

MODUL PELATIHAN PLC TINGKAT DASAR (MITSUBISHI Q-SERIES)



Josaphat Pramudijanto

LABORATORIUM PLC AA103
TEKNIK ELEKTRO OTOMASI
FAKULTAS VOKASI
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA-2017

Daftar Isi

1.	Pengantar	3
1.1.	Sejarah Perkembangan	3
1.2.	Bagian-bagian Sistem Kontrol Menggunakan PLC	3
1.3.	Pengertian Programmable Logic Controller (PLC)	4
1.4.	Tahapan Sistem yang Dikendalikan PLC	5
2.	Arsitektur PLC	6
2.1.	Sistem Input dan Output	6
2.2.	Modul Simulator PLC Q-Series	8
2.3.	Pemrograman PLC	9
3.	Pemrograman PLC dengan GX Developer	9
3.1.	Langkah-Langkah Pengoperasian GX-Developer	9
3.2.	Intruksi-Intruksi Dasar Pemrograman	11
3.3.	Latihan-latihan	14

1. Pengantar

PLC sangat vital keberadaannya untuk meningkatkan efisiensi dalam dunia perindustrian. Apalagi sekarang persaingan di dunia industri sudah sangat ketat misalkan persaingan perusahaan otomotif di dunia misal PT.Astra (Honda, Toyota, Daihatsu), PT. GM (General Motor), dan perusahaan Otomotif lainnya bersaing untuk memberikan performa yang baik dalam memberikan pelayanan kepada konsumennya seperti misalnya meliputi hal-hal berikut :

- a. Meningkatkan kualitas (Quality),
- b. Meminimkan Harga (Cost),
- c. Pengiriman produk (Delivery) yang sangat cepat sehingga tidak ada waktu tunggu,
- d. Keamanan dan keselamatan kerja (Safety) bagi pekerja sangat tinggi sehingga tidak rawan kecelakaan,
- e. Mengurangi operasi-operasi manual yang bisa membahayakan operator bahkan menimbulkan kematian dan beberapa aspek vital lainnya.

Untuk memberikan kriteria kinerja seperti yang dipaparkan diatas, PLC sangat bisa memberikan kendali yang besar untuk mewujudkannya. Meskipun dipasaran sudah banyak beredar berbagai macam merek PLC yang dipakai misalnya PLC merek Siemens, Allan Bradley, Omron, Mitsubishi dan merek lainnya. Pada kesempatan ini, kita akan membahas Cara mengoperasikan PLC Mitsubishi Q series mulai dari pengenalan hardware, dan bagaimana memrogramnya menggunakan modul simulator.

1.1. Sejarah Perkembangan

Programmable Logic Controller pertama kali dikembangkan oleh General Motor tahun 1968. Sistem kontrol ini digunakan sebagai alternatif untuk mengganti sistem relay control yang kompleks. Aplikasi PLC ini banyak digunakan dalam proses permesinan, pengepakan, material handling, assembly terotomasi dan sebagainya. Menurut IEC61131 bagian 1, pengertian PLC merupakan sistem elektronik yang beroperasi secara digital, menggunakan memori yang dapat diprogram untuk penyimpanan data sesuai keinginan pengguna, untuk melakukan fungsi khusus seperti logic, sequencing, timing, arithmetic; untuk melalui input baik analog maupun digital. Kontroler yang dapat diprogram dengan perangkat yang dirancang secara khusus mudah dapat diintegrasikan dengan sistem kontrol industri dan digunakan untuk menjalankan fungsi-fungsi yang diharapkan. Kontroler yang dapat diprogram sangat berguna bila dipergunakan pada operasi yang berulang-ulang dalam proses industri secara sekuensial. Pada pengontrolan secara sekuensial dilengkapi dengan relay, timer, dan counter. Sistem ini akan sangat sulit dilakukan untuk sistem produksi yang sering berubah dan dibongkar untuk kemudian dirancang kembali secara lengkap.

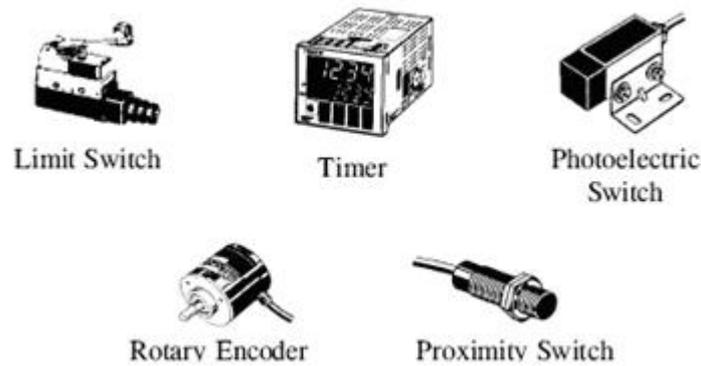
1.2. Bagian-bagian Sistem Kontrol Menggunakan PLC

Dalam sebuah sistem otomatis, PLC umumnya dianggap sebagai jantung dari sistem kontrol. Dengan program aplikasi kontrol (tersimpan dalam memori PLC) dalam pelaksanaan, PLC selalu memantau keadaan sistem melalui sinyal umpan balik perangkat input lapangan. Ini kemudian akan didasarkan pada logika program untuk menentukan tindakan yang akan dilakukan di lapangan perangkat output. PLC dapat digunakan untuk mengontrol sebuah tugas yang sederhana dan berulang-ulang, atau beberapa dari mereka mungkin saling berhubungan bersama-sama dengan pengontrol lain atau komputer pusat melalui semacam jaringan komunikasi, dalam rangka mengintegrasikan kontrol proses yang kompleks.

Input Device

Kepintaran sistem kontrol otomatis sangat tergantung pada kemampuan sebuah PLC

untuk membaca status sinyal input melalui perangkat lapangan.

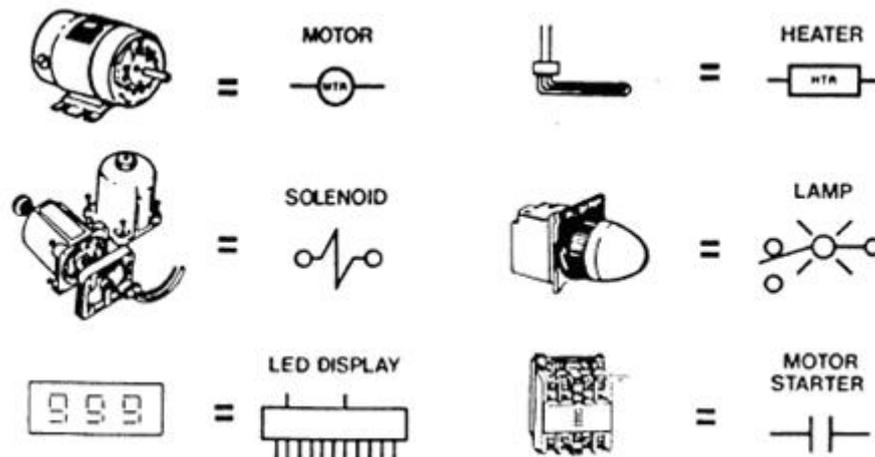


Gambar 1.1. Komponen Peralatan Kontrol yang Terhubung pada Input PLC

Push-buttons, keypad dan toggle switch, yang merupakan antarmuka manusia-mesin, adalah jenis alat manual input. Di sisi lain, untuk mendeteksi benda kerja, pemantauan bergerak mekanisme, memeriksa tekanan dan atau ketinggian cairan dan banyak lainnya, maka PLC akan menerima sinyal dari perangkat tertentu penginderaan otomatis seperti proximity switch, limit switch, photoelectric sensor, level sensor dan sebagainya. Jenis sinyal masukan ke PLC akan ON/OFF atau logika analog. Sinyal-sinyal input yang dihubungkan ke PLC melalui berbagai jenis modul masukan PLC.

Output Devices

Sebuah sistem otomatis atau sistem PLC hampir lumpuh tanpa sarana antarmuka ke peralatan lapangan. Beberapa perangkat yang paling sering dikendalikan motor, solenoida, indikator relay, buzzers dan beberapa peralatan lain. Aktivasi motor dan solenoida PLC dapat mengontrol sistem servo posisi lebih kompleks. Ini jenis perangkat output adalah mekanisme sistem otomatis dan begitu berpengaruh langsung terhadap kinerja sistem. Namun, perangkat output lain seperti lampu pilot, buzzers dan alarm yang hanya dimaksudkan untuk memberi tahu informasi telah tercapai.

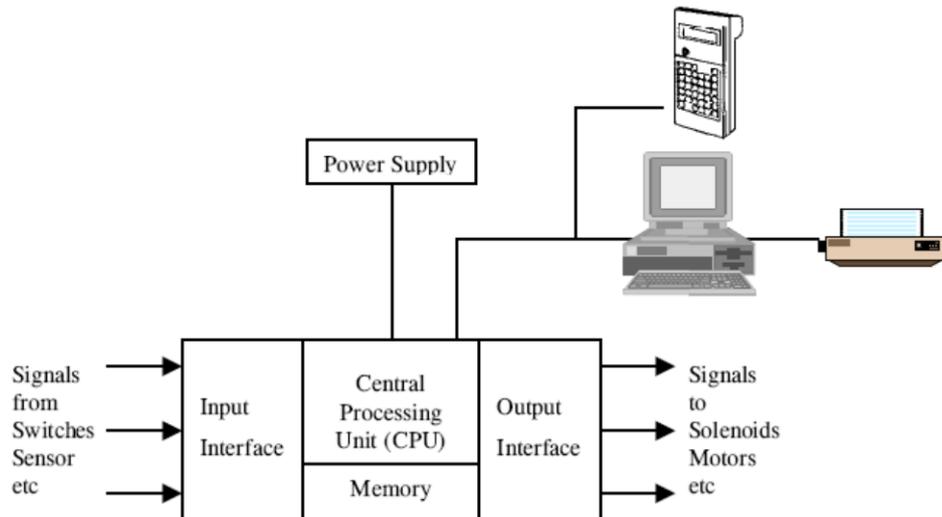


Gambar 1.2. Komponen Peralatan Kontrol yang Terhubung pada Output PLC

1.3. Pengertian Programmable Logic Controller (PLC)

Sebuah PLC terdiri dari Central Processing Unit (CPU) yang berisi program aplikasi dan Input dan Output Interface modul, yang terhubung langsung ke bagian I/O peralatan. Program ini mengontrol PLC sehingga ketika sinyal input dari perangkat input ternyata ON, respon yang tepat dibuat. Respons biasanya akan menyalakan sebuah sinyal keluaran ke beberapa perangkat

output.



Gambar 1.3. Diagram Fungsional PLC

Central Processing Unit

Central Processing Unit (CPU) adalah sebuah mikroprosesor yang mengkoordinasikan kegiatan PLC sistem. Hal ini mengeksekusi program, proses I/O & sinyal berkomunikasi dengan perangkat eksternal.

Memori

Ada berbagai jenis unit memori. Ini adalah wilayah yang memegang sistem operasi dan memori pengguna. Sistem operasi sebenarnya adalah sebuah perangkat lunak sistem yang mengkoordinasikan PLC. Program Ladder, Timer dan Counter Nilai disimpan dalam memori pengguna.

Perangkat Input dan Output

Bagian dari PLC yang berinteraksi dengan yang di kontrol.

1.4. Tahapan Sistem yang Dikendalikan PLC

Tujuh tahap untuk menyusun suatu sistem yang dikontrol dengan programmable controller:

1. Desain sistem

Peralatan atau sistem yang akan dikontrol harus didefinisikan terlebih dahulu. Sistem tersebut dapat berupa suatu mesin tunggal, sekelompok mesin-mesin atau suatu proses.

2. I/O assignment

Mengidentifikasi banyaknya input atau output yang akan dihubungkan dengan programmable controller.

3. Penulisan program

Memrogram berarti menuliskan Diagram Ladder dari flowchart atau algoritma kontrol yang berupa sequence chart atau time chart proses tersebut.

4. Programming

Menuliskan Diagram Ladder kedalam PLC yang akan digunakan untuk mengendalikan proses tersebut.

5. Editing

Melakukan perubahan-perubahan algoritma kontrol yang telah kita tuliskan, agar

sesuai dengan kemampuan PLC.

6. Monitoring

Melakukan proses uji coba program/simulasi dengan menghubungkan semua peralatan input dan output ke PLC. Pada tahapan ini dimungkinkan untuk proses editing lagi berdasarkan pada kondisi sebenarnya jika dihubungkan dengan proses.

7. Penyimpanan program

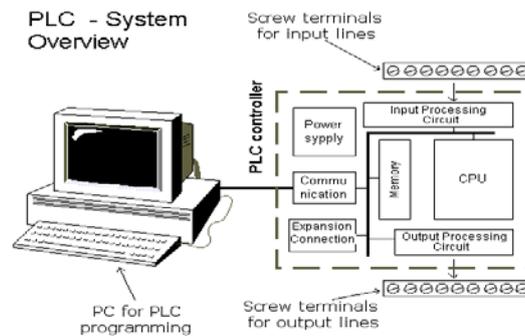
Melakukan proses pengimanan progam kedalam EPROM atau EEPROM. Dokumentasi secara cetak juga dilakukan.

2. Arsitektur PLC

Berikut ini mempelajari tentang komponen-komponen apa saja yang menyusun suatu konfigurasi yang terdapat pada programmable logic controller (PLC) .

Ada beberapa jenis hardware penyusun PLC, masing-masing adalah:

- CPU
- Perangkat pemrograman
- Power Supply
- Sistem I/O
- Komunikasi dan Ekspansi



Gambar 2.1. Hardware Penyusun PLC.

2.1. Sistem Input dan Output

Bagian dari PLC yang berinteraksi dengan lingkungan luar adalah bagian input dan output (I/O). Unit I/O merupakan penyanggah (interface) antara rangkaian elektronik PLC dengan lingkungan luar yang ingin berinteraksi (sensor, alarm, katup, limit switch, push button dll).

Untuk membuat pengkonversi sinyal dari PLC tersedia pilihan I/O untuk berbagai keperluan. Bagian ini secara garis besar terdiri dari:

- Modul Digital Input (DI)
- Modul Digital Output (DO)
- Modul Analog Input (AI)
- Modul Analog Output (AO)
- Modul untuk komunikasi (serial/paralel)

Kebanyakan PLC beroperasi internal antara 5-15 VDC (seperti tegangan TTL dan CMOS), sementara sinyal proses lebih besar, biasanya 24 VDC sampai 240 VAC dengan kemampuan arus beberapa ampere. Misal :

- Input switch 5V (level TTL), 24V, 110V, 240V
- Output switch 24V/1mA, 110V/1A, 240VAC/1A (triac), 240VAC/2A(relay)

2.1.1. Komponen Digital I/O

Perangkat hanya menghasilkan kondisi biner dengan pengolah sinyal berupa :

Contact Output

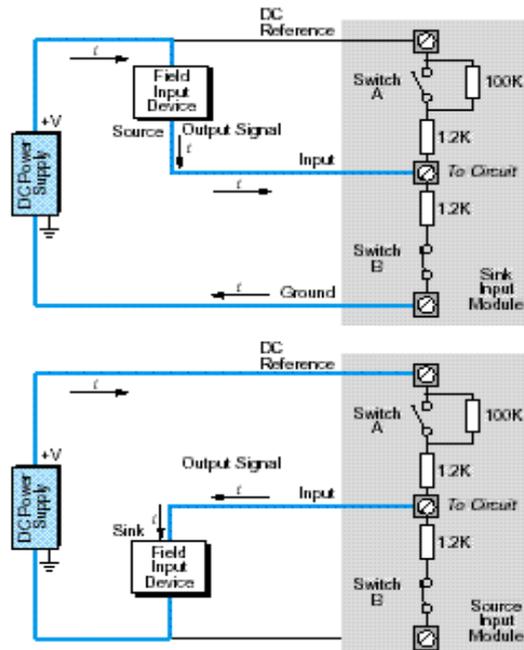
- Komponen untuk mengeluarkan sinyal adalah relay.
- Umumnya untuk beban arus yang relatif besar.
- Supply beban bisa AC/DC.
- Kecepatan respon rendah.
- Ada contact bounce, menyebabkan ada noise.

Solid State Output (SSR)

- Komponen yang digunakan solid-state
- **Beban AC:** outputnya menggunakan SCR. Respons lebih cepat dari contact output, dan lebih rendah dari transistor output.
- **Beban DC:** outputnya menggunakan Transistor. Respons sangat cepat untuk beban arus kecil. Frekwensi kerja cukup tinggi.

Sinyal output pada beban **DC** dapat berupa :

- Output **Tegangan:** memberikan sinyal tegangan sebagai output dan tahanan beban harus cukup tinggi dengan konsumsi arusnya kecil.
- Output **Arus:** mengolah sinyal arus (source/sink). Beban yang bekerja pada I/O-nya ada:
 - **current source** (PNP) : arus diberikan ke beban, beban dihubungkan keterminal output dan negatif catudaya.
 - **current sink** (NPN) : arus ditarik dari beban, beban dihubungkan keterminal positif catudaya dan terminal output.



Gambar 2.2 Modul Input Sink dan Modul Input Source

2.1.2. Modul Digital Input (DI)

Modul digunakan hanya untuk sinyal tegangan DC (searah), terutama model dengan komponen solid-state (SCR, transistor) jika dihubungkan dengan peralatan luar (sink/source). Jika modul ini memberikan arus ketika modul ini ON (aktif), maka modul ini dinamakan current source.

Jika modul ini menerima arus ketika modul ini ON (aktif), maka modul ini dinamakan current sink. Arus dari peralatan luar (mis. sensor) masuk ke modul DC input. Bagi peralatan luar berfungsi sebagai source, sedang modul DC input sebagai sink. Arus dari modul DC input ke peralatan luar (mis. katup). Bagi modul DC input berfungsi sebagai source, sedang peralatan luar sebagai sink.

Seringkali diperlukan memodifikasi pada sensor tipe sink yang dipakai pada modul input tipe sink. Perubahan modul sink ke source bisa dilakukan dengan menggunakan rangkaian berikut untuk arus Iout 50 mA.

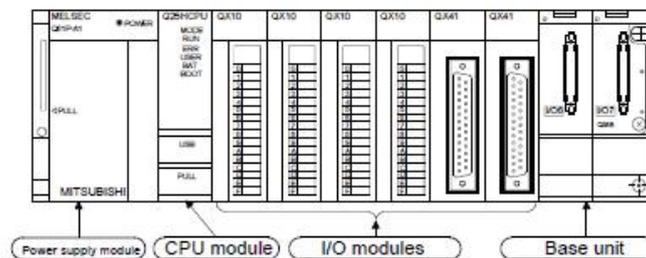
2.1.3. Modul Digital Output (DO)

Banyak sekali variasi output yang tersedia, PLC kompak dan mikro biasanya hanya memiliki satu jenis output, sedangkan modular sistem memungkinkan adanya variasi tegangan output dari PLC yang sama. Pada modul ini tegangan yang digunakan bisa AC dan atau DC. Tegangan output yang mampu diolah:

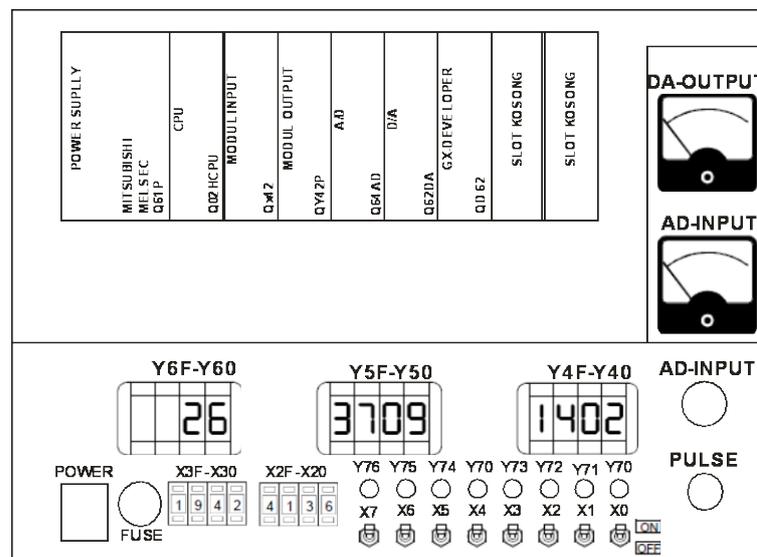
- 12 - 48 Volt AC/DC
- 120 Volt AC/DC
- 230 Volt AC/DC
- Contact (relay)
- Isolated output
- 5 - 50 Volt DC (sink/source)

2.2. Modul Simulator PLC Q-Series

Untuk belajar pemrograman dengan PLC Mitsubishi Q-Series ini modul PLC yang dipergunakan dipasang pada sebuah panel yang berupa simulator untuk mensimulasikan masukan sensor (switch) maupun keluaran berupa lampu. Secara lengkap bisa dilihat pada Gambar 2.3. dan Gambar 2.4.



Gambar 2.3 Susunan Modul PLC Q-Series.



Gambar 2.4 Simulator PLC Q-Series

2.3. Pemrograman PLC

Menurut IEC 61131-3 bahasa pemrograman yang digunakan untuk memprogram suatu PLC antara lain Ladder Diagram, Function Block Diagram, Statement List, Structure Text, dan Sequential Functional Chart. Tetapi dalam aplikasi nyatanya bahasa pemrograman PLC yang paling banyak digunakan adalah jenis ladder diagram.

IEC 61131-3 menetapkan lima bahasa pemrograman:

- Representasi **gambar/symbol**, dapat berupa
 - Ladder Diagram (**LAD**)
 - Diagram Blok Fungsi (Function Block Diagram / **FBD**)
 - Urutan Chart Fungsi (Sequential Function Chart / **SFC**)
- Tabel **perintah**, dapat berupa
 - Daftar Instruksi (Statement List / **STL**)
 - Teks Terstruktur (Structured Text / **ST**)

Tabel 2.1. Bahasa Pemrograman PLC

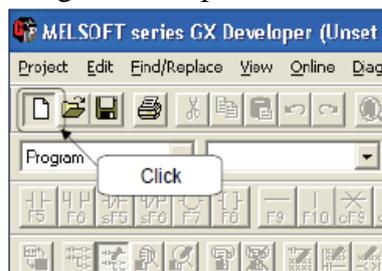
Ladder logic diagram	(LD)	Graphical	Discrete control
Function Block Diagram	(FBD)	Graphical	Continuous control
Sequential Function Chart	(SFC)	Graphical	Sequencing
Statement List	(STL)	Textual	Similar to ladder diagrams
Structured Text	(ST)	Textual	Complex logic, computations

3. Pemrograman PLC dengan GX Developer

GX Developer adalah software yang digunakan untuk memprogram PLC Mitsubishi. Software yang dipakai ini versi 8 dan rekomendasinya hanya bisa beroperasi secara maksimal di Windows XP. Untuk memudahkan belajar pemrograman Basic ini digunakan Ladder Diagram dengan pendekatan pemrograman sekuensial.

3.1. Langkah-Langkah Pengoperasian GX-Developer

- 1) Pilih [**MELSOFT Application**], Click [**GX Developer**]
- 2) Buat New Project, dengan cara seperti Gambar 3.1



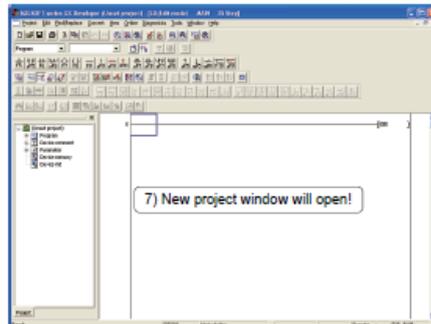
Gambar 3.1. Tampilan New Project

- 3) Konfirmasikan bahwa “PLC Series” adalah Q-CPU dan “PLC Type” adalah Q02H, seperti langkah dibawah :



Gambar 3.2. Konfirmasi PLC dan Tipe

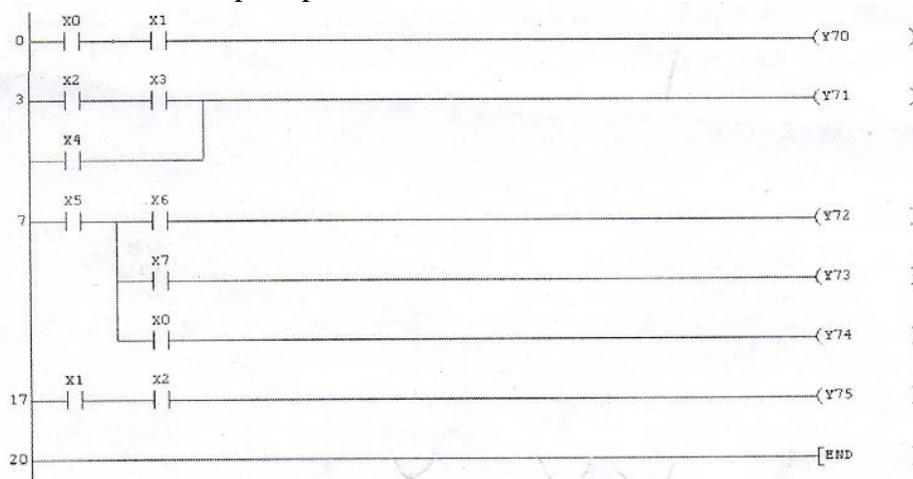
- 4) Akan muncul tampilan Window baru, sebagai tanda siap untuk mulai Program.



Gambar 3.3. Tampilan Siap Program

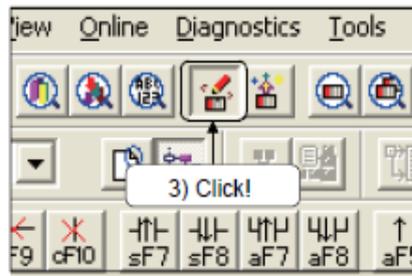
3.1.1 Latihan

- Buat Ladder seperti pada contoh di bawah berikut :



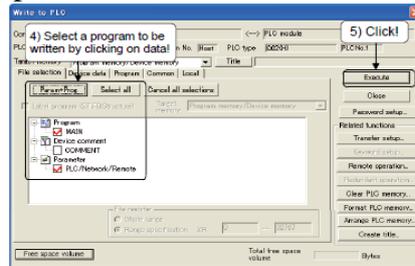
Gambar 3.4. Ladder Latihan 1

- 5) Setelah program ditulis, **Convert** Program dengan F4
- 6) Kemudian Simpan Program dengan **Ctrl S**
- 7) Sebelum melakukan proses Download, pastikan komunikasi PLC dengan PC terhubung secara Serial, setelah itu Download Program dengan cara :



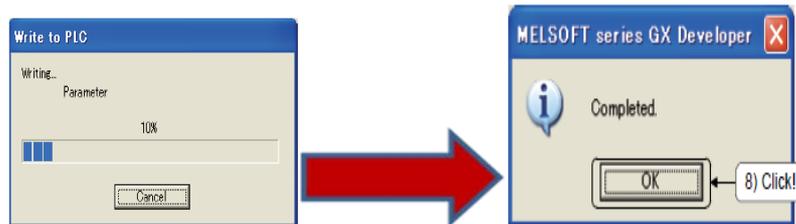
Gambar 3.5. Icon download

8) Akan muncul tampilan Window baru, kemudian lakukan sbb :



Gambar 3.6. Pilih Program dan Parameter

9) Setelah di **Execute**, maka indikasi bahwa proses Download sudah selesai akan muncul windows baru, dan kemudian Click **OK**



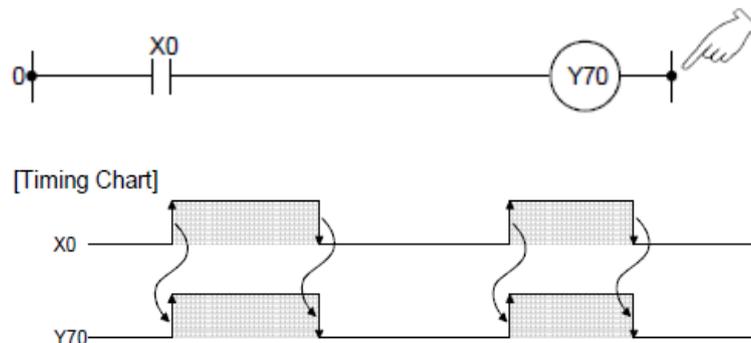
Gambar 3.7. Tampilan saat Proses Download dan Selesai Proses

10) Program telah berhasil di Download ke PLC, setelah itu kita cek apakah program sesuai dengan apa yang kita inginkan.

3.2. Intruksi-Intruksi Dasar Pemrograman

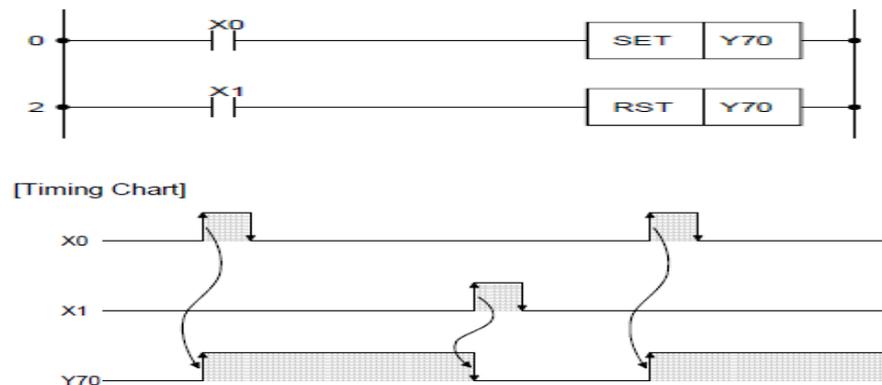
1. Kontak **Input(X)** dan **Output(Y)**

Intruksi-intruksi dasar dapat menggunakan perintah-perintah sbb



Intruksi kontak out(Y) berfungsi mengaktifkan output (Y70) jika menerima masukan dari X0 dan output akan OFF, jika input dalam keadaan OFF seperti yang ditunjukkan pada *timing chart* diatas.

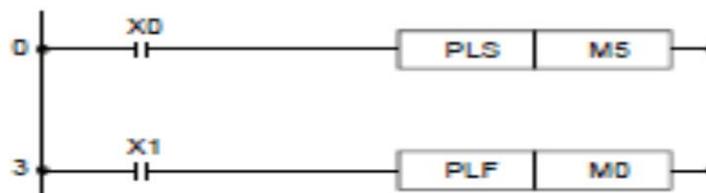
2. SET dan RST



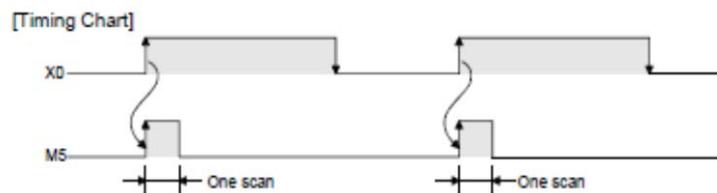
Gambar 3.9. Set dan Reset

Intruksi SET berfungsi mengaktifkan output jika menerima input X0 dan output akan tetap ON, meskipun input X0 dalam keadaan OFF. Untuk meng-OFF-kan output, maka digunakan intruksi RST, dengan cara mengaktifkan input X1.

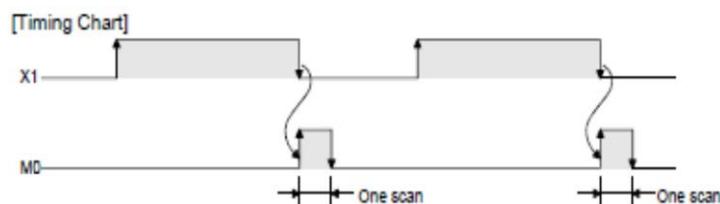
3. PLS (Pulse) dan PLF (Pulf)



PLS yaitu aktif ketika menerima input sesaat sinyal dalam kondisi naik.



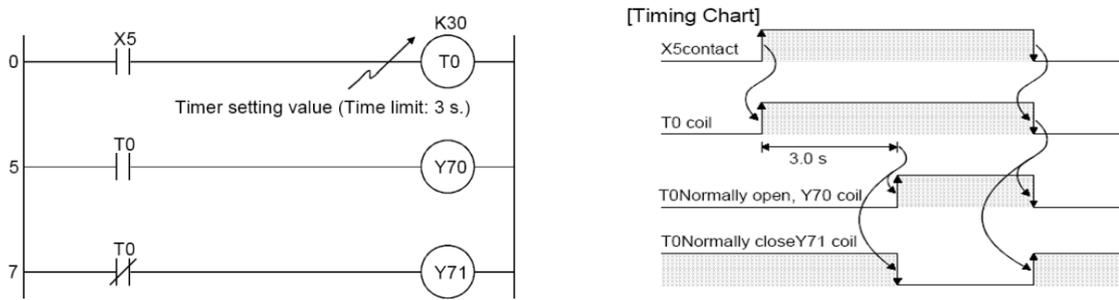
PLF yaitu aktif ketika menerima input sesaat sinyal dalam kondisi turun.



Gambar 3.10. Ladder PLS dan PLF

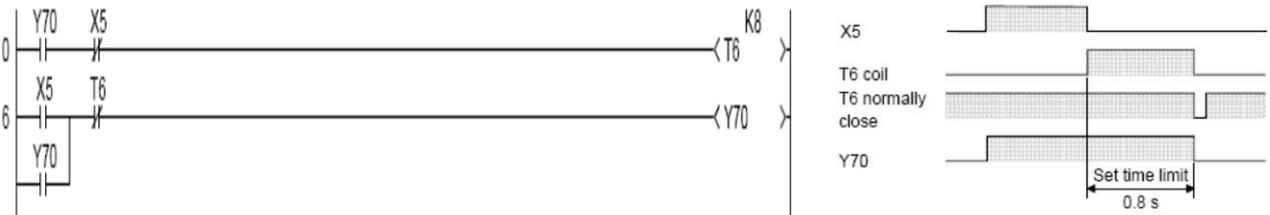
4. Timer (T)

Timer On Delay : Timer ini akan hidup setelah suatu periode waktu tunda yang telah ditetapkan. Nilai Timer 0,1 s/d 3276,7 detik (K1 sampai K32767).



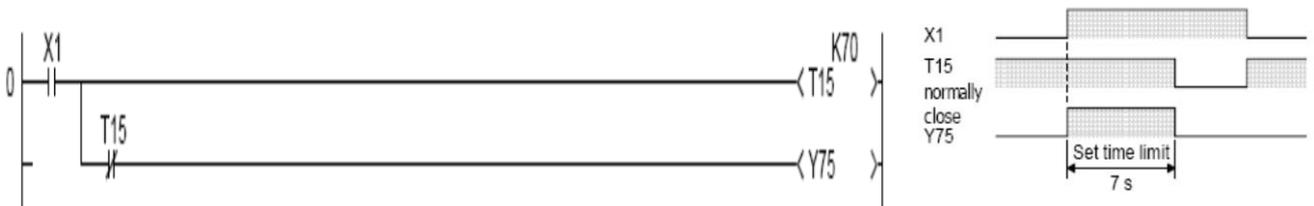
Gambar 3.11. Ladder Diagram dan Timing Diagram Timer On Delay

Timer Off Delay : Timer ini (T6) akan beroperasi ketika X5 di-OFF-kan.



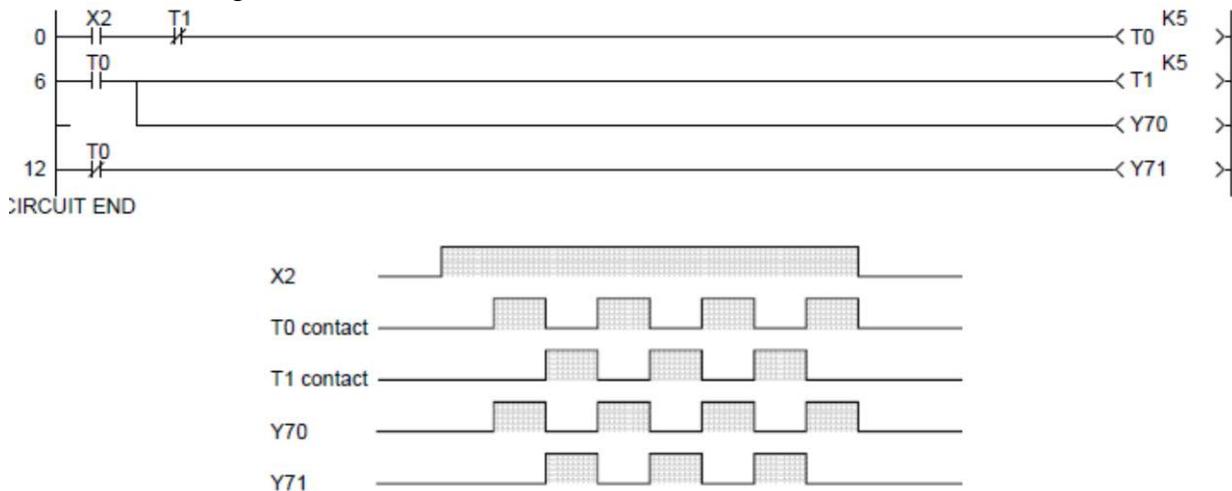
Gambar 3.12 Ladder Diagram dan Timing Diagram Timer Off Delay

One Shoot Timer : Timer ini akan berada dalam kondisi hidup selama periode waktu yang telah ditetapkan dan kemudian mati.



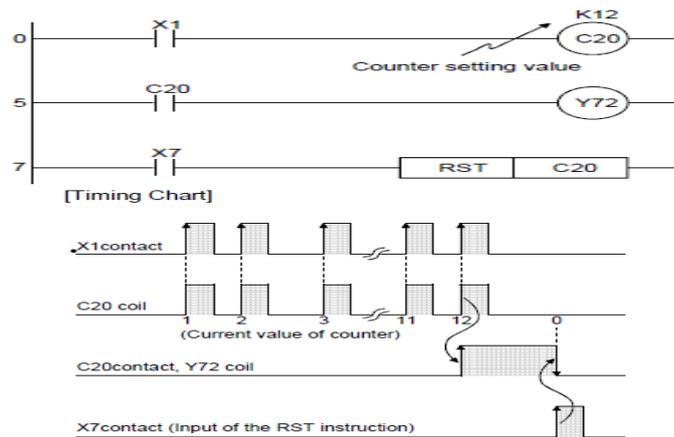
Gambar 3.13 Ladder Diagram dan Timing Diagram One Shoot Timer

Flip-flop : Timer ini mengatur waktu ON dan OFF output (Y70 dan Y71) secara bergantian.



Gambar 3.14 Ladder Diagram dan Timing Diagram Flip-flop

5. Counter (C)

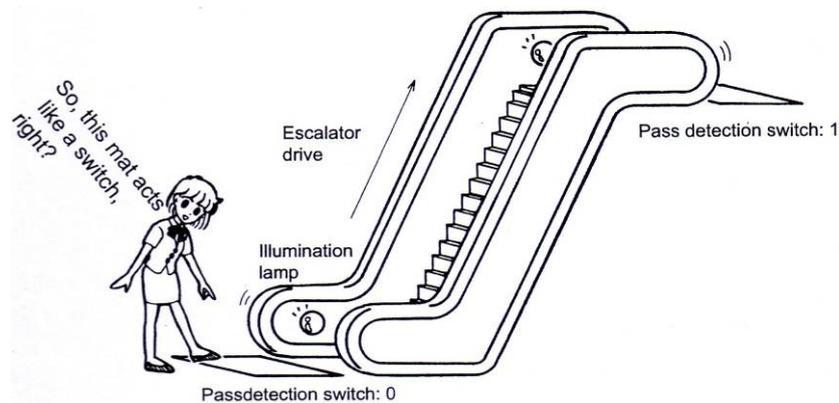


Gambar 3.15. Ladder Counter

Berdasar *timing diagram*, hitungan pada *counter* akan mulai saat sinyal input naik. Setelah nilai *counter* tercapai, sinyal input berikutnya tidak akan terhitung oleh *counter* atau nilai *default counter* tidak akan berubah hingga intruksi RST di eksekusi. Dengan intruksi RST, maka nilai *counter* akan berubah menjadi 0 (nol). Batas hitungan pada *counter* yang diperbolehkan dari K0 hingga K32767.

3.3. Latihan-latihan

3.3.1. Latihan 1: Control of Escalators



Gambar 3.16. Eskalator Naik

Keterangan:

1. Diasumsikan bahwa eskalator tidak bergerak sampai seseorang mendekatinya.
2. Ketika seseorang mendekati eskalator maka akan mengaktifkan *passdetection switch:0*, *illumination lamp* aktif, dan *escalator drive* mulai menggerakkan eskalator naik.
3. Orang yang mendekat eskalator tersebut akan dibawa naik dan ketika sampai diatas akan mengaktifkan *passdetection switch:1*, mematikan *illumination lamp* dan mematikan *escalator drive*.

Penjelasan:

1. Pada contoh di atas *passdetection switch:0* dan *passdetection switch:1* merupakan sensor tipe *non-hold*.

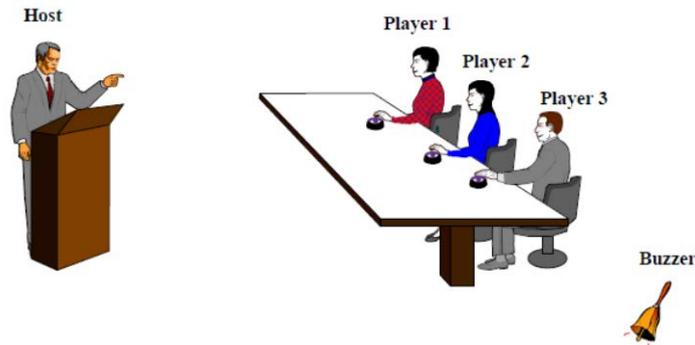
2. Asumsi yang lain pada contoh di atas bahwa orang yang akan naik eskalator satu persatu (jawa: gembrudug).

Pengalamatan I/O:

Masukan	Alat
	Passdetection switch:0
	Passdetection switch:1

Keluaran	Alat
	Escalator drive
	Illumination lamp

3.3.2. Latihan 2: Bel Kuis



Gambar 3.17. Acara Kuis

Keterangan:

1. Pembawa acara (Host) memberikan pertanyaan kepada 3 (tiga) peserta kuis;
2. Ke-tiga pemain berlomba-lomba untuk menekan tombol dalam rangka menjawab pertanyaan dari pembawa acara;
3. Buzzer akan berbunyi setelah ada salah seorang pemain berhasil menekan tombol untuk pertama kalinya;
4. Indikator lampu pada pemain tersebut (yang berhasil menekan tombol untuk pertama kali) akan dinyalakan dan hanya bisa dimatikan oleh saklar utama.

Penjelasan:

1. Pada contoh di atas tombol untuk peserta merupakan sensor tipe *non-hold*.

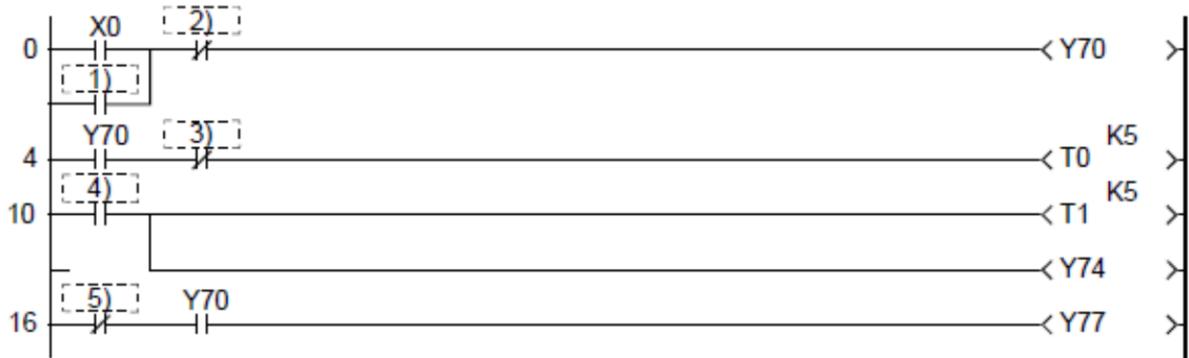
Pengalamatan I/O:

Masukan	Alat
	Player 1
	Player 2
	Player 3
	Reset

Keluaran	Alat
	Player 1 light
	Player 2 light
	Player 3 light
	Buzzer

3.3.4. Latihan 3: Lampu Berkedip

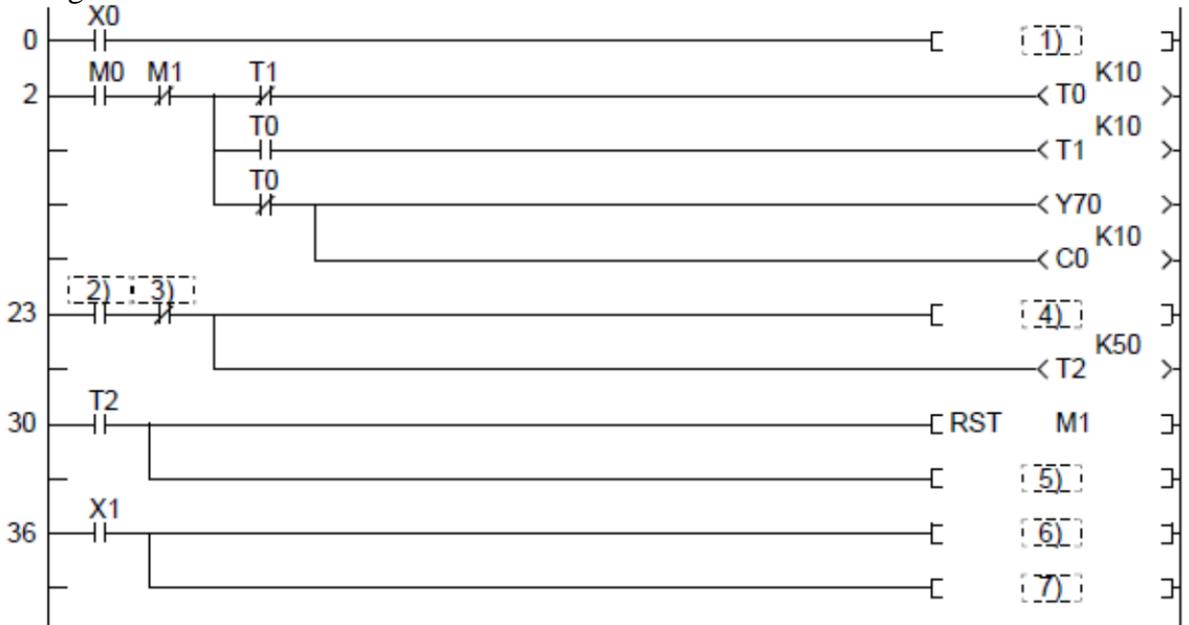
- 1) Ketika X0 aktif, Y70 akan tetap aktif, dan Y74 dan Y77 akan berkedip setiap 0,5 detik. Ketika X1 aktif, Y70 OFF dan lampu yang berkedip, Y74 dan Y77 juga berhenti.



Isilah titik-titik dibawah ini :

- 1..... 3..... 5.....
 2..... 4.....

Ketika X0 aktif, Y70 mulai berkedip setiap 1 detik dan berhenti berkedip selama 5 detik setelah berkedip 10 kali. Kemudian akan berkedip kembali. Kedipan akan berhenti dengan menekan tombol X1 ON.



Isilah titik-titik dibawah ini :

- 1..... 3..... 5..... 7.....
 2..... 4..... 6.....