

Mekatronika

Modul – 14

PLC dan Praktikum PLC

Hasil Pembelajaran :

Mahasiswa dapat memahami karakteristik PLC dan melaksanakan praktikum PLC

Tujuan

Bagian ini memberikan informasi mengenai karakteristik PLC dan penerapannya dalam praktikum

14.1 Pendahuluan

Alat kontrol logika yang dapat diprogram (PLC) memproses sinyal masukan binari untuk menghasilkan sinyal keluaran atau output, yang secara langsung mempengaruhi proses, pengoperasian dan urutan teknis.

Programmable Logic Controller (PLC) menjamin, bahwa semua langkah dalam rangkaian industrial tersebut diset dalam gerakan sesuai urutan yang benar dan dengan ketentuan waktu yang terkoordinasi.

PLC selalu memonitor status-status tertentu dalam instalasi tersebut (misalnya : suhu, tekanan, level dsb). Jika nilai yang telah ditetapkan terlewati atau tidak tercapai, maka PLC tersebut, akan memicu reaksi untuk mengatasi gejala tersebut atau memberi sinyal yang sesuai kepada staf operasional.

Konsep dari PLC adalah sesuai dengan namanya adalah sebagai berikut:

- *Programmable*
Menunjukkan kemampuannya yang dapat dengan mudah diubah-ubah sesuai program yang dibuat dan kemampuannya dalam hal memory program yang telah dibuat.
- *Logic*
Menunjukkan kemampuannya dalam memproses input secara aritmetik (ALU), yakni melakukan operasi membandingkan, menjumlahkan, mengalikan, membagi, mengurangi, dan negasi.
- *Controller*
Menunjukkan kemampuan dalam mengontrol dan mengatur proses sehingga menghasilkan output yang diinginkan.

14.2 PLC

Fungsi Programmable Logic Controller (PLC)

Fungsi dan kegunaan dari PLC dapat dikatakan hampir tidak terbatas. Tapi dalam prakteknya dapat dibagi secara umum dan secara khusus.

Secara umum fungsi dari PLC adalah sebagai berikut:

1. Kontrol Sekuensial

PLC memproses input sinyal biner menjadi output yang digunakan untuk keperluan pemrosesan teknik secara berurutan (sekuensial), di sini PLC menjaga agar semua step/langkah dalam proses sekuensial berlangsung dalam urutan yang tepat.

2. 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.

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 lain-lain.

Keuntungan PLC atas Kontrol Konvensional:

- Aplikasi Universal → Pemrograman yang ampuh
- Produksi yang besar → Mudah diubah
- Harga semakin murah → Commissioning mudah
- Bidang aplikasi baru → Text dan grafik

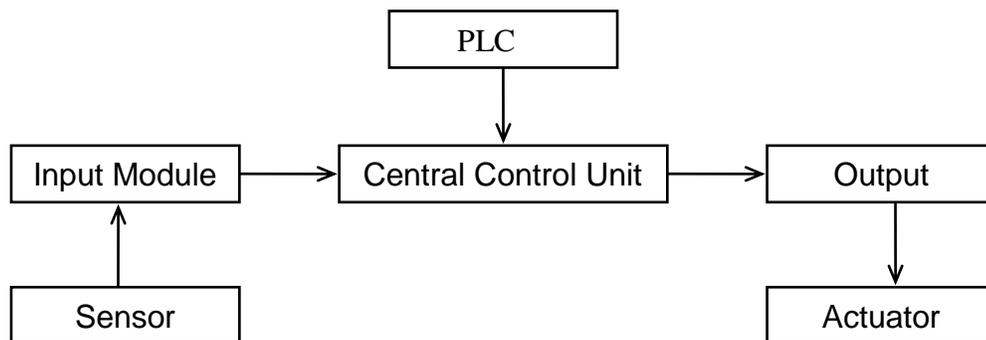
Sistem Komponen dari PLC

PLC atau biasa disebut PC (Programmable Controller) adalah suatu perangkat yang dapat dengan mudah diprogram dengan menggunakan pengontrol peralatan. PLC sederhana mempunyai komponen utama berupa CCU (Central Control Unit), Unit I/O, Programming Console, Rack atau Mounting Assembly dan catu daya.

Central Control Unit (CCU)

Central Control Unit merupakan unit pusat pengolah data yang digunakan untuk melakukan proses pengolahan data dalam PLC. Central Control Unit ini merupakan sebuah mikroprosesor.

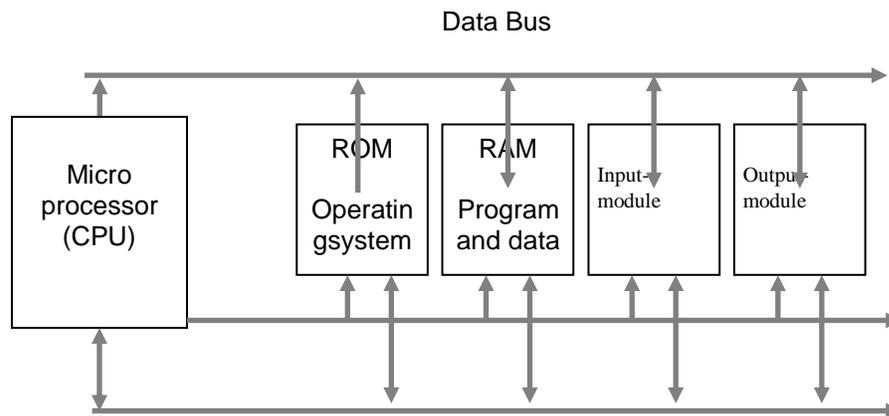
Sistem komponen dari sebuah PLC adalah seperti gambar di bawah ini:



Gambar 14-1. Sistem Komponen dari Sebuah PLC

Desain dari Central Control Unit

Gambar di bawah ini menunjukkan bagian dari mikroprosesor. Mikroprosesor terdiri dari Arithmetic Logic Unit (ALU), control unit dan sejumlah kecil memori unit yang disebut dengan register. Tugas dari ALU adalah untuk melakukan operasi aritmetika (penjumlahan, perkalian, pembagian, dsb) dan logika (operasi OR, AND, NOT, dsb).



Gambar 14-2. Desain dari Sebuah Mikroprosesor

Accumulator

Atau AC adalah register khusus yang dapat diakses secara langsung oleh ALU. Accumulator ini berfungsi untuk menyimpan data yang akan diproses dan data hasil pemrosesan.

Instruction Register

Berfungsi untuk menyimpan perintah yang telah dipanggil oleh program memori sampai perintah tersebut diterjemahkan dan dieksekusi.

Sebuah perintah (**Command**), terdiri dari dua bagian yaitu bagian operasi dan bagian alamat. Bagian operasi menunjukkan operasi logika mana yang akan dikerjakan sedangkan bagian alamat menunjukkan operand (sinyal input, sinyal output, flag, timer, dsb) dari operasi logika yang akan dikerjakan.

Program Counter

Program counter merupakan register yang berisi alamat dari perintah selanjutnya yang akan dikerjakan.

Control Unit

Control unit bertugas mengatur dan mengontrol seluruh urutan logika dari operasi yang dikehendaki selama proses eksekusi sebuah perintah.

Central Control Unit dari sebuah PLC

Central Control Unit digunakan untuk memproses sinyal input untuk mendapatkan sinyal output seperti dengan yang diharapkan. CCU memproses input berdasarkan dengan program yang telah diberikan. Sebuah program terdiri dari urutan-urutan yang harus dikerjakan.

Pemrosesan sebuah program oleh CCU sederhana dapat dijelaskan sebagai berikut:

- program memori berisi perintah-perintah yang harus dikerjakan, CCU mempunyai akses ke program memori.
- control unit menghendaki data input melalui modal input dan memprosesnya di ALU yang juga terhubung dengan perintah yang tersimpan dalam program memori.
- output data dikeluarkan oleh control unit melalui modul output ke sistem yang dikontrol.

Memori

Program yang dijalankan mendapat perhatian khusus selama proses operasi dan karenanya perlu suatu memori yang disebut memori program yang dapat dibaca oleh prosesor. Pemilihan memori program harus didasarkan atas pertimbangan-pertimbangan berikut ini:

- harus cukup sederhana dan mudah untuk memodifikasi atau membuat program baru.
- keamanannya terjamin, dalam hal program tidak akan berubah terhadap interferensi listrik atau bila listrik mati.
- harus cukup cepat atau tidak ada delay untuk operasi dengan prosesor.

Terdapat 3 jenis memori yang sering digunakan yaitu RAM, EPROM, dan EEPROM.

- **RAM**

RAM (Random Acces Memory) merupakan memori yang cepat dan bersifat volatile (data akan hilang bila arus listrik mati). RAM digunakan sebagai memori utama dalam PLC, dapat dibaca dan ditulisi. Untuk menjaga terhadap tegangan listrik yang mati, biasanya RAM dilengkapi dengan baterai yang tahan selama bertahun-tahun.

- **EPROM**

EPROM (Erasable Programmable Read Only Memory) adalah jenis memori yang cepat dan juga murah harganya, sama dengan memori RAM hanya saja EPROM bersifat non-volatile, artinya isi memori ini tetap ada walaupun supply tegangan hilang. Untuk keperluan modifikasi program maka memori ini harus dikosongkan isinya melalui penyinaran dengan sinar ultraviolet. Karena begitu kompleksnya proses penghapusan untuk memprogram ulang bahkan meskipun harganya murah, orang cenderung memilih RAM. Baru bila programnya sudah benar dan lengkap langsung bisa ditransfer ke EPROM secara permanen.

- **EEPROM**

EEPROM adalah memori yang mirip dengan memori EPROM, hanya saja untuk proses penghapusannya menggunakan arus listrik.

Modul Input dan Modul Output

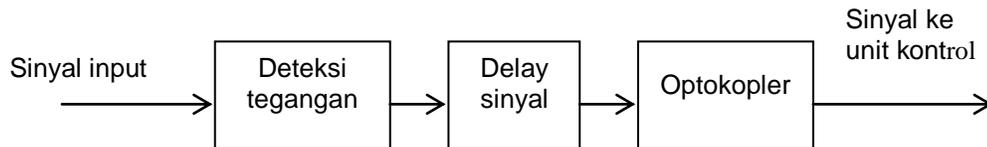
Fungsi dari sebuah modul input adalah untuk mengubah sinyal masukan dari sensor ke PLC untuk diproses di bagian CCU. Sedangkan modul output adalah kebalikannya, mengubah sinyal PLC kedalam sinyal yang sesuai untuk menggerakkan aktuator. Dari modul input dan output kita dapat menentukan jenis suatu PLC dari hubungan antara central control unit dengan modul input dan output yaitu compact PLC dan modular PLC. Compact PLC adalah bila input modul CCU dan output modul dikemas dalam suatu wadah. Modular PLC bila modul input, modul output dan CCU dikemas dalam satu wadah. Modular PLC bila modul input, modul output dan CCU dikemas secara tersendiri.

- **Modul input**

Modul input adalah modul tempat menghubungkan sensor-sensor dengan modul itu sendiri. Sinyal sensor tersebut selanjutnya akan diteruskan ke CCU. Fungsi terpenting dari sebuah modul input adalah sebagai berikut:

- mendeteksi sinyal masukan
- mengatur tegangan kontrol untuk batas tegangan logika masukan yang diijinkan

- melindungi peralatan elektronik yang sensitif terhadap tegangan luar
- Menampilkan sinyal masukan tersebut.
-



Gambar 14-3. Blok Diagram Modul Input

Deteksi tegangan error meyakinkan bahwa tegangan masuk masih dalam batas yang diijinkan atau tidak. Bila tegangannya terlalu tinggi akan diturunkan melalui dioda breakdown.

Delay sinyal meyakinkan apabila tegangan yang diterima sudah merupakan input yang sebenarnya atau bukan. Rangkaian ini mempertahankan tegangan input sesaat (1-20 ms) untuk membedakannya dengan sinyal-sinyal lain seperti tegangan interferensi.

Optokopler mengirimkan informasi sensor berupa cahaya dan menciptakan isolasi elektronik antara kontrol dan rangkaian logika. Selanjutnya melindungi komponen elektronik yang sensitif dan naiknya tegangan luar secara tiba-tiba. Terdapat optokopler yang mampu memberikan perlindungan terhadap tegangan sampai dengan 5 kV (5000 volt) yang sesuai dengan aplikasi untuk industri.

- **Modul output**

Modul output mengeluarkan sinyal dari CCU ke kontrol elemen yang diperlukan untuk menggerakkan aktuator sesuai dengan tugas yang telah diberikan. Fungsi terpenting dari modul output adalah sebagai berikut:

- mengatur tegangan kontrol untuk batas tegangan logika yang diijinkan.
- melindungi peralatan elektronik yang sensitif terhadap tegangan luar.
- memberikan penguatan sinyal output sebelum dikeluarkan sehingga cukup kuat menggerakkan aktuator.
- memberikan perlindungan terhadap arus hubung singkat (short-circuit) dan pembebanan relay atau elektronik.



Gambar 14-4. Blok Diagram Modul Output

Optokopler adalah bentuk dasar dari power secara elektronik yang memberikan perlindungan terhadap komponen elektronik dan juga berfungsi untuk pengatur tegangan output. Saat ini perlindungan terhadap short-circuit dan overload serta power amplification telah dikemas dalam satu rangkaian terpadu berupa modul-modul melalui hubungan transistor secara Darlington atau lainnya.

Amplifier berguna untuk menguatkan arus listrik output sehingga nantinya cukup kuat untuk menggerakkan aktuator.

Short circuit monitoring memonitor jika terjadi arus hubung singkat pada rangkaian luar dan memutuskan hubungan antara modul output dengan rangkaian luar.

Catu Daya

Sistem PLC memerlukan dua buah catu daya. Satu untuk keperluan peralatan output, sedangkan satunya untuk catu daya modul-modul PLC itu sendiri yang menggunakan arus DC. Arus DC ini dapat diperoleh dari rangkaian terintegrasi atau transistor. Jika sistem catu daya menggunakan IC TTL dapat dihasilkan tegangan 5V, tetapi jika menggunakan IC CMOS tegangan yang didapat akan dapat bervariasi dalam 3 sampai 18 Volt.

Komponen dari Sistem Kontrol yang Menggunakan PLC

PLC adalah Programmable Logic Controller yang tak lebih berupa sebuah komputer yang dirancang secara khusus untuk mengerjakan tugas-tugas tertentu. Komponen dari sistem kontrol yang menggunakan PLC adalah seperti berikut:

- Hardware

Hardware adalah semua komponen elektronik dan mekanik dari suatu CCU dan memori. Sedangkan modul input dan output adalah untuk menghubungkan antara CCU dan aktuator/sensor.

- **Software**
Dengan software kita bisa menentukan program untuk kondisi tertentu pada peralatan yang telah kita pasang. Dalam bahasa pemrograman hal tersebut adalah berarti menggambarkan kumpulan dari hubungan-hubungan rangkaian logika. Masing-masing program menjelaskan secara terpisah proses kontrol untuk sistem dan sebelum dieksekusi ditransfer dulu kedalam memori khusus dalam CCU (program memori). Jika diperlukan kita dapat memodifikasi program tersebut.
- **Sensor**
Sensor dapat berupa limit switch, level sensor, atau sensor temperatur. Sensor ini memberitahu PLC, status dari sistem yang dikontrol. Bila diperlukan sinyal non listrik dapat diubah menjadi sinyal listrik untuk keperluan input.
- **Aktuator**
Aktuator dapat berupa silinder pneumatik, motor listrik, dan sebagainya. Aktuator merupakan output dari PLC ke device dari sistem yang dikontrol.
- **Programmer**
Program yang telah ditulis programmer diterjemahkan kedalam bahasa mesin CCU, kemudian bahasa mesin tersebut ditransfer kedalam program memori untuk selanjutnya dikerjakan. Dewasa ini programmer adalah berupa komputer PC yang berisi sistem pemrograman atau peralatan khusus semacam itu yang lebih kecil ukurannya (hand-hold programming).
- **Diagram Rangkaian**
Cara menghubungkan input/output dengan PLC, terlebih dahulu harus dikenali komponen input seperti switch, maupun sensor, yang harus dihubungkan dengan alamat input PLC. Sedangkan komponen output seperti solenoid untuk menggerakkan valve dan silinder, lampu, coil magnetic contactor untuk menghubungkan sumber motor harus dihubungkan dengan alamat output PLC.

Dasar Pemograman PLC

Kontrol program adalah komponen utama dalam sistem yang bekerja secara otomatis. Kontrol program harus didesain secara sistematis, terstruktur dengan baik dan harus terdokumentasi agar bebas dari kesalahan, pemeliharaan mudah dan efektif dalam biaya. Untuk memrogram PLC dapat digunakan prosedur berikut untuk menyelesaikan permasalahan mengenai kontrol.

- Langkah 1 : Identifikasi masalah.
Definisi permasalahan harus menjabarkan problema kontrol secara tepat dalam bentuk yang detail. Informasi yang diperlukan yaitu skema posisi, skema sekuensial dan tabel kebenaran yang menerangkan hubungan antara masukan dan keluaran dan juga berguna untuk tes terhadap resiko pada saat instalasi.
- Langkah 2 : Allocation List
Allocation list berisi kondisi-kondisi program termasuk identifier atau alamat yang dipakai oleh keluaran atau masukan.
- Langkah 3 : Pembuatan program
Terdapat dua cara untuk membuat program, yaitu dengan menggunakan ladder diagram atau statement list. Pemrogram dapat menggunakan salah satu cara yang dikuasai.

Ladder diagram menggambarkan program dalam bentuk grafik. Diagram ini dikembangkan dari kontak-kontak relay yang terstruktur yang menggambarkan aliran arus listrik. Dalam ladder diagram ini terdapat dua buah garis vertikal dimana garis vertikal sebelah kiri dihubungkan dengan sumber tegangan positif catu daya, sedangkan garis sebelah kanan dihubungkan dengan sumber tegangan negatif catu daya. Diantara dua garis ini dipasang kontak-kontak yang menggambarkan kontrol dari saklar, sensor dan output. Satu baris dari diagram disebut dengan satu **rung**.

Statement List adalah bahasa pemrograman tingkat tinggi. Semua hubungan logika dan kontrol sekuens dapat diprogram dengan menggunakan perintah dalam bahasa ini. Perintah-perintah yang digunakan adalah mirip dengan bahasa tingkat tinggi Basic atau Pascal.

```
Misal:          IF    S1
                THEN SET  LAMPU
```

Pembuatan program PLC secara umum merupakan gabungan dari logika-logika sederhana diantaranya adalah logika AND, logika OR maupun Inversenya serta

pengunci. Logika AND merupakan instruksi yang harus selalu didahului sekurang-kurangnya satu kontak yang lain. Pada ladder diagram logika AND dapat dijelaskan sebagai kontak NO (Normally Open) dalam rangkaian seri dengan kontak-kontak sebelumnya. Instruksi AND memungkinkan banyaknya masukan dari sinyal-sinyal kondisi. Bila semua sinyal kondisi bernilai benar ("1" / "true") maka baris program tersebut akan dijalankan dan selain itu tidak dijalankan.

Logika OR dapat dijelaskan sebagai kontak NO (Normally Open) tunggal yang dihubungkan secara paralel dengan kontak pertama dari ladder diagram. Instruksi OR memungkinkan banyaknya masukan dari sinyal-sinyal kondisi. Bila salah satu atau semua sinyal kondisi bernilai benar ("1" / "true") maka baris program tersebut akan dijalankan dan selain itu tidak dijalankan.

Lembar Kerja

Alat dan Bahan

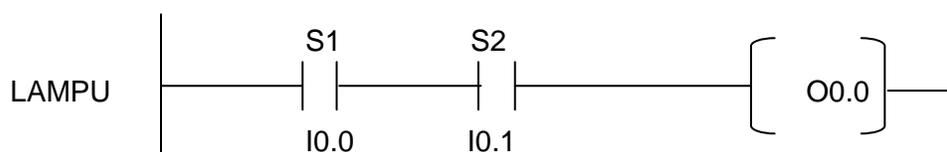
1. Unit PLC 1 buah
2. Unit input 1 buah
3. Unit output 1 buah
4. Komputer 1 set
5. Kabel Penghubung secukupnya

Percobaan Pertama : Logika AND

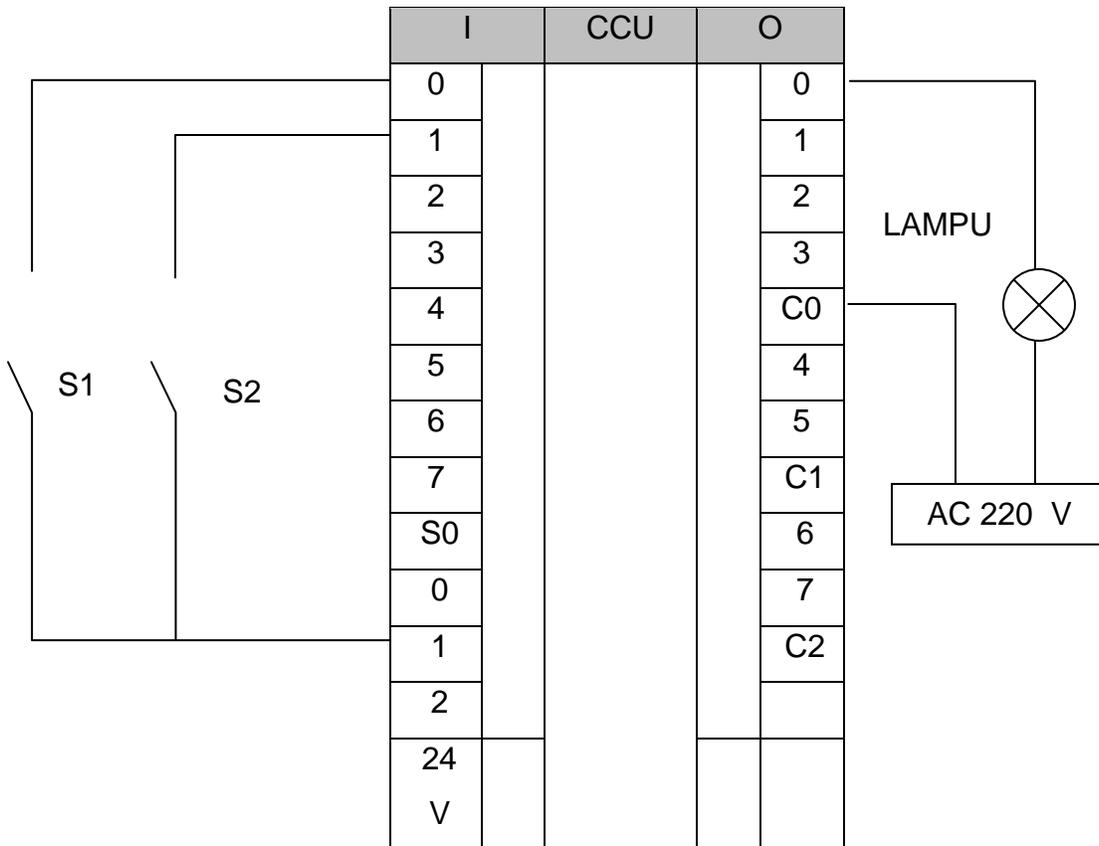
1. Buatlah project baru dengan nama LATIH-1 dan diskripsi LOGIKA AND !
2. Buat alocation list sebagai berikut !

Absolut Op.	Symbolic Op.	Deskripsi
I0.0	S1	Saklar "ON" / "1" saat ditekan
I0.1	S2	
O0.0	LAMPU	"ON" saat bernilai "1"

3. Buatlah ladder diagram berikut dengan komputer !



4. Buatlah rangkaian berikut !



5. Download lah ladder diagram ke PLC !
6. Jalankan PLC dengan menekan tombol RUN/STOP hingga pada posisi RUN !
7. Tekanlah S1, apakah lampu menyala ? ya / tidak *) !
8. Tekanlah S2, apakah lampu menyala ? ya / tidak *) !
9. Tekanlah S1 dan S2 secara bersamaan, apakah lampu menyala ? ya / tidak *) !
10. Buatlah kesimpulan dari pengamatan tersebut !
11. Buatlah statement list berikut dengan komputer !

```

IF      S1
AND    S2
THEN   SET      LAMPU
OTHRW RESET    LAMPU
    
```

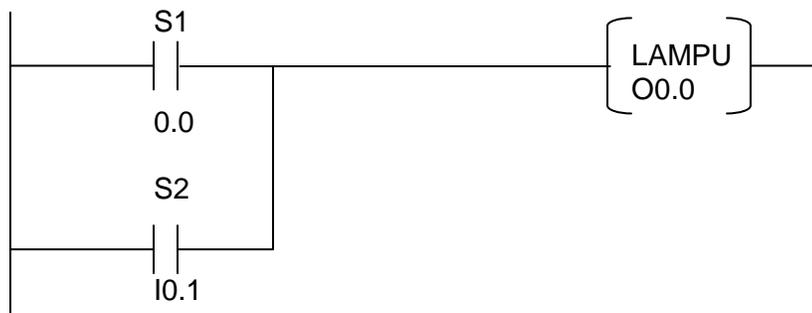
12. Ulangilah langkah 5 sampai 10 !

Percobaan Kedua : Logika OR

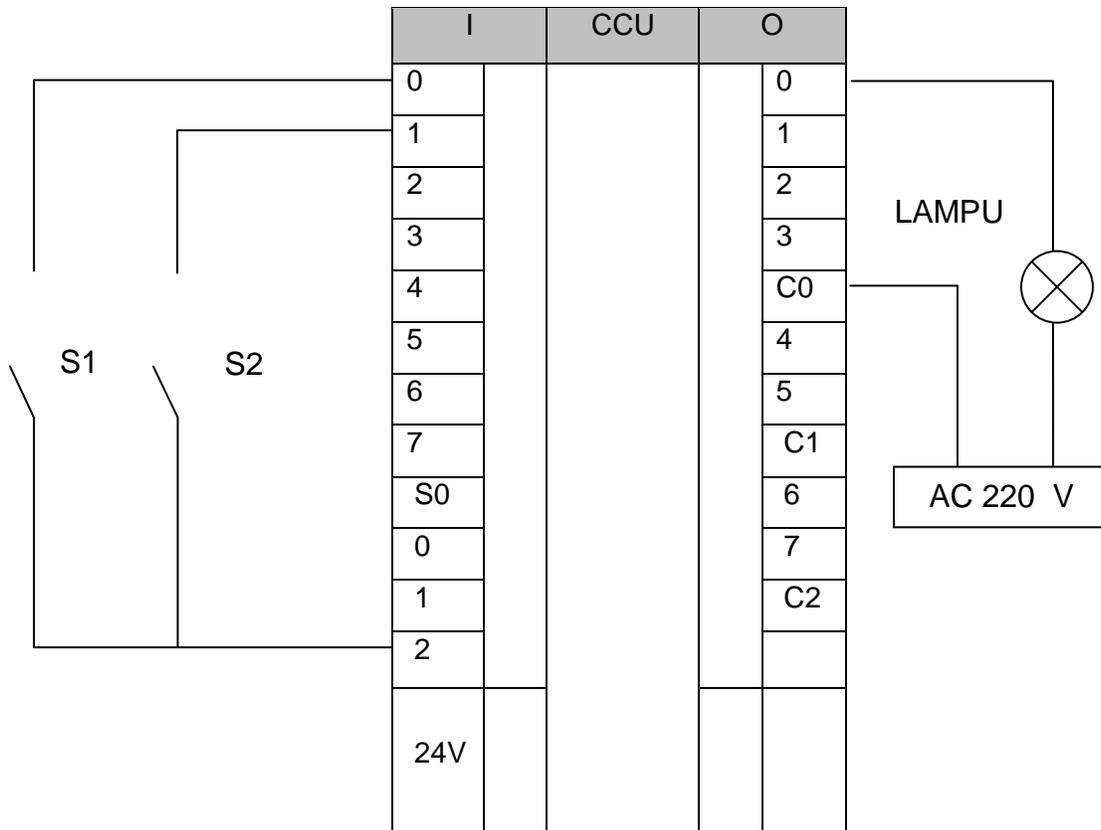
1. Buat project baru dengan nama LATIH-2 dan diskripsi LOGIKA OR!
2. Buat alocation list sebagai berikut !

Absolut Op.	Symbolic Op.	Deskripsi
I0.0	S1	Saklar "ON" / "1" saat ditekan
I0.1	S2	
O0.0	LAMPU	"ON" saat bernilai "1"

3. Buatlah ladder diagram berikut dengan komputer !



4. Buatlah rangkaian berikut !



5. Download lah ladder diagram ke PLC !
6. Jalankan PLC dengan menekan tombol RUN/STOP hingga pada posisi RUN !
7. Tekanlah S1, apakah lampu menyala ? ya / tidak *) !
8. Tekanlah S2, apakah lampu menyala ? ya / tidak *) !
9. Tekanlah S1 dan S2 secara bersamaan, apakah lampu menyala ? ya / tidak *) !
10. Buatlah kesimpulan dari pengamatan tersebut !
11. Buatlah statement list berikut dengan komputer !

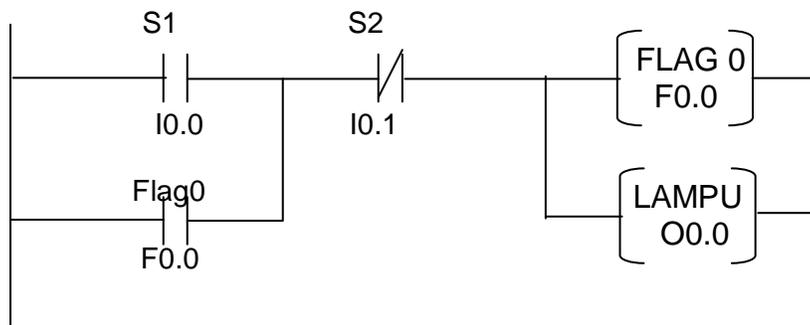
IF	S1		
OR	S2		
THEN	SET	LAMPU	
OTHRW	RESET	LAMPU	
12. Ulangilah langkah 5 sampai 10 di atas !

Percobaan Ketiga : FLAG

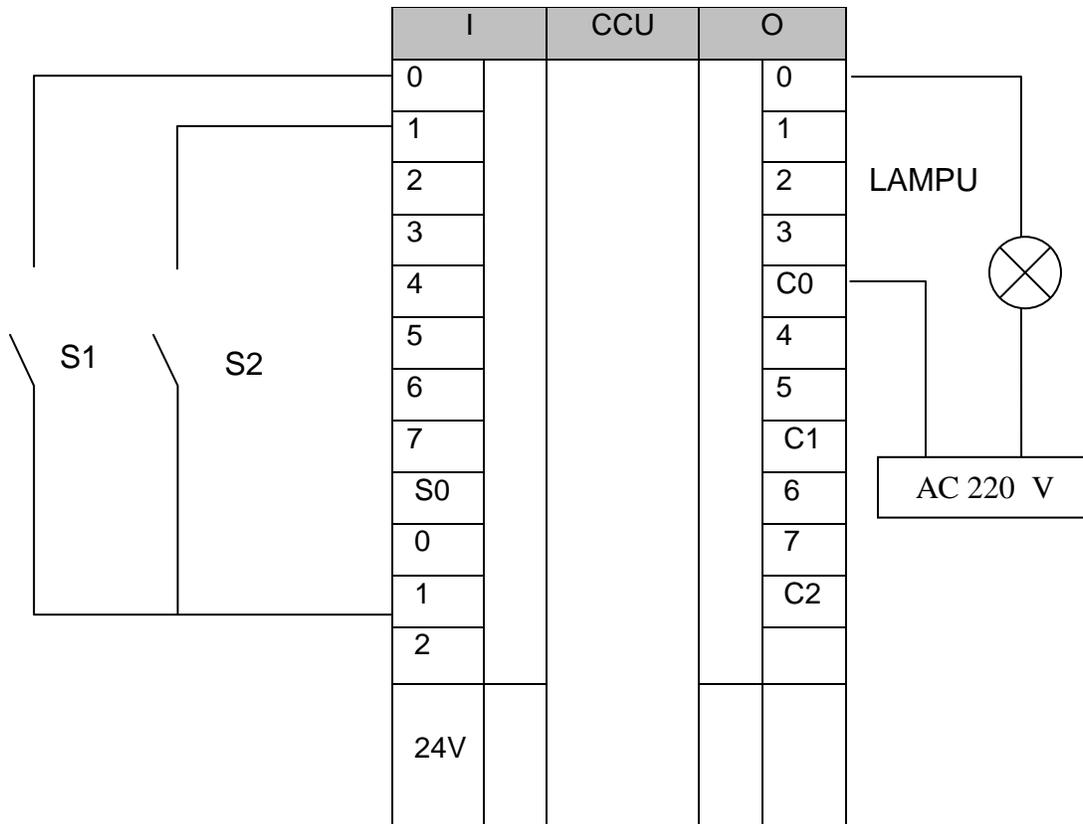
1. Buatlah project baru dengan nama LATIH-3 dan diskripsi FLAG !
2. Buatlah alocation list sebagai berikut !

Absolut Op.	Symbolic	Deskripsi
I0.0	S1	Saklar "ON" / "1" saat ditekan
I0.1	S2	
O0.0	LAMPU	"ON" saat bernilai "1"
F0.0	FLAG0	Penyimpan nilai "1" saat aktif

3. Buatlah ladder diagram berikut dengan komputer !



4. Buatlah rangkaian berikut !



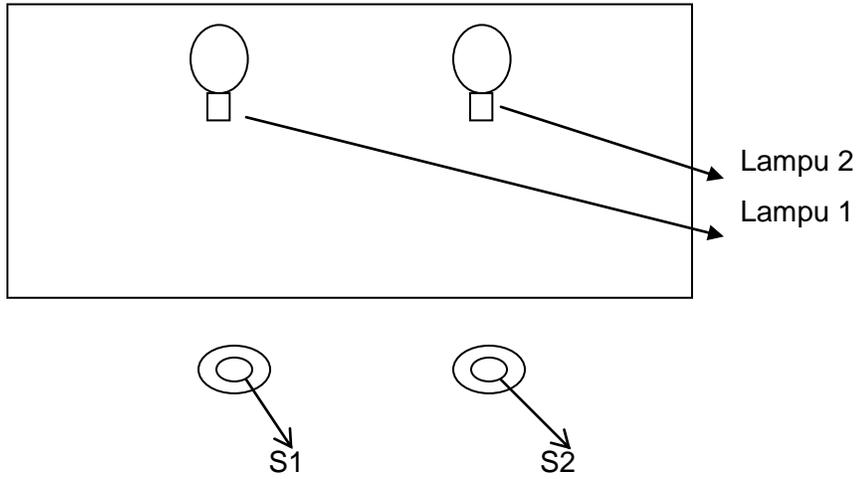
5. Download lah ladder diagram ke PLC !
6. Jalankan PLC dengan menekan tombol RUN/STOP hingga pada posisi RUN !
7. Tekanlah S1, apakah lampu menyala ? ya / tidak *) !
8. Tekanlah S2, apakah lampu menyala ? ya / tidak *) !
9. Tekanlah S1 dan S2 secara bersamaan, apakah lampu menyala ? ya / tidak *) !
10. Buatlah kesimpulan dari pengamatan tersebut !
11. Buatlah statement list berikut dengan komputer !


```

      IF          S1
      OR          FLAG0
      AND        N    S2
      THEN       SET      FLAG0
                SET      LAMPU
      OTHRW     RESET    LAMPU
      
```
12. Ulangilah langkah 5 sampai 10 di atas !

Lembar Latihan

Dua buah lampu dengan 2 saklar akan menyala secara bergantian. Jika saklar 1 ditekan Lampu 1 menyala dan Lampu 2 Mati dan sebaliknya jika saklar 2 ditekan maka Lampu 1 mati sedangkan Lampu 2 menyala.



Pemrograman PLC dengan Timer dan Counter

Dalam aplikasi sistem kontrol logika AND, logika OR dan kebalikannya saja tidak cukup untuk menyelesaikan permasalahan pengontrolan, namun memerlukan komponen penunda waktu atau lazim disebut dengan TIMER dan komponen penghitung yang disebut dengan COUNTER. Timer dapat di set untuk satuan detik, menit maupun jam. Timer akan mulai melakukan perhitungan waktu saat saklar reset ditekan. Pada saat perhitungan waktu timer sama dengan settingnya kontak NO pada timer akan berubah menjadi NC demikian pula kontak NC pada timer akan berubah menjadi NO. Untuk satu timer, kita dapat menggunakan beberapa kontak NO dan NC yang dimiliki timer dengan perubahan kondisi kontak yang bersamaan.

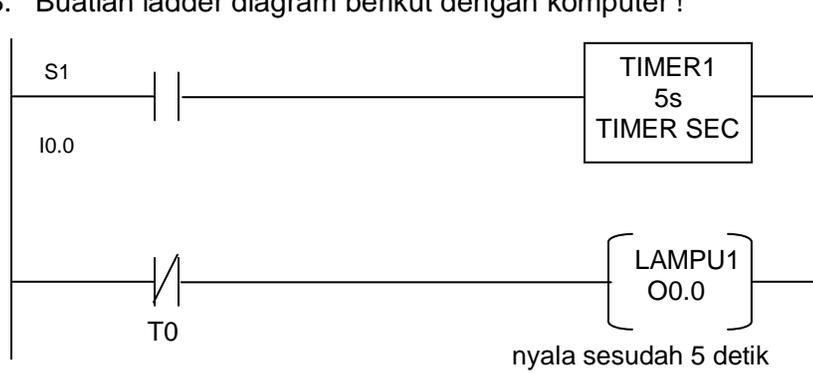
COUNTER pada pemrograman PLC digunakan untuk melakukan penghitungan sinyal masukan dengan setting nilai yang ada pada COUNTER yang dilakukan oleh pemrogram. Jika hitungan sinyal masukan sama dengan nilai setting maka Counter akan merubah posisi NO menjadi NC dan sebaliknya NC akan menjadi NO. Sinyal masukan akan dihitung satu selama posisi sinyal masukan OFF menjadi ON. Panjang dan pendeknya sinyal OFF atau ON tidak berpengaruh pada perhitungan counter. Untuk masing-masing counter, kita dapat menggunakan beberapa kontak NO dan NC yang dimiliki counter dengan perubahan kondisi kontak yang bersamaan.

Percobaan Pertama : Timer (On Delay)

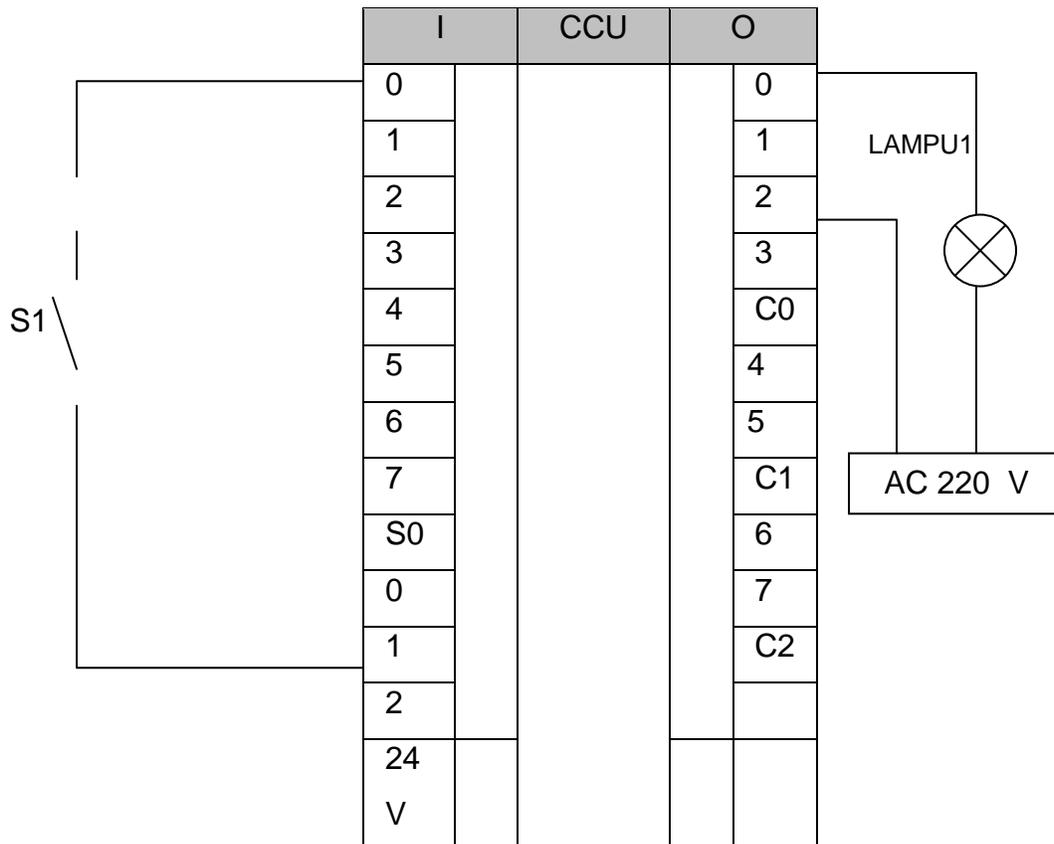
1. Buatlah project baru dengan nama LATIH-5 dan diskripsi TIMER !
2. Buat alocation list sebagai berikut !

Absolut Op.	Symbolic	Deskripsi
I0.0	S1	Saklar "ON" / "1" saat ditekan
T0	TIMER1	ON / "1" setelah n detik
TP0	TPRE0	Timer Preselect
O0.0	LAMPU1	"ON" saat bernilai "1"

3. Buatlah ladder diagram berikut dengan komputer !



4. Rangkailah seperti rangkaian berikut !



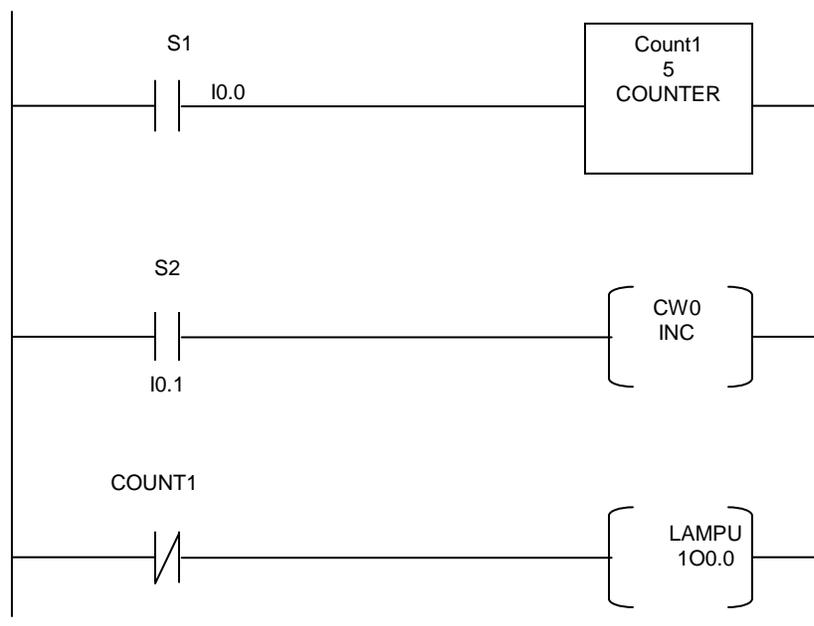
5. Download lah ladder diagram ke PLC !
6. Jalankan PLC dengan menekan tombol RUN/STOP hingga pada posisi RUN !
7. Tekanlah S1, apakah lampu1 menyala ? ya / tidak *) !
8. Apakah lampu1 menyala setelah 5 detik tombol ditekan ? ya / tidak *).

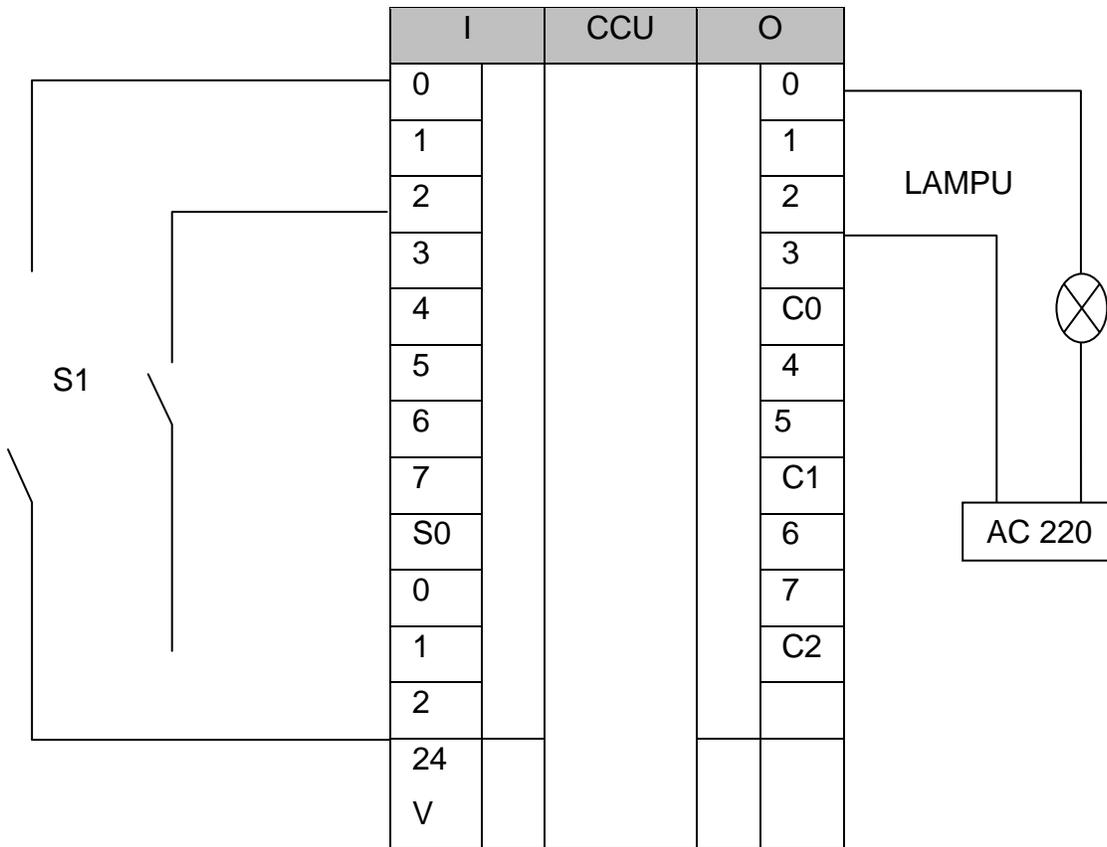
Percobaan Kedua : Counter

1. Buatlah project baru dengan nama LATIH-6 dan diskripsi COUNTER !
2. Buatlah allocation list sebagai berikut !

Absolut Op.	Symbolic Op.	Deskripsi
I0.0	S1	Saklar "ON" / "1" saat ditekan
I0.1	S2	
C0	COUNT1	ON / "1" setelah n kali
CW0	CWORD_0	Penghitung
CP0	CPRE0	Counter Preselect
O0.0	LAMPU	"ON" saat bernilai "1"

3. Buatlah ladder diagram berikut dengan komputer.
4. Rangkailah seperti gambar rangkaian berikut !





5. Download lah ladder diagram ke PLC !
6. Jalankan PLC dengan menekan tombol RUN/STOP hingga pada posisi RUN !
7. Tekanlah S1, apakah lampu menyala ? ya / tidak *) !
8. Tekanlah S2 sebanyak 5 kali, apakah lampu menyala ? ya / tidak *) !
9. Buatlah kesimpulan dari pengamatan tersebut !

Lembar Latihan

1. Sebuah unit produksi membutuhkan dua buah motor 3 phase yang berjalan berurutan dengan proses kerja sebagaimana berikut; motor satu berjalan ketika PLC (kontrol) mulai dijalankan dan 5 detik kemudian setelah tombol ditekan motor mati. Bersamaan dengan motor pertama mati motor kedua berjalan. Buatlah Allocation list dan Ladder diagram untuk kontrol motor tersebut !