

## BAB II

### KAJIAN TEORITIK DAN KERANGKA BERPIKIR

#### 2.1. Kajian Teoritik

##### 2.1.1. Pengertian Rancang Bangun

Rancang bangun adalah suatu istilah umum untuk membuat atau mendesain suatu objek dari awal pembuatan sampai akhir pembuatan. Dalam pengertian lain rancang bangun memiliki arti teknik *modeling* 3D objek-objek 3D primitif yang disusun sedemikian rupa untuk membentuk berbagai objek 3D yang lebih kompleks.<sup>1</sup> Rancang bangun berawal dari kata desain yang artinya perancangan, rancang, desain, bangun. Sedangkan merancang artinya mengatur, mengerjakan atau melakukan sesuatu dan perancangan artinya proses, cara, perbuatan merancang. Dapat disimpulkan arti kata desain adalah proses, cara, perbuatan dengan mengatur segala sesuatu sebelum bertindak atau merancang.

Dari kedua pengertian diatas dapat disimpulkan bahwa rancang bangun merupakan tahapan-tahapan untuk menghasilkan sebuah hasil yang diinginkan dengan cara membuat dan mendesain objek yang diinginkan dengan melalui serangkaian proses.<sup>2</sup>

##### 2.1.2. Pengertian Prototipe

Menurut Kamus Besar Bahasa Indonesia Prototipe memiliki arti model yang mula (model asli) yang menjadi contoh atau standar ukuran dari sebuah entitas.

---

<sup>1</sup>Bayu Adjie, *Modeling dan Animasi dengan 3D Studio MAX 7.X*, PT Elex Media Komputindo, Jakarta, h. 41.

<sup>2</sup>Wahyu, *Proposal Penelitian*, diakses dari [Wahyumau.blogspot.com/2012/12/proposal-penelitian](http://Wahyumau.blogspot.com/2012/12/proposal-penelitian), pada tanggal : 19 Oktober 2015 jam 20.18

Dalam bidang desain, sebuah prototipe dibuat sebelum dikembangkan atau justru dibuat khusus untuk pengembangan sebelum dibuat dalam skala sebenarnya atau sebelum diproduksi secara massal.

Prototipe bisa diartikan juga sebagai bentuk awalnya saja dan tidak menutup kemungkinan bisa dikembangkan menjadi skala yang lebih besar.<sup>3</sup>Jadi jangan heran apabila banyak prototipe yang dibuat dengan ukuran yang sangat kecil, ini bertujuan untuk membuat sebuah model awal program, rancangan perangkat-perangkat ataupun sebuah sistem. Terdapat 3 pendekatan utama prototipe, yaitu :

a. *Throw-Away*

Prototipe dibuat dan dites. Pengalaman yang diperoleh dari pembuatan prototipe digunakan untuk membuat produk akhir (*final*), kemudian prototipe tersebut dibuang (tidak dipakai).

b. *Incremental*

Produk akhirnya dibuat sebagai komponen-komponen yang terpisah. Desain produk akhirnya secara keseluruhan hanya ada satu tetapi dibagi dalam komponen-komponen lebih kecil yang terpisah (*independent*).

c. *Evolutionary*

Pada metode ini, prototipenya tidak dibuang tetapi digunakan untuk iterasi desain berikutnya. Dalam hal ini, sistem atau produk yang sebenarnya dipandang sebagai evolusi dari versi awal yang sangat terbatas menuju produk *final* atau produk akhir.

---

<sup>3</sup>Ian Sommerville, *Software Engineering (Rekayasa Perangkat Lunak)*, Erlangga, Jakarta, h. 175.

### **2.1.3. Bahan Bakar**

Bahan bakar adalah suatu materi yang bisa diubah menjadi energi. Biasanya bahan bakar mengandung energi panas yang dapat dilepaskan dan dimanipulasi. Kebanyakan bahan bakar digunakan manusia melalui proses pembakaran (reaksi redoks) dimana bahan bakar tersebut akan melepaskan panas setelah direaksikan dengan oksigen di udara. Proses lain untuk melepaskan energi dari bahan bakar adalah melalui reaksi eksotermal dan reaksi nuklir (seperti Fisi nuklir atau Fusi nuklir). Hidrokarbon (termasuk di dalamnya bensin dan solar) sejauh ini merupakan jenis bahan bakar yang paling sering digunakan manusia. Bahan bakar lainnya yang bisa dipakai adalah logam radioaktif.

#### **2.1.3.1. Jenis Bahan Bakar Minyak**

Bensin merupakan nama umum untuk beberapa jenis BBM yang diperuntukkan untuk mesin dengan pembakaran dengan pengapian. Di Indonesia terdapat beberapa jenis bahan bakar jenis bensin yang memiliki nilai mutu pembakaran berbeda. Nilai mutu jenis BBM bensin ini dihitung berdasarkan nilai RON (*Random Octane Number*). Berdasarkan RON tersebut maka BBM bensin dibedakan menjadi 3 jenis yaitu:

- a. Premium (RON 88) : Premium adalah bahan bakar minyak jenis distilat berwarna kekuningan yang jernih. Warna kuning tersebut akibat adanya zat pewarna tambahan. Penggunaan premium pada umumnya adalah untuk bahan bakar kendaraan bermotor bermesin bensin, seperti : mobil, sepeda motor, motor tempel dan lain-lain. Bahan bakar ini sering juga disebut motor gasoline atau petrol.

- b. Pertamax (RON 92) : ditujukan untuk kendaraan yang mempersyaratkan penggunaan bahan bakar beroktan tinggi dan tanpa timbal (*unleaded*). Pertamax juga direkomendasikan untuk kendaraan yang diproduksi diatas tahun 1990 terutama yang telah menggunakan teknologi setara dengan *electronic fuel injection* dan *catalytic converters*.
- c. Pertamax Plus (RON 95) : Jenis BBM ini telah memenuhi standar *performance International World Wide Fuel Charter* (WWFC). Ditujukan untuk kendaraan yang berteknologi mutakhir yang mempersyaratkan penggunaan bahan bakar beroktan tinggi dan ramah lingkungan. Pertamax Plus sangat direkomendasikan untuk kendaraan yang memiliki kompresi *ratio* > 10,5 dan juga yang menggunakan teknologi *Electronic Fuel Injection* (EFI), *Variable Valve Timing Intelligent* (VVTI), (VTI), *Turbochargers* dan *catalytic converters*.

Beberapa keunggulan dari Pertamax dan Pertamax Plus dibandingkan Premium adalah:

- a. Mempunyai bilangan oktan yang tinggi.  
Produsen mobil cenderung memproduksi kendaraan yang menggunakan perbandingan kompresi mesin yang tinggi. (Perbandingan kompresi mesin adalah perbandingan volume silinder sebelum dan sesudah kompresi). Hal ini dimaksudkan agar tenaga mesin menjadi besar dan kendaraan dapat melaju dengan kecepatan tinggi. Mesin demikian membutuhkan bensin dengan bilangan oktan yang tinggi.
- b. Meningkatkan kinerja mesin agar mesin makin bertenaga

Pertamax dan Pertamax Plus memiliki stabilitas oksidasi yang tinggi dan juga mengandung aditif generasi terakhir. Pembakaran bensin menjadi semakin sempurna sehingga kinerja mesin bertambah baik.

c. Bersifat ramah lingkungan

Pertamax dan Pertamax Plus tidak mengandung Pb yang bersifat racun. Pembakaran yang semakin sempurna juga dapat mengurangi kadar emisi gas polutan seperti CO dan Nox.

d. Lebih ekonomis dari segi harga bahan bakar dan biaya perawatan.

Pertamax dan Pertamax Plus sudah mengandung aditif sehingga praktis dan tepat takarannya. Aditif juga dapat melindungi mesin sehingga dapat menekan biaya perawatan.

#### **2.1.4. Pertamina**

Pertamina adalah singkatan dari Pertamina mini. Ini adalah kios kecil yang mengecer BBM kepada kendaraan bermotor khususnya kendaraan roda dua. Pertamina menjual bensin Premium dengan harga eceran Rp 8.000,- per liter, dimana biasanya harga resmi di pom bensin Pertamina adalah Rp 7.500,-

Selain di Jakarta, kini Pertamina menjamur hingga ke kawasan Bekasi, Tangerang dan Depok. Pertamina menjual bensinnya dari satu buah drum dengan dilengkapi alat *nozzle* seperti pada SPBU umumnya.

Pertamina sendiri dianggap membantu para pengendara motor yang kehabisan bensin di jalan yang letaknya jauh dari SPBU. Namun, kehadiran Pertamina ternyata bukanlah resmi dari Pertamina. Mengingat adanya aturan bahwa

pemerintah jelas tidak berhak mengeluarkan izin distribusi BBM oleh selain Pertamina. Dan adanya kesulitan yang dialami Pertamina dalam melarang penjualan bensin eceran ala pertamini ini.

Terlepas dari sisi positif dan negatifnya Pertamini ternyata merupakan usaha kreatif yang dapat membantu para pengendara yang kehabisan bahan bakar dan jauh jangkauan dari SPBU. Alangkah baiknya apabila inovasi ini dibarengi dengan izin yang jelas dan keamanan yang memenuhi standar. Cara kerja pertamini di jelaskan pada gambar 2.1 di bawah ini.



**Gambar 2.1 Pertamini**

1. Simpan bensin kedalam penampungan utama (seperti gambar No.1 pada gambar 2.1) dengan cara masukan bensin yang sudah di beli dari SPBU, melalui tempat mencurahkan BBM yang telah di sediakan (seperti gambar No.6 pada gambar 2.1)
2. Setelah bensin tersimpan ke dalam penampungan lalu gunakan Pompa (seperti gambar No.2 pada gambar 2.1), dengan alat pompa manual ini bensin akan naik ke penampungan ke 2. Caranya cukup diputar beberapa

kali pengayuh pompanya dan bensin akan mengisi penampungan ke 2 yang terletak dibagian atas (seperti gambar No.3 pada gambar 2.1)

3. Di tangki penampungan ke 2 ini sudah ada garis-garis pembatas ukuran 1 sampai 5 liter (maksimum 5 liter) seperti gambar 2.2 di bawah ini.



**Gambar 2.2 Tabung penampung**

4. Setelah tabung penampungan ke 2 terisi kini bahan bakar siap untuk di jual dengan cara bensin yang ada pada tabung penampung di salurkan ke konsumen dengan menggunakan *nozzle* (seperti gambar No.5 pada gambar 2.1) Melalui selang yang memiliki ini bensin dialirkan ke tangki motor/mobil. Tinggal tekan *nozzle* nya maka bensin akan mengalir.
5. Untuk penjualan di malam hari kios pertamini ini telah di lengkapi dengan *neon box* (seperti gambar No.4 pada gambar 2.1).
6. Jika di malam hari Kios Pertamina ingin di masukan ke dalam rumah atau memindahkan ketempat yang lebih aman, telah di pasang roda di bagian bawah (seperti gambar No.7 pada gambar 2.1).

## **2.1.5. PLC (*Programmable Logic Controller*)**

### **2.1.5.1. Sejarah PLC**

PLC pertama kali diperkenalkan pada tahun 1960-an. Alasan utama perancangan PLC adalah untuk menghilangkan beban ongkos perawatan dan penggantian sistem kontrol mesin berbasis *relay*. Bedford Associate (Bedford, MA) mengajukan usulan yang diberi nama MODICON (kepanjangan *Modular Digital controller*) untuk perusahaan-perusahaan mobil di Amerika. Sedangkan perusahaan lain mengajukan sistem berbasis komputer (PDP-8). MODICON 084 merupakan PLC pertama di dunia.

Saat kebutuhan produksi berubah maka demikian pula dengan sistem kontrolnya. Hal ini menjadi sangat mahal jika perubahannya terlalu sering. Karena relai merupakan alat mekanik, maka relai memiliki umur hidup atau masa penggunaan yang terbatas, yang akhirnya membutuhkan jadwal perawatan yang ketat. Pelacakan kerusakan atau kesalahan menjadi cukup membosankan jika banyak relai yang digunakan. Bayangkan saja sebuah panel kontrol yang dilengkapi dengan monitor ratusan hingga ribuan relai yang terkandung pada sistem kontrol tersebut. Bagaimana kompleksnya melakukan pengkabelan pada relai-relai tersebut. Dengan demikian, pengontrol baru ini harus memudahkan para teknisi perawatan dan teknisi lapangan melakukan pemrograman. Umur alat harus menjadi lebih panjang dan program proses dapat dimodifikasi atau dirubah dengan lebih mudah. Serta harus mampu bertahan dalam lingkungan industri yang keras.



Pada pertengahan tahun 1970-an, teknologi PLC yang dominan adalah sekuenser mesin-kondisi dan CPU berbasis *bit-slice*. Prosesor AMD 2901 dan 2903 cukup populer digunakan dalam MODICON dan PLC A-B. Mikroprosesor konvensional kekurangan daya dalam menyelesaikan logika PLC secara cepat, kecuali PLC kecil. Setelah mikroprosesor konvensional mengalami perbaikan dan pengembangan, PLC konvensional mulai banyak menggunakannya. Walaupun begitu, hingga saat ini ada yang masih berbasis pada AMD 2903.

Kemampuan komunikasi pada PLC mulai muncul pada awal-awal tahun 1973. Sistem yang pertama adalah *Modbus*-nya MODICON. Dengan demikian PLC bisa berkomunikasi dengan PLC lain dan bisa ditempatkan lebih jauh dari lokasi mesin sesungguhnya yang dikontrol. Sayangnya, kurangnya standarisasi mengakibatkan komunikasi PLC menjadi mimpi buruk untuk protokol-protokol dan jaringan-jaringan yang tidak kompatibel. Tetapi bagaimanapun juga, saat itu merupakan tahun yang hebat untuk PLC.

Pada tahun 1980-an dilakukan usaha untuk menstandarisasi komunikasi dengan protokol otomasi pabrik milik General Motor MAP (*Manufacturing Automation Protocol*). Saat itu juga merupakan waktu untuk memperkecil ukuran PLC dan pembuatan perangkat lunak pemrograman melalui pemrograman simbolik dengan komputer PC daripada terminal pemrogram atau penggunaan pemrogram genggam (*handheld programmer*). Sekarang PLC terkecil seukuran dengan sebuah kontrol relai tunggal. Pada tahun 1990-an dilakukan reduksi protokol baru dan modernisasi lapisan fisik dari protokol-protokol populer yang bertahan sampai dengan tahun 1980-an. Standar terakhir (IEC 1131-3).<sup>4</sup>

---

<sup>4</sup> <http://www.kelas-mikrokontrol.com/e-learning/plc/sejarah-plc.html> diakses pada tanggal : 20 Oktober 2015 jam 15.11.

Seiring perkembangan teknologi *solid state*, saat ini PLC telah mengalami perkembangan luar biasa, baik dari ukuran, kepadatan komponen serta dari segi fungsionalnya. Beberapa peningkatan perangkat keras dan perangkat lunak ini diantaranya adalah:

- a. Ukuran semakin kecil dan kompak.
- b. Jumlah *input-output* yang semakin banyak dan padat
- c. Waktu eksekusi program yang semakin cepat.
- d. Pemrograman relatif semakin mudah. Hal ini terkait dengan perangkat lunak pemrograman yang semakin *user friendly*.
- e. Memiliki kemampuan komunikasi dan sistem dokumentasi yang semakin baik.
- f. Jenis instruksi/fungsi semakin banyak dan lengkap.

#### **2.1.5.2. Pengertian PLC**

*Programmable Logic Controllers* (PLC) adalah komputer elektronik yang mudah digunakan dan memiliki fungsi kendali untuk berbagai tipe dan tingkat kesulitan yang beraneka ragam. Definisi *Programmable Logic Controller* menurut Capiel (1982) adalah “sistem elektronik yang beroperasi secara digital dan didesain untuk pemakaian di lingkungan industri, dimana sistem ini menggunakan memori yang dapat diprogram untuk penyimpanan secara *internal* instruksi-instruksi yang mengimplementasikan fungsi-fungsi spesifik seperti logika, urutan, perwaktuan, pencacahan dan operasi aritmatik untuk mengontrol mesin atau proses melalui modul-modul I/O digital maupun analog”.

Berdasarkan namanya, konsep PLC adalah sebagai berikut :

a. *Programmable*

Menunjukkan kemampuan dalam hal memori untuk menyimpan program yang telah dibuat yang dengan mudah diubah-ubah fungsi atau kegunaannya.

b. *Logic*

Menunjukkan kemampuan dalam memproses *input* secara aritmatik dan *logic* yakni melakukan operasi membandingkan, menjumlahkan, mengalikan, membagi, mengurangi, negasi, AND, OR, dan lain sebagainya.

c. *Controller*

Menunjukkan kemampuan dalam mengontrol dan mengatur proses sehingga menghasilkan *output* yang diinginkan.

PLC dirancang untuk menggantikan suatu rangkaian *relay* sekuensial 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. Jenis PLC dapat dilihat pada gambar 2.3 pada halaman berikutnya.



**Gambar 2.3** PLC Omron Sysmac CP1E

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*. 1 menunjukkan bahwa keadaan yang diharapkan terpenuhi sedangkan 0 berarti keadaan yang diharapkan tidak terpenuhi. PLC juga dapat diterapkan untuk pengendalian sistem yang memiliki *output* banyak.

### 2.1.5.3. Fungsi PLC

Fungsi dan kegunaan PLC sangat luas. Praktiknya, PLC dapat dibagi secara umum dan secara khusus. Secara umum fungsi PLC adalah sebagai berikut :

a. *Sekuensial Control*

PLC memproses *input* sinyal biner menjadi *output* yang digunakan untuk keperluan pemrosesan teknik secara berurutan. PLC menjaga agar semua langkah dalam proses berlangsung dalam urutan yang tepat.

b. *Monitoring Plant*

PLC secara terus menerus memonitor status suatu sistem (misalnya temperatur, tekanan, tingkat ketinggian) dan mengambil tindakan yang diperlukan sehubungan dengan proses yang dikontrol (misalnya nilai sudah melebihi batas) atau menampilkan pesan tersebut pada operator.

Sedangkan fungsi PLC secara khusus adalah dapat memberikan *input* ke CNC (*Computerized Numerical Control*). Beberapa PLC dapat memberikan *input* ke CNC untuk kepentingan pemrosesan lebih lanjut. CNC bila dibandingkan dengan PLC mempunyai ketelitian yang lebih tinggi dan lebih mahal harganya. CNC biasanya dipakai untuk proses *finishing*, membentuk benda kerja, *moulding* dan sebagainya.<sup>5</sup>Keuntungan menggunakan PLC adalah sebagai berikut:

a. Fleksibel

PLC dapat mengontrol lebih dari satu alat dan programnya mudah dimodifikasi dalam jangka waktu yang relatif singkat.

b. Jumlah kontak bantu tak terbatas

PLC mempunyai jumlah kontak yang banyak untuk setiap koil dalam pemrogramannya, dapat mencapai ratusan kontak untuk satu kontaktor, tergantung dari kapasitas memori.

---

<sup>5</sup><https://vavarivistava.wordpress.com/2012/12/06/plc/> diakses pada tanggal : 20 Oktober 2015 jam 19.15.

c. Mudah diamati

Operasi program PLC dapat dilihat selama operasi pada layar monitor. Jika operasi mengalami kesalahan maka dapat diketahui. Hal ini dikarenakan rangkaian logika terlihat terang pada layar monitor jika mendapat tegangan/terhubung.

d. Kecepatan operasi

Kecepatan operasi program PLC sangat cepat. Kecepatan operasi logika PLC dibatasi oleh *scan time*, yang membutuhkan beberapa mili/detik.

e. Metode pemrograman dengan diagram tangga

Pemrograman *ladder diagram* PLC dapat diselesaikan dalam model diagram tangga bukan dalam bentuk teks sehingga lebih mudah untuk dilihat dan dipahami.

#### 2.1.5.4. Prinsip Kerja PLC

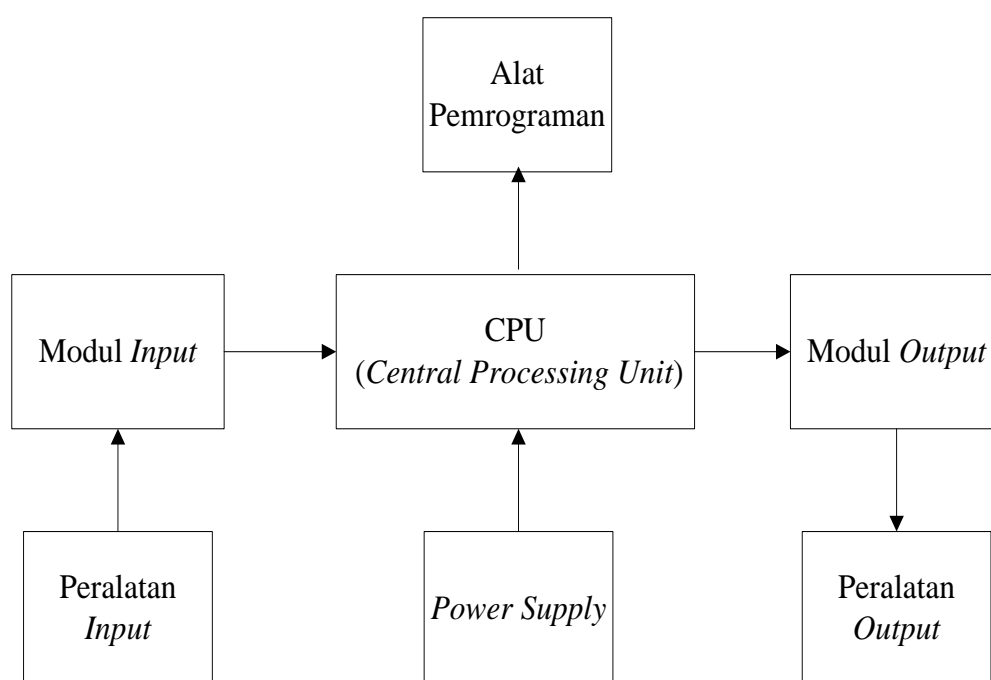
Prinsip kerja PLC dapat diuraikan sebagai berikut : PLC merupakan peralatan elektronik yang dibangun dari mikroprosesor untuk memonitor keadaan dari peralatan *input* dan kemudian dianalisa sesuai dengan kebutuhan perencanaan (*programmer*) untuk mengontrol keadaan *output*.<sup>6</sup>

Pada prinsipnya PLC bekerja dengan cara menerima data-data dari peralatan *input* luar seperti pada gambar 2.4 Peralatan *input* dapat berupa saklar, tombol, dan sensor. Data-data yang masuk dari peralatan input ini berupa sinyal diskrit atau *analog*. Modul *input* ini akan mengidentifikasi serta mengubah sinyal yang masuk tersebut ke dalam bentuk tegangan yang sesuai oleh CPU sehingga

---

<sup>6</sup>Handy Wicaksono, *Programmable Logic Controller (Teori, Pemrograman dan Aplikasinya dalam Otomasi Sistem)*, Graha Ilmu, Yogyakarta, h. 50.

menjadi sinyal-sinyal digital. Kemudian oleh CPU yang ada di dalam PLC, sinyal-sinyal digital tersebut akan diolah berdasarkan program yang telah disimpan dalam memori dan selanjutnya sinyal tersebut dikirim ke modul *output*. Bentuk sinyal digital ini akan diubah oleh modul *output* menjadi sinyal yang dapat digunakan untuk menjalankan peralatan *output* yang dapat berupa lampu, katup, motor, kontaktor, ataupun *relay*. Peralatan *output* inilah yang nantinya akan mengoperasikan sistem atau proses yang akan dikontrol.



**Gambar 2.4** Arsitektur PLC

Sebuah PLC berisi CPU (*Central Processing Unit*) atau otak dari PLC yang berisi sebuah aplikasi program, modul *interface input* dan *output* yang terhubung secara langsung ke *field I/O devices* dan lebih lengkapnya PLC dapat diartikan sebuah alat kontrol yang bekerja berdasarkan pada pemrograman dan eksekusi instruksi logika. PLC beroperasi dengan cara memeriksa *input* dari

sebuah proses guna mengetahui statusnya kemudian sinyal *input* ini diproses berdasarkan instruksi logika yang telah diprogram dalam memori.

Pada PLC juga dipersiapkan *internal input* dan *output* untuk proses dalam PLC sesuai dengan kebutuhan program, Dimana *internal input* dan *output* ini hanya sebagai *flag* dalam proses. Di dalam PLC juga dipersiapkan *timer* yang dapat dibuat dalam konfigurasi *on delay*, *off delay*, *on timer*, *off timer* dan lain-lain sesuai dengan programnya.

Untuk memproses, *timer* PLC tersebut memanggil berdasarkan alamatnya. Untuk melaksanakan fungsi sebagai kontrol sistem, PLC didukung oleh perangkat lunak yang merupakan bagian penting dari program PLC dan sebagai hasilnya berupa sinyal *output*. Sinyal *output* inilah yang dipakai untuk mengendalikan peralatan atau mesin. Antarmuka (*interface*) yang terpasang di PLC memungkinkan PLC dihubungkan secara langsung ke *actuator* atau *transducer* tanpa memerlukan *relay*.

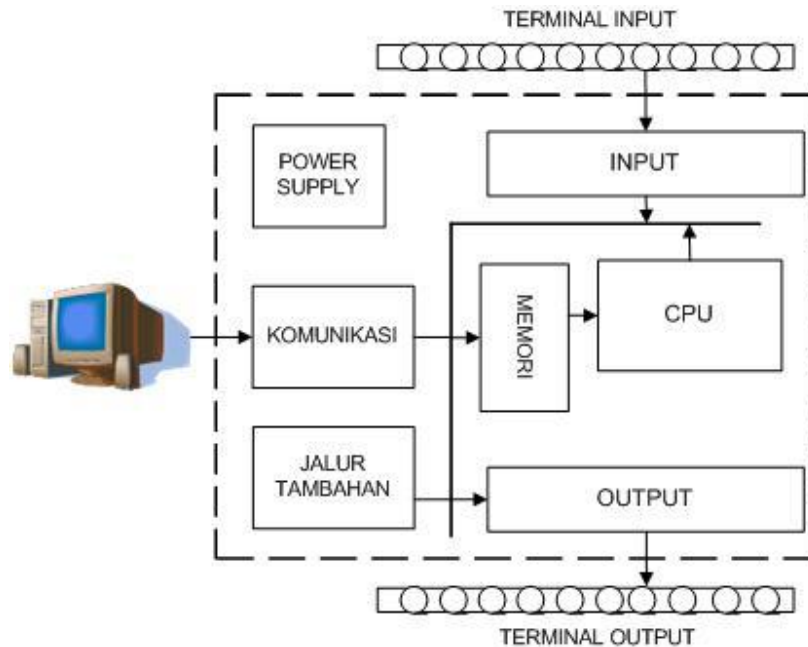
#### **2.1.5.5. Komponen pada PLC**

Pada kenyataannya PLC merupakan suatu sistem mikrokontroler yang digunakan untuk keperluan industri. PLC dapat dikatakan sebagai suatu perangkat keras dan lunak yang dapat dibuat untuk diaplikasikan dalam dunia industri.<sup>7</sup> Secara umum PLC memiliki bagian-bagian yang sama dengan komputer maupun mikrokontroler, yaitu CPU, Memori dan I/O. Susunan komponen PLC dapat dilihat pada Gambar 2.5 pada halaman berikutnya.

---

<sup>7</sup>Husanto Thomas, *op.cit.*, h. 6.





**Gambar 2.5 Komponen Pada PLC**

Sumber: Husanto, Thomas, ST, MT, PLC (*Programmable Logic Controller*) FP Sigma, C.V Andi Offse, Yogyakarta.

a. *Central Processing Unit (CPU)*

CPU berfungsi untuk mengambil instruksi dari memori, kemudian mendekodekannya lalu mengeksekusi instruksi tersebut. Selama proses tersebut CPU akan menghasilkan sinyal kontrol, memindahkan data ke I/O *port* atau sebaliknya, melakukan fungsi aritmatika dan logika juga mendeteksi sinyal dari luar CPU. CPU bertugas menghubungkan peralatan *input* dan *output*. CPU juga menangani masalah komunikasi data dengan piranti *eksternal*, interkoneksi antar bagian-bagian *internal* PLC, eksekusi program, manajemen memori, mengawasi atau mengamati masukan dan memberikan sinyal ke *output*. Kontroler PLC ini memiliki kemampuan untuk memeriksa memori secara teratur dan kompleks untuk mengetahui rusak atau

tidaknya memori pada PLC. Untuk mengetahui kesalahan atau kerusakan pada PLC dapat dilihat pada indikator LED PLC.

b. Memori

Memori sistem ini berfungsi untuk menyimpan sistem operasi dan juga untuk menyimpan program yang harus dijalankan dalam bentuk *biner*, hasil terjemahan *ladder diagram* yang dibuat oleh pemrogram. Data-data yang terdapat pada memori dapat diubah (dikosongkan atau dihapus). Pemrograman PLC biasanya dilakukan melalui kanal serial komputer yang dipakai untuk memprogram. Memori pengguna terdiri dari beberapa blok yang memiliki fungsi khusus. Beberapa bagian pada memori digunakan untuk menyimpan status *input* dan *output*. Status yang sesungguhnya dari *input* maupun *output* akan disimpan sebagai *logic*/bilangan 0 dan 1. Masing-masing *input* berkaitan dengan *bit* yang terdapat dalam memori. Bagian lain dalam memori digunakan untuk menyimpan variabel yang telah dituliskan di dalam program.

PLC menggunakan peralatan memori semi konduktor seperti RAM (*Random Acces Memory*), ROM (*Read Only Memory*), dan PROM (*Programmable Read Only Memory*). RAM mempunyai waktu akses yang cepat dan program-program yang terdapat didalamnya dapat diprogram ulang sesuai dengan keinginan pemakaiannya. RAM disebut juga sebagai *volatile memory*, maksudnya program-program yang terdapat mudah hilang jika *supply* listrik padam. Dengan demikian untuk mengatasi *supply* listrik yang padam tersebut maka diberi *supply* cadangan daya listrik berupa baterai yang disimpan pada RAM. Seringkali CMOS RAM dipilih untuk pemakaian *power*

yang rendah. Baterai ini mempunyai jangka waktu kira-kira lima tahun sebelum harus diganti.

c. *Pemrograman PLC*

*Controller PLC* dapat diprogram melalui komputer, namun ada juga cara lain untuk memprogramnya yaitu dengan cara memprogram manual atau yang biasa disebut dengan konsol (*console*). Untuk melakukan pemrograman manual ini dibutuhkan *software* yang sesuai dengan produk PLC nya, karena masing-masing produk PLC membutuhkan *software*-nya sendiri-sendiri. Dalam memprogram ulang PLC sangat penting diperhatikan bahwa saat sistem diperbaiki program yang benar dan sesuai harus disimpan kedalam PLC lagi. Penting dilakukan pemeriksaan program PLC apakah selama penyimpanan tidak terjadi perubahan atau sebaliknya, apakah program sudah berjalan benar atau tidak. Karena dengan pemeriksaan ini dapat menghindari situasi berbahaya dalam ruang produksi (pabrik).

d. *Power Supply PLC*

*Power supply* digunakan untuk memberikan aliran daya ke seluruh bagian PLC termasuk CPU, Memori, dan lain-lain. Rata-rata PLC bekerja dengan daya 24 VDC atau 220 VAC. Untuk PLC besar catu daya biasanya terpisah sebagai modul sendiri sedangkan untuk PLC kecil catu dayanya sudah menyatu dengan PLC nya.

e. *Input PLC*

Kecerdasan suatu sistem yang terotomasi sangat bergantung pada kemampuan sebuah PLC untuk membaca sinyal dari berbagai macam jenis

sensor dan piranti *input* yang lain. Untuk mendeteksi proses yang sedang terjadi maka dibutuhkan sensor-sensor yang tepat untuk masing-masing kondisi. Sinyal-sinyal yang telah terdeteksi sensor tersebut dapat berupa *logic* (*on* atau *off*) maupun *analog*. Pada PLC kecil biasanya hanya mampu menerima *input digital* saja namun untuk PLC besar mampu menerima *input analog* dari *unit* khusus yang sesuai dengan PLC nya.

Setiap *input* memiliki alamat dan nomor urutan khusus yang digunakan selama membuat program untuk memonitor satu persatu aktivitas *input* didalam program. Indikasi urutan status dari *input* ditandai LED (*Light Emitting Diode*) pada PLC atau modul *input*, hal ini dimaksudkan untuk memudahkan pengecekan proses pengoperasian *input* dari PLC itu sendiri.

f. *Interfacing Input*

*Interfacing input* berada dijalur *input* dan CPU. Tujuan elemen ini adalah untuk melindungi CPU dari sinyal-sinyal yang tidak diinginkan dan dapat merusak CPU tersebut. Modul *interfacing input* ini berfungsi untuk mengkonversi sinyal-sinyal *input* dari luar ke sinyal yang sesuai dengan tegangan kerja CPU. Contoh *input* tegangan pada sensor yaitu 22 VDC maka harus dikonversikan ke 5 VDC agar sesuai dengan tegangan kerja CPU.

g. *Output PLC*

Sistem terotomasi yang memiliki *input* pastinya juga memiliki *output*. Tidak akan lengkap jika tidak ada jalur *input* yang digunakan untuk menghubungkan ke alat-alat *eksternal*. Contoh alat yang banyak digunakan yaitu motor, solenoida, relai, lampu indikator, *speaker*, dan sebagainya.

*Output* ini dapat berupa *analog* maupun *digital*. *Output* digital akan dianalogikan seperti saklar yang dapat memutus atau menyambungkan jalur. Untuk *output analog* contohnya dapat digunakan merubah tegangan untuk pengendalian motor secara regulasi *linier* sehingga memperoleh kecepatan yang diinginkan.

#### h. *Interfacing Output*

Seperti halnya *interfacing input*, *output* juga memerlukan *interfacing* yang sama digunakan untuk memberikan perlindungan antara CPU dengan peralatan *eksternal* agar tidak terjadi kerusakan pada CPU-nya.

#### i. Jalur Tambahan

Setiap PLC biasanya memiliki jumlah *input* dan *output* yang terbatas. Jika diperlukan jumlah ini dapat ditambahkan dengan menggunakan modul *input* dan *output* tambahan.

### 2.1.5.6. Instruksi – Instruksi Dalam Pemrograman PLC

#### 1. Instruksi Dasar

Instruksi dasar merupakan instruksi yang digunakan untuk membuat rangkaian logika dari diagram tangga pada pemrograman PLC.<sup>8</sup> Instruksi dasar pada pemrograman diagram tangga ini ada enam, yaitu :

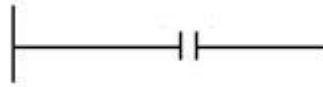
#### a. LD

LD atau singkatan dari *Load*, merupakan instruksi untuk memulai program garis atau blok pada rangkaian logika yang dimulai dengan kontak

---

<sup>8</sup>Husanto Thomas, *Op.Cit.*, hlm.47.

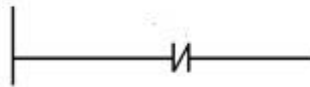
NO (*Normally Open*) *input* seperti terlihat pada gambar 2.6 gambar dibawah ini.



**Gambar 2.6 Instruksi LD**

b. NOT

Instruksi dasar NOT berfungsi untuk membentuk suatu kontak NC (*Normally Close*) *input* seperti pada gambar 2.7 di bawah ini.



**Gambar 2.7 Instruksi NOT**

c. OUT

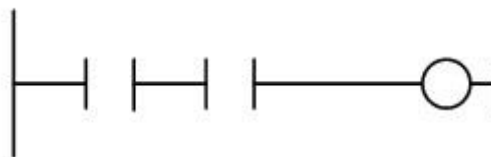
OUT merupakan instruksi untuk memasukkan program koil *output*. Kontak-kontak dari masing-masing koil *output* dapat digunakan beberapa kali sesuai yang diinginkan. Simbol intruksi OUT bisa dilihat pada gambar 2.8



**Gambar 2.8 Instruksi OUT**

d. AND

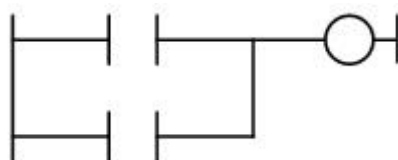
Instruksi AND ini digunakan untuk menghubungkan dua atau lebih kontak-kontak *input* atau *output* secara seri seperti pada gambar 2.9 di bawah ini.



**Gambar 2.9 Instruksi AND**

## e. OR

Instruksi dasar OR digunakan untuk menghubungkan dua atau lebih kontak-kontak *input* atau *output* secara paralel seperti terlihat pada gambar 2.10 di bawah ini.



**Gambar 2.10 Instruksi OR**

## f. END

Instruksi dasar END untuk menyatakan rangkaian kontrol yang dibuat telah berakhir. Instruksi END ini harus selalu dimasukkan dalam penulisan program, karena apabila akhir rangkaian kontrol tidak dilengkapi dengan instruksi END, maka program tersebut tidak akan dieksekusi oleh CPU.

## 2. Instruksi Gabungan

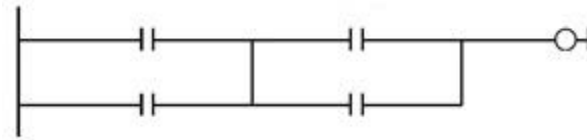
Instruksi gabungan merupakan suatu instruksi yang menggunakan dua buah instruksi dasar atau lebih untuk menggabungkan dua blok rangkaian dalam program.<sup>9</sup> Instruksi gabungan tersebut adalah sebagai berikut :

## a. AND LD

Instruksi ini merupakan gabungan dari instruksi AND dan LD yang digunakan untuk menggabungkan dua blok rangkaian dalam secara seri seperti terlihat pada gambar 2.11 pada halaman berikutnya.

---

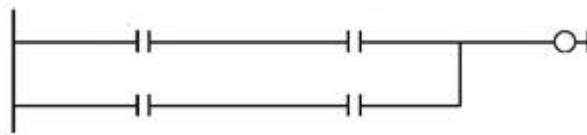
<sup>9</sup>Syufrijal, *op.cit.*, h. 6.



**Gambar 2.11 Instruksi AND LD**

b. OR LD

Instruksi ini digunakan untuk menggabungkan dua blok dalam rangkaian secara parallel seperti terlihat pada gambar 2.12 di bawah ini.

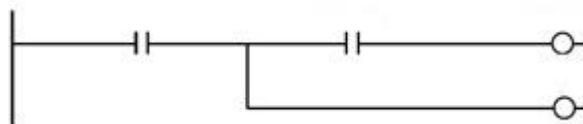


**Gambar 2.12 Instruksi OR LD**

c. Instruksi Garis Bercabang (*Temporary Relay*)

Instruksi garis bercabang merupakan suatu instruksi yang mempunyai sebuah garis yang terdiri dari dua instruksi atau lebih dan letaknya setelah *input*. Instruksi garis bercabang tersebut terdapat pada *temporary relay* (TR). *Temporary relay* adalah *relay* bantu yang digunakan pada rangkaian yang mempunyai dua atau lebih percabangan dari *relay output*, *timer* atau *counter*. Beberapa TR dapat digunakan sesuai dengan kebutuhan, akan tetapi tidak dapat melebihi dari jumlah yang ditetapkan pabrik. Dalam rangkaian, satu nomer TR hanya dapat digunakan sekali saja, sedangkan untuk pengalamatan TR lainnya digunakan pada rangkaian lainnya secara berurutan. Penggunaan TR dapat dilihat pada gambar 2.13 pada halaman berikutnya.





**Gambar 2.13 Instruksi Garis Bercabang**

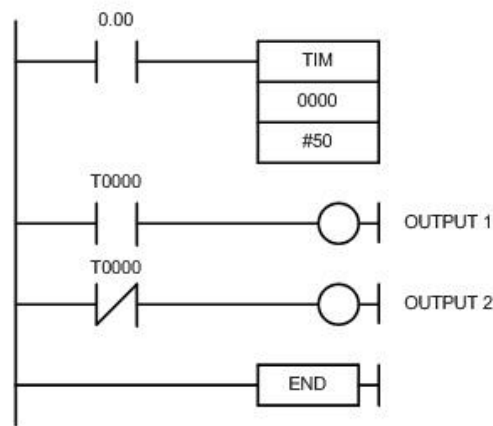
### 3. Instruksi *Timer* dan *Counter*

*Timer* PLC yang banyak digunakan dalam industri adalah *timer on-delay*. Keuntungan *timer* PLC adalah akurasi waktunya tinggi karena PLC menggunakan teknologi *solid-state*. Pada PLC Omron pengalaman *timer* dan *counter* digunakan secara bersama-sama yaitu dari nomer 0 – 127. Cara kerja *timer* dan *counter* hampir mirip, perbedaannya *timer* mencacah pulsa *internal* sedangkan *counter* mencacah pulsa dari luar.

#### a. *Timer*

*Timer* adalah suatu instruksi yang membuat suatu proses berhenti sesaat sebelum kembali melanjutkan proses.<sup>10</sup> *Timer* berfungsi untuk mengaktifkan suatu keluaran dengan interval waktu yang dapat diatur. Pengaturan waktu dilakukan melalui nilai *setting* (*preset value*). *Timer* pada PLC Omron diberi nomer dari 000–127 (T0 – T127). Instruksi *timer* ada 2 macam yaitu *timer* (TIM) dan *high timer* (TIMH). Bedanya pada pengukuran waktu TIM mempunyai pulsa *clock* lebih panjang dibanding TIMH. TIM mempunyai pulsa *clock* sebesar 0,1 detik sedangkan TIMH sebesar 0,01 detik. *Timer* tersebut akan bekerja bila diberi *input* dan mendapat pulsa *clock*. Untuk pulsa *clock* sudah disediakan oleh pembuat PLC. Penggunaan *timer* dapat dilihat pada gambar 2.14 pada halaman berikutnya.

<sup>10</sup>Husanto Thomas, *Op.Cit.*,h.63.



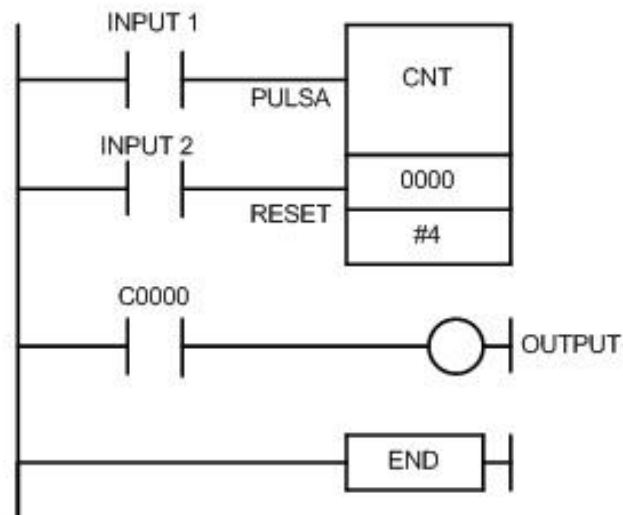
**Gambar 2.14 Contoh Penggunaan Timer**

Saat *input* 0.00 *off*, maka *output* 1 akan *off* dan *output* 2 *on*, tetapi pada saat *input* 0.00 *on* maka *timer* mulai mencacah pulsa dari 0 sampai *preset value* (selama 5 detik) maka akan mengaktifkan *output* 1 dan mematikan *output* 2. Akan tetapi apabila *input* *off* sebelum *timer* mencapai *preset value* maka *timer* akan *off* (*reset*) sehingga menyebabkan *output* 1 *off* dan *output* 2 *on* kembali.

*b. Counter*

Instruksi *counter* pada PLC berfungsi untuk menghitung setiap *input* yang masuk.<sup>11</sup> Pada PLC Omron terdapat *counter* yang diberi nomor dari 0 – 127 (C0 – C127). Penggunaan alamat *counter* ini digunakan bersama-sama dengan *timer*. Oleh sebab itu, dalam satu program, pemberian nomor *counter* tidak boleh sama dengan nomor *timer*. Cara kerja *counter* mirip dengan *timer*, perbedaannya *timer* mencacah pulsa *internal* sedangkan *counter* mencacah pulsa dari luar. Ada 2 sinyal *input* yang digunakan oleh *counter* yaitu sinyal pulsa dan sinyal *reset*. Penggunaan *counter* dapat dilihat pada gambar 2.15 pada halaman berikutnya.

<sup>11</sup>Husanto Thomas, *Op.Cit.*,h.68.



**Gambar 2.15 Contoh Penggunaan Counter**

*Counter* akan mulai mencacah pulsa dari 0 sampai *preset value* ketika terdapat sinyal *input 1* berupa pulsa dan kondisi *input 2* sebagai resetnya *off*. Bila cacahan *counter* sudah mencapai *preset value* yaitu sebanyak 4 kali maka *counter* akan mengaktifkan *output*. Akan tetapi bila *input 2* reset *on* sebelum *counter* mencapai *preset value* maka *counter* akan *off* (*reset*) dan *output* akan *off*.

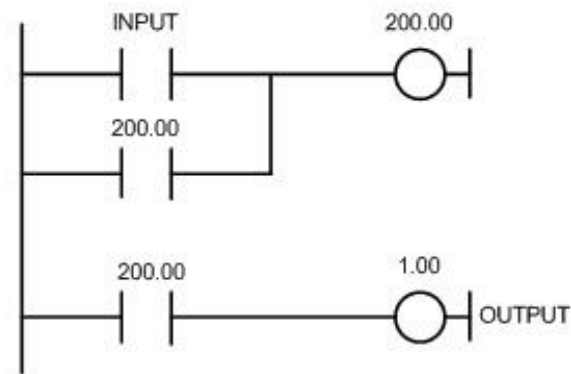
#### 4. *Internal Relay*

*Internal relay* adalah *general purpose relay* yang ada di dalam PLC yang tidak dapat diakses secara langsung untuk digunakan sebagai *input* maupun *output* seperti yang terdapat pada program komponen. *Internal relay* adalah *relay* semu yang merupakan *bit* digital yang disimpan pada *internal image register*. Dari sudut pandang pemrograman, semua PLC mempunyai satu *coil* dan mempunyai sejumlah kontak N/O dan N/C sesuai dengan yang diinginkan *programmer*, contoh penggunaan *internal relay* bisa dilihat pada gambar 2.16.

Semua PLC mempunyai *internal relay* akan tetapi penomeran dan jumlah maksimum yang diperbolehkan tergantung dari merek dan model PLC.

*Internal relay* memberi keleluasaan pada *programmer* untuk melaksanakan operasi *internal* yang lebih rumit tanpa memerlukan penggunaan biaya mahal untuk beberapa *output relay*. Dalam contoh pemrograman pada PLC Omron memakai simbol IR dengan penomeran sebagai berikut :

- a. 200 – 231
- b. 600 – 615
- c. 700 – 715
- d. 800 – 815
- e. 900 – 915



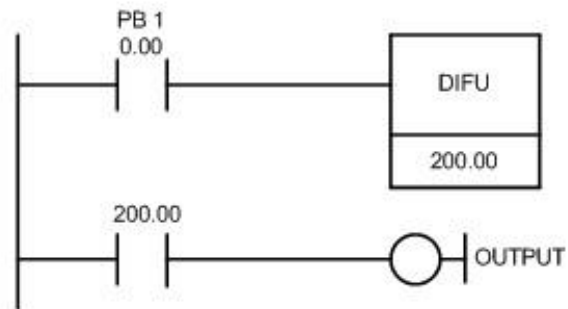
**Gambar 2.16 Internal Relay**

##### 5. DIFU (*Differential Up*)

Aplikasi kontrol ini berfungsi untuk meng-*on* kan *output* selama satu *scan*.<sup>12</sup> Aplikasi DIFU bisa dilihat pada gambar 2.17 Contoh program menggunakan DIFU :

<sup>12</sup>Husanto Thomas, *Op.Cit.*,h.59.

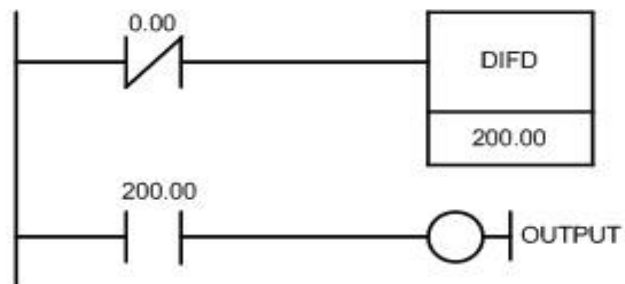
Apabila PB 1 ditekan maka *output* DIFU 200.00 akan *on* dan kontak DIFU 200.00 akan hidup hanya sekejap walaupun tombol PB 1 ditekan lama.



**Gambar 2.17 Contoh Penggunaan DIFU**

#### 6. DIFD (*Diferential Down*)

Aplikasi kontrol ini berfungsi untuk melewatkan satu pulsa *input* (*On-Off*) pada *output* sehingga *output* hanya akan *on* selama satu pulsa (satu *scan*) seperti terlihat pada gambar 2.18 Contoh program menggunakan DIFD : *Output* DIFD 200.00 akan *on* selama satu *scan* setelah *input* 0.00 *off*.



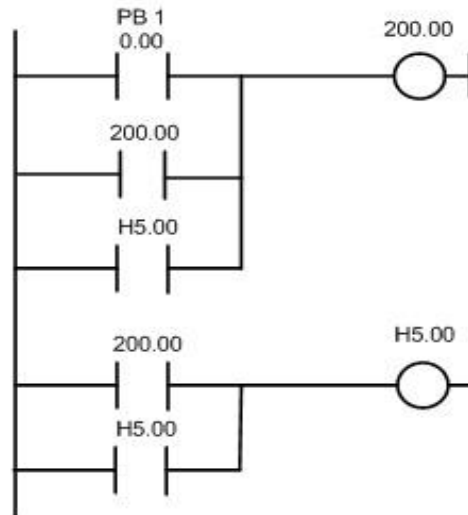
**Gambar 2.18 Contoh Penggunaan DIFD**

#### 7. *Holding Relay*

*Holding relay* adalah *relay internal* yang bisa dipakai untuk menahan sistem yang sedang bekerja walaupun aliran *supply power off* seperti yang terlihat pada gambar 2.19, misalnya jika sumber *power/PLN* mati, apabila pada

program dipasang *holding relay* maka proses bisa tetap lanjut tanpa harus memulai program dari awal lagi. Contoh penggunaan *holding relay* :

Apabila *input* 0.00 PB 1 dihidupkan maka *output* 200.00 dan H5.00 akan hidup, dan apabila sumber *power*/PLN mati maka *output* 200.00 masih tetap *ON* karena program ditahan oleh  *Holding Relay*.



**Gambar 2.19** Contoh Penggunaan  *Holding Relay*

#### 2.1.5.7. *Software CX-Programmer*

*CX-Programmer* merupakan salah satu bentuk perangkat lunak yang digunakan untuk memasukkan program ke dalam PLC. Berikut ini adalah langkah-langkah yang diperlukan dalam membuat program PLC melalui *CX-Programmer*, yaitu :

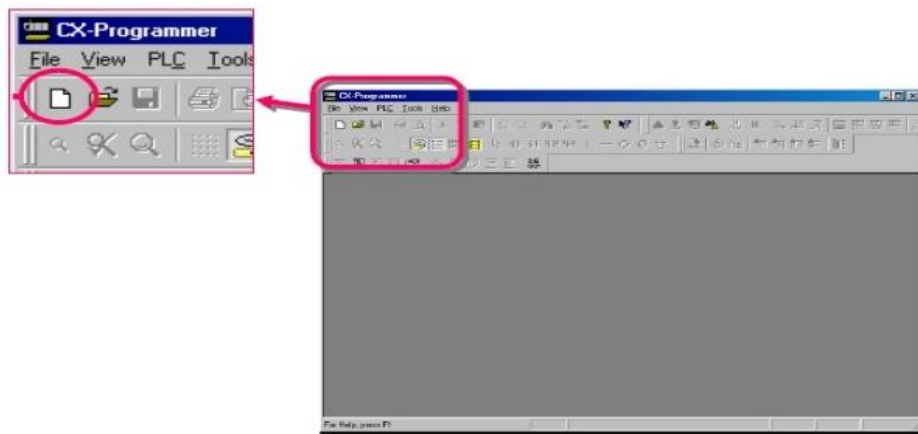
##### 1. *Install Software CX-Programmer*

Dalam penginstalan *CX-Programmer* perlu dipastikan untuk menutup semua *windows* program yang sedang aktif. Jika kita memiliki program *CX-Programmer* versi lama, *uninstall* terlebih dahulu sebelum meng-*install* *CX-Programmer* versi

terbaru. *CX-Programmer* dapat di-*install* mulai dari *OS Windows 95/98/NT 4.0 SP 6, Windows 2000/Me*, hingga *Windows XP* dan *7*.

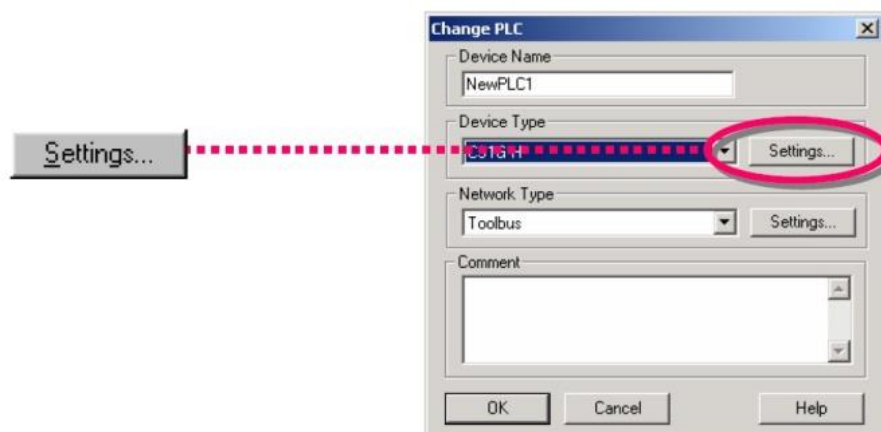
## 2. Membuat *Project* Baru

Untuk membuat *project* baru, langkah yang harus dilakukan pertama kali adalah dengan mengklik [*New*] pada *toolbar* di *CX-Programmer*, seperti pada gambar 2.20 di bawah ini.



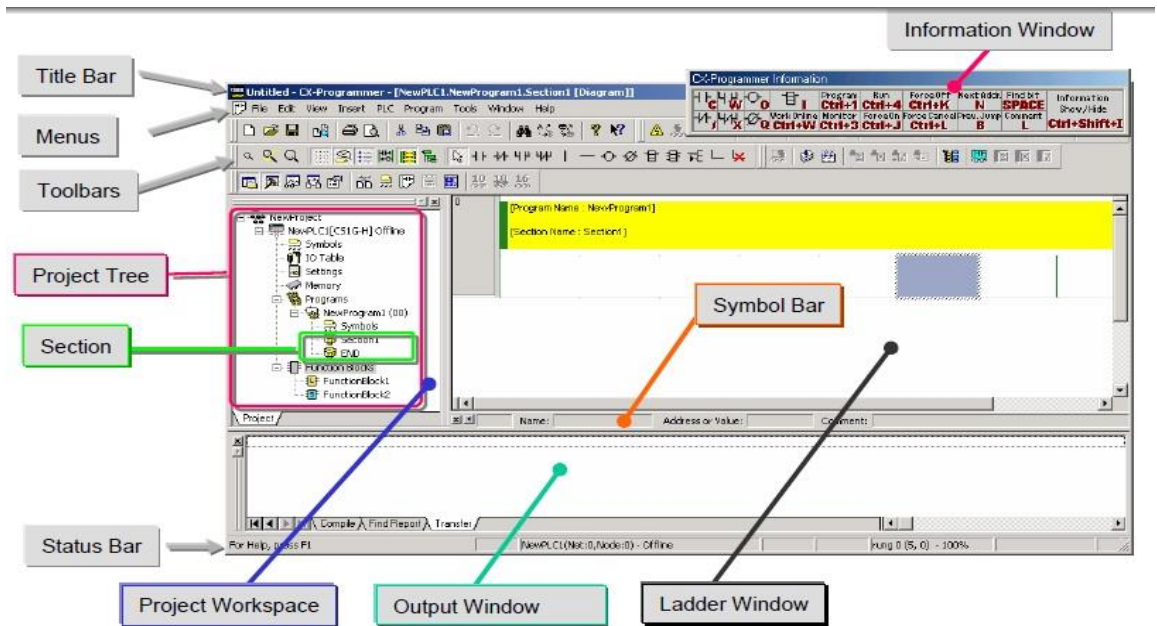
**Gambar 2.20 Tombol *New* pada *Toolbar***

Selanjutnya akan muncul tampilan seperti gambar 2.21 untuk memilih jenis PLC yang akan digunakan, kemudian klik tombol *settings* untuk menampilkan layar (*Device Type Settings*) dan mengubah jenis CPU PLC.



**Gambar 2.21 *Change PLC***

Kemudian klik tombol OK. Maka akan muncul *main window* seperti pada gambar 2.22 di bawah ini.



**Gambar 2.22 Main Window Software CX-Programme**

Fungsi masing-masing menu pada gambar 2.22 dijelaskan pada tabel 2.1 di bawah ini :

**Tabel 2.1. Fungsi Main Window Software CX-Programmer**

Nama	Fungsi
<i>Title Bar</i>	Memperlihatkan nama <i>file</i> yang telah di <i>save</i> pada <i>CX-Programmer</i>
<i>Menu</i>	Untuk memilih <i>item</i> menu
<i>Toolbars</i>	Untuk memilih fungsi yang akan digunakan. Pilih [View] → [Toolbars], untuk memperlihatkan <i>toolbars</i> .



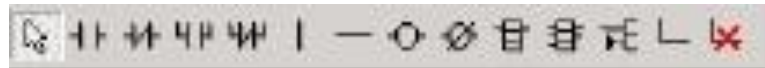
<i>Section</i>	Untuk membagi suatu program dalam beberapa <i>block</i>
<i>Project Workspace</i> <i>Project Tree</i>	Mengontrol program dan data. Dapat digunakan untuk meng- <i>copy</i> data dengan <i>Drag and Drop</i> di antara <i>project</i> yang berbeda atau dalam satu <i>project</i> .
<i>Ladder Window</i>	Layar untuk menulis dan mengedit <i>diagram ladder</i>
<i>Output Window</i>	Menunjukkan <i>error check</i>  Menunjukkan hasil pencarian <i>contacts/coils</i> di <i>list form</i>  Menunjukkan <i>error details</i> ketika terjadi kesalahan dalam suatu <i>file project</i> .
<i>Status Bar</i>	Menunjukkan informasi seperti nama PLC, <i>online/offline</i> , lokasi <i>cell</i> yang aktif.
<i>Information Window</i>	Layar <i>small window</i> untuk menunjukkan <i>basic shortcut keys</i> yang digunakan di <i>CX-Programmer</i> .  Munculkan pilih [View] -> [Information Window].
<i>Symbol Bar</i>	Menunjukkan nama, alamat atau nilai, dan penjelasan dari simbol yang dipilih kursor.

### 3. Membuat Program

Untuk membuat program dengan *ladder diagram*, dapat meng-klik simbol kontak, koil, garis atau fungsi yang diinginkan seperti yang terdapat pada

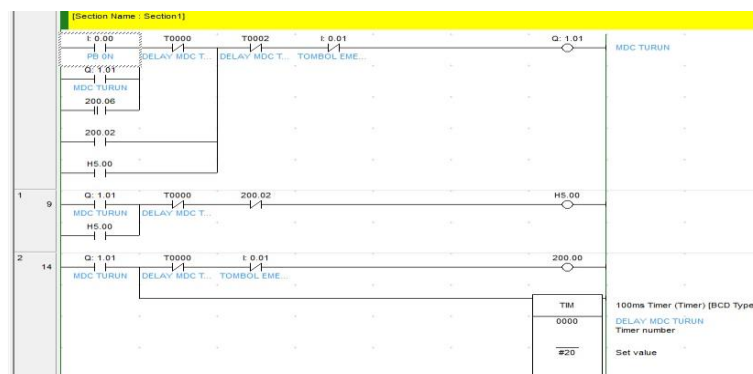
gambar 2.23 untuk membuat suatu program pada PLC, dilakukan langkah-langkah pemrograman sebagai berikut :

- a. Tentukan sistem apa yang akan dikontrol.
- b. Hitung jumlah *input/ output* dan diberi alamatnya.
- c. Membuat *ladder diagram*.
- d. *Test* dan *check* program.



**Gambar 2.23 Simbol Pada *Toolbar***

Lalu buat *ladder diagram* seperti yang terlihat pada gambar 2.24 sesuai logika yang diinginkan untuk ditransfer ke PLC kemudian dijalankan pada alat yang terhubung dengan PLC.<sup>13</sup>



**Gambar 2.24 Contoh *Ladder Diagram***

#### 4. Aturan-aturan pada Pemrograman *Ladder Diagram*

- a. *Output* dapat menjadi *input*, tetapi *input* tidak dapat menjadi *output*.

*Output* PLC dapat berubah menjadi *input*, di mana *input* tersebut baru akan aktif jika *output* diaktifkan. Hal ini dimungkinkan karena *output* tersebut

<sup>13</sup>Syufrijal, *Op.Cit.*,h.47.

merupakan bagian alamat dari PLC. Jadi yang dimanupilasi ialah alamat *output*, bukan peralatan *output* secara fisik.

Namun hal tersebut tidak berlaku untuk *input*, dimana *input* tidak dapat berubah menjadi *output*. Alasannya sederhana saja, untuk mengaktifkan *input*, diperlukan suatu tindakan atau perubahan fisik. Misalkan untuk mengaktifkan tombol harus dilakukan penekanan tombol secara fisik. Hal tersebut tidak bisa dilakukan dari program PLC (meskipun ada fitur *force* pada PLC yang memungkinkan pengecekan *input*, namun fitur tersebut tidak ditujukan dalam konteks pemrograman.

b. *Internal relay* dapat digunakan sebagai perantara

Pada era *relay*, seluruh peralatan *input* dan *output* akan dihubungkan dengan *relay* sebagai pengendali. Pada PLC, sebagai gantinya diberikan *relay virtual* yang disebut *internal relay*. Perbedaan *internal relay* dengan *input* (I) atau *output* (O) ialah tidak ada keharusan menghubungkan alat fisik tertentu pada alamat ini. Sedang pada alamat *input* atau *output*, penggunaan harus benar-benar menghubungkan peralatan secara fisik.

c. *Input* dapat muncul berkali-kali, namun *output* hanya boleh muncul 1 kali.

Seperti halnya *contact* pada *relay*, kontak di PLC dapat muncul berkali-kali dalam suatu *ladder diagram*. Ini adalah salah satu kelebihan PLC dibanding *relay*, karena jumlah *contact* maksimal yang umum beredar di pasaran ialah 4 *contact* saja. Sedangkan jumlah maksimal kontak pada PLC nyaris tak terbatas (hanya dibatasi oleh ketersediaan memori PLC saja).

### 2.1.6. Motor Listrik

Motor listrik termasuk kedalam kategori mesin listrik dinamis dan merupakan sebuah perangkat elektromagnetik yang mengubah energi listrik menjadi energi mekanik. Energi mekanik ini digunakan untuk, misalnya, memutar *impeller* pompa, *fan* atau *blower*, menggerakkan kompresor, mengangkat bahan, dll di industri dan digunakan juga pada peralatan listrik rumah tangga (seperti: *mixer*, bor listrik, kipas angin).

Motor listrik kadangkala disebut “kuda kerja” nya industri, sebab diperkirakan bahwa motor-motor menggunakan sekitar 70% beban listrik total di industri.

Mekanisme kerja untuk seluruh jenis motor listrik secara umum sama, yaitu:

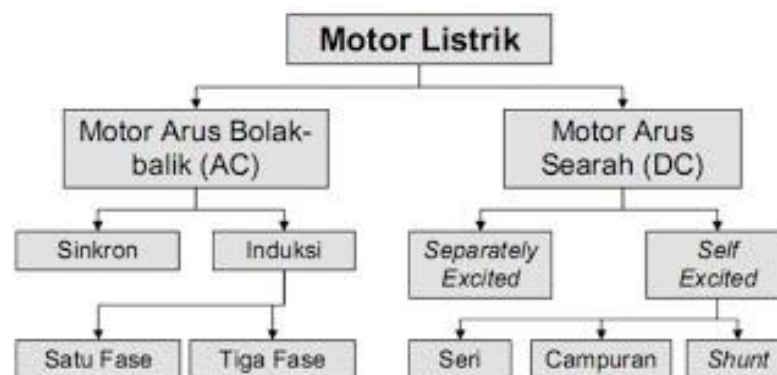
- a. Arus listrik dalam medan magnet akan memberikan gaya.
- b. Jika kawat yang membawa arus dibengkokkan menjadi sebuah lingkaran/*loop*, maka kedua sisi *loop*, yaitu pada sudut kanan medan magnet, akan mendapatkan gaya pada arah yang berlawanan.
- c. Pasangan gaya menghasilkan tenaga putar/ torsi untuk memutar kumparan.
- d. Motor-motor memiliki beberapa *loop* pada dinamanya untuk memberikan tenaga putaran yang lebih seragam dan medan magnetnya dihasilkan oleh susunan elektromagnetik yang disebut kumparan medan.

Dalam memahami sebuah motor listrik, penting untuk mengerti apa yang dimaksud dengan beban motor. Beban mengacu kepada keluaran tenaga putar/torsi sesuai dengan kecepatan yang diperlukan. Beban umumnya dapat dikategorikan kedalam tiga kelompok :

- a. Beban torsi konstan, adalah beban dimana permintaan keluaran energinya bervariasi dengan kecepatan operasinya, namun torsi nya tidak bervariasi. Contoh beban dengan torsi konstan adalah *conveyors*, *rotary kilns*, dan pompa *displacement* konstan.
- b. Beban dengan torsi variabel, adalah beban dengan torsi yang bervariasi dengan kecepatan operasi. Contoh beban dengan torsi variabel adalah pompa sentrifugal dan *fan* (torsi bervariasi sebagai kwadrat kecepatan).
- c. Beban dengan energi konstan, adalah beban dengan permintaan torsi yang berubah dan berbanding terbalik dengan kecepatan.

#### 2.1.6.1. Jenis Motor Listrik

Bagian ini menjelaskan tentang dua jenis utama motor listrik, motor DC dan motor AC. Motor tersebut diklasifikasikan berdasarkan pasokan *input*, konstruksi, dan mekanisme operasi, dan dijelaskan lebih lanjut dalam gambar 2.25 di bawah ini.



**Gambar 2.25 Klasifikasi Motor Listrik**

#### a. Motor DC/Arus Searah

Motor DC/ arus searah, sebagaimana namanya, menggunakan arus langsung yang tidak langsung/*direct-unidirectional*. Motor DC digunakan pada penggunaan khusus dimana diperlukan penyalaan torsi yang tinggi atau percepatan yang tetap untuk kisaran kecepatan yang luas.

Gambar 3 memperlihatkan sebuah motor DC yang memiliki tiga komponen utama:

##### a. Kutub medan

Secara sederhana digambarkan bahwa interaksi dua kutub magnet akan menyebabkan perputaran pada motor DC. Motor DC memiliki kutub medan yang stasioner dan dinamo 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.

##### b. Dinamo

Bila arus masuk menuju dinamo, maka arus ini akan menjadi elektromagnet. Dinamo yang berbentuk silinder, dihubungkan ke as penggerak untuk menggerakkan beban. Untuk kasus motor DC yang kecil, dinamo 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 dinamo.

### c. Kommutator

Komponen ini terutama ditemukan dalam motor DC. Kegunaannya adalah untuk membalikan arah arus listrik dalam dinamo. Kommutator juga membantu dalam transmisi arus antara dinamo dan sumber daya. Keuntungan utama motor DC adalah kecepatannya mudah dikendalikan dan tidak mempengaruhi kualitas pasokan daya. Motor DC ini dapat dikendalikan dengan mengatur:

1. Tegangan dinamo—meningkatkan tegangan dinamo akan meningkatkan kecepatan.
2. Arus medan – menurunkan arus medan akan meningkatkan kecepatan.

Motor DC tersedia dalam banyak ukuran, namun penggunaannya pada umumnya dibatasi untuk beberapa penggunaan berkecepatan rendah, penggunaan daya rendah hingga sedang, seperti peralatan mesin dan *rolling mills*, sebab sering terjadi masalah dengan perubahan arah arus listrik mekanis pada ukuran yang lebih besar. Juga, motor tersebut dibatasi hanya untuk penggunaan di area yang bersih dan tidak berbahaya sebab resiko percikan api pada sikatnya. Motor DC juga relatif mahal dibanding motor AC.

### b. Motor AC/Arus Bolak-Balik

Motor AC/arus bolak-balik menggunakan arus listrik yang membalikkan arahnya secara teratur pada rentang waktu tertentu. Motor listrik AC memiliki dua buah bagian dasar listrik: "*stator*" dan "*rotor*".

*Stator* merupakan komponen listrik statis. *Rotor* merupakan komponen listrik berputar untuk memutar as motor. Keuntungan utama motor DC

terhadap motor AC adalah bahwa kecepatan motor AC lebih sulit dikendalikan. Untuk mengatasi kerugian ini, motor AC dapat dilengkapi dengan penggerak frekuensi variabel untuk meningkatkan kendali kecepatan sekaligus menurunkan dayanya. Motor induksi merupakan motor yang paling populer di industri karena keandalannya dan lebih mudah perawatannya. Motor induksi AC cukup murah (harganya setengah atau kurang dari harga sebuah motor DC) dan juga memberikan rasio daya terhadap berat yang cukup tinggi (sekitar dua kali motor DC).

### **2.1.7. Pompa Air**

Pompa adalah mesin untuk menggerakkan fluida. Pompa menggerakkan fluida dari tempat bertekanan rendah ke tempat dengan tekanan yang lebih tinggi, untuk mengatasi perbedaan tekanan ini maka diperlukan tenaga (energi)<sup>14</sup>.

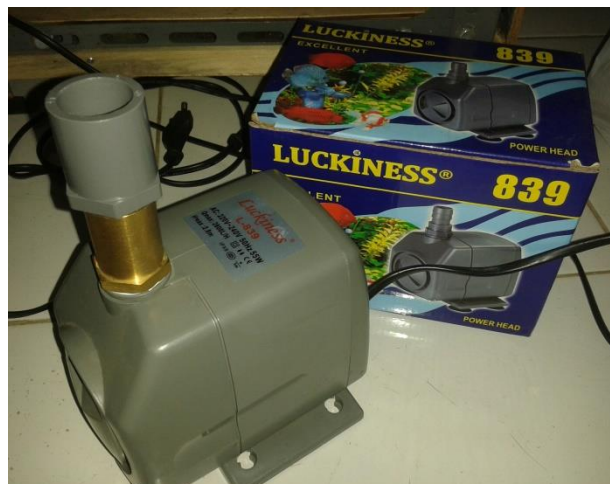
#### **2.1.7.1. Pompa Air Celup (*Submersible*)**

Sesuai namanya, pompa air listrik ini penggunaannya dicelupkan ke dalam air. Penggunaan yang umum adalah pompa air yang dipakai dalam aquarium untuk mengalirkan air ke tempat penyaringan air sehingga air aquarium terjaga kejernihannya untuk waktu yang lebih lama. Contoh pompa air celup seperti gambar 2.26 di halaman berikutnya.

---

<sup>14</sup> <https://id.wikipedia.org/wiki/Pompa>, diakses pada tanggal 28 November 2015 jam 11.51





**Gambar 2.26 Pompa Air Celup (*Submersible*)**

Cara kerjanya pun sama seperti pompa air listrik di atas, memanfaatkan daya Sentrifugal dari perputaran kipas *impeller* untuk mendorong air ke atas. Jenis pompa air celup ini cukup banyak jenisnya, tergantung keperluannya.

#### **2.1.8. *Flow meter***

*Flow meter* adalah sebuah instrument yang berguna untuk mengukur aliran dari suatu fluida baik *liquid*, *sludge*, maupun gas, baik bertemperatur rendah hingga bertemperatur tinggi<sup>15</sup>. Dalam memilih *flow meter* harus disesuaikan dengan kondisi *fluid* dan fungsi *flow meter* itu sendiri. Karakteristik dari fluida yang diukur oleh *flow meter* sangat luas sekali mulai dari tingkat *corosive* fluida dimana untuk fluida yang tingkat keasamannya tinggi mungkin lebih cocok jika digunakan *flow meter* bahan dari PVC / Non Logam. Gambar 2.27 pada halaman selanjutnya adalah contoh *flow meter* berbahan dari PVC.

<sup>15</sup> <https://wiratamaengineering.wordpress.com/category/flow-meter/>, diakses pada tanggal 28 November 2015, jam 13.47.



**Gambar 2.27 Flow Meter Sea (yf-s201)**

**Sumber:** <http://www.dx.com/>

Untuk fluida yang temperaturnya tinggi tentunya digunakan material lain. Untuk fluida yang diaplikasikan pada bahan makanan atau obat-obatan yang menuntut pipa dan *flow meter* yang harus *food grade* maka disarankan untuk menggunakan *stainless steel* SUS 316L. Sedangkan untuk lingkungan yang *corisive* seperti di laut maka *body* dan *flange flow meter* lebih baik jika menggunakan *stainless steel* karena dibutuhkan ketelitian dan pemahaman akan karakteristik *fluid*, manfaat serta fungsi lanjutan dari *flow meter*.

Ada beberapa variabel yang harus ditentukan dalam memilih *flow meter* pada saat penentuan *type* dan model yang cocok dengan aplikasi yang diharapkan, variabel pemilihan *flow meter* tersebut dapat dimasukkan ke dalam pertanyaan sebagai berikut :

1. Jenis *fluid* yang akan digunakan pada *flow meter*, jenis fluida yang diukur oleh *flow meter* dapat berupa : *Steam, air, water, wet gas, liquid gas, chemical, oil, solid, sludge, powder and cement.*

2. Hasil Pengukuran pada *flow meter* bisa berupa *flow rate* dan total volume fluida yang mengalir atau kedua-duanya.
3. *Viscosity* dari *fluid*, kebersihan dan kekotoran dari *fluid* (lumpur, banyak kotoran atau bersih) yang mengalir ke *flow meter*.
4. Tujuan dari *flow meter* : sebagai alat ukur *flow*, total volume, *control*, *switch*, pengiriman sinyal *electric* yang berfungsi sebagai *control* ataupun data ke komputer atau *handphone* melalui sms.
5. Ada juga *flow meter* yang bisa *moveable* yang bisanya banyak digunakan oleh *inspector* atau QC seperti Jenis *Portable* atau *Handheld* yang jenisnya *Ultrasonic Flow meter*.
6. *Body Material* yang digunakan *flow meter* bisa berupa *Carbon Steel*, *Stainless Steel* (310, 304, atau 316), PVC, Alumunium, *Brass* dan yang lainnya tergantung pada aplikasinya.
7. Perlu tidaknya *Display* pada *Flow Meter* (*Local Display* atau *Remote Display*).
8. Besaran (Maksimal dan Minimal) dari *flow Rate*, *pressure work*, *temperature* dari *fluid* yang akan diukur *flow meter*. Besaran *flow rate* bisa kg/mnt, ton/hour, gpm, m3/hour. Untuk besaran *pressure*-nya bisa bar, PSI, Ma, dan untuk besaran temperatur bisa derajat *celcius* atau lainnya. Karena untuk aplikasi dengan *temperature* dan *pressure* yang tinggi, pada dasarnya bisa dipesan sesuai dengan yang diinginkan.

9. Perlu tidaknya sistem kedap air pada *indicator/converter flow meter* (*Waterproof*) serta area yang mudah terbakar atau eksplosif.
10. Penggunaan untuk bahan kimia dan makanan seperti tingkat keasaman dari *fluid* atau perlu *food grade* untuk material *flow meter* yang sering digunakan di *industry* obat dan makanan dan minuman.
11. Ukuran dari pipa dimana *flow meter* ini di *install* termasuk menggunakan sistem sambungan *flange*, ulir atau wafer. Ukuran pipa sendiri terdiri dari 1/2", 3/5", 1", 1,5", 2", 4" s/d 45". Untuk *Connection Flange* bisa dipilih ANSI atau JIS.
12. Sistem Instalasi *flow meter* : *Vertical* atau *Horizontal*, atau bisa juga mengikuti arah aliran
13. Keterangan yang diperlukan dalam memilih jenis *flow meter* karena pada dasarnya *flow meter* bisa dibuat atau dipesan sesuai dengan permintaan *costumer*.

Begitu pentingnya penentuan parameter-parameter kerja fluida diatas untuk menghindari kesalahan pemilihan Jenis *flow meter* atau Model *flow meter*. Hal ini bertujuan agar *flow meter* tersebut dapat bekerja secara maksimal sesuai dengan yang diinginkan.

### **2.1.8.1. YF-S201 Hall Effect Water Flow Meter / Sensor**

Sensor ini duduk sejalan dengan garis air dan mengandung sensor kincir untuk mengukur berapa banyak cairan telah pindah melalui *flow meter*. Ada sebuah ruang magnetik sensor efek terpadu yang output pulsa listrik dengan setiap revolusi. Efek hall sensor disegel dari pipa air dan memungkinkan sensor untuk tetap aman dan kering.

Sensor ini dilengkapi dengan tiga kabel: merah (Power 5-24VDC), hitam (ground) dan kuning (output pulsa efek Hall). Dengan menghitung pulsa dari output dari sensor, dapat dengan mudah menghitung aliran air. Setiap pulsa adalah sekitar 2,25 mililiter. Catatan ini bukan sensor presisi, dan denyut nadi tidak bervariasi sedikit tergantung pada laju aliran, tekanan fluida dan orientasi sensor. Ini akan membutuhkan kalibrasi hati-hati, jika lebih dari 10% presisi diperlukan<sup>16</sup>. Namun, yang besar untuk tugas-tugas pengukuran dasar.

Kita miliki sebagai contoh sketsa Arduino yang dapat digunakan untuk dengan cepat menguji sensor, itu akan menghitung aliran perkiraan air dalam liter / jam.

Sinyal pulsa adalah gelombang persegi sederhana sehingga cukup mudah untuk login dan mengkonversi ke liter per menit dengan menggunakan rumus berikut.

$$\text{Pulse frequency (Hz)} / 7.5 = \text{flow rate in L/min.}$$

Fitur:

Model: YF-S201

---

<sup>16</sup> <http://www.hobbytronics.co.uk/yf-s201-water-flow-meter> diakses pada tanggal : 24 Januari 2016 jam 20.16.

*Sensor Type: Hall effect*

Tegangan kerja: 5 sampai 18V DC (min diuji bekerja tegangan 4.5V)

Max menarik arus: 15mA @ 5V

*Output Type: 5V TTL*

Kerja *Flow Rate*: 1 sampai 30 Liter / Menit

Bekerja Kisaran suhu: -25 sampai + 80 °C

Bekerja Kelembaban Range: 35% -80% RH

Akurasi: ± 10%

Maksimum tekanan air: 2.0 MPa

Duty cycle *Output*: 50% -10% +

Waktu naik *Output*: 0.04us

*Output* waktu jatuh: 0.18us

*Flow rate pulse characteristics: Frequency (Hz) = 7.5 \* Flow rate (L/min)*

Pulsa per Liter: 450

Daya tahan: minimum 300.000 siklus

Panjang kabel: 15cm

1/2 "koneksi pipa nominal, 0,78" diameter luar, 1/2 "benang

Ukuran: 2.5 "x 1.4" x 1.4 "

Rincian sambungan:

Kabel merah: 5 V

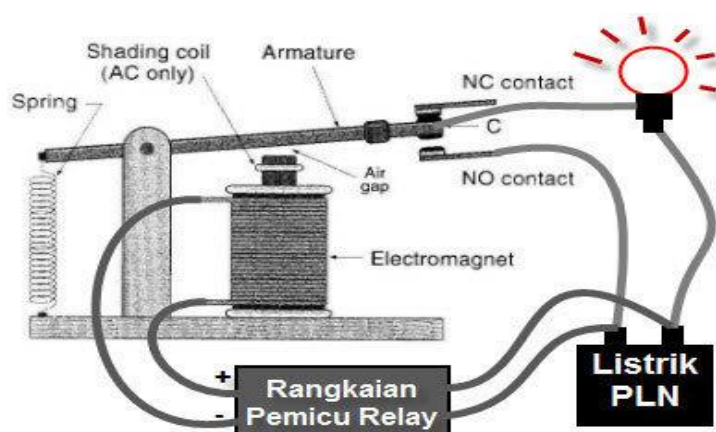
Kabel hitam: GND

Kabel kuning: PWM *output*

### **2.1.9. Relai**

Relai adalah suatu rangkaian *switching* magnetik yang bekerja bila mendapat catu dari rangkaian *trigger*. Relai memiliki tegangan dan arus nominal yang harus dipenuhi *output* rangkaian pengemudinya. Arus yang digunakan pada rangkaian adalah arus DC. Relai adalah komponen listrik yang bekerja berdasarkan prinsip induksi medan elektromagnetis. Jika sebuah penghantar dialiri oleh arus listrik, maka di sekitar penghantar tersebut timbul medan magnet. Medan magnet yang dihasilkan oleh arus listrik tersebut selanjutnya diinduksikan ke logam

ferromagnetis. Logam ferromagnetis adalah logam yang mudah terinduksi medan elektromagnetis. Ketika ada induksi magnet dari lilitan yang membelit logam, logam tersebut menjadi "magnet buatan" yang sifatnya sementara.<sup>17</sup> Cara ini kerap digunakan untuk membuat magnet non permanen. Sifat kemagnetan pada logam ferromagnetis akan tetap ada selama pada kumparan yang melilitinya teraliri arus listrik. Sebaliknya, sifat kemagnetannya akan hilang jika suplai arus listrik ke lilitan diputuskan. Gambar 2.28 di bawah ini menunjukkan ilustrasi kerja dari sebuah relai.



**Gambar 2.28 Ilustrasi dari sebuah relai**

Sumber : [www.elangsakti.com](http://www.elangsakti.com)

Konstruksi dalam suatu relai terdiri dari lilitan kawat (*coil*) yang dililitkan pada inti besi lunak. Jika lilitan kawat mendapatkan arus, inti besi lunak menghasilkan medan magnet dan menarik *switch* kontak. *Switch* kontak mengalami gaya tarik magnet sehingga berpindah posisi ke kutub lain atau terlepas dari kutub asalnya. Keadaan ini akan bertahan selama arus mengalir pada kumparan relai. Dan relai akan kembali ke posisi semula yaitu *normally-off*, bila tidak ada lagi arus yang mengalir padanya. Posisi normal relai tergantung pada

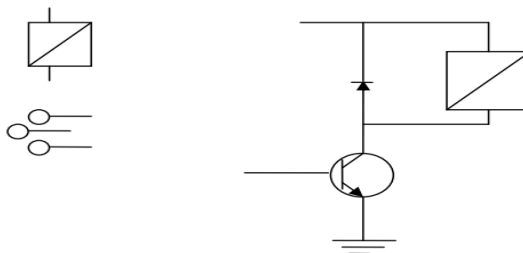
<sup>17</sup> <http://teknikelektronika.com/pengertian-relay-fungsi-relay/>, diakses 10 September 2015 jam 14.20

jenis relai yang digunakan. Dan pemakaian jenis relai tergantung pada keadaan yang diinginkan dalam suatu rangkaian/sistem.

Menurut kerjanya, relai dapat dibedakan menjadi:

1. *Normaly Open* (NO) : Saklar akan tertutup bila dialiri arus.
2. *Normaly Close* (NC) : Saklar akan terbuka bila dialiri arus.
3. *Change Over* (CO) : Relai ini mempunyai saklar tunggal yang normalnya tertutup.

Yang mana bila kumparan 1 dialiri arus maka saklar akan terhubung ke terminal A, sebaliknya bila kumparan 2 dialiri arus maka saklar akan terhubung ke terminal B. Analogi rangkaian relai yang digunakan adalah saat basis transistor ini dialiri arus maka transistor dalam keadaan tertutup yang dapat menghubungkan arus dari kolektor ke *emitter* yang mengakibatkan relai terhubung. Sedangkan fungsi dioda disini adalah untuk melindungi transistor dari tegangan induksi yang bisa mencapai 100 sampai 150 volt dimana tegangan ini dapat merusak transistor. Gambar 2.39 pada halaman berikutnya merupakan simbol relai dan relai dalam skema rangkaian listrik.



**Gambar 2.29 Simbol Relai dan Relai dalam Rangkaian**

Sumber : [www.elangsakti.com](http://www.elangsakti.com)



### 2.1.10. Saklar *Push Button*

Saklar tekan/tombol (*push button*) seperti yang terlihat pada gambar 2.30 pada halaman selanjutnya, ada dua jenis yaitu tombol tekan *Normally Open* (NO) dan tombol tekan *Normally Close* (NC). Konstruksinya tombol tekan ada beberapa jenis, yaitu jenis tunggal *on* dan *off* dibuat secara terpisah dan ada juga yang dibuat satu tempat. Jenis ini untuk satu tombol dapat untuk *on* dan *off* tergantung keinginan penggunaannya.

Saklar *push button* berfungsi sebagai pemberi sinyal masukan pada rangkaian listrik, ketika/selama bagian knopnya ditekan maka alat ini akan bekerja sehingga kontak-kontaknya akan terhubung untuk jenis *normally open* dan akan terlepas untuk jenis *normally close*. Sebaliknya, ketika knopnya dilepas kembali maka akan bekerja kebalikan dari sebelumnya. Tombol tekan *push button* tunggal terdiri dari dua terminal, sedangkan tombol tekan *push button* ganda terdiri dari empat terminal.<sup>18</sup> Untuk membuktikan pada terminal *push button* bisa digunakan alat ukur multimeter/ohm meter, pada umumnya pemakaian terminal jenis NO digunakan untuk menghidupkan rangkaian, dan jenis NC digunakan untuk mematikan rangkaian, namun semuanya tergantung pada kebutuhan mesin.

---

<sup>18</sup>Gestiyawaati, "Pushbutton, Limit Switch, Relay," Sugestiku, diakses dari, <http://sugestiku.blogspot.com/2013/01/push-button-limit-switch-relay.html>, diakses pada tanggal: 4 November 2015 jam 21.12



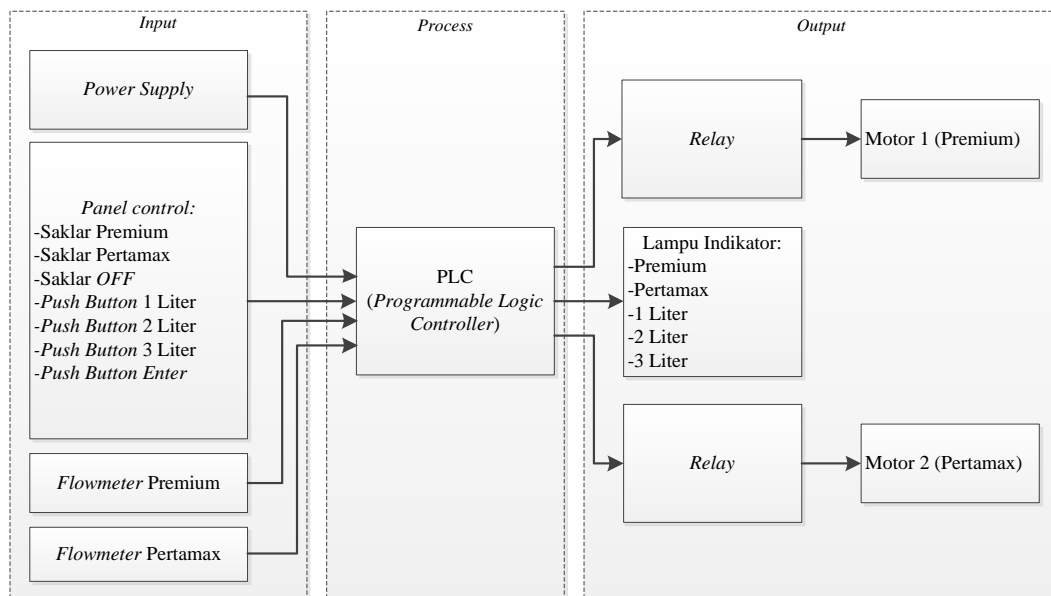
**Gambar 2.30 Saklar *Push Button***

Sumber: <http://sugestiku.blogspot.com/2013/01/push-button-limit-switch-relay.html>

## **2.2 Kerangka Berpikir**

Penelitian dalam pembuatan prototipe pengisian bahan bakar minyak dengan menggunakan motor listrik sebagai pemompa yang di dasari pada penggunaan PLC sebagai alat pengendali otomatis yang lebih cepat, praktis dan efisien dalam penjualan bahan bakar eceran. PLC berfungsi sebagai alat pengontrol pada motor listrik.

Prototipe pengisian bahan bakar minyak premium dan pertamax pada pertamini dengan menggunakan motor listrik sebagai pemompa otomatis berbasis PLC dengan sistem ini alat bekerja otomatis memompa bahan bakar dari tong penyimpanan sementara ke tabung penampung sebelum di pasarkan ke masyarakat. Langkah awal untuk melakukan pembuatan alat diawali dengan memahami skema diagram blok dari alat yang akan dibuat pada gambar 2.31 di halaman berikutnya :



**Gambar 2.31 Blok Diagram Sistem Kerja Alat**

Gambar 2.31 menerangkan alur kerja antar *hardware* dan juga antar blok. Alat pertamini ini dilengkapi dengan sistem kontrol berbasis PLC. Pada *prototype* pengisian bahan bakar minyak menggunakan motor listrik sebagai pemompa otomatis yang berbasis PLC ini berfungsi untuk memompa bahan bakar minyak ke tabung ukur satuan liter, dimana blok *input* terdapat *panel control*, yang terdiri dari saklar mode *off*, mode premium dan mode pertamax, lalu *push button* 1 liter, *push button* 2 liter, *push button* 3 liter, dan *push button enter*, yang berfungsi sebagai kendali untuk mengatur *system* pemompaan pada *prototype* yang akan dipompa dari penampungan utama ke tabung takaran dalam satuan liter. Blok *input* berfungsi untuk memberikan masukan berupa sinyal dan data yang akan diolah oleh PLC.

Blok proses merupakan blok yang berfungsi untuk mengolah masukan dari sinyal *input*. PLC merupakan inti dari blok diagram. PLC bertugas merespon dan

mengolah masukan yang diterima dari *selector switch*. Sinyal yang diterima dari *selector switch* akan menjadi tanda bagi PLC untuk memulai bekerja.

Blok *output* merupakan keluaran dari PLC. Respon yang dilakukan PLC akan memulai kerja *relay* dan *system* pemompa bahan bakar yang digerakkan oleh motor pompa.