

BAB II

KERANGKA TEORITIS DAN KERANGKA BERPIKIR

2.1 Kerangka teoritis

2.1.1 PLC (*Programmable Logic Controller*)

Sebuah PLC (*Programmable Logic Controller*) adalah sebuah alat yang digunakan untuk menggantikan rangkaian sederetan *relay* yang ada pada sistem kontrol konvensional. PLC bekerja dengan cara mengamati masukan (melalui sensor), kemudian melakukan proses dan melakukan tindakan sesuai yang dibutuhkan, berupa menghidupkan atau mematikan keluaran. Program yang digunakan adalah berupa *ladder diagram* yang kemudian harus dijalankan oleh PLC. Dengan kata lain PLC menentukan aksi apa yang harus dilakukan pada *instrument* keluaran yang berkaitan dengan status suatu ukuran atau besaran yang diamati. Proses yang dikontrol ini dapat berupa regulasi variabel secara *continue* seperti pada sistem-sistem servo, atau hanya melibatkan kontrol dua keadaan (*on/off*) saja, tetapi dilakukan secara berulang-ulang seperti umum dijumpai pada mesin pengeboran, sistem konveyor dan lain sebagainya.

PLC secara bahasa berarti pengontrol logika yang dapat diprogram, tetapi pada kenyataannya, PLC secara fungsional tidak lagi terbatas pada fungsi-fungsi logika saja. Sebuah PLC dewasa ini juga dapat melakukan perhitungan-perhitungan aritmatika yang relatif kompleks, fungsi komunikasi, dokumentasi dan lain sebagainya. PLC banyak digunakan pada aplikasi-aplikasi industri, misalnya pada proses pengepakan, perakitan otomatis dan lain-lain. Hampir semua aplikasi kontrol listrik membutuhkan PLC. Alasan utama perancangan PLC

adalah untuk menghilangkan beban ongkos perawatan dan penggantian sistem kontrol mesin berbasis *relay*. Adapun ciri atau karakteristik PLC memiliki beberapa aspek sebagai berikut :

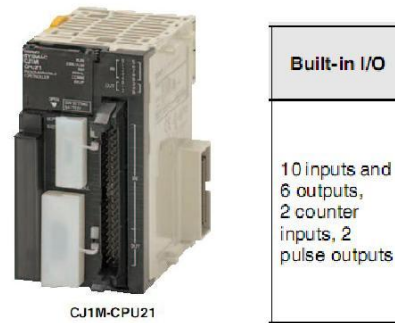
- a. PLC sebenarnya suatu sistem berbasis mikroprosesor yang memiliki fungsi-fungsi dan fasilitas utama dari sebuah mikro komputer.
- b. PLC diprogram melalui *programming unit* yang bisa berupa terminal komputer dengan VDU (*Video Display Unit*) dan *keyboard* atau dengan terminal portable khusus (mirip kalkulator dengan tampilan LCD). Pada saat ini PLC dapat diprogram melalui PC.
- c. PLC mengontrol suatu alat berdasarkan status masukan/keluaran suatu alat dan program.

Sehingga pengertian PLC yang awalnya berfungsi menggantikan peran *relay*, dapat diartikan sesuai kata penyusunnya adalah sebagai berikut :

- a. *Programmable* yaitu menunjukkan kemampuannya yang dapat dengan mudah diubah-ubah sesuai program yang dibuat dan kemampuannya dalam hal memori program yang telah dibuat.
- b. *Logic* yaitu menunjukkan kemampuannya dalam memproses *input* secara aritmatik (ALU) dengan melakukan proses membandingkan, menjumlahkan, mengkalikan, membagi, dan mengurangi.
- c. *Controller* yaitu menunjukkan kemampuannya dalam mengontrol dan mengatur proses sehingga menghasilkan *output* yang diinginkan.

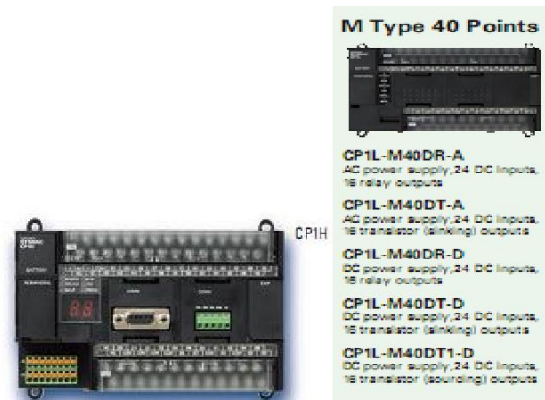
Selanjutnya berdasarkan jumlah *input/output* yang dimilikinya ini, secara umum PLC dapat dibagi menjadi tiga kelompok besar yaitu :

- a. PLC mikro, PLC dapat dikategorikan mikro jika jumlah *input/output* pada PLC ini kurang dari 32 terminal. Berikut ini adalah gambar salah satu jenis PLC mikro yang ditunjukkan pada gambar 2.1 di bawah ini.



Gambar 2.1 PLC *mikro* OMRON *type* CJ1M.¹

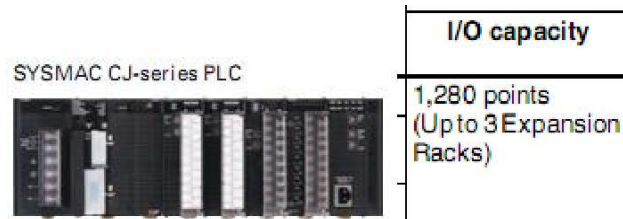
- b. PLC *mini*, kategori ukuran mini adalah jika PLC tersebut memiliki jumlah *input/output* antara 32 sampai 128 terminal. Berikut ini adalah gambar salah satu jenis PLC *mini* yang ditunjukkan pada gambar 2.2 di bawah ini.



Gambar 2.2 PLC *mini* OMRON *type* CP1L .

¹Katsuhiko Ogata, Belajar PLC Omron, terjemahan Ir. Edi Leksono, (Jakarta: Erlangga,1989), h. 4

- c. PLC *large*, PLC ukuran ini dikenal juga dengan PLC tipe *rack* dimana PLC dapat dikategorikan sebagai PLC besar jika jumlah *input/output* nya lebih dari 128 terminal. Berikut ini adalah gambar salah satu jenis PLC *mini* yang ditunjukkan pada gambar 2.3 di bawah ini.



Gambar 2.3 PLC *large* OMRON type CJ series

2.1.2 Fungsi PLC

PLC ini dirancang untuk menggantikan satu rangkaian *relay* sequensial dalam suatu sistem kontrol. Selain dapat diprogram, alat ini juga dapat dikendalikan, dan dioperasikan oleh orang yang tidak memiliki pengetahuan di bidang pengoperasian komputer secara khusus. PLC ini memiliki bahasa pemrograman yang mudah dipahami dan dapat dioperasikan bila program yang telah dibuat dengan menggunakan *software* yang sesuai dengan jenis PLC yang digunakan sudah dimasukkan.

Alat ini bekerja berdasarkan *input-input* yang ada dan tergantung dari keadaan pada suatu waktu tertentu yang kemudian akan meng-*on* atau meng-*off* kan *output-output*. PLC juga dapat diterapkan untuk pengendalian sistem yang memiliki *output* banyak. Banyak hal yang dapat dilakukan oleh PLC, yaitu :

2.1.2.1 Fungsi ini meliputi :

- a. Pengganti *relay control logic*.
- b. *Timers/counters*.
- c. Pengganti pengendali yang berupa papan rangkaian elektronik.
- d. Pengendali mesin dan proses.

2.1.2.2 Meliputi Fungsi-Fungsi :

- a. Operasi aritmatik (+, -, x, :).
- b. Penanganan informasi.
- c. Kontrol analog (suhu, tekanan, dan lain-lain).
- d. PID (*Propositional Integrator Differentiator*).
- e. Servo motor control.
- f. *Stepper motor* control.

2.1.2.3 Pengaplikasian Fungsi-fungsi :

- a. Proses monitor dan alarm.
- b. Monitor dan diagnosa kesalahan .
- c. Antarmuka dengan komputer (RS 232 / RCS 485).
- d. Antarmuka dengan *printer* / ASCII.
- e. Jaringan kerja otomasi pabrik.
- f. *Local Area Network*.
- g. *Wide Are Network*.²

² Ibid., h. 6

2.1.3 Kelebihan PLC

Sistem kontrol menggunakan PLC mempunyai banyak keuntungan dibandingkan sistem kontrol menggunakan peralatan kontrol yang dirangkai secara listrik seperti *relay* atau kontaktor yaitu :

- a. PLC didesain untuk bekerja dengan kehandalan yang tinggi dan jangka waktu pemakaian yang lama pada lingkungan industri.
- b. Jika sebuah aplikasi kontrol yang kompleks dan menggunakan banyak *relay*, maka akan lebih murah apabila kita menggunakan/memasang satu buah PLC sebagai alat kontrol.
- c. PLC dapat dengan mudah diubah-ubah dari satu aplikasi ke aplikasi lain dengan cara memprogram ulang sesuai yang kita inginkan.
- d. PLC dapat melakukan diagnosa dan menunjukkan kesalahan apabila terjadi gangguan sehingga ini sangat membantu dalam melakukan pelacakan gangguan.
- e. PLC juga dapat berkomunikasi dengan PLC lain termasuk juga dengan komputer. Sehingga kontrol dapat ditampilkan di layar komputer, didokumentasikan, dan gambar kontrol dapat didokumentasikan dan dicetak dengan menggunakan *printer*.
- f. Mudah dalam melakukan pelacakan gangguan kontrol.

PLC mempunyai kemampuan menggantikan logika dan pengerjaan sirkuit kontrol *relay* yang merupakan instalasi langsung. Rangkaian kontrol cukup dibuat secara *software*. Pengkabelan hanya diperlukan untuk menghubungkan peralatan *input* dan *output*. Hal ini mempermudah dalam mendesain dan memodifikasi rangkaian, karena cukup dengan mengubah program PLC.

2.1.4 Struktur Unit PLC

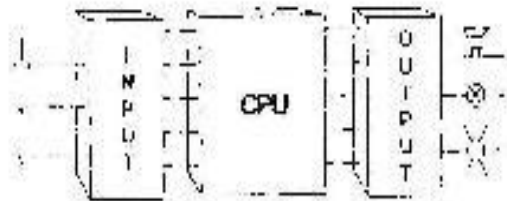
Secara umum PLC terdiri dari dua komponen utama (gambar 2.4) yaitu :

2.1.4.1 *Central Processing Unit (CPU)*

Unit processor atau CPU (*Central Processing Unit*) adalah unit yang berisi mikroprosesor yang mengolah sinyal-sinyal *input* dan melaksanakan pengontrolan, sesuai dengan program yang disimpan di dalam memori, lalu mengkomunikasikan keputusan-keputusan yang diambilnya sebagai sinyal-sinyal kontrol ke *interface output*. Fungsi CPU adalah mengatur semua proses yang terjadi di PLC. Ada tiga komponen utama penyusun CPU ini, yaitu *processor*, *memory*, dan *power supply*.

2.1.4.2 *Sistem Antarmuka Input/Output*

Pada umumnya informasi data pada PLC dinyatakan dalam bentuk tegangan listrik antara 5-15 VDC, sedangkan sistem tegangan di luar bervariasi antara 24-240 VDC maupun AC. Unit I/O dimaksudkan untuk *interfacing* antara besaran kedua tersebut. Adapun komponen utama PLC ditunjukkan pada gambar 2.4 di bawah ini.



Gambar 2.4 Blok diagram PLC

Konfigurasi fisik PLC terbagi menjadi dua yaitu sebagai berikut :

a. *Fixed*

Terdiri dari bagian *processor*, masukan dan keluaran, catu daya dalam satu unit.

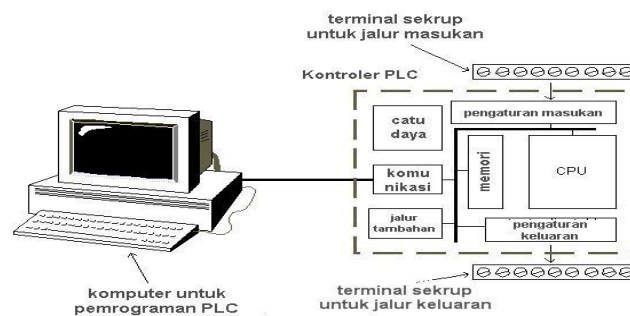
b. *Modular*

PLC modular terdiri dari *chassis* di mana catu daya, CPU dan semua modul masukan dan keluaran sebagai perangkat keras yang dapat dipasang dan dilepas secara terpisah. Untuk mengetahui blok diagram keseluruhan PLC ditunjukkan pada gambar 2.5. Dalam proses pengolahan data di dalam memori PLC, PLC dapat memberikan sinyal-sinyal *troubleshooting* melalui indikator lampu atau kondisi *error* pada *ladder diagram* pada PLC. Adapun *troubleshooting* dalam PLC adalah sebagai berikut :

a. *Faults indicators.*

b. *Run/stop indicators.*

c. *Input/output status indicators.*



Gambar 2.5 Blok diagram keseluruhan PLC

2.1.5 Data dan Memori PLC

Aturan penulisan memori PLC adalah :

- a. *Word* atau *channel* yang terdiri dari 16 bit, ditulis XXX.
- b. Bit atau *contact* yang terdiri dari 1 bit, ditulis XXXXX .
- c. Dua angka yang paling belakang (di garis bawah) menunjukkan nomor *contact* dan sisa angka yang di depan menunjukkan nomor *channel*.

Memori juga merupakan elemen yang terdapat pada CPU yang berupa IC (*integrated circuit*). Karakteristik memori ini mudah dihapus dengan meng-*off* kan catu daya. Seperti halnya sistem komputer, *memory* PLC terdiri atas RAM dan ROM. Kapasitas *memory* antara satu PLC dengan yang lain berbeda-beda tergantung pada *type* dan pabrik pembuatnya. Beberapa pabrik menyatakan ukuran *memory* dalam *byte*, ada juga yang *kilo byte*, dan ada pula yang dinyatakan dengan jumlah instruksi yang dapat disimpan :

1. *Random Acces Memory*

Random Acces Memory mempunyai singkatan kode RAM. Program yang ditulismemunya disimpan dalam RAM yang ada di dalam PLC sehingga dapat diubah/di-edit melalui *programming unit*. Kerugian penyimpanan di RAM adalah program dan data akan hilang ketika *power supply* mati. Untuk mengatasi hal ini, RAM dapat *diback-up* dengan *battery lithium*, sehingga meskipun *power supply* mati, program dan data tidak hilang. Umumnya bila *battery* tidak rusak, program dan data disimpan selama 5 tahun.

2. *Read Only Memory*

Read Only Memory mempunyai singkatan kode ROM. Semua data yang ada dapatdibaca, tetapi tidak dapat ditulisi, karena termasuk data *non volatile* yang

tersedia secara permanen. Supaya program dalam RAM bisa dieksekusi harus ada *operating system (PLC)*. *Operating system* ini dibuat oleh pabrik pembuat PLC yang disimpan dalam ROM dan hanya dapat dibaca oleh *processor*. Dalam beberapa PLC tidak menggunakan ROM tetapi EPROM atau EEPROM. Pengguna dapat juga menyimpan program di sebagian tempat di EEPROM atau dikenal sebagai *flash memory*.

Sehingga secara garis besar ada tiga fungsi *memory* yaitu untuk menyimpan informasi yang diperlukan untuk menjalankan program, untuk menyimpan program (*program storage*), untuk menyimpan pesan (*program message*). Memori PLC terdiri dari :

a. *Internal Relay*

Internal Relay mempunyai singkatan kode IR. *Relay* mempunyai pembagian fungsi seperti IR *input*, IR *output*, dan juga IR *work area* (untuk pengolahan data pada program). IR *input* dan IR *output* adalah IR yang berhubungan dengan terminal *input* dan *output* pada PLC. Sedangkan IR *work area* tidak dihubungkan ke terminal PLC, akan tetapi berada dalam *internal memory* PLC dan fungsinya untuk pengolahan logika program kiat (manipulasi program).

Ada juga IR yang difungsikan untuk *SYSMAC Bus Area*, *special I/O unit area*, *optical I/O unit area*, dan *group 2 high density I/O unit area*. *Sysmac bus area* berfungsi untuk komunikasi data PLC antara CPU PLC dan I/O unit PLC hanya menggunakan 2 kabel saja (RS 485), maksimal 200 meter. *Special I/O unit area* merupakan IR yang digunakan oleh *special I/O unit* PLC (contoh: *analog input*, *analog output* dan lain-lain), untuk mengatur, menyimpan dan

mengolah datanya. *Optical I/O Unit area* adalah IR yang digunakan untuk mengolah dan menyimpandata dari *optical I/O unit PLC.Group 2 high density I/O unit area* adalah IR untuk menyimpan dan mengolah data dari *high density I/O unit group 2*.

b. *Special Relay*

Special Relay mempunyai singkatan kode SR. *Special relay* adalah *relay* yang mempunyai fungsi-fungsi khusus seperti untuk *flags* (misalnya pada instruksi penjumlahan terdapat kelebihan digit pada hasilnya (*cary flags*)), kontrol bit PLC, informasi kondisi PLC, dan *system clock* (pulsa 1 detik, 0,2 detik, dan seterusnya).

c. *Auxiliary Relay*

Auxiliary Relay mempunyai singkatan kode AR. Terdiri dari flags dan bit untuktujuan-tujuan khusus. Dapat menunjukkan kondisi PLC yang disebabkan oleh kegagalan sumber tegangan, kondisi *special I/O*, kondisi *input/output unit*, kondisi CPU PLC, kondisi memori PLC dan lain-lain.

d. *Holding Relay*

Holding Relay mempunyai singkatan kode HR. Dapat difungsikan untk menyimpandata (bit-bit penting) karena tidak akan hilang walaupun sumber tegangan PLC mati.

e. *Link relay*

Link Relay mempunyai singkatan kode LR. Digunakan untuk data *link* pada PLC *link system*. Artinya untuk tukar menukar informasi antar dua PLC atau lebih dalam suatusistem kontrol yang saling berhubungan satu dengan yang lain dan menggunakan banyak PLC (minimum 2 PLC).

f. *Temporary Relay*

Temporary Relay mempunyai singkatan kode TR. Berfungsi untuk penyimpanan sementara kondisi logika program pada *ladder diagram* yang mempunyai titik percabangan khusus.

g. *Timer/Counter*

Timer relay mempunyai singkatan kode TR dan *counter relay* mempunyai singkatan kode CNT. Untuk mendefinisikan suatu sistem waktu tunda/*time delay (timer)* ataupun untuk penghitung (*counter*). Untuk *timer* mempunyai orde 100 ms, ada yang mempunyai orde 10 ms yaitu TIMH (15). Untuk TIM 000 s/d TIM 015 dapat dioperasikan secara *interrupt* untuk mendapatkan waktu yang lebih presisi.

h. *Data Memory*

Data memory disingkat dengan kode DM. *Data memory* berfungsi untuk penyimpanan data-data program karena isi DM tidak akan hilang (*reset*) walaupun sumber tegangan PLC mati. Macam-macam DM adalah sebagai berikut :

1. *DM Read/write*

Pada DM ini bisa dihapus dan ditulis oleh program yang kita buat. Jadi sangat berguna untuk manipulasi program.

2. *DM special I/O unit*

DM ini berfungsi untuk menyimpan dan mengolah hasil dari *special I/O* unit, mengatur, dan mendefinisikan sistem kerja *special I/O* unit.

3. *DM history log*

Pada DM disimpan informasi-informasi penting pada saat PLC terjadi kegagalan sistem operasionalnya. Pesan-pesan kesalahan sistem PLC yang disimpan adalah berupa kode-kode angka tertentu.

4. *DM link test area*

Berfungsi untuk menyimpan informasi-informasi yang menunjukkan status dari sistem *link* PLC.

5. *DM setup*

Berfungsi untuk *setup* kondisi *default* (kondisi kerja saat PLC aktif). Pada DM inilah kemampuan kerja suatu PLC didefinisikan untuk pertama kalinya sebelum PLC tersebut diprogram dan dioperasikan pada suatu sistem kontrol. Tentu saja *setup* PLC tersebut disesuaikan dengan sistem kontrol yang bersangkutan.

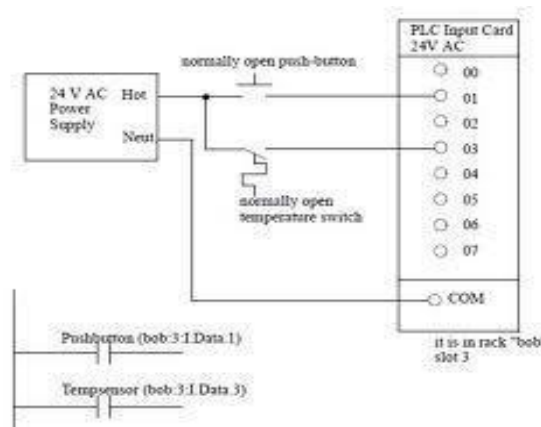
i. *Upper Memory*

Upper memory mempunyai singkatan kode yaitu UM. Memori ini berfungsi untuk menyimpan dan menjalankan program kita (*user program*). Kapasitasnya tergantung pada masing-masing tipe PLC yang dipakai.

DM tidak mempunyai *contact*, yang ada hanya *channel/word* saja. DM dapat difungsikan untuk penyimpanan data-data penting yang tidak boleh hilang waktu *power* padam, atau untuk manipulasi program kita. Memori yang sifatnya dapat menyimpan data program jika listrik mati adalah DM dan HR, sedangkan yang lain akan kembali *reset* (hilang). Pemrograman PLC ada dua macam yaitu dengan *ladder diagram* atau dengan mnemonic.

2.1.6 Power Supply PLC

Unit ini berfungsi untuk memberikan sumber daya pada PLC. Kebanyakan PLC bekerja dengan catu daya 24 VDC atau 220 VAC. Sumber tegangan yang dibutuhkan oleh CPU, memori dan rangkaian lain adalah sumber tegangan DC, umumnya untuk komponen digital diperlukan tegangan searah 5 volt. *Port power supply* PLC ditunjukkan pada gambar 2.6 di bawah ini.



Gambar 2.6 *Port power supply* PLC

2.1.7 I/O Modul

Modul *input* mempunyai beberapa fungsi diantaranya :

- a. Mendeteksi ketika sinyal diterima dari sensor.
- b. Mengkonversi sinyal *input* menjadi level tegangan yang bisa diterima *processor*.
- c. Mengirim sinyal ke *indikator input* PLC sehingga bisa diketahui *input* mana yang sedang menerima sinyal. Modul *output* mempunyai beberapa fungsi diantaranya :
 - a. *Output unit* pada PLC juga berfungsi sebagai *interface* terhadap peralatan luar.

- b. *Output* PLC bertindak sebagai *switch* terhadap *power supply* untuk mengoperasikan peralatan *output* (misal : *relay*, *solenoid valve* dan lain-lain).
- c. Komponen yang biasa dipakai PLC sebagai bagian *output unit* adalah *relay* untuk AC/DC, TRIAC untuk AC saja, dan transistor atau FET untuk DC saja.

2.1.8 *Programming Panel* / Peralatan Pemrograman

Piranti pemrograman menyediakan sarana *printer* sehingga pemakai dapat berkomunikasi dengan rangkaian kontrol yang dapat diprogram. Ini memungkinkan pemakai untuk meng-*enter*, meng-*edit* dan memonitor program dengan terhubung ke unit *processor* dan mengizinkan akses ke memori pemakai.

Terdapat tiga jenis *programmer*/monitor yang bisa digunakan :

- a. Jenis yang paling sederhana berukuran satu genggam tangan, bentuknya mirip sebuah kalkulator. Namun selain angka, pada *keypad* nya terdapat simbol-simbol untuk pemrograman grafik. Jika *keypad* ini dioperasikan ke mode monitor, operasi yang berlangsung pada PLC dapat kita amati.



Gambar 2.7 Unit *mini programmer* untuk memprogram PLC

- b. Pada beberapa PLC terdapat *keypad* yang sudah dilengkapi dengan monitor LCD, sehingga selain dari lampu indikator, kita juga dapat mengamatinya melalui layar LCD pada PLC tersebut.

- c. Jenis yang ketiga adalah dengan menggunakan IBM-PC beserta perangkat lunaknya. Simulasi program dapat dilakukan dengan PC, kemudian jika telah dianggap sesuai/benar dapat dijalankan ke *interface* PLC.

2.1.9 Dasar Pemrograman PLC

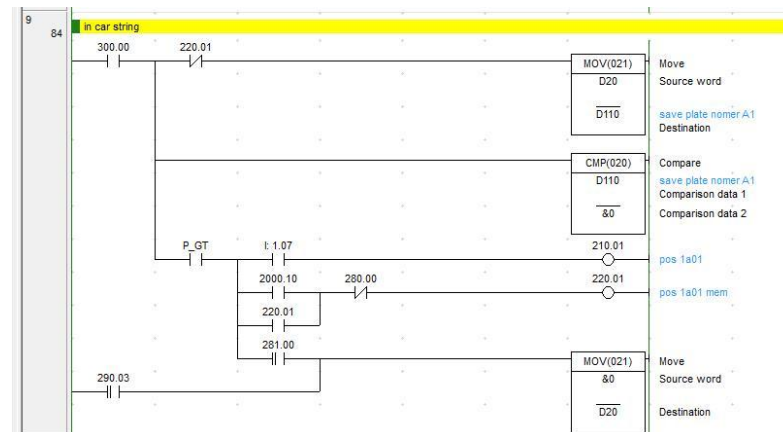
Pada dasarnya PLC tidak dapat melakukan apa-apa tanpa adanya program di dalam memori proses. Program PLC dimasukkan ke dalam memori dengan menggunakan peralatan pemrograman PLC yang sesuai, peralatan pemrograman PLC itu diantaranya :

- a. *Hand-held Unit*.
- b. Terminal video.
- c. Komputer pribadi/PC.

2.1.10 Ladder Diagram/Diagram Ladder

Ladder diagram terdiri dari garis vertikal yang disebut garis *bar*. Instruksi yang dinyatakan dengan simbol digambarkan dan disusun sepanjang garis horizontal dimulai dari kiri dan dari atas ke bawah.

Ladder diagram digunakan untuk menggambarkan rangkaian listrik dan dimaksudkan untuk menunjukkan urutan kejadian, bukan hubungan kabel antar komponen. Pada *ladder diagram* memungkinkan elemen-elemen elektrik dihubungkan sedemikian rupa sehingga keluaran (*output*) tidak hanya terbatas pada ketergantungan terhadap masukan (*input*) tetapi juga terhadap logika. Untuk mengetahui contoh *ladder diagram* dapat ditunjukkan pada gambar 2.8 pada halaman berikutnya.



Gambar 2.8 Contoh *ladder diagram*

Ladder languages merupakan bahasa pemrograman yang menuliskan instruksi kontrol secara grafis. Untuk menggambar *ladder language*/diagram ada beberapa ketentuan yang perlu diperhatikan yaitu :

- a. Daya mengalir dari kiri ke kanan.
- b. *Output* ditulis pada bagian yang paling kanan.
- c. Tidak ada kontak yang diletakkan di sebelah kanan *output*.
- d. Setiap *output* disisipkan satu kali dalam setiap program.

Ladder diagram memuat beberapa blok yang dapat mempresentasikan aliran program dan fungsi seperti :

1. *Contact*

Contact dapat berupa kontak *input* (sakelar, *push button*), kontak *internal variabel* (*relay* otomatis) dan lain-lain, ada 4 macam tipe kontak yaitu :

- a. Kontak *NO* (*Normally Open*) adalah kontak yang terdapat pada *ladder diagram* dimana pada saat keadaan sistem belum bekerja kondisi kontak dalam keadaan terbuka.

- b. Kontak *NC (Normally Close)* adalah kontak yang terdapat pada *ladder diagram* di mana pada saat keadaan sistem belum bekerja kondisi kontak dalam keadaan tertutup.
- c. Kontak *rising edge* adalah kontak yang terdapat pada *ladder diagram* di mana pada saat pada saat keadaan sistem mulai bekerja kondisi kontak berubah dari logika “0” menjadi logika “1”.
- d. Kontak *falling edge* adalah kontak yang terdapat pada *ladder diagram* di mana pada saat keadaan sistem mulai bekerja kondisi kontak berubah dari logika “1” menjadi logika “0”

2. Coil

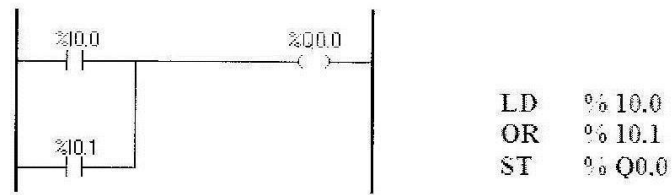
Coil secara umum menyatakan *output*, ada 4 macam tipe *coil* yaitu :

- a. *Coil*.
- b. *Negatif coil*.
- c. *SET coil*.
- d. *RESET coil*.

2.1.11 Instruction List Language

Instruction List Language merupakan penulisan program berbasis teks. Adapun ciri dari *instruction list* :

- a. Bahasa program dapat ditulis baris per baris.
- b. Masing-masing baris program merupakan instruksi yang dapat dimengerti dan dapat dieksekusi *controller*.
- c. *Instruction list* mirip dengan bahasa *assembler* pada mikroprocessor.



Ladder diagram

Instruction list language

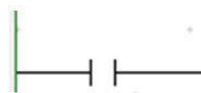
Gambar 2.9 Ladder diagram dan instruction list

2.1.11.1 Instruksi - Instruksi Dasar PLC

Semua instruksi (perintah program) yang ada di bawah ini, merupakan instruksi paling dasar pada PLC. Berikut ini merupakan instruksi-instruksi dasar pada PLC.

1. LOAD

- a. Instruksi *load* pada PLC mempunyai singkatan kode LD. Instruksi ini dibutuhkan jika urutan kerja (*sequence*) pada suatu sistem kontrol hanya membutuhkan satu kondisi *logic* saja dan sudah dituntut untuk mengeluarkan satu *output*.
- b. Logikanya seperti *contact NO relay*.
- c. *Ladder diagram* simbol *Load* ditunjukkan pada gambar 2.10 di bawah ini.

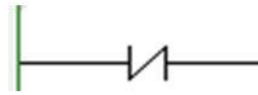
Gambar 2.10 Instruksi *Load*

- d. *Operand data area*

B (BIT) : IR, SR, AR, HR, TC, LR, TR.

2. LOAD NOT

- a. Instruksi *Load Not* pada PLC mempunyai singkatan kode LD NOT. Instruksi ini dibutuhkan jika urutan kerja (*sequence*) pada suatu sistem kontrol hanya membutuhkan satu kondisi *logic* saja dan sudah dituntut untuk mengeluarkan satu *output*.
- b. Logikanya seperti *contact NC relay*.
- c. *Ladder diagram* simbol *Load not* ditunjukkan pada gambar 2.11.



Gambar 2.11 Instruksi *Load Not*

- d. *Operand data area*

B (BIT) : IR, SR, AR, HR, TC, LR.

3. AND

- a. Instruksi *And* pada PLC mempunyai singkatan kode AND. Instruksi ini dibutuhkan jika urutan kerja (*sequence*) pada suatu sistem kontrol membutuhkan lebih dari satu kondisi *logic* yang harus terpenuhi semuanya untuk mengeluarkan satu *output*.
- b. Logikanya seperti *contact NO relay*.
- c. *Ladder diagram* simbol *And* ditunjukkan pada gambar 2.12 di bawah ini.



Gambar 2.12 Instruksi *And*

- d. *Operand data area*

B (BIT) : IR, SR, AR, HR, TC, LR.

4. AND NOT

- a. Instruksi *And Not* pada PLC mempunyai singkatan kode *And not*. Instruksi ini dibutuhkan jika urutan kerja (*sequence*) pada suatu sistem kontrol membutuhkan lebih dari satu kondisi *logic* yang harus terpenuhi semuanya untuk mengeluarkan satu *output*.
- b. Logikanya seperti *contact NC relay*.
- c. *Ladder diagram* simbol *And not* ditunjukkan pada gambar 2.13.



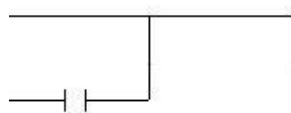
Gambar 2.13 Instruksi *And not*

- d. *Operand data area*

B (BIT) : IR, SR, AR, HR, TC, LR.

5. OR

- a. Instruksi *OR* pada PLC mempunyai singkatan kode *OR*. Instruksi ini dibutuhkan jika urutan kerja (*sequence*) pada suatu sistem kontrol hanya membutuhkan salah satu saja dari beberapa kondisi logika untuk mengeluarkan satu *output*.
- b. Logikanya seperti *contact NO relay*.
- c. *Ladder diagram* simbol *OR* ditunjukkan pada gambar 2.14.



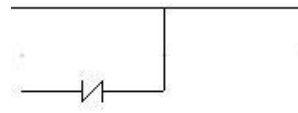
Gambar 2.14 Instruksi *OR*

- d. *Operand data area*

B (BIT) : IR, SR, AR, HR, TC, LR.

6. OR NOT

- Instruksi *Or not* pada PLC mempunyai singkatan kode *Or not*. Instruksi ini dibutuhkan jika urutan kerja (*sequence*) pada suatu sistem kontrol hanya membutuhkan salah satu saja dari beberapa kondisi logika untuk mengeluarkan satu *output*.
- Logikanya seperti *contact NC relay*.
- Ladder diagram* simbol *Out* ditunjukkan pada gambar 2.15.



Gambar 2.15 Instruksi *Or not*

- Operand data area*

B (BIT) : IR, SR, AR, HR, TC, LR.

7. OUT

- Instruksi *Out* pada PLC mempunyai singkatan kode *OUT*. Instruksi ini berfungsi untuk mengeluarkan *output* jika semua kondisi logika *ladder diagram* sudah terpenuhi.
- Logikanya seperti *contact NO relay*.
- Ladder diagram* simbol *Out* ditunjukkan pada gambar 2.16.



Gambar 2.16 Instruksi *Out*

- Operand data area*

B (BIT) : IR, HR, LR, TR.

8. OUT NOT

- a. Instruksi *Out not* pada PLC mempunyai singkatan kode *Out not*. Instruksi ini berfungsi untuk mengeluarkan *output* jika semua kondisi logika *ladder diagram* tidak terpenuhi.
- b. Logikanya seperti *contact NC relay*.
- c. *Ladder diagram* simbol *Out not* ditunjukkan pada gambar 2.17.

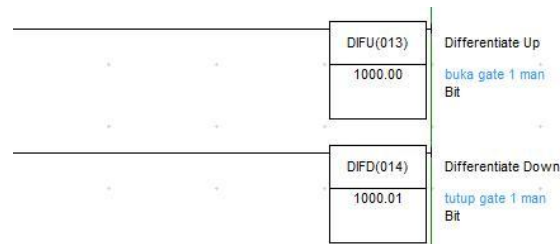


Gambar 2.17 Instruksi *Out not*

- d. *Operand data area*
B (BIT) : IR, HR, LR, TR.

9. Differentiate Up dan Differentiate Down

- a. Instruksi *Differentiate Up* pada PLC mempunyai singkatan kode DIFU (13) dan *Differentiate Down* (DIFD (14). Instruksi DIFU (13) dan DIFD (14) berfungsi untuk mengubah kondisi logika bit *operand* dari *off* menjadi *on* selama 1 *scantime*. 1 *scan time* adalah waktu yang dibutuhkan oleh PLC untuk menjalankan program dimulai dari alamat 00000 sampai instruksi END (01). DIFU (13) sifatnya mendeteksi transisi naik dari *input*, dan DIFD (14) mendeteksi transisi turun dari *input*.
- b. *Ladder diagram* simbol *DIFU* dan *DIFD* ditunjukkan pada gambar 2.18 pada halaman selanjutnya.



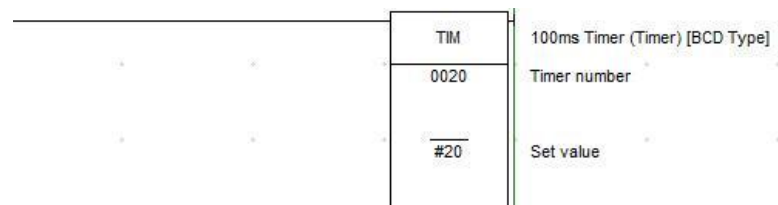
Gambar 2.18 Instruksi DIFU dan DIFD

c. *Operand data area*

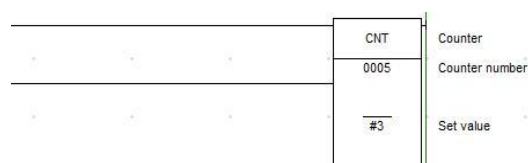
B (BIT) : IR, AR, HR, LR.

10. *Timer dan Counter*

- a. Instruksi *Timer* pada PLC mempunyai singkatan kode TIM dan *counter* pada PLC mempunyai kode CNT. Nilai *timer/counter* pada PLC bersifat *countdown* (menghitung mundur) dari nilai awal yang ditetapkan oleh program. Setelah hitungan mundur tersebut mencapai angka nol, maka NO *timer/counter* akan ON.
- b. *Ladder diagram* simbol *Timer* ditunjukkan pada gambar 2.19.

Gambar 2.19 Instruksi *timer*

- c. *Ladder diagram* simbol *counter* ditunjukkan pada gambar 2.20.

Gambar 2.20 Instruksi *counter*

d. *Operand data area*

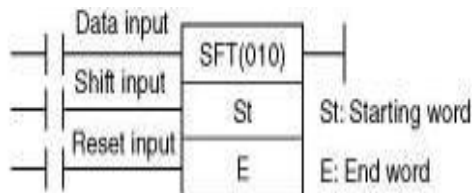
SV (Set Value) : IR, AR, DM, HR, LR, #.

11. *Shift Register*

a. Instruksi *Shift register* pada PLC mempunyai singkatan kode SFT (10).

Instruksi ini berfungsi untuk menggeser data dari bit yang paling rendah tingkatannya ke bit yang paling tinggi tingkatannya. Data *input* akan mulai digeser pada saat transisi naik dari *clock input*.

b. *Ladder diagram* simbol *shift register* ditunjukkan pada gambar 2.21.



Gambar 2.21 Instruksi *Shift register*

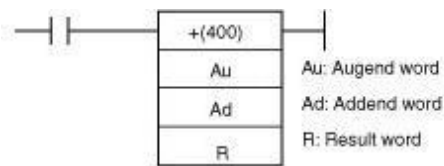
c. *Operand data area*

St (alamat awal): CIO,WR, HR. E (alamat akhir) : CIO, WR, HR.

12. *Increment dan Decrement*

a. Instruksi *Increment* pada PLC mempunyai singkatan kode INC (38) dan *decrement* pada PLC mempunyai singkatan kode DEC (39). Instruksi INC (38) dan DEC (39) merupakan instruksi BCD. INC (38) berfungsi untuk menambah data BCD dengan 1. Sedangkan instruksi DEC (39) berfungsi untuk mengurangi data BCD dengan 1.

b. *Ladder diagram* simbol *Increment* ditunjukkan pada gambar 2.22 pada halaman selanjutnya.

Gambar 2.22 Instruksi *Increment*

c. *Operand data area*

Au : CIO, WR, HR, AR, TC, DM, EM.

Ad : CIO, WR, HR, AR, TC, DM, EM.

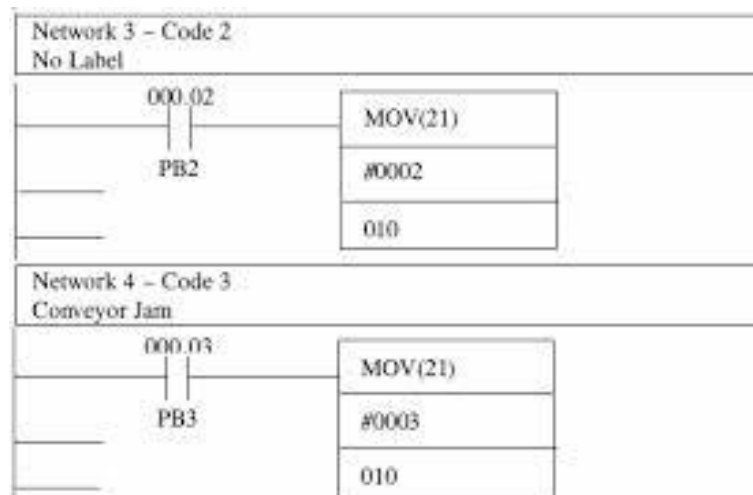
R : CIO, WR, HR, AR, TC, DM, EM.

13. *Move*

a. Instruksi *Move* pada PLC mempunyai singkatan kode MOVE (21).

Instruksi MOV (21) berfungsi untuk memindahkan data *channell* (16 bit data) dari alamat memori asal ke alamat memori tujuan. Atau untuk mengisi suatu alamat memori yang ditunjuk dengan data bilangan (*hexadecimal* atau BCD).

b. *Ladder diagram* simbol *Move* ditunjukkan pada Gambar 2.23.

Gambar 2.23 Instruksi Move^[4].

c. *Operand data area*

St (data awal) : CIO, WR, HR, AR, TC,

DM, EM. E (data akhir) : CIO, WR, HR,

AR, TC, DM, EM.

14. *Compare*

a. Instruksi *Compare* pada PLC mempunyai singkatan kode CMP (20).

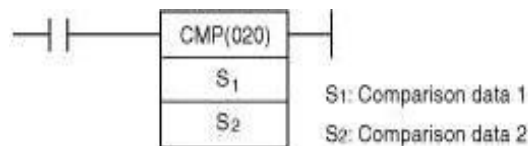
Instruksi ini berfungsi untuk membandingkan dua data 16 bit dan mempunyai *output* berupa bit > (lebih dari), bit = (sama dengan), bit < (kurang dari). Ketiga bit tersebut terdapat pada *special relay* yaitu :

a. 25505 yaitu bit >

b. 25506 yaitu bit =

c. 25507 yaitu bit <

b. *Ladder diagram* simbol *Compare* ditunjukkan pada Gambar 2.24.



Gambar 2.24 Instruksi *compare*

c. *Operand data area*

Cp1 (data compare 1) : CIO, WR, HR, AR, TC, DM, EM.

Cp2 (data compare 2) : CIO, WR, HR, AR, TC, DM, EM.

2.1.11.2 Spesifikasi PLC OMRON CP1L-M40DR-A

Unit PLC dibuat dalam banyak model/tipe. Pemilihan suatu tipe PLC harus mempertimbangkan, yang dibedakan sebagai berikut :

1. Jenis catu daya.
2. Jumlah terminal *input/output*.
3. Tipe rangkaian *output*.

Berikut ini spesifikasi dari PLC OMRON CP1L-M40DR-A dan bentuk fisik PLC-CP1L-M40DR-A :(lihat gambar 2.25)



Gambar 2.25 Bentuk fisik PLC CP1L-M40DR-A

1. Jenis catu daya : Tegangan 1 fasa 220 VAC.
2. Jumlah terminal *input/output* : 40 terminal.
 - a. *Input* : 26 terminal.
 - b. *Output* : 14 terminal.
3. Tipe rangkaian *output* :
 - a. *Input* : 24 VDC *relay*.
 - b. *Output* : 24 VDC *relay*.

Untuk menyambungkan koneksi komunikasi antara SCADA dan PLC, maka diperlukan tambahan 2 unit *board* pada PLC OMRON CP1L-M40DR-A menggunakan type CP1W-CIFO1 dengan *board* RS-232C. Berikut ini gambar

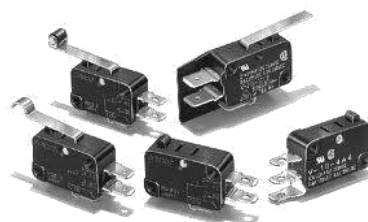
dari *option board* CP1W-CIF01 ditunjukkan pada gambar 2.26



Gambar2.26 CP1W-CIF01

2.1.13 *Limit Switch*

Limit switch adalah alat bantu sakelar yang bisa menghasilkan perubahan status dari *on* ke *off* atau dari 0 ke 1 atau sebaliknya bila mengenai atau melewati batas yang ditentukan. Di dalam penggunaan sistem parkir ini, *limit switch* merupakan sensor yang digunakan untuk mendeteksi saat ada mobil atau kendaraan parkir pada lokasi yang diberikan. Pada pembuatan miniatur sistem parkir ini menggunakan 20 *limit switch*. *Limit switch* ditunjukkan pada Gambar 2.27.



Gambar 2.27 *Limit switch*

2.1.13 *Power Supply Unit*

Merupakan alat yang digunakan untuk mengubah arus listrik AC menjadi arus listrik DC. Pada pembuatan miniatur tugas akhir ini memanfaatkan sumber

listrik 24 VDC pada *power supply* untuk sumber tegangan *motor stepper*, *driver motor stepper*, dan *fiber optic*. Pada pembuatan miniatur sistem parkir ini menggunakan 1 pcs *power supply*. *Power supply* ditunjukkan pada gambar 2.28 .



Gambar 2.28 *Power supply unit*

2.1.14 *Wireless*

Wireless LAN (WLAN) adalah teknologi LAN yang menggunakan frekuensi dan transmisi radio sebagai media penghantarnya, pada area tertentu, menggantikan fungsi kabel. Pada umumnya WLAN digunakan sebagai titik distribusi di tingkat pengguna akhir, melalui sebuah atau beberapa perangkat yang disebut dengan *Access Point (AP)*, berfungsi mirip hub dalam terminologi jaringan kabel *ethernet*. Di tingkat *backbone*, sejumlah AP tersebut tetap dihubungkan dengan media kabel. WLAN dimaksudkan sebagai solusi alternatif media untuk menjangkau pengguna yang tidak terlayani oleh jaringan kabel, serta untuk mendukung pengguna yang sifatnya bergerak atau berpindah-pindah (mobilitas). Di tunjukan pada gambar 2.29 pada halaman selanjutnya.



Gambar 2.29 *Wireless LT8900*