READ STATUS FOR FINS/TCP CONNECTION		Same
		IP ADDRESS TABLE WRITE		Same
		IP ADDRESS WRITE		Same
		IP ADDRESS TABLE READ		Same
		IP ROUTING TABLE READ		Same
PROTOCOL STATUS READ			Same	
MEMORY STATUS READ			Same	
SOCKET STATUS READ			Same	
ADDRESS DATA READ		Same		
IP ADDRESS READ		Same		

1-3 Part Names and Dimensions

1-3-1 Part Names

Front

■ Version 1.4 or Lower



■ CPU Functional Element

Name	Description
LED Indicators	Indicate the status of the CPU Functional Element. (Refer to CPU Functional Element Indicators on page 10 for details.)
DIP switch	Sets initial settings for the CPU Functional Element.
Peripheral port	Connects to a computer running Support Software or a Programming Console.
Memory Card Indicators	MCPWR (green): Lit when power is supplied to Memory Card. BUSY (orange): Lit when Memory Card is being accessed.
Memory Card power supply switch	Stops the power supply to the Memory Card so that the Memory Card can be removed.
Memory Card connector	The connector where a Memory Card can be inserted. (The label on the Memory Card must face to the right.)
Memory Card eject button	Pressed to remove the Memory Card.
RS-232C port	Connects to a serial communications device.

■ Ethernet Functional Element

Name	Description
LED Indicators	Indicate the status of the Ethernet Functional Element. (Refer to Ethernet Functional Element Indicators on page 12 for details.)
Unit number switch	Sets the unit number for Ethernet functionality. Setting range: 0 to F hex
Node number switches	Set the FINS node address for Ethernet as a 2-digit hexadecimal number. Setting range: 01 to FE hex
Ethernet connector	Connects to Ethernet through a twisted-pair cable.
Sliders	Connect to and lock the adjacent Units.
Mounting pins for DIN Track	Used to mount and lock the Unit to a DIN Rack.

Note When you are not using the peripheral port or RS-232C port on the CPU Functional Element, cover it with the enclosed cover to protect it from dust.

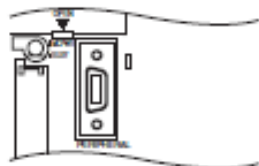
CPU Functional Element Indicators

The following table describes the LED indicators located on the front panel of the CPU functional element.

Indicator	Color	Status	Meaning
RUN	Green	ON	PLC is operating normally in MONITOR or RUN mode.
		Flashing	System download mode error or DIP switch settings error.
		OFF	PLC has stopped operating while in PROGRAM mode, or has stopped operating due to a fatal error, or is downloading data from the system.
ERR/ALM	Red	ON	A fatal error has occurred (including FALS instruction execution), or a hardware error (watchdog timer error) has occurred. The CPU functional element will stop operating, and the outputs from all Output Units will turn OFF.
		Flashing	A non-fatal error has occurred (including FAL instruction execution). The CPU functional element will continue operating.
		OFF	CPU functional element is operating normally.
INH	Orange	ON	Output OFF Bit (A50015) has been turned ON. The outputs from all Output Units will turn OFF.
		OFF	Output OFF Bit (A50015) has been turned OFF.
PRPHL	Orange	Flashing	CPU functional element is communicating (sending or receiving) via the peripheral port.
		OFF	CPU functional element is not communicating via the peripheral port.
COMM	Orange	Flashing	CPU functional element is communicating (sending or receiving) via the RS-232C port.
		OFF	CPU functional element is not communicating via the RS-232C port.
BKUP	Orange	ON	User program and parameter area data is being backed up to flash memory in the CPU functional element or being restored from flash memory. Note Do not turn OFF the power supply to the PLC while this indicator is lit.
		OFF	Data is not being written to flash memory.



Indicator	Color	Status	Meaning
MCPWR	Green	ON	Power is being supplied to the Memory Card.
		Flashing	Flashes once: Easy backup read, write, or verify normal Flashes five times: Easy backup write malfunction Flashes three times: Easy backup write warning Flashes continuously: Easy backup read or verify malfunction
		OFF	Power is not being supplied to the Memory Card.
		Flashing	Memory Card is being accessed.
BUSY	Orange	Flashing	Memory Card is being accessed.
		OFF	Memory Card is not being accessed.

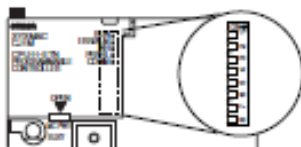


DIP Switch

The CJM1 CPU Unit with Ethernet Functions has an 8-pin DIP switch that is used to set basic operational parameters for the Unit, as shown in the following table.

Pin no.	Setting	Function	Usage	Default
1	ON	Writing disabled for user program memory. (See note.)	Used to prevent programs from being accidentally overwritten from Programming Devices (Including Programming Console).	OFF
	OFF	Writing enabled for user program memory.		
2	ON	The user program is automatically transferred from the Memory Card when power is turned ON.	Used to store the programs in the Memory Card to switch operations, or to automatically transfer programs at power-up (Memory Card ROM operation). Note When pin 7 is ON and pin 8 is OFF, easy backup reading from the Memory Card is given priority, so even if pin 2 is ON, the user program is not automatically transferred from the Memory Card when power is turned ON.	OFF
	OFF	The user program is not automatically transferred from the Memory Card when power is turned ON.		
3	---	Not used.	---	OFF
4	ON	Peripheral port communications parameters set in the PLC Setup are used.	Turn ON to use the peripheral port for a device other than Programming Console or CX-Programmer (Peripheral bus only).	OFF
	OFF	Peripheral port communications parameters set using Programming Console or CX-Programmer (Peripheral bus only) are used.		
5	ON	RS-232C port communications parameters set using a CX-Programmer (Peripheral bus only) are used.	Turn ON to use the RS-232C port for a Programming Device.	OFF
	OFF	RS-232C port communications parameters set in the PLC Setup are used.		
6	ON	User-defined pin. Turns OFF the User DIP Switch Pin Flag (A30512).	Set pin 6 to ON or OFF and use A30512 in the program to create a user-defined condition without using an I/O Unit.	OFF
	OFF	User-defined pin. Turns ON the User DIP Switch Pin Flag (A30512).		
7	ON	Writing from the CPU functional element to the Memory Card.	Press and hold the Memory Card Power Supply Switch for three seconds.	OFF
		Restoring from the Memory Card to the CPU functional element.	To read from the Memory Card to the CPU functional element, turn ON the PLC power. This operation is given priority over automatic transfer (pin 2 is ON) when power is ON.	
	OFF	Verifying contents of Memory Card.	Press and hold the Memory Card Power Supply Switch for three seconds.	
8	OFF	Always OFF.		OFF

- Note**
- The following data cannot be overwritten when pin 1 is ON:
 - All parts of the user program (programs in all tasks)
 - All data in the parameter area (such as the PLC Setup and I/O table)
 When pin 1 is ON, the user program and parameter area will not be cleared when the memory clear operation is performed from a Programming Device.
 - The CPU functional element will not enter any mode except PROGRAM mode after backing up data to a Memory Card using DIP switch pin 7. To enter RUN or MONITOR mode, turn OFF the power supply, turn OFF pin 7, and then restart the PLC. This will enable changing the operating mode as normal.



Note The language displayed for the CJ-series CPU Units is not set on the DIP switch, but rather is set using Programming Console keys.

Ethernet Functional Element Indicators

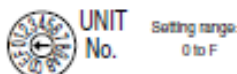
The status of the indicators show the operating status of the Ethernet functional element, as shown below.



Indicator	Color	Status	Meaning
RUN (Transfer speed)	Green	Not lit	Operation stopped Hardware error
		Lit	Normal operation
100M (Transfer speed)	Green	Not lit	10 Mbps (10Base-T)
		Lit	100 Mbps (100Base-TX)
ERC (Ethernet functional element error)	Red	Not lit	Ethernet functional element normal
		Lit	Node address not between 1 and 254 A hardware (e.g., internal memory) error has occurred.
ERH (CPU functional element error)	Red	Not lit	CPU functional element normal
		Lit	An error has occurred at the CPU functional element. There is an error in the I/O table, unit number, unit setup, or routing table settings.
		Flashing	An illegal IP address has been set. With automatic address generation, the rightmost two digits of the IP address do not match the node address.
SD (Send Data)	Yellow	Not lit	Not sending data (ready to send)
		Lit	Sending data
RD (Receive Data)	Yellow	Not lit	Not receiving data (ready to receive)
		Lit	Receiving data
LNK (Link status)	Yellow	Not lit	Link not established between hubs.
		Lit	Link established between hubs.
TCP (TCP socket in use)	Yellow	Not lit	None of the eight TCP sockets provided for socket services is in use.
		Lit	At least one of the eight TCP sockets provided for socket services is in use.
FTP (FTP Server Service)	Yellow	Not lit	FTP server on standby.
		Lit	FTP server operating (one user connected).
HOST (Server connection status)	Yellow	Not lit	Server (DNS, SMTP, POP3, or SNMP) not being used.
		Flashing	An error has occurred in the server connection.
		Lit	The server connection is normal.

Setting the Unit Number for the Ethernet Functional Element

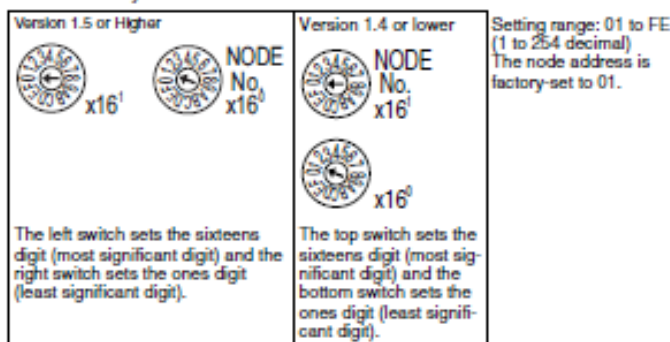
The unit number is used to identify individual CPU Bus Units when more than one CPU Bus Unit is mounted to the same PLC. Use a small screwdriver to make the setting, taking care not to damage the rotary switch. The unit number is factory-set to 0.



- Note**
1. Turn OFF the power supply before setting the unit number.
 2. If the unit number is being set for the first time or changed, then I/O tables must be created for the PLC.
 3. With CS-series and CJ-series PLCs, dedicated areas are automatically allocated in the CIO Area and DM Area according to the unit numbers that are set. For details, refer to *SECTION 4 Ethernet Unit Memory Allocations (W420)*.

Setting the Node Address for the Ethernet Functional Element

With the FINS communications service, when there are multiple Ethernet functional elements or Ethernet Units connected to the Ethernet network, the Ethernet functional elements and Ethernet Units are identified by node addresses. Use the node address switches to set the node address between 01 and FE hexadecimal (1 to 254 decimal). Do not set a number that has already been set for another node on the same network.



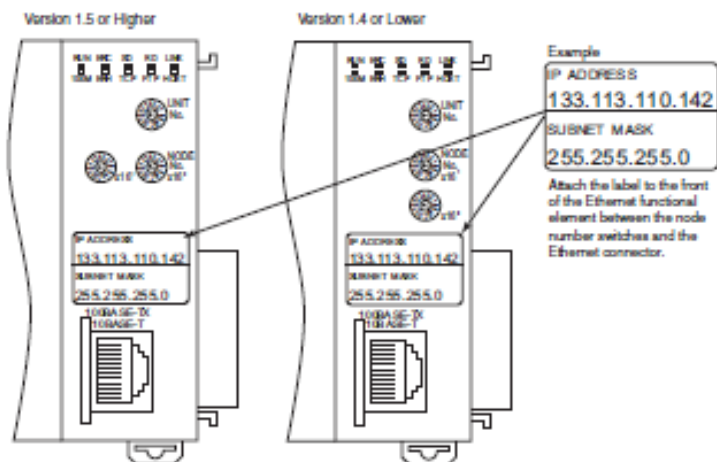
- Note** Turn OFF the power supply before setting the node address.

Relation to IP Addresses

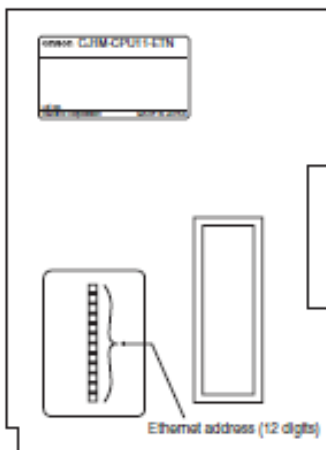
When using the automatic generation method (either dynamic or passive) for address conversion, set the rightmost byte of the IP address set for the Ethernet functional elements or Ethernet Unit to the same value as the node address. If this is not possible, either the IP address table method or the combined method must be used for address conversion. (For details, refer to the *Ethernet Units Construction of Networks Operation Manual (W420)*.) If the FINS communications service is not used on the Ethernet network, then it is all right to overlap with other Ethernet functional elements or Ethernet Units. The setting, however, must be made within a range of 01 to FE. If a value outside of this range is set, the ERC indicator will light.

An IP address label is provided with the Ethernet functional element. Writing the IP address and the subnet mask on this label and then attaching it to the front of the Ethernet functional element allows the IP address and subnet mask to be easily confirmed.

Front of Ethernet Functional Element



Each communications device connected to the Ethernet network is allocated a unique Ethernet address. For the Ethernet functional element, this Ethernet address is shown on the right side of the Unit as a 12-digit hexadecimal number.



Note The Ethernet address can also be checked using the FINS command.

A-13. Datasheet Input

CJ-series Input Units **CJ1W-ID/IA**

CGM_CJ1W-ID_IA_DS_E_11_8

A Wide Range of Basic Input Units for High Speed Input and Different Applications

- Receive ON/OFF signals from external devices into the PLC System to update I/O memory in the CPU Unit.
- New high-speed input models CJ1W-ID212 and CJ1W-ID288 are now available. These units can help to increase system throughput.



CJ1W-ID212



CJ1W-ID288

Features

- High-speed input models are available, meeting versatile applications.
 - ON Response Time: 15µs, OFF Response Time: 90µs
- Use 24-VDC, 100-VAC, and 200-VAC models to connect to devices with different types of outputs.
- The 24-VDC models can be connected to devices with either NPN or PNP outputs. There is no need to select the polarity.¹⁾
- A digital filter in the Unit can be set from 0 to 32 ms to reduce the influence of external noise.
- Either a Fujitsu or MIL connector interface can be used.²⁾
- Several models of Terminal Block Conversion Units are available, making it easy to connect to external devices.






¹⁾ The same polarity is used for the same common.
²⁾ For models with 32 or 64 inputs.

Ordering Information

International Standards

- The standards are abbreviated as follows: U: UL, U1: UL (Class I Division 2 Products for Hazardous Locations), C: CSA, UC: cULus, UC1: cULus (Class I Division 2 Products for Hazardous Locations), CU: cUL, N: NK, L: Lloyd, and CE: EC Directives.
- Contact your OMRON representative for further details and applicable conditions for these standards.

Input Units

Unit type	Product name	Specifications					Current consumption (A)		Model	Standards
		I/O points	Input voltage and current	Commons	External connection	No. of words allocated	5 V	24 V		
CJ1 Basic I/O Units	 DC Input Units	8 inputs	12 to 24 VDC, 10 mA	Independent contacts	Removable terminal block	1 word	0.06	–	CJ1W-ID201	UL1, N, L, CE
		16 inputs	24 VDC, 7 mA	16 points, 1 common	Removable terminal block	1 word	0.06	–	CJ1W-ID211	
	 	16 inputs (High speed)	24 VDC, 7 mA	16 points, 1 common	Removable terminal block	1 word	0.12	–	CJ1W-ID212	N, L, CE
		32 inputs	24 VDC, 4.1 mA	16 points, 1 common	Fujitsu connector	2 words	0.06	–	CJ1W-ID221	
	 	32 inputs	24 VDC, 4.1 mA	16 points, 1 common	MEI connector	2 words	0.06	–	CJ1W-ID222	UL1, N, L, CE
		32 inputs (High speed)	24 VDC, 4.1 mA	16 points, 1 common	MEI connector	2 words	0.25	–	CJ1W-ID223	
	 	64 inputs	24 VDC, 4.1 mA	16 points, 1 common	Fujitsu connector	4 words	0.06	–	CJ1W-ID261	N, L, CE
		64 inputs	24 VDC, 4.1 mA	16 points, 1 common	MEI connector	4 words	0.06	–	CJ1W-ID262	
	 AC Input Units	8 inputs	200 to 24 VAC, 10 mA (200 V, 50 Hz)	8 points, 1 common	Removable Terminal Block	1 words	0.06	–	CJ1W-IA201	UL1, N, L, CE
		16 inputs	100 to 120 VAC, 7 mA (100 V, 50 Hz)	16 points, 1 common	Removable Terminal Block	1 words	0.06	–	CJ1W-IA111	

Accessories

Connectors are not included for models with connectors. Either use one of the applicable connector listed below or use an applicable Connector-Terminal Block Conversion Unit or I/O Relay Terminal. For details on wiring methods, refer to *External Interfaces*.

Applicable Connectors

Fujitsu Connectors for 32-input, 32-output, 64-input, 64-output, 32-input/32-output, and 16-input/16-output Units

Size	Connection	Remarks	Applicable Units	Model	Standard
40-pin Connector	Soldered	PCN-361J023-AU Connector	Fujitsu Connector; CJ1W-ID031 (32 inputs); 1 per Unit CJ1W-ID051 (32 inputs); 2 per Unit CJ1W-ID051 (32 outputs); 1 per Unit CJ1W-ID051 (32 outputs); 2 per Unit CJ1W-ID051 (32 inputs, 32 outputs); 2 per Unit	C500-C5404	-
		PCN-360C023-B Connector			
	Crimped	PCN-361J023-AU Heating Connector		C500-C5405	
		PCN-360C023-B Connector			
Pressure welded	PCN-361J023-AUP	C500-C5403			
34-pin Connector	Soldered	PCN-361J023-AU Connector	Fujitsu Connector; CJ1W-ID031 (16 inputs, 16 outputs); 2 per Unit	C500-C5241	-
		PCN-360C023-B Connector			
	Crimped	PCN-361J023-AU Socket Connector		C500-C5242	
		PCN-360C023-B Connector			
Pressure welded	PCN-361J023-AUP	C500-C5243			

ML Connectors for 32-input, 32-output, 64-input, 64-output, 32-input/32-output, and 16-input/16-output Units

Size	Connection	Remarks	Applicable Units	Model	Standard
40-pin Connector	Pressure welded	FRCS-4028-IT03	ML Connector; CJ1W-ID030030 (32 inputs); 1 per Unit CJ1W-ID030030 (32 outputs); 1 per Unit CJ1W-ID050 (32 inputs); 2 per Unit CJ1W-ID050 (32 outputs); 2 per Unit CJ1W-ID050 (32 inputs, 32 outputs); 2 per Unit	X04M-4030-T	-
	Crimped	-		X06N-401*	
34-pin Connector	Pressure welded	FRCS-4028-IT03	ML Connector; CJ1W-ID030030 (16 inputs, 16 outputs); 2 per Unit	X04M-3030-Y	-
Crimped	-	X06N-301*			

* Crimp Contacts are also required. Refer to page 20 for details.



Applicable Connector-Terminal Block Conversion Units

Type	Series	Number of pins	Wiring method	Terminal type	Size			Mounting		Conduit terminals	Blender mountable	Isolates	IO Units	Model *	Standard
					Depth (mm)	Height (mm)	Width (mm)	SN Tracks	Cones						
PLC	XWR	34	Phillips screw	M3	50	40.85	136.7	Yes	No	No	No	No	CJ1W-ID031 CJ1W-ID051	XWR-P34QD-C1	-
													CJ1W-ID030 CJ1W-ID030 CJ1W-ID050		
			Screw (for up)	M3 (European type)	50	44.01	98.5						CJ1W-ID031 CJ1W-ID051	XWR-E34QD-C1	
													CJ1W-ID030 CJ1W-ID030 CJ1W-ID050		
Push-in spring	Clamp	50	44.01	98.5	CJ1W-ID031 CJ1W-ID051	XWR-P34QD-C1									
					CJ1W-ID030 CJ1W-ID030 CJ1W-ID050		XWR-P34QD-C2								

Note: For the combination of input Units with Connector-Terminal Block Conversion Units, refer to 2. Connecting Connector-Terminal Block Conversion Units.

* Representative models only. For details, refer to the XWR series catalog (Cat. No. 0077).

Connecting Cables for Connector-Terminal Block Conversion Units

Appearance	Connectors	Cable length [m]	Model
	One 40-pin Fujitsu Connector to One 40-pin ML Connector	0.5	XW22-000PF
		1	XW22-100PF
		1.5	XW22-150PF
		2	XW22-200PF
		3	XW22-300PF
	One 40-pin ML Connector to One 40-pin ML Connector	0.5	XW22-000PM
		1	XW22-100PM
		1.5	XW22-150PM
		2	XW22-200PM
		3	XW22-300PM
5	XW22-500PM		

Applicable I/O Relay Terminals

Type	Series	Specifications				Size (terminal mounting)			Mounting		Model	Status			
		Classification	Relay	Number of points	Relay coil current (max.)	Rated voltage	Height (mm)	Width (mm)	Depth (mm)	Pin Track			Screw		
Push-in Plus terminal block	G70V	Inputs	DC	NPN	16 (3/7:16) × 16	50 mA	32 VDC	143	40	50	Yes	Yes	G70V-SD16P-14	V, C, C2 (VDE certified)	
			PnP	NPN	16 (3/7:16) × 16	50 mA							G70V-SD16P-14		
													G70V-SD16P-C16-16		
		Relay (16/16)	DC	NPN	16 (3/7:16) × 16	84 point, 10 A (max.)	32 VDC	143	40	50	Yes	Yes	G70V-SD16P-14		
			PnP	NPN	16 (3/7:16) × 16	84 point, 10 A (max.)							G70V-SD16P-14		
													G70V-SD16P-C16-16		
Standard	G770	Inputs	AC	NPN	16 (3/7:16) × 16	100/110 VAC	160	55	60	Yes	No	G770-IA16 AC100/110	V, C		
						200/220 VAC						G770-IA16 AC200/220			
						115 VDC						G770-DB16 DC115			
			DC	NPN	16 (3/7:16) × 16	115 VDC	G770-DB16 DC115								
						125 VDC	G770-DB16 DC125								
						180 VDC	G770-DB16 DC180								
		Relay (16/16)	AC	NPN	16 (3/7:16) × 16	100/110 VAC	160	55	60	Yes	No	No	G770-IA16 AC100/110		
						200/220 VAC							G770-IA16 AC200/220		
						115 VDC							G770-DB16 DC115		
			DC	NPN	16 (3/7:16) × 16	115 VDC	G770-DB16 DC115								
						125 VDC	G770-DB16 DC125								
						180 VDC	G770-DB16 DC180								
High-capacity socket	G70A-1 (Socket only)	Inputs	Relay	NPN	16 (3/7:16) × 16	110 VDC max., 240 VAC max., 50	202	75	62	Yes	No	G70A-2DC16-6	V, C, C2 (VDE certified)		
						32 VDC						G70A-2DC16-6			
						32 VDC						G70A-2DC16-6			
		Relay (16/16)	NPN	16 (3/7:16) × 16	110 VDC max., 240 VAC max., 50	202	75	62	Yes	No	No	G70A-2DC16-6			
					32 VDC							G70A-2DC16-6			
					32 VDC							G70A-2DC16-6			
Space-saving	Vertical type G700-V	Inputs	NPN	16 (3/7:16) × 16	3 A or 3 A/5	135	40	67	Yes	Yes	Yes	G700-V90016	V, C, C2 (VDE certified)		
					0.5 A							G700-V90016			
					0.5 A							G700-V90016			
		Relay (16/16)	NPN	16 (3/7:16) × 16	3 A	135	40	67	Yes	Yes	Yes	Yes		G700-V90016	
					0.5 A									G700-V90016	
					0.5 A									G700-V90016	
	Flat type G700	Inputs	NPN	16 (3/7:16) × 16	3 A	68	40	42	Yes	Yes	Yes	Yes	G700-90008	-	
					3 A								G700-90016		
					3 A								G700-90016-1		
		Relay (16/16)	NPN	16 (3/7:16) × 16	3 A	136	57	36	Yes	Yes	Yes	Yes	Yes		G700-P0M16
					0.5 A										G700-P0M16
					0.5 A										G700-P0M16-1
High-capacity, space-saving	G700K	Inputs	Relay	NPN	16 (3/7:16) × 16	10 A	136	40	55	Yes	Yes	Yes	G700K-90008	-	
						32 VDC							G700K-90008		

*1. G70A is a I/O terminal socket product. Relay is not provided with the socket. Be sure to order a relay, timer separately.

*2. Each relay to be mounted must incorporate a coil that has proper specifications within the maximum rated voltage range.

*3. Eight or fewer points ON; 5 A, Nine or more points ON; 3 A.

*4. Internal common at terminal block. No internal connections.

*5. Internal common at terminal block. Internal I/O common 16 points internally connected.

*6. Internal common at terminal block. Every 2 points internally connected at terminal block middle row.

Note: 1. For the combination of Input Units with I/O Relay Terminal and Connecting Cables, refer to 3. Connecting I/O Relay Terminals.

2. Please refer to each Datasheet about details.

3. When the G770 is used with an AC rated voltage, three rated currents can be used. If a coil voltage of 110 or 220 VAC is used, 50 Hz cannot be used.

Cables for I/O Relay Terminals

Type	Name	I/O Classification	Appearance	Cable length L (mm)	Models
Fujitsu connectors (34 pins)	Cables with Connectors (1-1) XMS2-ROCO	16 I/O points		1,000	XW32-R100C
				1,500	XW32-R150C
				2,000	XW32-R200C
				3,000	XW32-R300C
				5,000	XW32-R500C
Fujitsu connectors (48 pins)	Cables with Connectors (1-2) XMS2-ROCO-C XMS2-ROCO-C	32 input points		(A) 1,000 (B) 750	XW32-R0100C-75
		32 output points		(A) 1,500 (B) 1,250	XW32-R0150C-125
				(A) 2,000 (B) 1,750	XW32-R0200C-175
		32 input points		(A) 3,000 (B) 2,750	XW32-R0300C-275
				(A) 5,000 (B) 4,750	XW32-R0500C-475
		32 output points		(A) 1,000 (B) 750	XW32-R0100C-75
				(A) 1,500 (B) 1,250	XW32-R0150C-125
		32 input points		(A) 2,000 (B) 1,750	XW32-R0200C-175
				(A) 3,000 (B) 2,750	XW32-R0300C-275
		32 output points		(A) 5,000 (B) 4,750	XW32-R0500C-475
(A) 1,000 (B) 750	XW32-R0100C-75				
ML connectors (38 pins)	Cables with Connectors (1-1) XMS2-ROCO XMS2-ROCO	16 I/O points		250	XW32-R025C
				300	XW32-R030C
				250	XW32-R025C
				300	XW32-R030C
ML connectors (48 pins)	Cables with Connectors (1-2) XMS2-ROCO-C-D1, XMS2-ROCO-C-D1	32 I/O points		(A) 500 (B) 250	XW32-R0500C-25-D1
				(A) 750 (B) 500	XW32-R0750C-50-D1
				(A) 1,000 (B) 750	XW32-R01000C-75-D1
				(A) 1,500 (B) 1,250	XW32-R01500C-125-D1
				(A) 2,000 (B) 1,750	XW32-R02000C-175-D1
				(A) 3,000 (B) 2,750	XW32-R03000C-275-D1
				(A) 5,000 (B) 4,750	XW32-R05000C-475-D1
				(A) 500 (B) 250	XW32-R0500C-25-D1
				(A) 750 (B) 500	XW32-R0750C-50-D1
				(A) 1,000 (B) 750	XW32-R01000C-75-D1
				(A) 1,500 (B) 1,250	XW32-R01500C-125-D1
				(A) 2,000 (B) 1,750	XW32-R02000C-175-D1
				(A) 3,000 (B) 2,750	XW32-R03000C-275-D1
				(A) 5,000 (B) 4,750	XW32-R05000C-475-D1

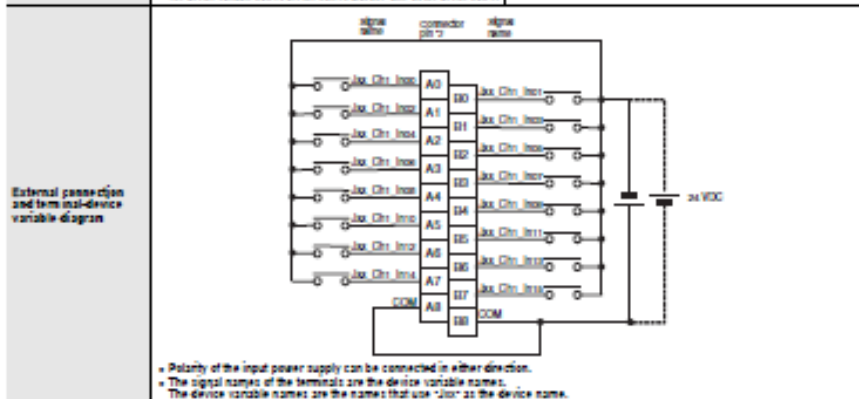
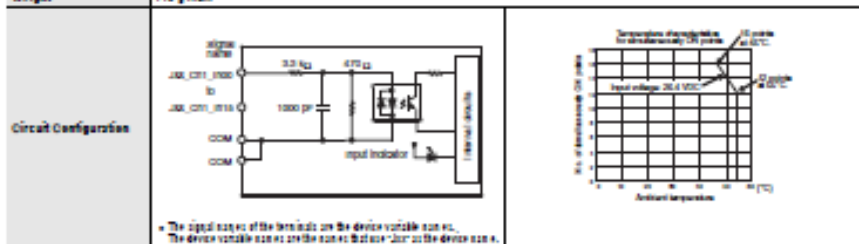
Note: Refer to the Datasheet for the XMS2-R Cables for I/O Relay Terminals (Cat. No. G120).

Mountable Racks

Model	NJ system		CJ system (CJ1, CJ2)		CP1H PLC	MLJ system	
	CPU Rack	Expansion Rack	CPU Rack	Expansion Backplane		MLJ Controller	Expansion Backplane
CJ1W-ID001	10 Units	10 Units (per Expansion Rack)	10 Units	10 Units (per Expansion Backplane)	Not supported	Not supported	10 Units (per Expansion Backplane)
CJ1W-ID011							
CJ1W-ID012							
CJ1W-ID021							
CJ1W-ID022							
CJ1W-ID023							
CJ1W-ID027							
CJ1W-ID028							
CJ1W-ID029							
CJ1W-ID031							
CJ1W-IA111							

CJ1W-ID211 DC Input Unit (24 VDC, 16 Points)

Name	16-point DC Input Unit with Terminal Block
Model	CJ1W-ID211
Rated Input Voltage	24 VDC
Rated Input Voltage Range	20.4 to 28.4 VDC
Input Impedance	3.3 kΩ
Input Current	7 mA typical (at 24 VDC)
ON Voltage/ON Current	14.4 VDC min./3 mA min.
OFF Voltage/OFF Current	5 VDC max./1 mA max.
ON Response Time	0.8 ms max. (Can be set to between 0 and 30 ms in the Setup.) *1
OFF Response Time	0.8 ms max. (Can be set to between 0 and 30 ms in the Setup.) *1
Number of Circuits	16 (16 points/common, 1 circuit)
Number of Significantly ON Points	100% simultaneously ON (at 24 VDC) (Refer to the following illustration.)
Insulation Resistance	20 MΩ min. between external terminals and the GND terminal (100 VDC)
Dielectric Strength	1,000 VAC between the external terminals and the GND terminal for 1 minute at a leakage current of 10 mA max.
Internal Current Consumption	60 mA max.
Weight	110 g max.



- The ON response time will be 20 μs maximum and OFF response time will be 200 μs maximum even if the response time are set to 0 ms due to internal element delays.
- Terminal numbers A0 to A0 and B0 to B0 are used in the external connection and terminal-device variable diagrams. They are not printed on the Units.

A-14. Datasheet Output

CJ-series Output Units **CJ1W-OC/OA/OD**

CSM_CJ1W-OUTPUT_DS_C_4_1

A Wide Range of Basic Output Units for High Speed Output and Different Applications

- These Output Units receive the results of output instructions from the CPU Unit and perform ON/OFF control for external devices.
- High-speed Output models CJ1W-OD213 and CJ1W-OD234 can help to increase system throughput.



CJ1W-OD213



CJ1W-OD234

Features




- High-speed output models are available, making versatile applications.
ON Response Time: 15 μ s, OFF Response Time: 80 μ s
 - Output Units are available with any of three output types: relay contact outputs, triac outputs, or transistor outputs.
 - For transistor outputs, select from sinking outputs or sourcing outputs.
 - Output Units with load short-circuit protection are also available.^{*1}
 - Select the best interface for each application: Fujitsu connectors or MIL connectors.^{*2}
 - A wide variety of Connector-Terminal Block Conversion Units are available to allow you to easily wire external output devices.
- ^{*1}. The following Units have load short-circuit protection: CJ1W-OC232, CJ1W-OD204, CJ1W-OD212, and CJ1W-OD232.
^{*2}. Available for models with 32 outputs or 64 outputs.

Ordering Information

International Standards

- The standards are abbreviated as follows: UL: UL, UL: UL (Class I Division 2 Products for Hazardous Locations), C: CSA, UC: cULus, UC1: cULus (Class I Division 2 Products for Hazardous Locations), CUL: cUL, N: NR, L: Lloyd, and CE: EC Directive.
- Contact your OMRON representative for further details and applicable conditions for these standards.

Output Units

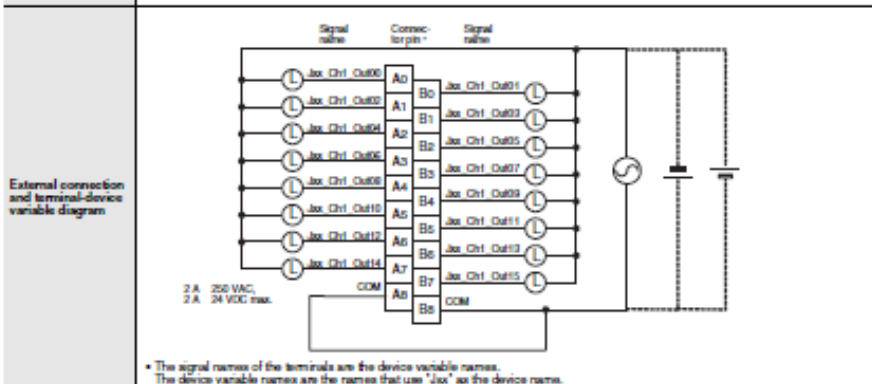
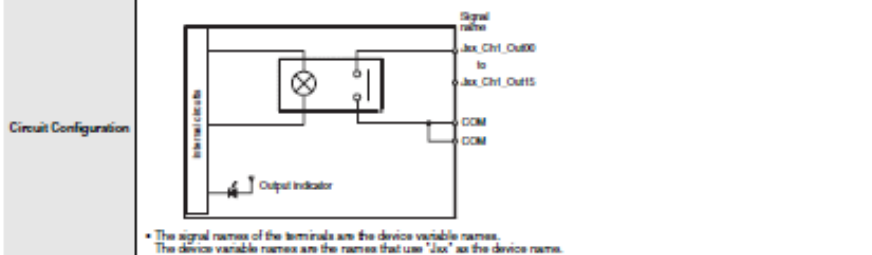
Unit type	Product name	Specifications					No. of words allocated	Current consumption (A)		Model	Standards	
		Output type	IO points	Maximum switching capacity	Commons	External connection		5 V	24 V			
CJ1 Basic I/O Units	 Relay Contact Output Units	-	8 outputs	250 VAC/24 VDC, 2 A	Independent 1 contacts	Removable terminal block	1 words	0.00	0.040 max.	CJ1W-OC201	UC1, N, L, CE	
		-	16 outputs	250 VAC/24 VDC, 2 A	16 points, 1 common	Removable terminal block	1 words	0.11	0.096 max.	CJ1W-OC211		
	 Triac Output Unit	-	8 outputs	250 VAC, 0.6 A	8 points, 1 common	Removable terminal block	1 words	0.22	-	CJ1W-OA201		
	 Transistor Output Units	Sinking	8 outputs	12 to 24 VDC, 2 A	4 points, 1 common	Removable terminal block	1 words	0.00	-	CJ1W-OD201		
		Sinking	8 outputs	12 to 24 VDC, 0.5 A	8 points, 1 common	Removable terminal block	1 words	0.10	-	CJ1W-OD203		
		Sinking	16 outputs	12 to 24 VDC, 0.5 A	16 points, 1 common	Removable terminal block	1 words	0.10	-	CJ1W-OD211		
		Sinking	16 outputs (High speed)	24 VDC, 0.5 A	16 points, 1 common	Removable terminal block	1 words	0.15	-	CJ1W-OD213		N, L, CE
		Sinking	32 outputs	12 to 24 VDC, 0.5 A	16 points, 1 common	Fujitsu connector	2 words	0.14	-	CJ1W-OD231		UC1, N, L, CE
		Sinking	32 outputs	12 to 24 VDC, 0.5 A	16 points, 1 common	MIL connector	2 words	0.14	-	CJ1W-OD233		
		Sinking	32 outputs (High speed)	24 VDC, 0.5 A	16 points, 1 common	MIL connector	2 words	0.22	-	CJ1W-OD234		N, L, CE
		Sinking	64 outputs	12 to 24 VDC, 0.3 A	16 points, 1 common	Fujitsu connector	4 words	0.17	-	CJ1W-OD251		UC1, N, L, CE
		Sinking	64 outputs	12 to 24 VDC, 0.3 A	16 points, 1 common	MIL connector	4 words	0.17	-	CJ1W-OD253		
		Sourcing	8 outputs	24 VDC, 2 A Short-circuit protection	4 points, 1 common	Removable terminal block	1 words	0.11	-	CJ1W-OD202		UC1, N, L, CE
		Sourcing	8 outputs	24 VDC, 0.5 A Short-circuit protection	8 points, 1 common	Removable terminal block	1 words	0.10	-	CJ1W-OD204		
		Sourcing	16 outputs	24 VDC, 0.5 A Short-circuit protection	16 points, 1 common	Removable terminal block	1 words	0.10	-	CJ1W-OD212		
Sourcing	32 outputs	24 VDC, 0.5 A Short-circuit protection	16 points, 1 common	MIL connector	2 words	0.15	-	CJ1W-OD232				
Sourcing	64 outputs	12 to 24 VDC, 0.3 A	16 points, 1 common	MIL connector	4 words	0.17	-	CJ1W-OD252				

Accessories

Connectors are not included for models with connectors. Either use one of the applicable connector listed below or use an applicable Connector-Terminal Block Conversion Unit or I/O Relay Terminal. For details on wiring methods, refer to [External Interface](#).

CJ1W-OC211 Contact Output Unit (16 Points)

Name	16-point Contact Output Unit with Terminal Block
Model	CJ1W-OC211
Max. Switching Capacity	2 A 250 VAC (cosφ = 1), 2 A 250 VAC (cosφ = 0.4), 2 A 24 VDC (0 A/Unit)
Min. Switching Capacity	1 mA 5 VDC
Relays	NY-24W-K-IE (Fujiel; Takamizawa Components, Ltd.). Cannot be replaced.
Service Life of Relay	Electrical: 150,000 operations (24 VDC, resistive load)/ 100,000 operations (250 VAC, cosφ = 0.4, inductive load) Mechanical: 25,000,000 operations Service life will vary depending on the connected load.
ON Response Time	15 ms max.
OFF Response Time	15 ms max.
Number of Circuits	16 points/common, 1 circuit
Insulation Resistance	20 MΩ between external terminals and the GR terminal (500 VDC)
Dielectric Strength	2,000 VAC between the external terminals and the GR terminal for 1 minute at a leakage current of 10 mA max.
Internal Current Consumption	110 mA 5 VDC max. 96 mA 24 VDC max. (6 mA × No. of ON points)
Weight	170 g max.



* Terminal numbers A0 to A8 and B0 to B8 are used in the external connection and terminal-device variable diagrams. They are not printed on the Units.

A-15. Power Supply unit

CJ-series Power Supply Unit CJ1W-PA/PD

CJ1W-PA/PD, PD, DR, S, A, J

Equipped with the RUN output for checking the operation status, as well as the replacement notification function for easy maintenance.

Lineup of five models including the AC power supply (25W).



CJ1W-PA205C



CJ1W-PA202

- Stable power supply is available from the CJ-series CPU Unit to each I/O Units via the dedicated bus. The most suitable Power Supply Unit can be selected among various I/O specifications and/or options.

Features





- The replacement notification function can prevent an overflow of the system due to the power lifetime. (CJ1W-PA205C only)
- Power Supply Unit provides wide variations according to the system scale up to maximum 25 W.
- Power Supply Unit provides wide variations according to the power supply (100 to 240 V AC/24 V DC) status.
- The output contact during operation enables you to check the CPU operation. (CJ1W-PA205R only)
- Conforming to the SEMI-F47 Standard. *
- * AC input type, 200 V min. input.

Ordering Information

International Standards

- The standards are abbreviated as follows: U: UL, U1: UL(Class I Division 2 Products for Hazardous Locations), C: CSA, UC: cULus, UC1: cULus (Class I Division 2 Products for Hazardous Locations), CU: cUL, N: NK, L: Lloyd, and CE: EC Directives.
- Contact your OMRON representative for further details and applicable conditions for these standards.

Power Supply Units

Product name		Power supply voltage	Output capacity			Options			Model	Standards
			5-VDC output capacity	24-VDC output capacity	Total power consumption	24-VDC service power supply	RUN output	Maintenance forecast monitor		
AC Power Supply Unit		100 to 240 VAC	5 A	0.8 A	25 W	No	No	Yes	CJ1W-PA205C	UL1, N, L, CE
	Yes						No	CJ1W-PA205R		
			2.8 A	0.4 A	14 W		No	No	No	
DC Power Supply Unit		24 VDC	5 A	0.8 A	25 W	No	No	CJ1W-PD025		
			2 A	0.4 A	19.6 W	No	No	CJ1W-PD022	UL1, CE	

Note: This unit cannot be used with the Machine Automation Controller NJ-series.

Accessories

There is no accessory for the CJ series Power Supply Unit.

Specifications

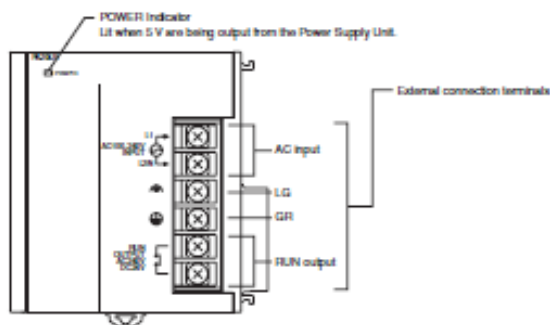
Item	Specifications				
	CJ1W-PA200R	CJ1W-PA200C	CJ1W-PA202	CJ1W-PD02S	CJ1W-PD02Z
Supply voltage	100 to 240 V AC (wide-range), 50/60 Hz				24 VDC
Operating voltage and frequency ranges	85 to 264 V AC, 47 to 63 Hz			19.2 to 28.8 V DC	21.6 to 26.4 V DC
Power consumption	100 VA max.		50 VA max.	50 W max.	35 W max.
Inrush current *1	At 100 to 120 V AC: 15 A/6 ms max. for cold start at room temperature At 200 to 240 V AC: 30 A/6 ms max. for cold start at room temperature			At 100 to 120 V AC: 20 A/6 ms max. for cold start at room temperature At 200 to 240 V AC: 40 A/6 ms max. for cold start at room temperature	
Output capacity *7	5.0 A, 5 V DC (including supply to CPU Unit)		2.8 A, 5 V DC (including supply to CPU Unit)	5.0 A, 5 V DC (including supply to CPU Unit)	2.0 A, 5 V DC (including supply to CPU Unit)
	0.8 A, 24 V DC		0.4 A, 24 V DC	0.8 A, 24 V DC	0.4 A, 24 V DC
	Total: 25 W max.		Total: 14 W max.	Total: 25 W max.	Total: 19.6 W max.
Output terminal (service supply)	Not provided.				
RUN output *2	Contact configuration: SPST-NO Switch capacity: 250 V AC, 2 A (resistive load) 120 V AC, 0.5 A (inductive load), 24 V DC, 2 A (resistive load) 24 V DC, 2 A (inductive load)	Not provided.			
Replacement notification function	Not provided.	With Alarm output (open-collector output) 30 V DC max., 50 mA max.	Not provided.		
Insulation resistance	20 M Ω min. (at 500 V DC) between AC external and GR terminals *3	<ul style="list-style-type: none"> • 20 MΩ min. (at 500 V DC) between all external terminals and GR terminal *3, and between all alarm output terminals. • 20 MΩ min. (at 250 V DC) between all alarm output terminals and GR terminal *3. 		20 M Ω min. (at 500 V DC) between AC external and GR terminals *3	20 M Ω min. (at 500 V DC) between DC external and GR terminals *3
Dielectric strength *4	2,300 V AC 50/60 Hz for 1 min between AC external and GR terminals *3 Leakage current: 10 mA max.	<ul style="list-style-type: none"> • 2,300 V AC, 50/60 Hz for 1 min between all external terminals and GR terminal *3 and between all alarm output terminals with a leakage current of 10 mA max. • 1,000 V AC, 50/60 Hz for 1 min between all alarm output terminals and GR terminal *3 with a leakage current of 10 mA max. 		2,300 V AC 50/60 Hz for 1 min between AC external and GR terminals *3 Leakage current: 10 mA max.	1,000 V AC, 50/60 Hz for 1 min between DC external and GR terminals *3 Leakage current: 10 mA max.
	1,000 V AC, 50/60 Hz for 1 minute between DC external and GR terminals *3 Leakage current: 10 mA max.				
Noise immunity	2 kV on power supply line (conforming to IEC61000-4-4)				
Vibration resistance	Conforms to IEC60068-2-6 5 to 8.4 Hz with 3.5-mm amplitude, 8.4 to 150 Hz Acceleration of 0.8 m/s ² for 100 min in X, Y, and Z directions (10 sweeps of 10 min each = 100 min)				
Shock resistance	Conforms to IEC60068-2-27 147 m/s ² , 3 times in X, Y, and Z directions (100 m/s ² for Relay Output Units)				
Ambient operating temperature	0 to 55°C				
Ambient operating humidity	10% to 90% (with no condensation) *5		10% to 90% (with no condensation)		
Atmosphere	Must be free from corrosive gases.				
Ambient storage temperature	-20 to 70°C (excluding battery)		-20 to 75°C *5		
Grounding	Less than 100 Ω				
Enclosure	Mounted in a panel.				
Weight	All models are each 5 kg max.				

Item	Specifications				
	CJ1W-PA205R	CJ1W-PA205C	CJ1W-PA202	CJ1W-PD025	CJ1W-PD022
Model					
CPU Rack dimensions	90.7 to 490.7 x 90 x 65 mm (W x H x D) (not including cables) Note: $W = a + b + 20 \times n + 31 \times m + 14.7$ a : Power Supply Unit: PA205R and PA205C = 80; PA202 = 45; PD025 = 60; PD022=27 b : CPU Unit: CJ1-H or CJ1 = 62; CJ1M-CPU11C = 31; CJ1M-CPU11T = 49 The total width is given by the following: $W = 156.7 + a \times 20 + m \times 31$, where n is the number of 32-point I/O Units or I/O Control Units and m is the number of other Units.				
Safety measures	Conforms to CE/UL and EC Directives.				

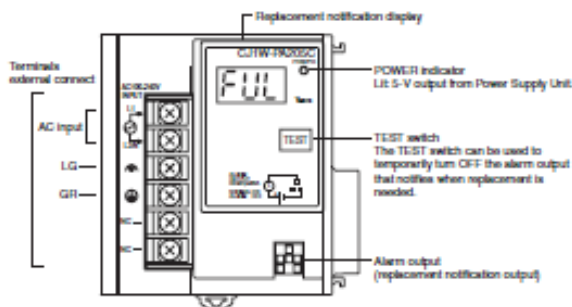
1. Disconnect the Power Supply Units LG terminal from the GR terminal when testing insulation and dielectric strength. Testing the insulation and dielectric strength with the LG terminal and the GR terminals connected will damage internal circuits in the CPU Unit.
2. Supported only when mounted to CPU Rack.
3. The inrush current is given for a cold start at room temperature. The inrush control circuit uses a thermistor element with a low-temperature current control characteristic. If the ambient temperature is high or the PLC is hot-started, the thermistor will not be sufficiently cool, and the inrush currents given in the table may be exceeded by up to twice the given values. When selecting fuses or breakers for external circuits, allow sufficient margin in shut-off performance.
4. Maintain an ambient storage temperature of -25 to 30°C and relative humidity of 25% to 70% when storing the Unit for longer than 3 months to keep the replacement notification function in optimum working condition.
5. Change the applied voltage gradually using the adjuster on the Tester. If the full dielectric strength voltage is applied or turned OFF using the switch on the Tester, the generated impulse voltage may damage the Power Supply Unit.
6. CJ1W-PD022 is not insulated between the primary DC power and secondary DC power.
7. Internal components in the Power Supply Unit will deteriorate or be damaged if the Power Supply Unit is used for an extended period of time exceeding the power supply output capacity or if the outputs are shorted.

External Interface

CJ1W-PA205R



CJ1W-PA205C



AC Input

Supply 100 to 240 V AC (allowable: 85 to 264 V AC). (Voltage selection is not required.)

DC Input

Supply 24 V DC.

Model	Allowable power supply voltage fluctuation range
CJ1W-PD025	19.2 to 28.8 VDC ($\pm 20\%$)
CJ1W-PD022	21.6 to 26.4 VDC ($\pm 10\%$)

LG

Ground to a resistance of 100 Ω or less to increase noise resistance and avoid electric shock.

GR

Ground to a resistance of 100 Ω or less to avoid electric shock.

RUN Output (CJ1W-PA205R Only)

The internal contact turns ON when the CPU Unit is operating (RUN or MONITOR mode). The Power Supply Unit must be in the CPU Rack to use this output.

Alarm Output (CJ1W-PA205C Only)

The alarm output is used to notify when Power Supply Unit replacement is required. The output is normally ON. The output turns OFF when the time until replacement is 6 months or less.

Wiring**Wire****For AC/DC power supply**

Recommended wire size	AWG14 to 20 (0.517 to 2.08mm ²)
-----------------------	--

For grounding wire

Recommended wire size	2 mm ² min.
-----------------------	------------------------

For alarm output

Recommended wire size	Use	Pushing strength (clamping operation)	Pulling strength (holding force)	Length of stripped section
AWG 22 to 18 (0.32 to 0.82 mm ²)	Connecting to PLC terminal block models	30 N max.	30 N min.	7 to 10 mm
	Connecting to PLC connector models		10 N min.	

Example: Calculating Total Current and Power Consumption

Example: When the Following Units are Mounted to a CJ-series CPU Rack Using a CJ1W-PA202 Power Supply Unit

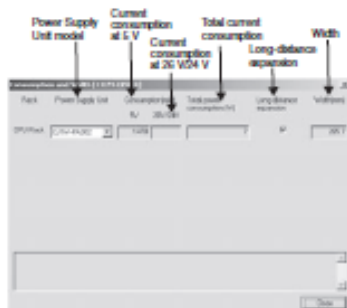
Unit type	Model	Quantity	Voltage group	
			5 V	24 V
CPU Unit	CJ1M-CPU13	1	0.580 A	–
I/O Control Unit	CJ1W-IC101	1	0.020 A	–
Basic I/O Units (Input Units)	CJ1W-ID211	2	0.080 A	–
	CJ1W-ID231	2	0.090 A	–
Basic I/O Units (Output Units)	CJ1W-OC201	2	0.090 A	0.048 A
Special I/O Unit	CJ1W-DA041	1	0.120 A	–
CPU Bus Unit	CJ1W-CLK23	1	0.350 A	–
Current consumption	Total		$0.580 + 0.020 + 0.080 \times 2 + 0.090 \times 2 + 0.090 \times 2 + 0.120 + 0.350$	$0.048 \text{ A} \times 2$
	Result		1.59 A ($\leq 2.8 \text{ A}$)	0.096 A ($\leq 0.4 \text{ A}$)
Power consumption	Total		$1.59 \times 5 \text{ V} = 7.95 \text{ W}$	$0.096 \text{ A} \times 24 \text{ V} = 2.304 \text{ W}$
	Result		$7.95 + 2.304 = 10.254 \text{ W} (\leq 14 \text{ W})$	

Note: For details on Unit current consumption, refer to Ordering Information.

Using the CX-Programmer to Display Current Consumption and Width

CPU Rack and Expansion Rack current consumption and width can be displayed by selecting Current Consumption and Width from the Options Menu in the CS/CJ/CP Table Window. (The width can be displayed for the CJ/CP Series only.) If the capacity of the Power Supply Unit is exceeded, it will be displayed in red characters. For details, refer to the CX-Programmer Operation Manual (Cat. No. W446).

Example:



A-16. RFID

UHF Integrative RFID Reader with TCPIP/RJ45



Overview

UHF Integrative Reader is a high performance device which can read and write all UHF tags or cards in a long distance. It is simple to install and use. Now it is widely applied in many RFID applications, like parking and access control.

Main Features

- Compatibility: ISO18000-6B, ISO18000-6C(EPC Gen 2) protocol;
- Frequency Band: 902-928MHz or 865-868MHz;
- RF Power: 0-30dbm adjustable;
- Interface: TCPIP/RJ45, RS232/UART, RS485, Weigand26/34;
- Read Range: 1-6meters depends on tag and environment.
- Provide free demo for testing and free SDK for development;

Applications

- Intelligent traffic management such as ETC, custom clearance, parking lot, and automatic weighing
- Logistics such as container management, pallet management, and asset tracking
- Ticketing
- Access control

Specifications

Model No.	CT-I809
Type	UHF Integrative RFID Reader
Working frequency	902-928Mhz or 865-868Mhz
Working mode	FHSS hopping and fixed frequency
Support protocol	ISO18000-6C/EPC Gen2 ,ISO18000-6B
Transmit power	0 ~ 30dBm adjustable
Antenna mode	Built-in 8dbi polarization antenna
Read &Write distance	read 1-6m, write 1-3m, tag and environment dependent
Data Interface	TCP/IP/RJ45, RS232/UART, RS485, Weigand26/34
Read prompt	Buzzer
Power supply	single +9V DC
Unit size	235mmx235mmx57mm
Net weight	900g
Operating temperature	-10 ~ +60°C
Storage temperature	-25 ~ +80°C
Humidity	5%~95%(non-condensing)
Material	ABS Waterproof

Cable Connector

Color	Type
Red	+9V
Black	GND
Yellow	Weigand DATA0
Blue	Weigand DATA1
Purple	RS485 R+
Orange	RS485 R-
Brown	GND
White	RS232 RXD
Green	RS232 TXD
Grey	Trigger input (TTL level)

A-17. Tag RFID



TECHNOLOGY HIGHLIGHTS:

- High quality, high performance ISO card
- UHF EPC Class 1 Gen2 and ISO 18000-6C
- 128 bit EPC number, 512 bit user memory
- Worldwide UHF frequency coverage
- MIFARE™ DESFire™ EV1 is NFC Tag Type 4 compliant
- Optional 3-Tech LF, HF, UHF

APPLICATION AREAS:

- Hands free or parking access
- Closed payment systems
- NFC applications
- Access control
- Loyalty schemes

SPECIFICATIONS

	ISO Card UHF or UHF / MIFARE
Base Model Number	6A7401 (UHF), 6C3401 (UHF + MIFARE DESFire EV1 4K) 6C2401 (UHF + MIFARE Classic EV1 1K)
Dimensions	ISO card: 3.4 x 2.1 x 0.03 in (85.6 x 54 x 0.76 mm)
Material	PVC (default, others on request)
Color	White
Chip Type	Monza 4QT + (optional MIFARE DESFire EV1 4K or MIFARE EV1 1K)
Memory	UHF: 128 bit EPC, 512 bit user memory, 96 bit TID MIFARE DESFire: 4KB (optional: 256B, 2KB, 8KB), MIFARE EV1 1KB (optional: 4KB)
Operating Frequency	860-960 MHz (Worldwide) + 13.56 MHz (MIFARE)
Standards	UHF: EPC Class 1 Gen 2, ISO 18000-6C MIFARE DESFire: ISO 14443 / NFC Tag Type 4 (if NDEF formatted); ISO 10373, ISO 7816-1
Anti Collision	Yes
CHEMICAL AND MECHANICAL RESISTANCE	
Water IP68	68° F / 20° C (1m, 24h)
Resistances	Acetic acid water, artificial perspiration, carbonated sodium water, ethylene glycol, fuel B, salt mist, salt water, sugared water; humidity 95% at 122° F (50° C), 24 h
Storage Temperature	-31° to +122° F (-35° to +50° C)
Operating Temperature	-31° to +122° F (-35° to +50° C)
Shock / Fatigue Temperature	-31° to +176° F (-35° to +80° C), 50 cycles, 5 min soaking time, 20 sec transition
Force	Dynamic bending and torsion, 4 x 250 bends
Reading Distance	UHF up to 39 ft / 12 m (ZW reader ERP, free space)
Options	Artwork; encoding; laser engraving; magnetic stripe; other contactless, contact- or memory-chips; signature panel and optical security features; UHF only cards support an optional horizontal punch slot
Prelaminates	For card manufacturers this technology is alternatively available in form of 410 µm (16 mil) prelinates



“Halaman ini sengaja dikosongkan”

RIWAYAT PENULIS



Penulis bernama lengkap Fredrico Aditrio Pohan lahir di Medan Sumatera Utara pada tanggal 16 Januari 1998. Penulis merupakan anak ketiga dari tiga bersaudara. Penulis bertempat tinggal di JL. Jawa Indah II no 51 GKB Gresik. Penulis menyelesaikan pendidikan dasar SD Negeri Petrokimia Gresik lulus pada tahun 2009, SMP Semen Gresik lulus pada tahun 2012, SMA Semen Gresik lulus pada tahun 2015 dan masih memulai masa kuliah di Departemen Teknik Elektro Otomasi Institut Teknologi Sepuluh Nopember pada tahun 2015