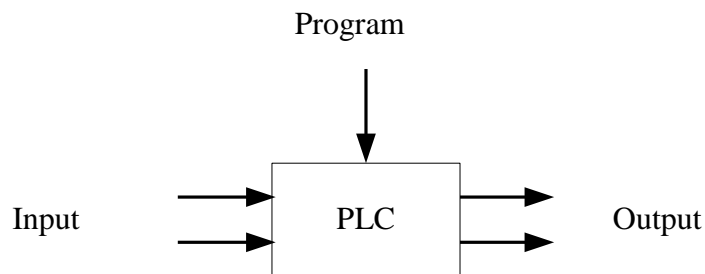




BAB II TINJAUAN PUSTAKA

2.1 Programmable Logic Controller (PLC)¹

PLC (*Programmable logic controller*) merupakan suatu bentuk khusus pengontrolan berbasis mikroprosesor yang memanfaatkan memori yang dapat diprogram untuk menyimpan intruksi-intruksi dan untuk mengimplementasikan fungsi – fungsi logika, sequencing, pewaktuan (*timing*), pencacahan (*counting*) dan aritmetika guna mengontrol mesin-mesin dan proses-proses seperti (gambar 2.1) dan dirancang untuk dioperasikan oleh para insinyur yang memiliki pengetahuan mengenai bahasa pemrograman. Piranti ini dirancang sedemikian rupa agar tidak hanya programmer komputer saja yang dapat membuat dan mengubah program- programnya. Oleh karena itu, para perancang PLC telah menempatkan program awal di dalam piranti ini (pre-program) yang memungkinkan program-program kendali dimasukkan dengan menggunakan suatu bentuk bahasa pemrograman yang sederhana dan intuitif.



Gambar 2.1 Sebuah *programmable logic controller*

Istilah logika (*logic*) dipergunakan karena pemrograman yang harus dilakukan sebagian besar berkaitan dengan pengimplementasian operasi- operasi logika dan penyambungan saklar. Perangkat-perangkat input, yaitu sensor-sensor semisal saklar dan perangkat-perangkat Output di dalam sistem yang di kendali, misalnya motor, katup, dll yang disambungkan ke PLC. Sang operator kemudian

¹William Bolton, *Programmable Logic Controller (PLC) Sebuah Pengantar Edisi Ketiga* (Jakarta: Erlangga, 2003), hlm. 03

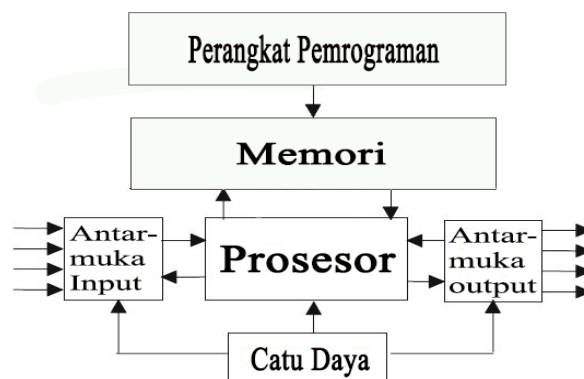


memasukan serangkaian intruksi, yaitu sebuah program ke dalam memory PLC. Perangkat pengontrol tersebut kemudian memantau input-input dan output-output sesuai dengan intruksi-intruksi di dalam program dan melaksanakan aturan-aturan kendali yang telah diprogramkan.

PLC serupa dengan komputer namun, bedanya komputer dioptimalkan untuk tugas-tugas perhitungan dan penyajian data, sedangkan PLC dioptimalkan untuk tugas-tugas pengontrolan dan pengoperasian di dalam lingkungan industri. Dengan demikian PLC memiliki karakteristik :

1. Kokoh dan dirancang untuk tahan terhadap getaran, suhu, kelembaban, dan kebisingan.
2. Antarmuka input dan output telah tersedia secara built-in didalamnya.
3. Mudah diprogram dan menggunakan sebuah bahasa pemrograman yang mudah dipahami, yang sebagian besar berkaitan dengan operasi-operasi logika dan penyambungan.

Perangkat PLC pertama kali dikembangkan pada tahun 1969. Dewasa ini PLC secara luas digunakan dan telah dikembangkan dari unit-unit kecil yang berdiri sendiri (*self-contained*) yang hanya mampu menangani sekitar 20 input/output menjadi sistem-sistem modular yang dapat menangani input/output dalam jumlah besar, menangani input/output analog maupun digital, dan melaksanakan mode-mode kontrol proporsional integral derivatif.



Gambar 2.2 Sistem PLC (*Programmable Logic Controller*)



2.1.1 Hardware

Umumnya, sebuah sistem PLC memiliki lima komponen dasar. Komponen-komponen ini adalah unit processor, memori, unit catu daya, bagian antarmuka input/output, dan perangkat pemrograman.

1. *Unit processor* atau *central processing unit* (unit pengolahan pusat) (*CPU*) adalah unit yang berisi mikroprocessor yang menginterpretasikan sinyal-sinyal input dan melaksanakan tindakan-tindakan pengontrolan, sesuai dengan program yang tersimpan di dalam memori, lalu mengkomunikasikan keputusan-keputusan yang diambilnya sebagai sinyal kontrol ke antarmuka output.
2. *Unit catu daya* diperlukan untuk mengkonversikan tegangan AC sumber menjadi tegangan rendah DC (5V) yang dibutuhkan oleh *processor* dan rangkaian-rangkaian di dalam modul-modul antarmuka input dan output.
3. *Perangkat pemrograman* dipergunakan untuk memasukkan program yang dibutuhkan ke dalam memori. Program tersebut dibuat dengan menggunakan perangkat ini dan kemudian dipindahkan ke dalam unit memori PLC.
4. *Unit memori* adalah tempat dimana program yang digunakan untuk melaksanakan tindakan-tindakan pengontrolan oleh mikroprocessor disimpan.
5. *Bagian input dan output* adalah antarmuka di mana prosesor menerima informasi dari dan mengkomunikasikan informasi kontrol ke perangkat-perangkat eksternal. Sinyal-sinyal input, oleh karenanya, dapat berasal dari saklar-saklar.

Tahap dasar untuk penyiapan awal untuk memudahkan dan memasukkan program dalam PLC dengan mempersiapkan daftar seluruh peralatan input dan output beserta lokasi I/O bit, penempatan lokasi word dalam penulisan data. Untuk pemrograman sebuah *Programmable Logic Controller* terlebih dahulu kita harus mengenal atau mengetahui tentang organisasi dan memorinya.

Ilustrasi dari organisasi memori adalah sebagai peta memori (memori map), yang spacenya terdiri dari kategori *User Programable* dan *Data Table*. *User Program* adalah dimana program *Logic Ladder* dimasukkan dan disimpan yang berupa instruksi – instruksi dalam format *Logic Ladder*. Setiap instruksi



memerlukan satu word didalam memori.

2.1.2 PLC Omron CP1E-E40 SDR-A²

Merupakan Jenis dari PLC Omron seri CP1E, sedangkan arti dari E40 merupakan jumlah dari output dan input yang terdapat pada PLC. PLC jenis ini dapat di implementasikan pada penggerak mekanisme alat industri, alat rumah tangga, dan tugas teknik lainnya, yang mana bersifat logika elektronika.



Gambar 2.3 PLC Omron CP1E-E40 SDR-A

PLC Omron seri CP1E memiliki I/O sebanyak 40 yang dimana 24 input bisa diubah menjadi analog, yaitu bekerja dengan tegangan 5 sampai 24 volt dan memiliki output sebanyak 16 yang dimana masing-masing output tersebut juga memiliki internal relay yang bekerja dengan arus hingga 10A. PLC Omron seri CP1E bekerja dengan tegangan yang bisa diubah 100 sampai 240 VAC, Program memory: 2Ksteps (EEPROM), Data memory DM: 2Kwords. Dan memiliki minimal tegangan kerja 5 VDC dan maksimum tegangan kerja 24 VDC pada input PLC. Kemudian pada masing-masing output PLC memiliki internal relay yang memiliki maksimum arus kerja sebesar 10 A. PLC Omron seri CP1E memiliki sistem program dengan menggunakan software pemrograman CX-Programmer.

² Omron, "SYSMAC CP-Series CP1E CPU Units Introduction Manual". Introduction Manual, omron, hlm. 03.

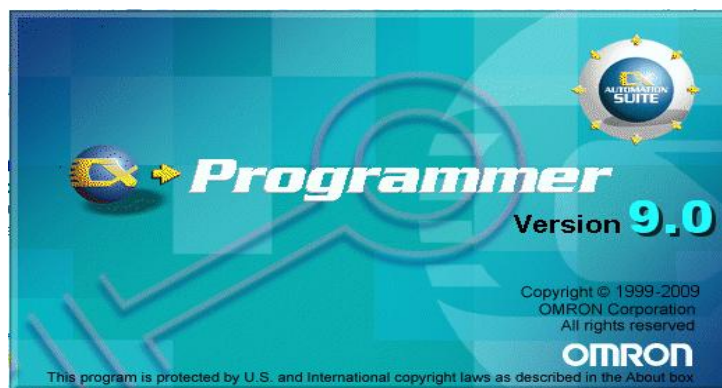


2.1.3 Software CX-Programmer³

CX-Programmer merupakan software khusus untuk memprogram PLC buatan OMRON. CX Programmer ini sendiri merupakan salah satu software bagian dari CX-One. Dengan CX-Programmer ini kita bisa memprogram aneka PLC buatan omron dan salah satu fitur yang saya suka yaitu adanya fitur simulasi tanpa harus terhubung dengan PLC, sehingga kita bisa mensimulasikan ladder yang kita buat, dan simulasi ini juga bisa kita hubungkan dengan HMI PLC Omron yang telah kita buat dengan menggunakan CX-Designer (bagian dari CX-One).

Software ini beroperasi di bawah sistem operasi Windows, oleh sebab itu pemakai software ini diharapkan sudah familier dengan sistem operasi Windows antara lain untuk menjalankan software program aplikasi, membuat file, menyimpan file, mencetak file, menutup file, membuka file, dan keluar dari (menutup) software program. Ada beberapa persyaratan minimum yang harus dipenuhi untuk bisa mengoperasikan CX Programmer secara optimal yaitu: Komputer IBM PC/AT kompatibel

- CPU Pentium I minimal 133 MHz
- RAM 32 Mega bytes
- Hard disk dengan ruang kosong kurang lebih 100 MB
- Monitor SVGA dengan resolusi 800 x 600

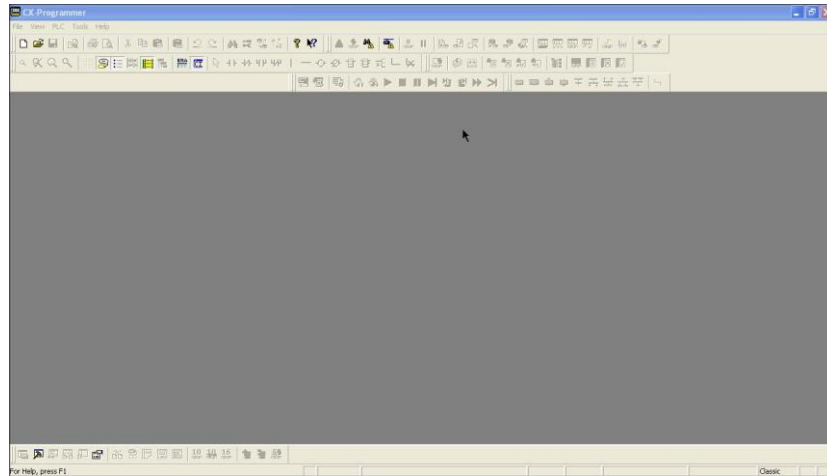


Gambar 2.4 CX-Programmer Version 9.0 Omron

³Musbikhin, "Pengantar CX Programmer (seri Belajar PLC)", diakses dari <http://www.musbikhin.com/pengantar-cx-programmer-seri-belajar-plc>, pada tanggal 25 Mei 2016 pukul 15.32 WIB

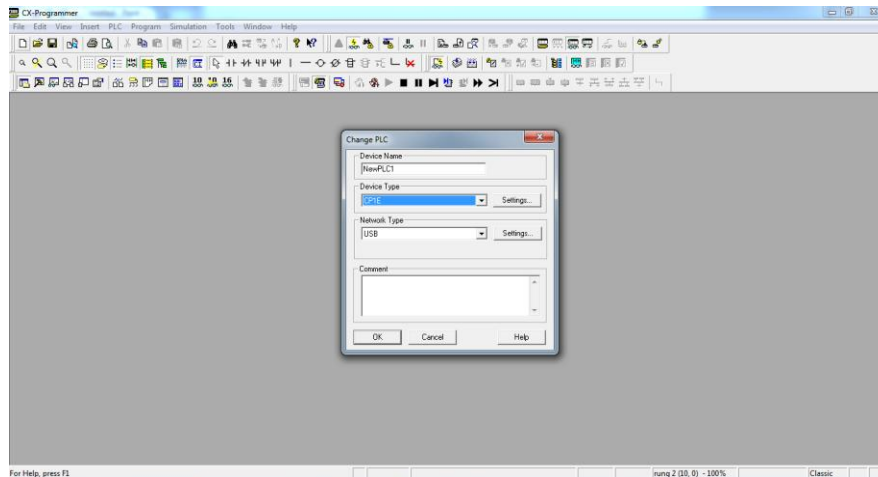


Berikut tampilan dari CX-Programmer saat pertama kali dibuka :



Gambar 2.5 Tampilan Pertama Program CX-Programmer Version 9.0 Omron

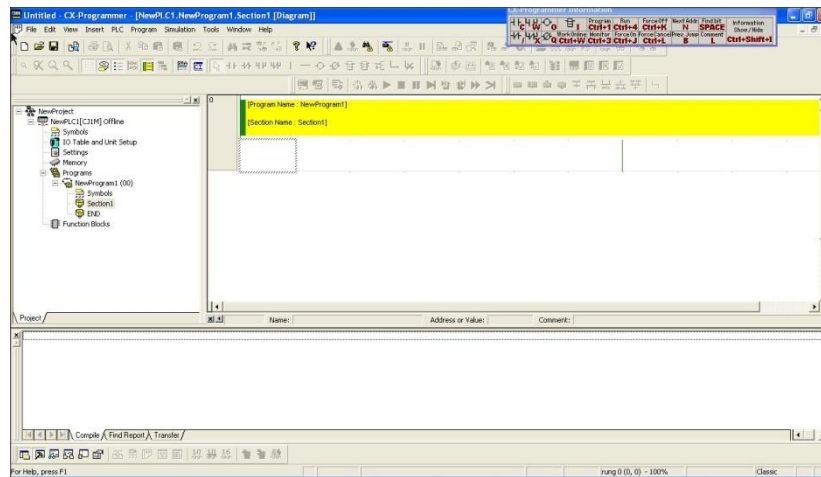
Untuk memulai menggunakan CX-Programmer ini yaitu pada menu pilih file -> new atau bisa langsung pada toolbar klik gambar kertas putih untuk memulai membuat project baru, kalo untuk membuka file project yang sudah dibuat sebelumnya yaitu pilih file -> open atau pada toolbar pilih gambar disamping kertas putih maka akan muncul tampilan berikut :



Gambar 2.6 Tampilan Pemilihan Device PLC Pada Program CX-Programmer
Version 9.0 Omron

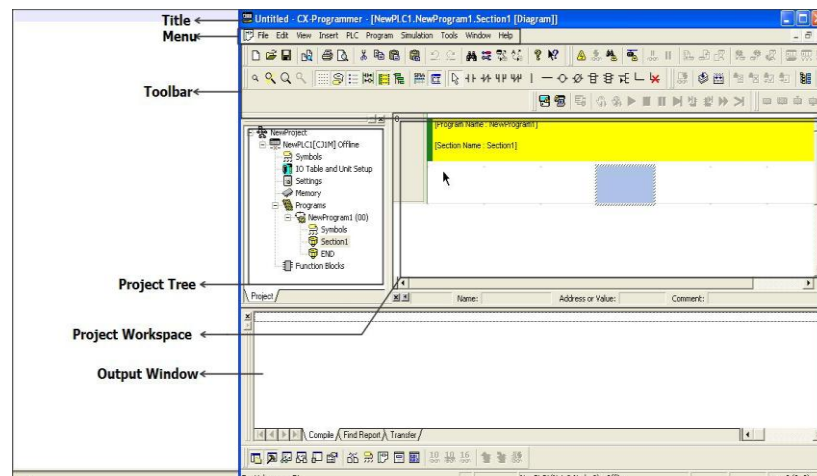


Setelah memilih tipe PLC yang akan digunakan, misalnya PLC CP1E dan Network type yang akan digunakan yaitu USB untuk setting lebih dalam bisa diklik setting, kemudian klik OK maka akan tampil tampilan berikut :



Gambar 2.7 Tampilan Project Program CX-Programmer Version 9.0 Omron

Keterangan detail untuk tampilan tersebut yaitu:



Gambar 2.8 Tampilan Keterangan Project Program CX-Programmer Version 9.0 Omron

**Title Bar :**

Menunjukkan nama file atau data tersimpan dan dibuat pada CX- Programmer

Menus :

Pilihan Untuk memilih Menu

Toolbar :

Pilihan untuk memilih fungsi dengan menekan tombol.

Select[view] Toolbar, Kemudian dapat memilih toolbar yang ingin ditampilkan.

Section :

Dapat membagi program kedalam beberapa blok. Masing-masing blok dapat dibuat atau ditampilkan.

Project WorkSpace Project Tree :

Mengatur program dan data. Dapat membuat duplikat dari setiap elemen dengan melakukan Drag dan Drop diantara proyek yang berbeda atau melalui suatu proyek.

Ladder Window :

Layar sebagai tampilan atau membuat diagram tangga.

Output Window :

Menunjukkan informasi error saat melakukan compile (error check).

Menunjukkan hasil dari pencarian kontak / koil didalam list form.

Menunjukkan detail dari error yang ada pada saat loading suatu proyek.

Status Bar :

Menunjukkan suatu informasi seperti nama PLC, status on line/offline, lokasi dari cell yang sedang aktif.

Information Window :

Menampilkan window yang menunjukkan shortcut key yang digunakan pada CX – Programmer.

Symbol Bar :

Menampilkan nama, alamat atau nilai dan comment dari simbol yang sedang dipilih cursor



2.1.4 Program PLC

Suatu software yang berfungsi sebagai pengontrol otomatis yang berupa softcontact yang diimplementasikan kedalam suatu bentuk bilangan logika. Sehingga dapat mengatur sistem suatu alat industri elektronika dan mekanik.

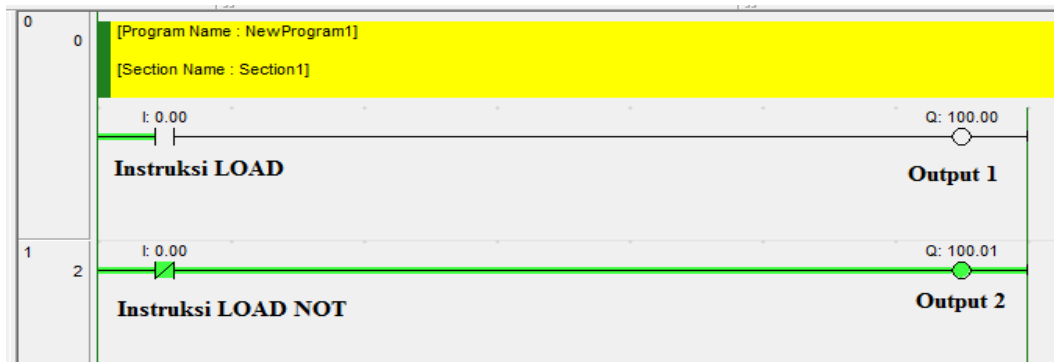
Ada 2 sistem pemrograman pada PLC Omron CP1E-E40SDR-A :

1. *Function Block Diagram* : Jenis Teknik Pemrograman Logic yang tersusun dari block-block diagram dalam 1 fungsi blok diagram khusus.
2. *Ladder Diagram* : Jenis Teknik Pemrograman Logic yang disusun dalam satuan-satuan kontak untuk menghasilkan fungsi tertentu dalam menghasilkan logika yang terdiri dari kontak NC, NO, Timer, Counter, dan lain-lain.

2.1.5 Instruksi Dasar Pada PLC

1. LOAD (LD) dan LOAD NOT (LD NOT)

Kondisi pertama yang mengawali sembarang blok logika di dalam diagram tangga berkaitan dengan instruksi LOAD (LD) atau LOAD NOT (LD NOT). Contoh instruksi ini ditunjukkan pada gambar 2.9.



Gambar 2.9 Ladder Diagram Instruksi LD dan LD NOT

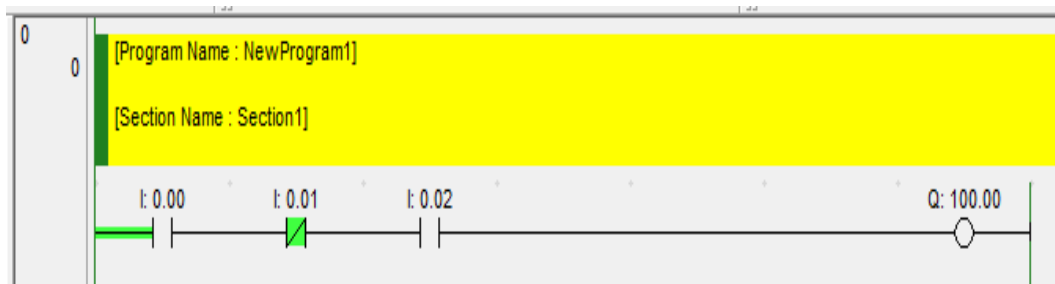
Tabel 2.1 Kode Mnemonic Instruksi LD dan LD NOT

Alamat	Instruksi	Operan
00000	LD	0.00
00001	OUT	100.00
00002	LD NOT	0.00
00003	OUT	100.01



2. AND dan AND NOT

Jika terdapat dua atau lebih kondisi yang dihubungkan seri pada garis instruksi yang sama maka kondisi pertama menggunakan instruksi LD atau LD NOT, dan sisanya menggunakan instruksi AND atau AND NOT. Gambar menunjukkan suatu penggalan diagram tangga yang mengandung tiga kondisi yang dihubungkan secara seri pada garis instruksi yang sama dan berkaitandengan instruksi LD, AND NOT, dan AND. Masing-masing instruksi tersebut membutuhkan satu baris kode mnemonic.



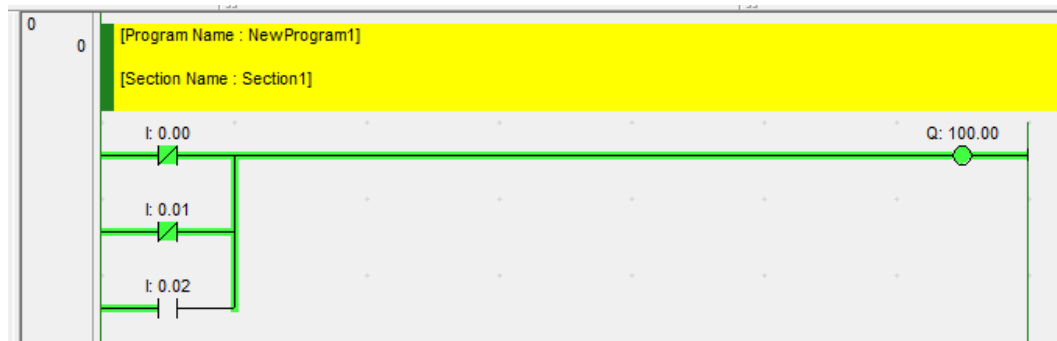
Gambar 2.10 Ladder Diagram Instruksi AND dan AND NOT

Tabel 2.2 Kode Mnemonic Instruksi AND dan AND NOT

Alamat	Instruksi	Operan
00000	LD	0.00
00001	AND NOT	0.01
00002	AND	0.02
00003	OUT	100.00

3. OR dan OR NOT

Jika dua atau lebih kondisi yang dihubungkan paralel, artinya dalam garisinstruksi yang berbeda kemudian bergabung lagi dalam satu garis instruksi yang sama maka kondisi pertama terkait dengan instruksi LD dan LD NOT dan sisanya berkaitan dengan instruksi OR dan OR NOT. Gambar menunjukkan tiga buah instruksi yang berkaitan dengan instruksi LD NOT, OR NOT, dan OR. Masing-masing instruksi tersebut membutuhkan satu baris kode mnemonic



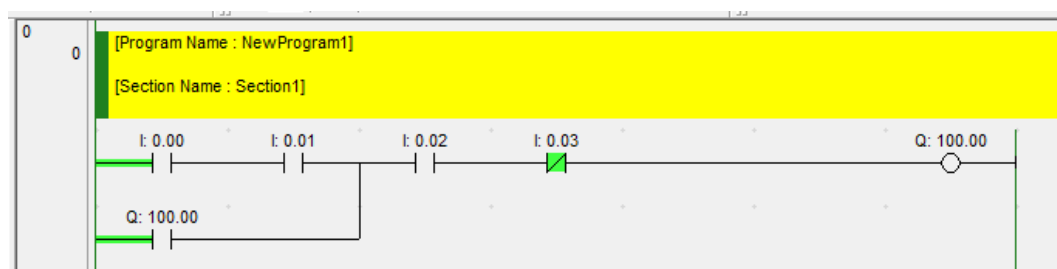
Gambar 2.11 Ladder Diagram Instruksi OR dan OR NOT

Tabel 2.3 Kode Mnemonic Instruksi OR dan OR NOT

Alamat	Instruksi	Operan
00000	LD NOT	0.00
00001	OR NOT	0.01
00002	OR	0.02
00003	OUT	100.00

4. Kombinasi instruksi AND dan OR

Jika instruksi AND dan OR digabung atau dikombinasikan dalam suatu rangkaian tangga yang kompleks maka bisa dipandang satu persatu, artinya bisa dilihat masing-masing hasil gabungan dua kondisi menggunakan instruksi AND atau OR secara sendiri-sendiri kemudian menggabungkannya menjadi satu kondisi menggunakan instruksi AND atau OR yang terakhir. Gambar 2.11 menunjukkan contoh diagram tangga yang mengimplementasikan cara seperti tersebut di atas.



Gambar 2.12 Ladder Diagram Instruksi AND dan OR



Tabel 2.4 Kode Mnemonic Instruksi AND dan OR

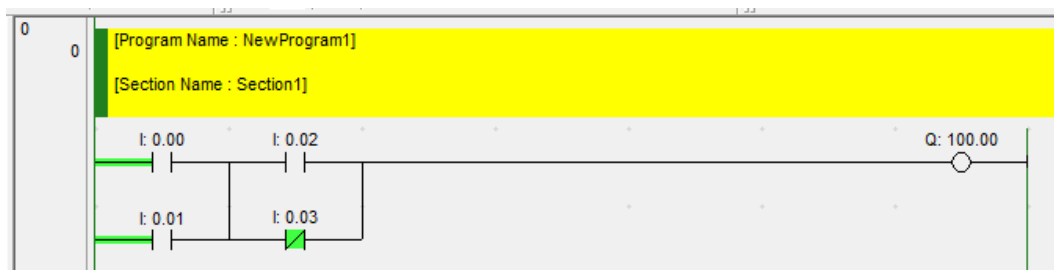
Alamat	Instruksi	Operan
00000	LD	0.00
00001	AND	0.01
00002	OR	100.00
00003	AND	0.02
00004	AND NOT	0.03
00005	OUT	100.00

5. Instruksi-instruksi Blok Logika

Instruksi-instruksi blok logika tidak berhubungan dengan suatu kondisi tertentu pada diagram tangga, melainkan untuk menyatakan hubungan antar blok-blok logika, misalnya instruksi AND LD akan meng-AND-logik-kan kondisi eksekusi yang dihasilkan oleh dua blok logika, demikian juga dengan OR LD untuk meng-OR logikkan kondisi eksekusi yang dihasilkan dua blok logika.

a. AND LOAD (AND LD)

Gambar 2.12 menunjukkan contoh penggunaan blok logika AND LD yang terdiri atas dua blok logika, yang akan menghasilkan kondisi ON jika blok logika kiri dalam kondisi ON (salah satu dari 0.00 atau 0.01 yang ON) dan blok logika kanan juga dalam keadaan ON (0.02 dalam kondisi ON atau 0.03 dalam kondisi OFF).



Gambar 2.13 Ladder Diagram Instruksi AND LOAD (AND LD)

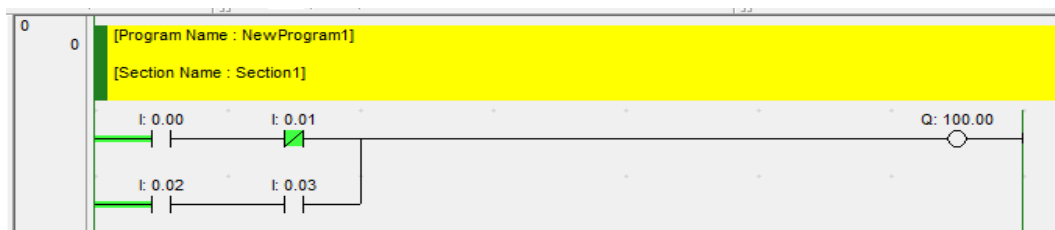


Tabel 2.5 Kode Mnemonic Instruksi AND LOAD (AND LD)

Alamat	Instruksi	Operan
00000	LD	0.00
00001	OR	0.01
00002	LD	0.02
00003	OR NOT	0.03
00004	AND LD	-
00005	OUT	100.00

b. OR LOAD (OR LD)

Instruksi ini digunakan untuk meng-OR-logik-kan dua blok logika. Gambar 2.13 menunjukkan contoh penggunaan blok logika OR LD yang terdiri atas dua blok logika. Kondisi eksekusi ON akan dihasilkan jika blok logik atas atau blok logika bawah dalam kondisi ON. Artinya, 0.00 dan kondisi ON dan 0.01 dalam kondisi OFF atau 0.02 dan 0.03 dalam kondisi ON).



Gambar 2.14 Ladder Diagram Instruksi OR LOAD (OR LD)

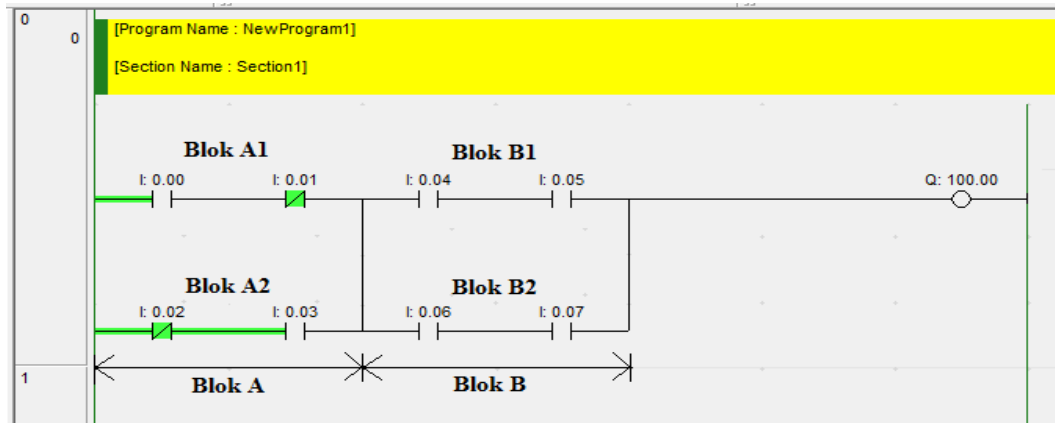
Tabel 2.6 Kode Mnemonic Instruksi OR LOAD (OR LD)

Alamat	Instruksi	Operan
00000	LD	0.00
00001	AND NOT	0.01
00002	LD	0.02
00003	AND	0.03
00004	OR LD	-
00005	OUT	100.00



c. Logika Kompleks

Untuk membuat kode mnemonik diagram tangga yang kompleks, caranya dengan cara membagi membagi diagram tersebut ke dalam blok-blok logika yang besar, kemudian membagi lagi blok yang besar tersebut menjadi blok- blok logika yang lebih kecil, demikian seterusnya hingga tidak perlu lagi dibuat blok yang lebih kecil lagi. Blok-blok ini kemudian masing-masing dikodekan, mulai dari yang kecil, dan digabungkan satu per satu hingga membentuk diagram tangga yang asli. Instruksi blok logika AND LD dan OR LD hanya digunakan untuk menggabungkan dua blok logika saja (blok logika yang digabungkan berupa hasil penggabungan sebelumnya, atau hanya sebuah kondisi tunggal). Gambar 2.14 memperlihatkan suatu contoh diagram tangga yang kompleks, yang dapat dibagi dua blok besar (blok A dan B). Blok A dapat dibagi lagi menjadi dua blok yang lebih kecil (blok A1 dan A2), dan blok B dibagi menjadi dua blok yang lebih kecil, yaitu blok B1 dan B2. Kemudian blok-blok logika yang kecil ini ditulis terlebih dahulu, diawali dengan menuliskan blok A1 (alamat 00000 dan 00001) dan blok A2 (alamat 00002 dan 00003), kemudian digabung menggunakan instruksi blok logik OR LD (alamat 00004). Selanjutnya blok B1 dituliskan (alamat 00005 dan 00006) dilanjutkan dengan blok B2 (alamat 00007 dan 00008) dan digabung dengan instruksi blok logik OR LD (alamat 00009). Hasilnya berupa blok A dan blok B yang kemudian juga digabung menggunakan blok logika AND LD (alamat 00010).



Gambar 2.15 Ladder Diagram Contoh Penggunaan Instruksi Blok Logika Kompleks

Tabel 2.7 Kode Mnemonic Contoh Penggunaan Instruksi Blok Logika Kompleks

Alamat	Instruksi	Operan
00000	LD	0.00
00001	AND NOT	0.01
00002	LD NOT	0.02
00003	AND	0.03
00004	OR LD	-
00005	LD	0.04
00006	AND	0.05
00007	LD	0.06
00008	AND	0.07
00009	OR LD	-
00010	AND LD	-
00011	OUT	100.00

6. Intruksi Kendali Bit

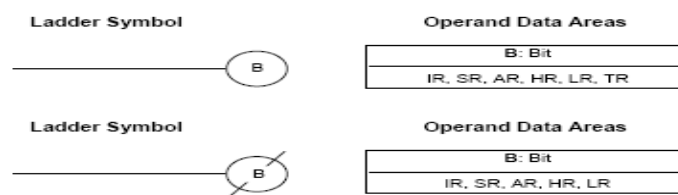
Terdapat instruksi dasar yang dapat digunakan untuk mengontrol status bit secara individual, yaitu DIFFERENTIATE UP (DIFU), DIFFERENTIATE DOWN (DIFD) instruksi ini dituliskan di sisi paling kanan diagram tangga dan



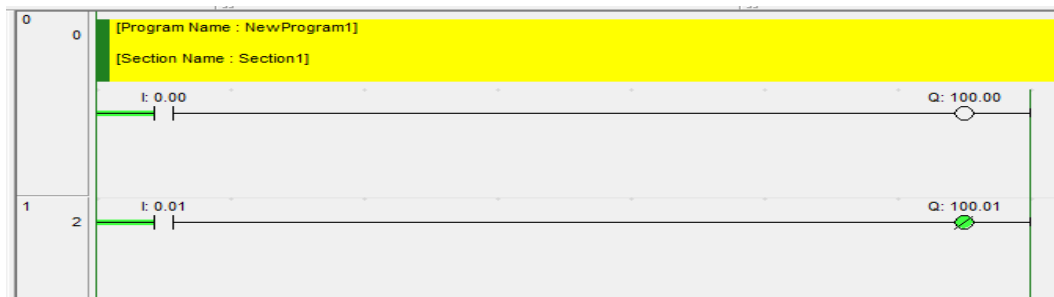
membutuhkan sebuah alamat bit sebagai operan. Selain instruksi-instruksi ini digunakan untuk membuat bit-bit keluaran ON atau OFF dalam area IR (ke piranti eksternal), ini juga digunakan untuk mengontrol bit-bit lainnya.

7. Instruksi OUTPUT (OUT) dan OUTPUT NOT (OUT NOT)

Instruksi ini digunakan untuk mengontrol operan yang berkaitan dengan kondisieksekusi (apakah ON atau OFF). Dengan menggunakan instruksi OUT, maka bit operanakan menjadi ON jika kondisi eksekusinya juga ON, sedangkan OUT NOT akanmenyebabkan bit operan menjadi ON jika kondisi eksekusinya OFF. Gambar 2.15 memperlihatkan simbol tangga dan area data operan dari instruksi OUT dan OUT NOT, sedangkan Gambar 2.16 memperlihatkan contoh implementasi kedua instruksi tersebut.



Gambar 2.16 Simbol Tangga Dan Area Data Operan Instruksi OUT dan OUT NOT



Gambar 2.17 Ladder Diagram Instruksi LD OUT dan LD OUT NOT

Tabel 2.8 Kode Mnemonic Instruksi LD OUT

Alamat	Instruksi	Operan
00000	LD	0.00
00001	OUT	100.00



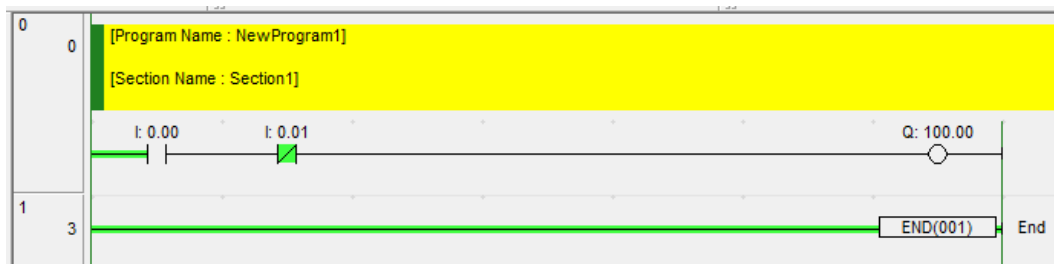
Tabel 2.9 Kode Mnemonic Instruksi LD OUT NOT

Alamat	Instruksi	Operan
00000	LD	0.01
00001	OUT NOT	100.01

8. Instruksi END

Instruksi END merupakan instruksi terakhir yang harus dituliskan atau digambarkan dalam diagram tangga. CPU pada PLC akan mengerjakan semua instruksi dalam program dari awal (baris pertama) sampai ditemui instruksi END yang pertama, sebelum kembali lagi mengerjakan instruksi dalam program dari awal (artinya instruksi-instruksi yang ada di bawah instruksi END akan diabaikan). Instruksi END tidak memerlukan operan dan tidak boleh diawali dengan suatu kondisi seperti pada instruksi lainnya.

Suatu diagram tangga atau program PLC harus diakhiri dengan instruksi END, jika tidak maka program tidak dijalankan sama sekali. Angka yang dituliskan pada instruksi END pada kode mnemonic merupakan kode fungsinya. Gambar 2.18 memperlihatkan contoh penggunaan instruksi END.



Gambar 2.18 Ladder Diagram Instruksi END

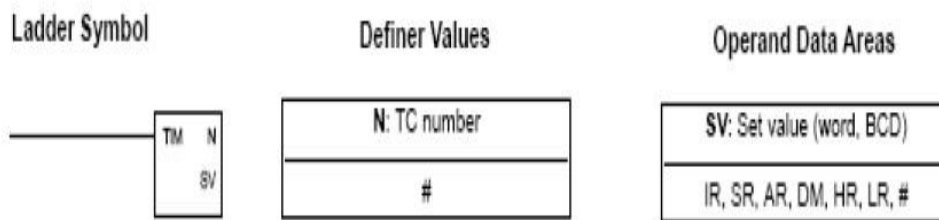
Tabel 2.10 Kode Mnemonic Instruksi END

Alamat	Instruksi	Operan
00000	LD	0.00
00001	AND NOT	0.01
00002	OUT	100.00
00003	END(001)	-

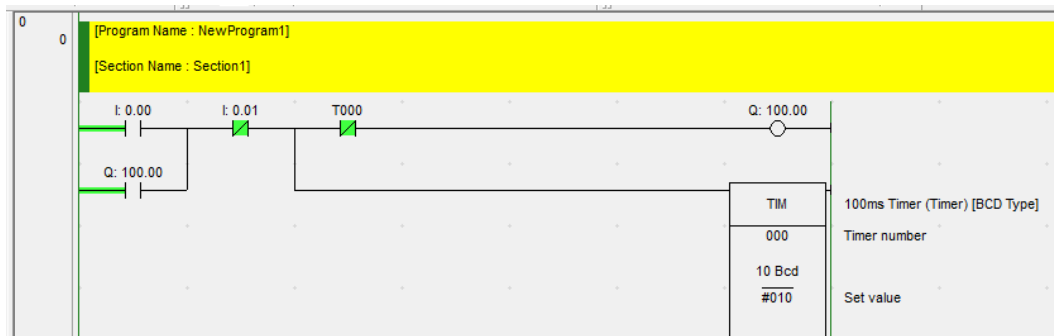


9. Instruksi TIMER (TIM)

Instruksi TIM dapat digunakan sebagai timer (pewaktu) ON-delay pada rangkaian relai. Gambar 2.18 memperlihatkan simbol tangga dan area data operan dari instruksi TIM. Instruksi TIM membutuhkan angka timer (N), dan nilai set (SV) antara 0000 sampai 9999 (artinya 000,0 sampai 999,9 detik).



Gambar 2.19 Simbol Ladder Diagram Dan Area Data Operan Dari Instruksi TIMER (TIM)



Gambar 2.20 Ladder Diagram Penggunaan Instruksi Blok Logika TIMER (TIM)

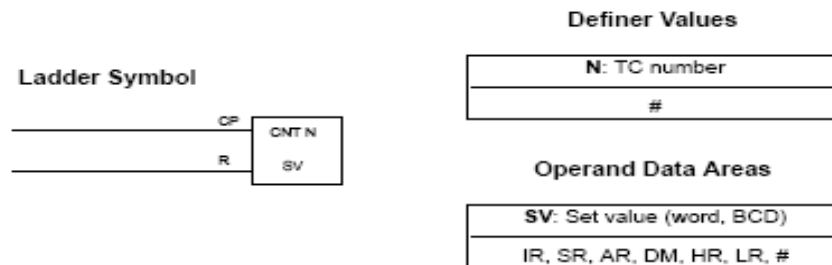
Tabel 2.11 Kode Mnemonic Instruksi Blok Logika TIMER (TIM)

Alamat	Instruksi	Operan
00000	LD	0.00
00001	OR	100.00
00002	AND NOT	0.01
00003	TIM	000 #010
00004	AND NOT	T000
00005	OUT	100.00

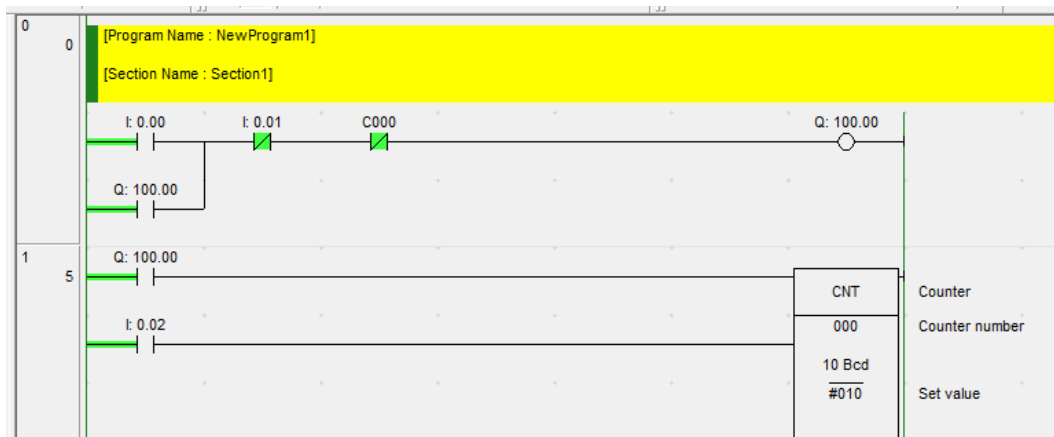


10. Instruksi COUNTER (CNT)

CNT yang digunakan di sini adalah counter penurunan yang diset awal. Penurunan satu hitungan setiap kali saat sebuah sinyal berubah dari OFF ke ON. Counter harus diprogram dengan input hitung, input reset, angka counter, dan nilai set (SV) Nilai set ini adalah 0000 sampai 9999. Gambar 2.20 memperlihatkan simbol tangga dan area data operan dari instruksi CNT. Dan Gambar 2.21 memperlihatkan contoh penggunaan instruksi END.



Gambar 2.21 Simbol Ladder Diagram Dan Area Data Operan Dari Instruksi COUNTER (CNT)



Gambar 2.22 Ladder Diagram Penggunaan Instruksi COUNTER (CNT)



Tabel 2.12 Kode Mnemonic Instruksi COUNTER (CNT)

Alamat	Instruksi	Operan
00000	LD	0.00
00001	OR	100.00
00002	AND NOT	0.01
00003	AND NOT	C000
00004	OUT	100.00
00005	LD	100.00
00006	LD	0.02
00006	CNT	000 #010

2.1.6 Perangkat – Perangkat Input⁴

Pada dasarnya PLC memerlukan sebuah input untuk melakukan perintah yang sudah diprogram dan di download ke PLC. Umumnya PLC hanya memerlukan input oleh sebuah saklar mekanis dan sensor.

Sebuah saklar mekanis menghasilkan sinyal (atau sinyal-sinyal) hidup / mati sebagai akibat dari tertutup atau terbukanya saklar oleh suatu input mekanis. Dipasaran tersedia saklar-saklar dengan kontak normally open (normal terbuka) (NO) atau normally closed (normal tertutup) (NC) atau kontak yang dapat diatur sesuai kebutuhan dengan memilih kontak-kontak yang tepat.

2.1.7 Perangkat-perangkat Output⁵

Port-port output sebuah PLC dapat berupa tipe relay atau tipe isolator-optik dengan transistor atau tipe triac, bergantung pada perangkat yang tersambung padanya, yang akan dikontrol. Secara umum, sinyal digital dari salah satu kanal output sebuah PLC digunakan untuk mengontrol sebuah actuator yang pada gilirannya mengontrol sebuah proses. Istilah actuator digunakan untuk perangkat yang mengubah sinyal listrik menjadi gerakan-gerakan mekanik yang

⁴William Bolton, *Programmable Logic Controller (PLC) Sebuah Pengantar Edisi Ketiga* (Jakarta: Erlangga, 2003), hlm. 15.

⁵William Bolton, *Programmable Logic Controller (PLC) Sebuah Pengantar Edisi Ketiga* (Jakarta: Erlangga, 2003), hlm. 24.



kemudian digunakan untuk mengontrol proses. Berikut ini adalah beberapa contoh dari output PLC.

2.2. Tombol Tekan

Prinsip kerja tombol tekan hampir sama dengan saklar tekan yang digunakan pada instalasi penerangan, bedanya jika saklar tekan jenis yang mempunyai togel akan langsung mengikat/mengunci, sedangkan pada tombol tekan tidak ada. Jadi tombol tekan setelah ditekan tidak akan mengunci, tetapi kembali keadaannya semula. Ada dua kontak yang dapat dilakukan oleh tombol tombol tekan, yaitu :

1. Kontak NO (Normally Open) Biasanya berwarna hijau.
2. Kontak NC (Normally Close) Biasanya berwarna Merah.



Gambar 2.23 Kontak NO



Gambar 2.24 kontak NC

2.3 Saklar Pemilih (Selector Switch)

Saklar jenis ini pada umumnya tersedia dua, tiga atau empat pilihan posisi, dengan berbagai tipe knop. Saklar pemilih biasanya dipasang pada panel kontrol untuk memilih jenis operasi yang berbeda, dengan rangkaian yang berbeda pula. Saklar pemilih memiliki beberapa kontak dan setiap kontak dihubungkan oleh kabel menuju rangkaian yang berbeda, misal untuk rangkaian putaran motor cepat dan untuk rangkaian putaran motor lambat.

Selector Switch, alat ini di gunakan untuk memilih, banyak sekali type selector switch, tapi biasanya hanya dua tipe yang sering di gunakan, yaitu 2 posisi, (ON-OFF/Start-Stop/0-1,) dan 3 posisi (ON-OFF-ON/Auto-Off-



Manual) dengan selector switch, kondisi peralatan dapat langsung di ketahui dari penunjukan tangkai selector switch, dengan selector switch, rangkaian ON-OFF lebih sederhana, karena selector switch tidak seperti tombol tekan yang hanya kontak sementara.



2.25 Selector Switch

2.4 Detektor Ketinggian Cairan

Sensor-sensor tekanan dapat digunakan untuk memantau ketinggian cairan di dalam sebuah tangki. Seringkali sebuah sensor dipakai hanya untuk memberikan sinyal ketika ketinggian cairan di dalam sebuah wadah mencapai titik tertentu. Saklar apung (*float switch*) yang digunakan untuk tujuan ini terdiri dari sebuah benda apung yang berisi sebuah magnet yang dapat bergerak dan sebuah saklar *reed*. Ketika benda apung naik dan turun mengikuti ketinggian cairan, magnet akan menyambungkan atau memutuskan saklar reed, di mana saklar reed yang tersambung ke sebuah rangkaian pada gilirannya akan menghasilkan sinyal hidup / mati.

Tank Water Level Float Sensor Switches atau Saklar Pelampung Air adalah saklar untuk mendeteksi jika air dalam suatu wadah / tangki / bak sudah mencapai ketinggian pada titik tertentu (sesuai dengan posisi saklar ini). Prinsip kerja saklar ini adalah menggunakan reed switches (saklar lidi) didalam batang dan magnet didalam pelampung yang berada disekeliling batang. Saat air mengangkat pelampung maka magnet akan mengaktifkan atau menonaktifkan reed switch. Saklar pelampung air ini dapat berfungsi secara NC atau NO dengan melepas pelampung dan membalik arahnya.⁶

⁶Anonim, Tank Water Level Float Sensor Switches Saklar Pelampung Air diakses dari <https://www.google.com/search.tank+water+level+float+sensor+switches+saklar+pelampung+air>



Gambar 2.26 Saklar Pelampung Air

2.5 Lampu Tanda

Lampu tanda atau biasa disebut juga *pilot lamp* digunakan pada peralatan kontrol untuk menandai bekerja atau tidaknya suatu peralatan atau rangkaian, dapat juga sebagai kondisi/keadaan beban. Jika lampu tanda dipergunakan untuk menandai suatu peralatan yang sedang bekerja, maka lampu tanda dipasang seri pada kontak NO, sedangkan apabila lampu tanda digunakan untuk menandai tidak bekerjanya suatu peralatan, maka lampu tanda dipasang paralel pada kontak NC pada rangkaian yang mengontrol peralatan tersebut. Jika lampu tanda dipergunakan untuk menandai keadaan suatu peralatan/beban, maka lampu tanda mempergunakan warna-warna yang berbeda-beda bergantung pada kondisi peralatan/beban yang ditandai. Tabel 2.1 dibawah ini merupakan warna- warna yang menunjukkan fungsi dari lampu tanda.

Tabel 2.13 Fungsi Warna Lampu Tanda

Kondisi peralatan/Beban	Warna lampu
Kondisi tidak normal, beban lebih bahaya.	Merah
Hati-hati, perhatian	Kuning
Posisi siap, mulai beroperasi	Hijau
Beroperasi normal	Putih

Lampu tanda tidak jauh berbeda dengan lampu penerangan biasa, biasanya lampu ini mempunyai tahanan dalam yang besar sehingga dayanya rata-



rata kecil. Lampu tanda juga sama seperti lampu penerangan biasa yang mempunyai bentuk bermacam-macam yang biasa dilihat pada gambar 2.27 dibawah ini.



Gambar 2.27 Lampu Tanda

2.6 Relay

Relay adalah suatu piranti yang bekerja berdasarkan elektromagnetik untuk menggerakkan sejumlah kontaktor (saklar) yang tersusun. Kontaktor akan tertutup (On) atau terbuka (Off) karena efek induksi magnet yang dihasilkan kumparan (induktor) ketika dialiri arus listrik. Berbeda dengan saklar dimana pergerakan kontaktor (On/Off) dilakukan manual tanpa perlu arus listrik.

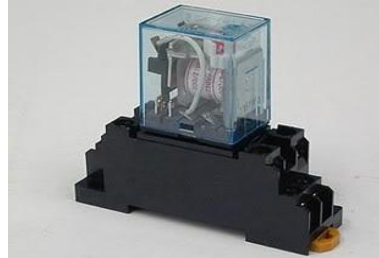
Sebagai komponen elektronika, relay mempunyai peran penting dalam sebuah sistem rangkaian elektronika dan rangkaian listrik untuk menggerakkan sebuah perangkat yang memerlukan arus besar tanpa terhubung langsung dengan perangkat pengendali yang mempunyai arus kecil. Dengan demikian relay dapat berfungsi sebagai pengaman.

Ada beberapa jenis relay berdasarkan cara kerjanya yaitu:

- Normally On : Kondisi awal kontaktor tertutup (On) dan akan terbuka (Off) jika relay diaktifkan dengan cara memberi arus yang sesuai pada kumparan (coil) relay. Istilah lain kondisi ini adalah Normaly Close (NC).
- Normally Off : Kondisi awal kontaktor terbuka (Off) dan akan tertutup jika relay diaktifkan dengan cara memberi arus yang sesuai pada kumparan (coil) relay. Istilah lain kondisi ini adalah Normaly Open (NO).
- Change-Over (CO) atau Double-Throw (DT) : Relay jenis ini memiliki dua



pasang terminal dengan dua kondisi yaitu Normaly Open (NO) dan Normaly Close(NC).



Gambar 2.28 Relay

2.7 Buzzer

Buzzer adalah sebuah komponen elektronika yang berfungsi untuk mengubah getaran listrik menjadi getaran suara. Pada dasarnya prinsip kerja buzzer hampir sama dengan loud speaker, jadi buzzer juga terdiri dari kumparan yang terpasang pada diafragma dan kemudian kumparan tersebut dialiri arus sehingga menjadi elektromagnet, kumparan tadi akan tertarik ke dalam atau keluar, tergantung dari arah arus dan polaritas magnetnya, karena kumparan dipasang pada diafragma maka setiap gerakan kumparan akan menggerakkan diafragma secara bolak-balik sehingga membuat udara bergetar yang akan menghasilkan suara. Buzzer biasa digunakan sebagai indikator bahwa proses telah selesai atau terjadi suatu kesalahan pada sebuah alat (alarm).



Gambar 2.29 Buzzer



2.8 Motor Arus Searah (DC)⁷

Motor listrik merupakan perangkat elektromagnetis yang mengubah energi listrik menjadi energi mekanik. Energi mekanik ini digunakan untuk, misalnya memutar *impeller* pompa, *fan* atau *blower*, menggerakkan kompresor, mengangkat bahan dan lain-lain. Motor listrik digunakan juga di rumah (*mixer*, bor listrik, kipas angin) dan di industri. Motor listrik kadangkala disebut “kuda kerja” nya industri sebab diperkirakan bahwa motor menggunakan sekitar 70% beban listrik total di industri.

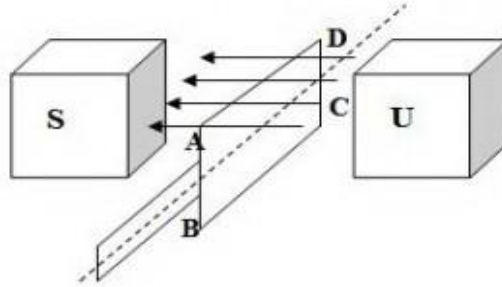
Motor DC memerlukan suplai tegangan yang searah pada kumparan medan untuk diubah menjadi energi mekanik. Kumparan medan pada motor DC disebut stator (bagian yang tidak berputar) dan kumparan jangkar disebut rotor (bagian yang berputar). Jika terjadi putaran pada kumparan jangkar dalam pada medan magnet, maka akan timbul tegangan (GGL) yang berubah-ubah arah pada setiap setengah putaran, sehingga merupakan tegangan bolak-balik. Prinsip kerja dari arus searah adalah membalik fasa tegangan dari gelombang yang mempunyai nilai positif dengan menggunakan komutator, dengan demikian arus yang berbalik arah dengan kumparan jangkar yang berputar dalam medan magnet. Bentuk motor paling sederhana memiliki kumparan satu lilitan yang bisa berputar bebas di antara kutub-kutub magnet permanen. Untuk melihat bentuk serta komponen dari motor DC dapat dilihat pada Gambar 2.4. Motor DC memiliki 2 bagian dasar :

- Bagian yang tetap/stasioner yang disebut stator. Stator ini menghasilkan medan magnet, baik yang dibangkitkan dari sebuah koil (elektro magnet) ataupun magnet permanen.
- Bagian yang berputar disebut rotor. Rotor ini berupa sebuah koil dimana arus listrik mengalir.

⁷ Zuhal, *Dasar Teknik Tenaga Listrik dan Elektronika Daya* (Jakarta: Gramedia, 1988). hlm. 57.



2.8.1 Momen Putar



Gambar 2.30 Momen Putar Pada Motor DC

Kumparan ABCD terletak dalam medan magnet serba sama dengan kedudukan sisi aktif AD dan CB yang terletak tepat lurus arah *fluks* magnet. Sedangkan sisi AB dan DC ditahan pada bagian tengahnya, sehingga apabila sisi AD dan CB berputar karena adanya gaya lorentz, maka kumparan ABCD akan berputar.

Hasil perkalian gaya dengan jarak pada suatu titik tertentu disebut momen, sisi aktif AD dan CB akan berputar pada porosnya karena pengaruh momen putar (T). Setiap sisi kumparan aktif AD dan CB pada gambar diatas akan mengalami momen putar sebesar :

$$T = F \cdot r \dots\dots\dots (2.1)$$

Dimana :

T = momen putar (Nm)

F = gaya tolak (newton)

r = jarak sisi kumparan pada sumbu putar (meter)

Pada daerah dibawah kutub-kutub magnet besarnya momen putar tetap karena besarnya gaya lorentz. Hal ini berarti bahwa kedudukan garis netral sisi-sisi kumparan akan berhenti berputar. Supaya motor dapat berputar terus dengan baik, maka perlu ditambah jumlah kumparan yang digunakan. Kumparan-kumparan harus diletakkan sedemikian rupa sehingga momen putar yang dialami setiap sisi kumparan akan saling membantu dan menghasilkan putaran yang baik. Dengan pertimbangan teknis, maka kumparan-kumparan yang berputar tersebut



dililitkan pada suatu alat yang disebut jangkar, sehingga lilitan kumparan itu pun disebut lilitan jangkar.

Adapun arus listrik yang melewati kumparan akan menyebabkan terbentuknya GGL lawan (E_b) pada kumparan sebesar :

$$E_b = \frac{\Phi \cdot n \cdot P}{a} \dots\dots\dots (2.2)$$

Persamaan tegangan secara umum dapat ditulis sebagai berikut :

$$E_b = K.n. \Phi \dots\dots\dots (2.3)$$

Dimana :

E_b = GGL lawan (volt)

P = jumlah kutub-kutub motor

a = jumlah cabang sisi kumparan

n = jumlah penghantar

Φ = *fluks* per kutub (maxwell)

2.8.2 Bagian-Bagian Motor DC

Bagian Atau Komponen Utama Motor DC adalah sebagai berikut :

1. Kutub Medan Magnet

Secara sederhana digambarkan bahwa interaksi dua kutub magnet akan menyebabkan perputaran pada motor DC. Motor DC memiliki kutub medan yang stasioner dan kumparan motor DC yang menggerakkan bearing pada ruang diantara kutub medan. Motor DC sederhana memiliki dua kutub medan: kutub utara dan kutub selatan. Garis magnetik energi membesar melintasi bukaan diantara kutub-kutub dari utara ke selatan. Untuk motor yang lebih besar atau lebih kompleks terdapat satu atau lebih elektromagnet. Elektromagnet menerima listrik dari sumber daya dari luar sebagai penyedia struktur medan.

2. Kumparan Motor DC

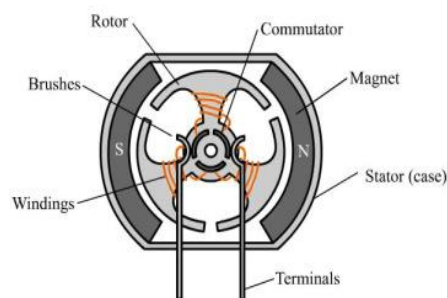
Bila arus masuk menuju kumparan motor DC, maka arus ini akan menjadi elektromagnet. kumparan motor DC yang berbentuk silinder, dihubungkan ke as penggerak untuk menggerakkan beban. Untuk kasus motor DC yang kecil, kumparan motor DC berputar dalam medan magnet yang dibentuk oleh kutub-



kutub, sampai kutub utara dan selatan magnet berganti lokasi. Jika hal ini terjadi, arusnya berbalik untuk merubah kutub-kutub utara dan selatan kumparan motor DC.

3. Kommutator Motor DC

Komponen ini terutama ditemukan dalam motor DC. Kegunaannya adalah untuk membalikan arah arus listrik dalam kumparan motor DC. Commutator juga membantu dalam transmisi arus antara kumparan motor DC dan sumber daya.



Gambar 2.31 Komponen Motor DC

Keuntungan utama motor DC adalah sebagai pengendali kecepatan, yang tidak mempengaruhi kualitas pasokan daya. Motor ini dapat dikendalikan dengan mengatur:

- Tegangan dinamo – meningkatkan tegangan dinamo akan meningkatkan kecepatan
- Arus medan – menurunkan arus medan akan meningkatkan kecepatan.

Hubungan antara kecepatan, flux medan dan tegangan dinamo ditunjukkan dalam persamaan berikut:

$$E = K\Phi N \dots\dots\dots (2.4)$$

$$T = K\Phi I_a \dots\dots\dots (2.5)$$

$$K = \frac{P.Z}{2\pi a} \dots\dots\dots (2.6)$$

Dimana:

E = gaya elektromagnetik yang dikembangkan pada terminal dinamo (volt)

Φ = flux medan yang berbanding lurus dengan arus medan

N = kecepatan dalam RPM (putaran per menit)

T = torque elektromagnetik



I_a = arus dinamo

K = konstanta persamaan

2.9 Bendungan

Bendungan adalah konstruksi yang dibangun untuk menahan laju air menjadi waduk, danau, atau tempat rekreasi. Sering kali bendungan juga digunakan untuk mengalirkan air ke sebuah pembangkit listrik tenaga air. Kebanyakan dam juga memiliki bagian yang disebut pintu air untuk membuang air yang tidak diinginkan secara bertahap atau berkelanjutan.



Gambar 2.32 Bendungan

2.9.1 Fungsi Bendungan

1. Sebagai pembangkit listrik

Listrik tenaga air adalah sumber utama listrik di dunia. Banyak Negara memiliki sungai dengan aliran air yang memadai dan dapat dibendung.

2. Untuk menstabilkan aliran air /irigasi.

Bendungan sering digunakan untuk mengontrol dan menstabilkan aliran air untuk pertanian dan irigasi. Bendungan dapat membantu menstabilkan atau mengembalikan tingkat air danau. Bendungan dapat digunakan untuk menyimpan air yang membantu kebutuhan manusia secara langsung.

3. Untuk mencegah banjir

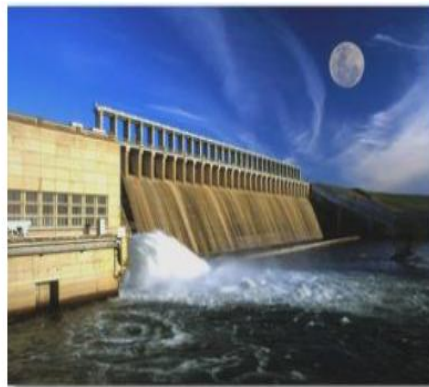
Bendungan diciptakan untuk pengendalian banjir.



2.9.2 Komponen Bendungan

1. Badan bendungan (*body of dams*)

Adalah tubuh bendungan yang berfungsi sebagai penghalang air. Bendungan umumnya memiliki tujuan untuk menahan air, sedangkan struktur lain seperti pintu air atau tanggul digunakan untuk mengelola atau mencegah aliran air ke dalam daerah tanah yang spesifik.



Gambar 2.32 Badan Bendungan

2. Pondasi (*foundation*)

Adalah bagian dari bendungan yang berfungsi untuk menjaga kokohnya bendungan.



Gambar 2.33 Pondasi Bendungan



3. Pintu air (*gates*)

Digunakan untuk mengatur, membuka dan menutup aliran air di saluran baik yang terbuka maupun tertutup. Bagian yang penting dari pintu air adalah :

a. Daun pintu (*gate leaf*)

Adalah bagian dari pintu air yang menahan tekanan air dan dapat digerakkan untuk membuka, mengatur dan menutup aliran air.

b. Rangka pengatur arah gerakan (*guide frame*)

Adalah alur dari baja atau besi yang dipasang masuk ke dalam beton yang digunakan untuk menjaga agar gerakan dari daun pintu sesuai dengan yang direncanakan.



Gambar 2.34 Pintu Air

c. Angker (*anchorage*)

Adalah baja atau besi yang ditanam di dalam beton dan digunakan untuk menahan rangka pengatur arah gerakan agar dapat memindahkan muatan dari pintu air ke dalam konstruksi beton.

d. *Hoist*

Adalah alat untuk menggerakkan daun pintu air agar dapat dibuka dan ditutup dengan mudah.



Gambar 2.35 Hoist

Berdasarkan cara pengoperasiannya, pintu air dibedakan menjadi 3 macam, yaitu :

1. Pintu Air Manual

Penggunaan pintu air secara manual sering kita jumpai pada pengaturan irigasi pada persawahan dan aliran dengan tekanan kecil. Pintu air manual ini masih memerlukan tenaga manusia untuk mengatur aliran air dengan menutup dan membuka pintu air ini.

2. Pintu Air Semi Otomatis

Penggunaan pintu air semi otomatis banyak digunakan pada bendungan yang bertekanan tinggi. Pintu air semi biasanya menggunakan sistem hidrolik untuk membuka dan menutup pintu air, sehingga masih menggunakan tenaga manusia untuk mengontrolnya.

3. Pintu Air Otomatis

Pintu air *full* otomatis digunakan untuk pengendalian banjir pada bangunan pelimpah pada suatu bendungan bertekanan tinggi. Pintu akan bekerja apabila debit air melebihi batas tertentu akan membuka sendiri secara otomatis. Buka tutup pintu air otomatis merupakan bangunan berserta instalasinya yang berfungsi mengatur aliran air yang masuk ke bendungan atau waduk, berdasarkan level ketinggian air pada hulu bendungan. Dengan melihat kondisi sekarang ini cuaca tidak dapat ditebak. Hujan dan badai angin sering datang dengan cepat dan bersamaan. Serta hujan yang terjadi di hulu yang mengakibatkan aliran air yang besar, sangatlah penting adanya alat yang dapat membuka dan menutup aliran air pada bendungan yang dapat bekerja sewaktu-waktu dengan cepat dengan gerakan



membuka, mengatur dan menutup sendiri secara otomatis.

2.10 Debit Aliran Air⁸

Debit aliran merupakan jumlah volume air yang mengalir dalam waktu tertentu melalui suatu penampang air, sungai, saluran, pipa atau kran. Aliran air dikatakan memiliki sifat ideal apabila air tidak dapat dimanfaatkan dan berpindah tanpa mengalami gesekan, hal ini berarti pada gerakan air tersebut memiliki kecepatan yang tetap pada masing-masing titik dalam pipa dan gesekannya beraturan akibat pengaruh gravitasi bumi.

Pengukuran merupakan salah satu hal yang sangat penting dalam suatu sistem pengolahan air. Pada praktiknya, terdapat beberapa metode yang digunakan untuk mengetahui debit air pada saluran terbuka, diantaranya.

1. *Dilution Timed*
2. *Gravimetric*
3. *Weir atau flume*
4. *Area velci*

Dari beberapa teknik pengukuran diatas penulis menggunakan metode pengukuran *Timed Gravimetric*, alasannya karena pada metode ini cara pengukurannya sangat sesuai digunakan pada pengujian yang akan dilakukan dengan menggunakan wadah yang telah diketahui volumenya kemudian dilakukan pengukuran waktu yang diperlukan untuk mengosongkan bak penampungan tersebut menggunakan stopwatch.

Untuk menentukan debit air menggunakan persamaan:

$$Q = \frac{V}{t} \dots\dots\dots (2.7)$$

Dimana:

Q = Debit (liter/s)

V = Volume (liter)

t = Waktu (s)

⁸Nuryanti Pratiwi, *Pengaturan Flow Debit Air pada Kran Otomatis Menggunakan Sensor Infrared* (Bandung: UNIKOM, 2013). hlm. 27.



Untuk menentukan volume air pada bak menggunakan persamaan

$$V = p \times l \times h \dots\dots\dots (2.8)$$

Dimana:

V = Volume (liter)

p = Panjang bak penampungan

l = Lebar bak penampungan

h = Tinggi air

Sedangkan untuk menentukan lama pembukaan menggunakan persamaan

$$\text{Lama pembukaan} = \frac{\text{Volume air yang akan dibuang}}{\text{Debit air}} \dots\dots\dots (2.9)$$

2.11 Hukum II Newton

Besarnya percepatan a berbanding lurus dengan besarnya gaya F dan berbanding terbalik dengan konstanta k yang merupakan ukuran kuantitas benda yang besarnya selalu tetap, selanjutnya disebut massa benda. Hukum ini dikenal sebagai hukum II Newton, dan secara matematis dapat ditulis sebagai berikut:

$$F = m \cdot a \dots\dots\dots (2.10)$$

dengan:

F = gaya (newton)

m = massa (kg)

a = percepatan (m/s^2)

Berat suatu benda (w) adalah besarnya gaya tarik bumi terhadap benda tersebut dan arahnya menuju pusat bumi (vertikal ke bawah).

Hubungan massa dan berat :

$$w = m \cdot g \dots\dots\dots (2.11)$$

dengan :

w = gaya berat. (N).

m = massa benda (kg).

g = percepatan gravitasi (m/s^2).

Perbedaan antara massa dan berat

a) Massa (m) merupakan besaran skalar besarnya di sembarang tempat untuk



suatu benda yang sama selalu tetap.

b) Berat (w) merupakan besaran vektor di mana besarnya tergantung pada tempatnya (percepatan gravitasi pada tempat benda berada).

2.12 Usaha Dan Daya

1. Usaha

Dalam ilmu fisika, usaha mempunyai arti, jika sebuah benda berpindah tempat sejauh s karena pengaruh F yang searah dengan perpindahannya maka usaha yang dilakukan sama dengan hasil kali antara gaya dan perpindahannya, secara matematis dapat ditulis sebagai berikut:

$$W = F \cdot s \dots\dots\dots(2.12)$$

atau

$$W = m \cdot g \cdot h \dots\dots\dots(2.13)$$

dengan :

W = usaha (joule)

s = Perpindahan (m)

2. Daya

Daya (P) adalah usaha yang dilakukan tiap satuan waktu, secara matematis didefinisikan sebagai berikut:

$$P = \frac{W}{t} \dots\dots\dots(2.14)$$

dengan :

W = Usaha (joule)

t = Waktu (sekon)