

BAB I

PENDAHULUAN

1.1 Latar Belakang

Latar belakang cuci tangan yang higienis kerap terlupakan oleh banyak orang, padahal mencuci tangan dengan air dan sabun terutama pada saat-saat penting yaitu sebelum memegang makanan dapat mencegah penularan virus dan bakteri penyebab penyakit infeksi terutama diare atau penularan penyakit lainnya. Dengan adanya fasilitas mencuci tangan di sekolah-sekolah, kampus, supermarket, rumah makan, rumah sakit dan tempat-tempat sarana umum lainnya yang memiliki wastafel dengan air mengalir, dan sabun serta tisu untuk sekali pakai untuk mengeringkan tangan, dapat memberikan dampak positif terutama pada kesehatan. Wastafel merupakan suatu hal yang tidak asing lagi dalam perlengkapan kamar mandi, juga sarana umum lainnya. Wastafel berfungsi untuk mencuci tangan atau hanya sekedar membersihkan muka. Walaupun fungsinya sederhana, tapi wastafel ini dirancang menarik tanpa harus memutar atau menekan kran dan sabun. Saat sensor dilewati oleh tangan, kran air dan sabun akan keluar dan berhenti sendiri otomatis ketika selesai digunakan. Karena, di dalam rumah sakit dan restoran juga sarana umum lainnya proses pencucian tangan masih banyak dilakukan secara manual sangatlah tidak efisien, kurang akurat kebersihannya dan membutuhkan waktu yang relatif lama. Sebenarnya jika proses pelayanan tersebut dapat di otomatiskan akan sangat menguntungkan, baik itu bagi

sinilah penulis tertarik untuk membuat alat otomatisasi kran pencuci tangan menggunakan PLC (*Programmable Logic Controller*) omron CPM1A sebagai perangkat softwearnya dan infrared. Sehingga kenyamanan dan keefisienan sebuah wastafel dirasakan jadi sangat penting.

1.2 Rumusan Masalah

Berdasarkan hal di atas, dalam penulisan ini penulis akan membuat alat yang bekerja berdasarkan sensor infra merah sebagai pengirim dan sekaligus penerima data untuk mengalirkan air dan sabun. Alat yang dibuat ini bekerja berdasarkan intensitas gerakan dan suhu tubuh manusia tentunya yang kemudian masuk pada sensor tersebut untuk menentukan aktif atau tidaknya kran air dan sabun pada wastafel. Dalam hal ini, akan digunakan sensor PIR yang berfungsi sebagai sensor jarak yang kemudian akan diteruskan ke rangkaian melalui sebuah PLC (*Programmable Logic Controller*) omron CPM1A sebagai perangkat lunak untuk mengatur kinerja aliran air dan sabun sesuai yang diinginkan.

1.3 Batasan Masalah

Adapun batasan masalah pada pembuatan alat kran dan sabun otomatis adalah sebagai berikut :

1. Perancangan dan pembuatan alat mekanik yang akan digunakan untuk alat mencuci tangan otomatis.
2. Perancangan rangkaian softwear atau program PLC (*Programmable*

pengolah data dan pengatur kerja perangkat keras lainnya dalam sistem aliran air dan sabun.

3. Perancangan dan cara menggunakan rangkaian sensor yang akan digunakan untuk mendeteksi benda yang akan melewati sensor.
4. Perancangan perangkat lunak untuk mengatur jarak atau waktu relay dalam aliran air dan sabun sesuai yang diinginkan.
5. Jarak benda 15cm – 25cm dan waktu aliran air 2s – 50s serta sabun 3,5s.

1.4 Tujuan dan Manfaat

1.4.1 Tujuan

Tujuan yang hendak dicapai dalam pembuatan alat ini adalah membuat suatu alat yang dapat memberikan keefisienan dalam menjaga kesehatan tubuh dengan menggunakan PLC (*Programmable Logic Controller*) Omron CPM1A sebagai pengolah dan pengatur perangkat keras lainnya.

1.4.2 Manfaat

Manfaat yang diperoleh dari tugas akhir ini adalah :

1. Bagi Mahasiswa
 - a. Memperoleh kesempatan terjun ke dalam dunia praktis dan menerapkan ilmu yang telah diperoleh untuk menyelesaikan permasalahan dalam dunia nyata.
 - b. Penelitian ini sebagai sarana pengembangan bagi diri sendiri dan memperluas wawasan di dunia nyata.

2. Bagi pihak Universitas Muhammadiyah Yogyakarta “ UMY “

Sebagai tambahan pustaka bagi jurusan Teknik Elektro, Fakultas Teknik, dan Universitas Muhammadiyah Yogyakarta “ UMY “ pada umumnya.

3. Bagi Sarana Umum

Bila hasil penelitian dan pembuatan alat ini benar-benar di realisasikan, akan sangat bermanfaat bagi khalayak ramai tentunya dalam menjaga kesehatan.

1.5 Metode Penelitian

Karya tulis ini disusun dengan tahap-tahap :

1. Studi Kepustakaan (*Library Study*) yaitu studi yang dilakukan untuk mendapat pengetahuan dengan cara mengumpulkan data baik dari internet maupun dari buku-buku.
2. Studi Lapangan (*Field Study*) yaitu dengan cara mengumpulkan komponen yang diperlukan untuk perancangan, merakit komponen dan melakukan pengujian bertahap.
3. Metode Observasi yaitu pengamatan terhadap alat yang akan dibuat, dengan cara melakukan percobaan baik secara langsung maupun secara tidak langsung.
4. Metode Perancangan yaitu tahap perancangan yang akan dibuat, mengenai perencanaan rangkaian, tata letak komponen-komponen

BAB II

LANDASAN TEORI

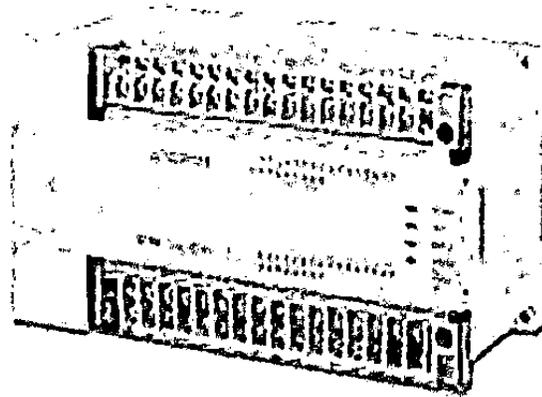
2.1 Tinjauan Pustaka

Untuk karya yang berhubungan dengan alat ini adalah :

- *AUTO CRANE FOR SINK*. Hikman Sari, *Under Graduate Program Faculty of Industrial Technology GUNADARMA University Library* (2009). Pada karya ini untuk otomatisasi kran air menggunakan Photo dioda sebagai sensor cahaya dan IC LM358 yang berfungsi sebagai komparator tegangan.
- DYAH AYU WULANDARI NIM J0D005026. *Sistem Otomatisasi Kran Pencuci Tangan*. PROGRAM STUDI DIII, FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM (2008). UNIVERSITAS DIPONEGORO SEMARANG. Pada karya ini otomatisasi kran air dan sabun menggunakan infrared sebagai sensor posisi tangan. Menggunakan relay sebagai actuator. Menggunakan mikrokontoller AT89S51 sebagai pengolah data.
- ESA KURNIAWAN (11246/108.EI). *User Manual Kran Air Otomatis*. SMA NEGERI 3 BONYOLANGU TULUNG AGUNG Tahun pelajaran 2010/2011. Pada karya ini Otomatisasi kran air dan sabun menggunakan LDR sebagai sensor yang bekerja apabila

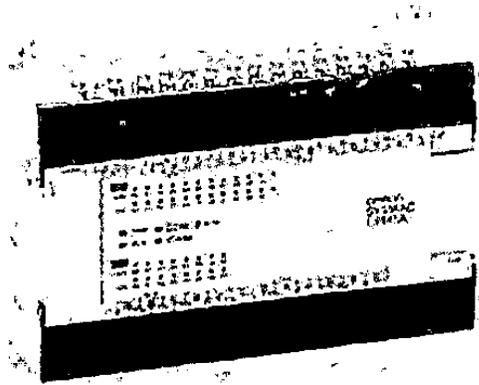
2.2 Dasar Teori

2.2.1 *Programmable Logic Controller (PLC)*

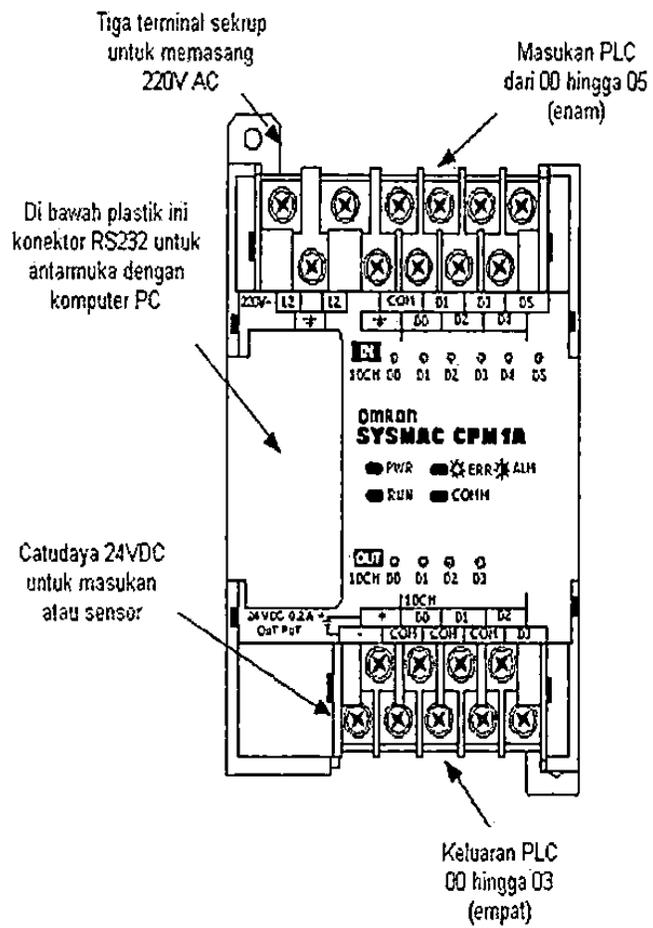


Gambar 2.1 Jenis-Jenis PLC

PLC (*Programmable Logic Controller*) adalah suatu peralatan elektronika yang bekerja secara digital memiliki memori yang dapat di program, menyimpan perintah-perintah untuk melakukan fungsi-fungsi khusus seperti *logic*, *sequencing*, *timming*, *counting* dan *aritmatik* untuk mengontrol berbagai jenis motor atau proses melalui modul input output analog atau digital. Perangkat lunak programnya sama sekali berbeda dengan dengan bahasa komputer seperti Pascal, Basic, C dan lain-lain, programnya menggunakan apa yang dinamakan sebagai diagram tangga atau *ladder diagram*. Didalam PLC berisi rangka elektronika yang dapat difungsikan seperti *contact relay* (baik NO maupun NC). Pada PLC OMRON CPM1A memiliki 6 masukan (D0-D5) dan 4 keluaran (Q0-Q3) total 10 jalur keluaran/masukan (10



Gambar 2.2 PLC Omron CPM1A



Gambar 2.3 Rangkaian PLC Omron CPM1A

Berikut adalah fasilitas yang ada pada PLC Omron CPM1A :

- Terdapat 4 lampu indikator yaitu, PWR (hijau), RUN (hijau), ERR/ALM (merah) dan COMM (kuning)
- Unit Power Supply (*Switching Power Supply*), yang berfungsi untuk memberikan tegangan pada blok CPU PLC
- CPU (*Central Processing Unit*) PLC
- Terdapat fasilitas untuk melakukan hubungan komunikasi dengan komputer maupun laptop, melalui RS-232C atau yang lebih dikenal dengan port serial.
- Terdapat 3 Memori Unit yaitu : *RAM (Random Acces Memory)*, *EPROM (Eraseable Programmable Read Only Memory)*, *EEPROM (Electrical Eraseable Programmable Read Only Memory)*. Yang berfungsi untuk pembuatan, pengaturan perintah-perintah dan penyimpanan program PLC
- Memiliki 6 input (D0-D5) dan 4 output (O0-O3) total terdapat 10 jalur input dan outputnya
- Input Point Digital, DC 24V input, DC 5V input / TTL (Transistor Transistor Logic), AC/DC 24V input, AC 110V input, AC 220V input
- Input Point Linear, 0 – 10 V DC, -10 V DC – 10 V DC, 4 – 20 mA DC
- Output Point Digital, Relay Output, AC 110 V output, AC 220

2.2.2.2 Deskripsi Data Area PLC Omron CPM1A

Data area		Words	Bits	Function
IR area ¹	Input area	IR 000 to IR 009 (10 words)	IR 00000 to IR 00915 (160 bits)	These bits can be allocated to the external I/O terminals.
	Output area	IR 010 to IR 019 (10 words)	IR 01000 to IR 01915 (160 bits)	
	Work area	IR 200 to IR 231 (32 words)	IR 20000 to IR 23115 (512 bits)	Work bits can be freely used within the program.
SR area		SR 232 to SR 255 (24 words)	SR 23200 to SR 25515 (384 bits)	These bits serve specific functions such as flags and control bits.
TR area		---	TR 0 to TR 7 (8 bits)	These bits are used to temporarily store ON/OFF status at program branches.
HR area ²		HR 00 to HR 19 (20 words)	HR 0000 to HR 1915 (320 bits)	These bits store data and retain their ON/OFF status when power is turned off.
AR area ²		AR 00 to AR 15 (16 words)	AR 0000 to AR 1515 (256 bits)	These bits serve specific functions such as flags and control bits.
LR area ¹		LR 00 to LR 15 (16 words)	LR 0000 to LR 1515 (256 bits)	Used for a 1:1 data link with another PC.
Timer/Counter area ²		TC 000 to TC 127 (timer/counter numbers) ³		The same numbers are used for both timers and counters.
DM area	Read/write ²	DM 0000 to DM 0999 DM 1022 to DM 1023 (1,002 words)	---	DM area data can be accessed in word units only. Word values are retained when the power is turned off.
	Error log	DM 1000 to DM 1021 (22 words)	---	Used to store the time of occurrence and error code of errors that occur. These words can be used as ordinary read/write DM when the error log function isn't being used.
	Read-only ⁴	DM 6144 to DM 6599 (456 words)	---	Cannot be overwritten from program.
	PC Setup ⁴	DM 6600 to DM 6655 (56 words)	---	Used to store various parameters that control PC operation.

Tabel 2.1 Data Area PLC Omron CPM1A

- Daerah IR

Bagian memori ini digunakan untuk menyimpan status keluaran dan masukan PLC. Beberapa bit berhubungan langsung dengan terminal masukan (input) dan keluaran (output) yang dinamakan terminal sekrup. Daerah memori IR terbagi atas tiga macam area, yaitu area masukan (*input area*), area keluaran (*output area*), dan area kerja (*work area*). Untuk mengakses memori ini, cukup dengan menuliskan

1. 000 untuk masukan, 010 untuk keluaran, dan 200 untuk

Number of I/O Terminals on the CPU Unit	CPU Unit Terminals		Expansion I/O Unit Terminals		Power Supply	Model Number
	Inputs	Outputs	Inputs	Outputs		
10	6 points: 00000 to 00005	4 points: 01000 to 01003	12 points: 00100 to 00111	8 points: 01100 to 01107	AC	CPM1-10CDR-A
					DC	CPM1-10CDR-D
20	12 points: 00000 to 00011	8 points: 01000 to 01007	12 points: 00100 to 00111	8 points: 01100 to 01107	AC	CPM1-20CDR-A
					DC	CPM1-20CDR-D
30	18 points: 00000 to 00011 00100 to 00105	12 points: 01000 to 01007 01100 to 01103	12 points: 00200 to 00211	8 points: 01200 to 01207	AC	CPM1-30CDR-A
			12 points: 00200 to 00211	8 points: 01200 to 01207	DC	CPM1-30CDR-D
			36 points: 00200 to 00211 00300 to 00311 00400 to 00411	24 points: 01200 to 01207 01300 to 01307 01400 to 01407	AC	CPM1-30CDR-A-V1 (Available soon)
			36 points: 00200 to 00211 00300 to 00311 00400 to 00411	24 points: 01200 to 01207 01300 to 01307 01400 to 01407	DC	CPM1-30CDR-D-V1 (Available soon)

Tabel 2.2 IR Area PLC Omron CPM1A

- Daerah SR

Daerah SR merupakan bagian khusus dari lokasi memori yang digunakan sebagai bit-bit kontrol dan status (*flag*), paling sering digunakan untuk pencacah dan interupsi. Misalnya, SR253.13 (atau 253.13 saja) adalah *Always ON* Flag atau nilainya selalu ON selama PLC dihidupkan, sedangkan SR253.14 (atau 253.14 saja) adalah *Always OFF* Flag atau nilainya selalu OFF selama PLC dihidupkan. SR255.04 (atau 255.04 saja) digunakan sebagai *Flag CARRY* (CY). SR255.05, SR255.06, dan SR255.07 masing-masing digunakan untuk menyimpan status Lebih Besar Dari (*Greater Than*), Sama Dengan (*Equals*). Dan Lebih Kecil Dari (*Less Than*) hasil dari fungsi perbandingan CMP.

- Daerah TR

Saat pindah ke sub-program selama eksekusi program maka semua

dalam daerah TR ini. Untuk PLC CPM1A (juga CPM2A) hanya memiliki 8 bit, yaitu TR0 sampai TR7.

- Daerah HR

Bit-bit pada daerah HR ini digunakan untuk menyimpan data dan tidak akan hilang walaupun PLC sudah tidak mendapat catu daya atau PLC sudah dimatikan, karena menggunakan baterai. Untuk PLC CPM1A (juga CPM2A) sendiri, daerah ini terdiri dari 20 word (HR00 sampai HR19) atau 320 bit (HR00.00 sampai HR19.15). Bit-bit HR ini dapat digunakan bebas didalam program sebagaimana bit-bit kerja (work bits).

- Daerah AR

Ini merupakan daerah lain yang juga digunakan untuk menyimpan bit-bit kontrol dan status, seperti status PLC, kesalahan, waktu sistem dan lain sejenisnya. Dan seperti daerah HR, daerah AR juga dilengkapi dengan baterai, sehingga data-data kontrol maupun status tetap akan tersimpan walaupun PLC sudah dimatikan. Untuk PLC CPM1A, daerah ini terdiri dari 16 word (AR00 sampai AR15) atau 256 bit (AR00.00 sampai AR15.15). Misalnya, AR08 bit 00 sampai 03

... ..

03 – Kesalahan Overrun.

- Daerah LR

Digunakan sebagai pertukaran data saat dilakukan koneksi atau hubungan dengan PLC yang lain. Untuk PLC CPM1A dan CPM2A, daerah ini terdiri atas 16 word (LR00 sampai LR15) atau 256 bit (LR00.00 sampai LR15.15).

- Daerah Pewaktu/Pencacah (*Timer/Counter*) – T/C Area

Daerah ini digunakan untuk menyimpan nilai-nilai pewaktu atau pencacah. Untuk PLC CPM1A terdapat 128 lokasi (TC000 sampai TC127), sedangkan pada PLC CPM2A terdapat 226 lokasi (TC000 sampai TC225).

- Daerah DM

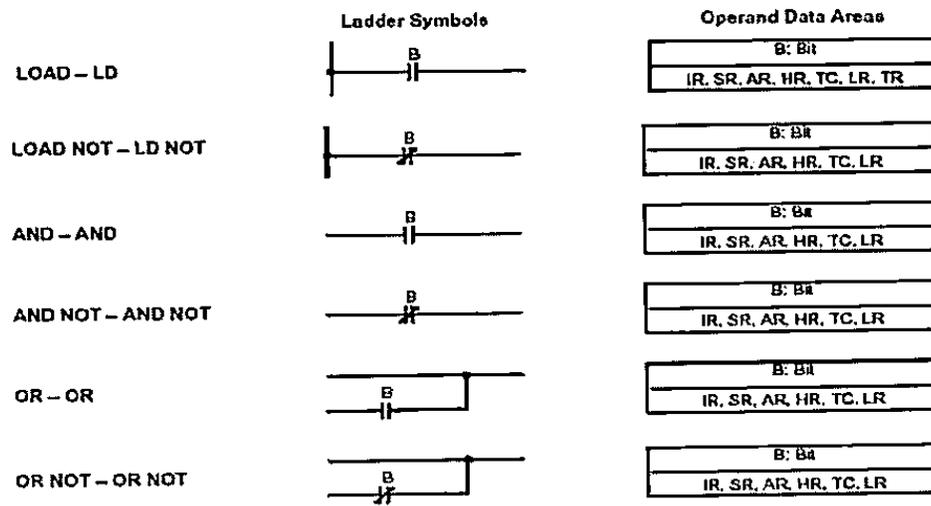
Berisikan data-data yang terkait dengan pengaturan komunikasi dengan komputer dan data pada saat ada kesalahan.

2.2.2.3 Kode Instruksi Pada PLC

Pada PLC omron CPM1A terdapat kode-kode instruksi sebagai berikut :

1. Instruksi-instruksi Tangga (*Ladder Instructions*)

Instruksi-instruksi tangga adalah instruksi-instruksi yang terkait dengan kondisi-kondisi didalam diagram tangga. Instruksi-instruksi tangga, baik yang independen maupun yang kombinasi atau gabungan

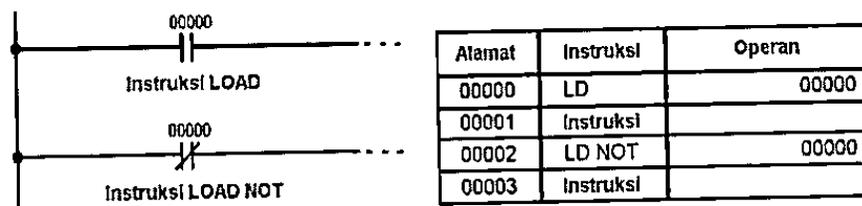


Gambar 2.4 Kode Mnemonik, Diagram Tangga dan Area Data Operan

Dari Instruksi- instruksi Tangga.

a. LOAD (LD) dan LOAD NOT (LD NOT)

Kondisi pertama yang mengawali sembarang blok logika di dalam diagram tangga berkaitan dengan instruksi LOAD (LD) atau LOAD NOT (LD NOT).



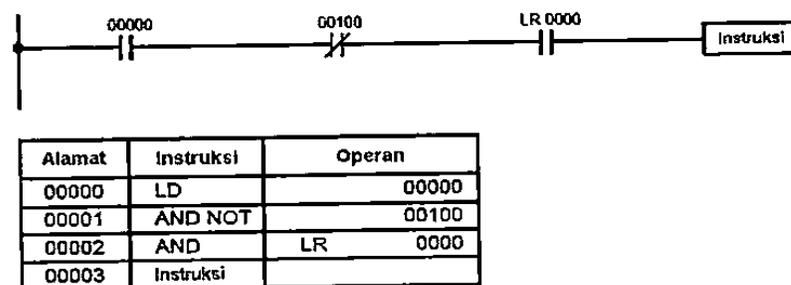
Gambar 2.5 Contoh Instruksi LD dan LD NOT, Tabel 2.3 Area Operan

LD dan LD NOT

b. AND dan AND NOT

Jika terdapat dua atau lebih kondisi yang dihubungkan seri pada garis instruksi yang sama maka kondisi pertama menggunakan instruksi LD atau LD NOT, dan sisanya menggunakan instruksi AND

tangga yang mengandung tiga kondisi yang dihubungkan secara seri pada garis instruksi yang sama dan berkaitan dengan instruksi LD, ANDNOT, dan AND. Masing-masing instruksi tersebut membutuhkan satu baris kode mnemonik.

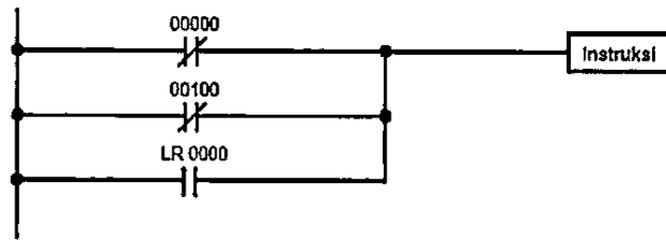


Gambar 2.6 Contoh Penggunaan Instruksi AND dan AND NOT, Tabel

2.4 Area Operan AND dan AND NOT

c. OR dan OR NOT

Jika dua atau lebih kondisi yang dihubungkan paralel, artinya dalam garis instruksi yang berbeda kemudian bergabung lagi dalam satu garis instruksi yang sama maka kondisi pertama terkait dengan instruksi LD dan LD NOT dan sisanya berkaitan dengan instruksi OR dan OR NOT. Gambar 2.9 menunjukkan tiga buah instruksi yang berkaitan dengan instruksi LD NOT, OR NOT, dan OR. Masing-



Alamat	Instruksi	Operan
00000	LD NOT	00000
00001	OR NOT	00100
00002	OR	LR 0000
00003	Instruksi	

Gambar 2.7 Contoh Penggunaan Instruksi OR dan OR NOT, Tabel 2.5

Area Operan OR dan OR NOT

d. Kombinasi instruksi AND dan OR

Jika instruksi AND dan OR digabung atau dikombinasikan dalam suatu rangkaian tangga yang kompleks maka bisa dipandang satu persatu, artinya bisa dilihat masing-masing hasil gabungan dua kondisi menggunakan instruksi AND atau OR secara sendiri-sendiri kemudian menggabungkannya menjadi satu kondisi menggunakan instruksi AND atau OR yang terakhir. Gambar 2.10 menunjukkan contoh diagram tangga yang mengimplementasikan cara seperti tersebut



Alamat	Instruksi	Operan
00000	LD	00000
00001	AND	00001
00002	OR	00200
00003	AND	00002
00004	AND NOT	00003
00005	Instruksi	

Gambar 2.8 Contoh Penggabungan Instruksi AND dan OR, Tabel 2.6

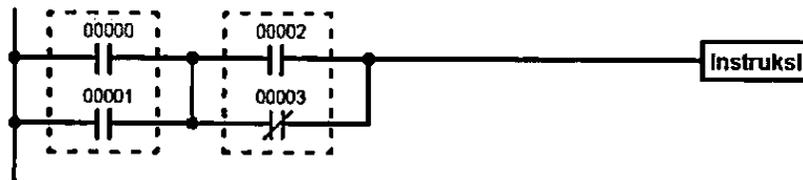
Area Operan AND dan OR

2. Instruksi-instruksi Blok Logika

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

a. AND LOAD (AND LD)

Gambar 2.11 menunjukkan contoh penggunaan blok logik AND LD yang terdiri atas dua blok logik, yang akan menghasilkan kondisi ON jika blok logik kiri dalam kondisi ON (salah satu dari IR000.00 atau IR000.01 yang ON) dan blok logik kanan juga dalam keadaan ON (IR000.02 dalam kondisi ON atau IR000.03 dalam kondisi OFF)



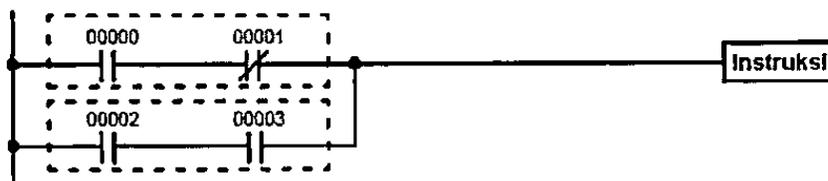
Alamat	Instruksi	Operan
00000	LD	00000
00001	OR	00001
00002	LD	00002
00003	OR NOT	00003
00004	AND LD	---

Gambar 2.9 Contoh Penggunaan Instruksi Blok Logika AND LD,

Tabel 2.7 Area Operan AND LD

b. OR LOAD (OR LD)

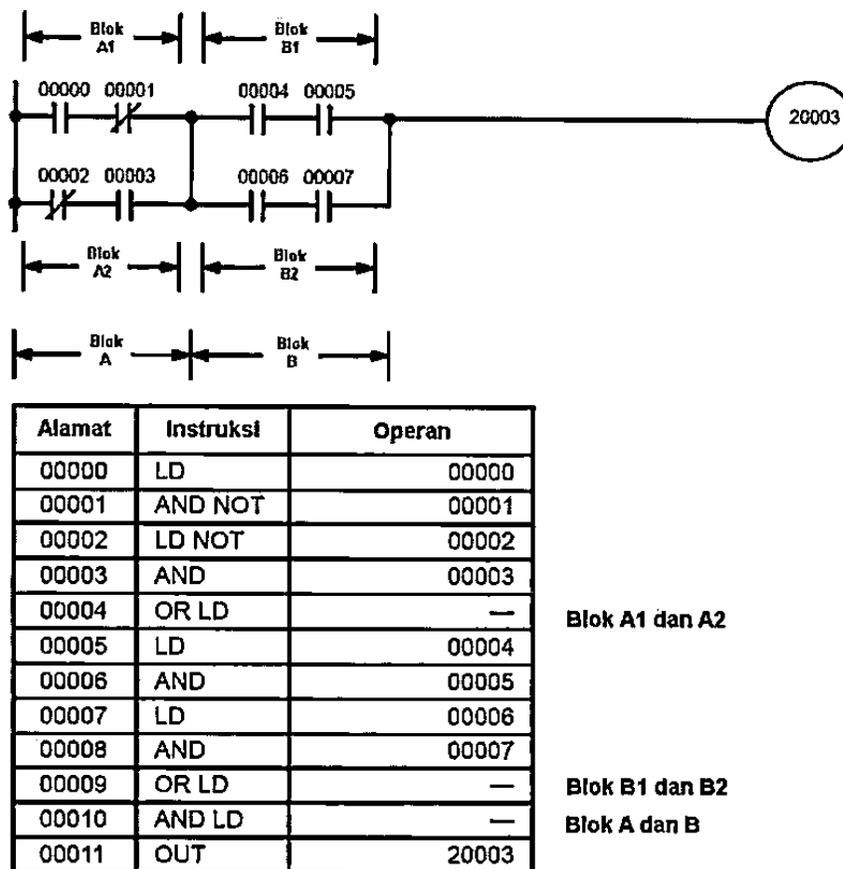
Instruksi ini digunakan untuk meng-OR-logik-kan dua blok logika. Gambar 2.12 menunjukkan contoh penggunaan blok logik OR LD yang terdiri atas dua blok logik. Kondisi eksekusi ON akan dihasilkan jika blok logik atas atau blok logik bawah dalam kondisi ON. Artinya, IR000.00 dalam kondisi ON dan IR000.01 dalam kondisi OFF atau IR000.02 dan IR000.03 dalam kondisi ON).



Alamat	Instruksi	Operan
00000	LD	00000
00001	AND NOT	00001
00002	LD	00002
00003	AND	00003
00004	OR LD	---

Gambar 2.10 Contoh Penggunaan Instruksi Blok Logika OR LD,

Untuk membuat kode mnemonik diagram tangga yang kompleks, caranya dengan cara membagi-bagi diagram tersebut ke dalam blok-blok logik yang besar, kemudian membagi lagi blok yang besar tersebut menjadi blok-blok logik yang lebih kecil, demikian seterusnya hingga tidak perlu lagi dibuat blok yang lebih kecil lagi. Blok-blok ini kemudian masing-masing dikodekan, mulai dari yang kecil, dan digabungkan satu persatu hingga membentuk diagram tangga yang asli. Instruksi blok logik AND LD dan OR LD hanya digunakan untuk menggabungkan dua blok logik saja (blok logik yang digabungkan berupa hasil penggabungan sebelumnya, atau hanya sebuah kondisi tunggal). Gambar 2.13 memperlihatkan suatu contoh diagram tangga yang kompleks, yang dapat dibagi dua blok besar (blok A dan B). Blok A dapat dibagi lagi menjadi dua blok yang lebih kecil (blok A1 dan A2), dan blok B dibagi menjadi dua blok yang lebih kecil, yaitu blok B1 dan B2. Kemudian blok-blok logik yang kecil ini ditulis terlebih dahulu, diawali dengan menuliskan blok A1 (alamat 00000 dan 00001) dan blok A2 (alamat 00002 dan 00003), kemudian digabung menggunakan instruksi blok logik OR LD (alamat 00004). Selanjutnya blok B1 dituliskan (alamat 00005 dan 00006) dilanjutkan dengan blok B2 (alamat 00007 dan 00008) dan digabung dengan instruksi blok logik OR LD (alamat 00009). Hasilnya berupa blok A dan blok B yang kemudian juga digabung menggunakan blok logik



Gambar 2.11 Contoh Diagram Tangga Yang Kompleks, Tabel 2.9

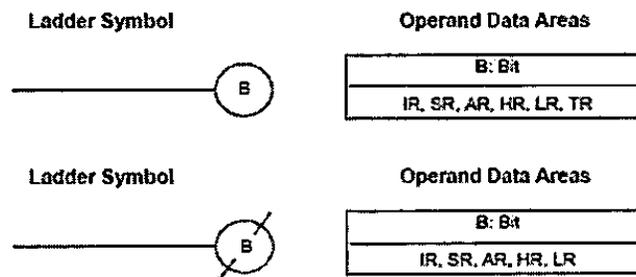
Area Operan Yang Kompleks

3. Instruksi Kontrol Bit

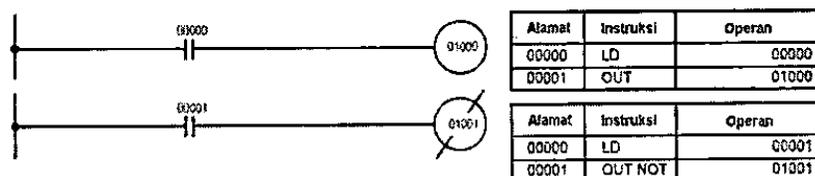
Terdapat tujuh instruksi dasar yang dapat digunakan untuk mengontrol status bit secara individual, yaitu *OUTPUT*, *OUTPUTNOT*, *SET*, *RESET*, *DIFFERENTIATE UP (DIFU)*, *DIFFERENTIATE DOWN (DIFD)* dan *KEEP*. Semua instruksi ini dituliskan disisi paling kanan diagram tangga dan membutuhkan sebuah alamat bit sebagai operan. Selain instruksi-instruksi ini digunakan untuk membuat bit-bit keluaran ON atau OFF dalam area IP (operasi sistem), mereka juga digunakan untuk mengontrol bit-

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

Instruksi ini digunakan untuk mengontrol operan yang berkaitan dengan kondisi eksekusi (apakah ON atau OFF). Dengan menggunakan instruksi OUT, maka bit operan akan menjadi ON jika kondisi eksekusinya juga ON, sedangkan OUT NOT akan menyebabkan bit operan menjadi ON jika kondisi eksekusinya OFF. Gambar 2.14 memperlihatkan simbol tangga dan area data operan dari instruksi OUT dan OUT NOT, sedangkan Gambar 2.15 memperlihatkan contoh implementasi kedua instruksi tersebut.



Gambar 2.12 Simbol Tangga dan Area Data Operan Instruksi OUT dan OUT NOT



Gambar 2.13 Contoh Penggunaan Instruksi OUT dan OUT NOT,

Tabel 2.10 Area Operan OUT dan OUT NOT

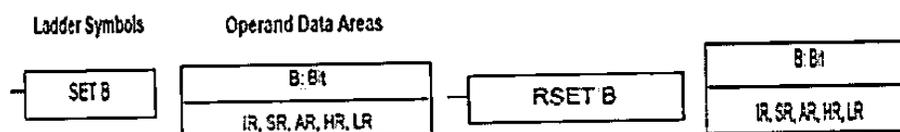
b. SET dan RESET

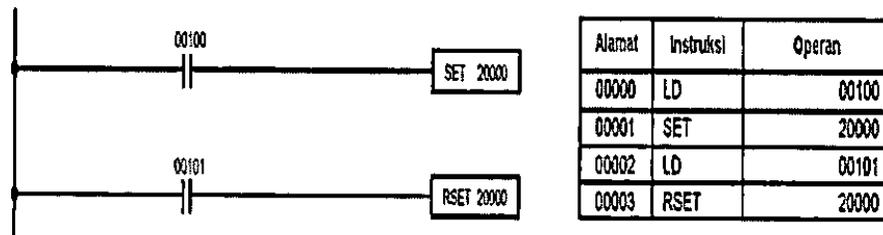
Instruksi SET dan RESET hampir sama dengan instruksi

OUTPUT 1. OUTPUT NOT hampir sama dengan instruksi SET dan RESET

ini mengubah kondisi status bit operan saat kondisi eksekusinya ON. Kedua instruksi ini tidak akan mengubah kondisi status bit jika kondisi eksekusinya OFF. Instruksi SET akan meng-ON-kan bit operan saat kondisi eksekusinya ON, tetapi tidak seperti instruksi OUT, SET tidak akan meng-OFF-kan bit operan walaupun kondisi eksekusinya sudah menjadi OFF (setelah ON). Sedangkan instruksi RESET akan meng-OFF-kan bit operan saat kondisi eksekusinya ON, tetapi tidak seperti instruksi OUT NOT, RESET tidak akan meng-ON-kan bit operan walaupun kondisi eksekusinya sudah OFF (setelah ON).

Gambar 2.16 memperlihatkan simbol tangga dan area data operan dari instruksi SET dan RESET, sedangkan Gambar 2.17 memperlihatkan contoh penggunaan kedua instruksi tersebut di atas. Pada diagram tangga tersebut, IR200.00 akan ON jika kondisi IR001.00 ON dan akan terus ON tidak tergantung kondisi IR001.00 selanjutnya, hingga kondisi IR001.01 menjadi ON sehingga me-RESET IR200.00 (menjadi OFF).





Gambar 2.15 Contoh Penggunaan Instruksi SET dan RESET, Tabel

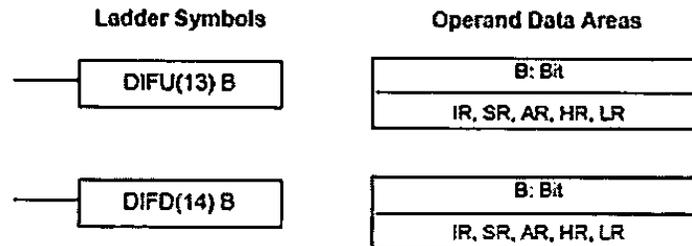
2.11 Area Operan SET dan RESET

c. DIFFERENTIATE UP (DIFU) dan DIFFERENTIATE DOWN (DIFD)

Instruksi DIFU dan DIFD digunakan untuk meng-ON-kan bit operan hanya satu siklus saja, atau dengan kata lain hanya sesaat saja. Instruksi DIFU digunakan untuk meng-ON-kan bit operan sesaat saja (hanya satu siklus) saat terjadi transisi kondisi eksekusi dari OFF ke ON. Sedangkan instruksi DIFD digunakan untuk tujuan yang sama dengan DIFU, tetapi pada saat terjadi transisi kondisi eksekusi dari ON ke OFF (kebalikan transisinya DIFU). Gambar 2.18 memperlihatkan simbol tangga dan area data operan dari instruksi DIFU dan DIFD, sedangkan Gambar 2.19 memperlihatkan diagram tangga sederhana yang menggunakan instruksi DIFU dan DIFD. Pada diagram tangga tersebut, IR200.01 akan ON untuk satu siklus (sekuat), jika terjadi transisi perubahan kondisi eksekusi pada IR000.00 dari OFF menjadi ON. Saat DIFU (13) 200.01 dikerjakan lagi untuk siklus berikutnya, IR200.01 tetap akan OFF (tidak tergantung lagi pada status IR000.00). IR200.02 akan ON untuk satu siklus saja jika terjadi transisi perubahan

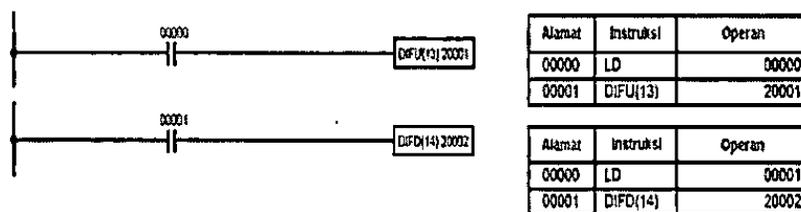
1. kondisi eksekusi pada IR000.01 dari ON menjadi OFF. Saat DIFD (14)

200.02 dikerjakan lagi untuk siklus berikutnya, IR200.02 akan tetap OFF (tidak tergantung lagi status IR000.01).



Gambar 2.16 Simbol Tangga dan Area Data Operan Dari Instruksi

DIFU dan DIFD



Gambar 2.17 Contoh Penggunaan Instruksi DIFU dan DIFD, Tabel

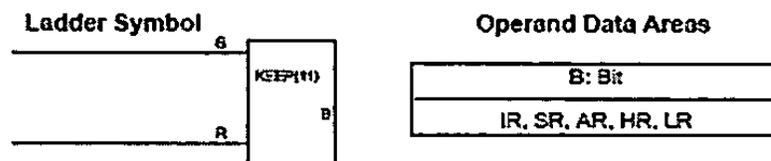
2.12 Area Operan DIFU dan DIFD

d. Instruksi KEEP

Instruksi KEEP digunakan untuk menyimpan status suatu bit operan berbasis pada dua kondisi eksekusi. Untuk keperluan ini, instruksi KEEP dihubungkan ke dua garis instruksi. Garis instruksi pertama digunakan untuk meng-ON-kan bit operan, sedangkan garis instruksi kedua digunakan untuk meng-OFF-kan bit operan, hal tersebut akan terjadi jika kondisi eksekusi pada garis instruksi yang terkait ON. Bit operan instruksi KEEP akan dijaga ON atau OFF-nya walaupun ada didalam bagian diagram yang mengandung INTER

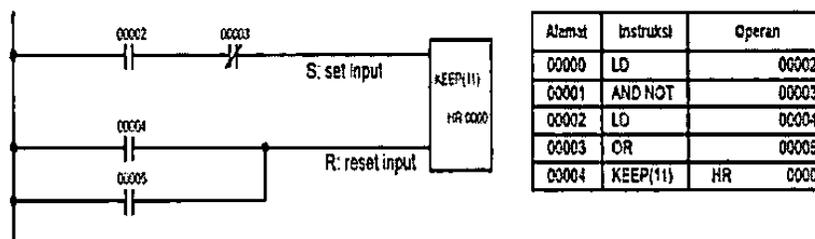
LOCK. Gambar 2.20 memperlihatkan simbol tangga dan area data

operan dari instruksi KEEP, sedangkan Gambar 2.21 memperlihatkan contoh diagram tangga yang menggunakan instruksi KEEP11. HR00.00 akan ON saat IR000.02 dalam kondisi ON dan IR000.03 dalam kondisi OFF, HR00.00 akan tetap ON hingga IR000.04 atau IR000.05 dalam kondisi ON. Karena instruksi ini memerlukan lebih dari satu garis instruksi maka garis-garis instruksi dikodekan terlebih dahulu (alamat 00000 sampai 00003) sebelum menuliskan instruksi KEEP-nya (alamat 00004).



Gambar 2.18 Simbol Tangga dan Area Data Operan Dari Instruksi

KEEP



Gambar 2.19 Contoh Penggunaan Instruksi KEEP(11), Tabel 2.13

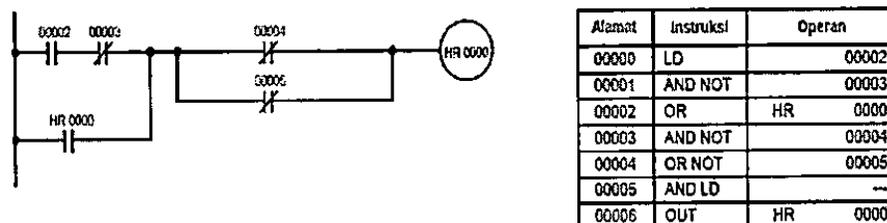
Area Operan KEEP(11)

Dalam merancang suatu program diagram tangga, untuk membuat

... lebih efektif ada dua hal yang perlu diperhatikan dan dapat

- o Bit-bit Penyimpan (Self-maintained)

Walaupun instruksi KEEP dapat digunakan untuk membuat bit-bit yang bisa menahan sendiri, kadangkala dibutuhkan membuat bit-bit yang bisa menahan sendiri dengan cara yang lain, sehingga bit-bit tersebut bisa di-OFF-kan pada bagian program yang mengandung INTERLOCK. Untuk membuat bit-bit tersebut, bit operan suatu instruksi OUT dibuat sebagai suatu kondisi untuk OUT yang sama dalam suatu bentuk penggabungan OR, sehingga bit operan OUT akan tetap ON atau OFF hingga muncul perubahan kondisi pada bit-bit yang lain. Sebagai contoh, kita bisa mengubah diagram tangga pada gambar 2.22 (yang menggunakan instruksi KEEP) dengan menghilangkan instruksi KEEP tersebut dan meletakkan bit operan OUT ke suatu garis instruksi tersendiri yang kemudian digabung dengan OR. Hasilnya ditunjukkan pada Gambar 2.23.



Gambar 2.20 Contoh Diagram Tangga Yang Menggunakan Bit Penyimpan, Tabel 2.14 Hasil Data Operan

- o Bit-bit Kerja (Work Bits) atau Relai Internal

Dalam pemrograman diagram tangga, mengkombinasikan kondisi-

seringkali sulit dilakukan. Kesulitan ini dapat ditangani dengan cara menggunakan beberapa bit untuk memicu instruksi lain secara tidak langsung. Bit-bit I/O dan bit-bit terdedikasi lainnya tidak dapat digunakan sebagai bit-bit kerja. Semua bit-bit yang ada di dalam area IR yang tidak dialokasikan untuk I/O dan beberapa bit yang tidak digunakan dalam area AR, dapat digunakan sebagai bit-bit kerja.

Bit-bit kerja biasanya sering digunakan bersama-sama dengan instruksi-instruksi OUT, OUT NOT, DIFU, DIFD, dan KEEP. Bit kerja pertama kali langsung digunakan oleh instruksi-instruksi tersebut sebagai bit operan sehingga kemudian bisa digunakan sebagai kondisi yang akan menentukan instruksi-instruksi lainnya. Bit-bit kerja dapat digunakan untuk menyederhanakan diagram tangga pada saat suatu kombinasi kondisi perlu diulang-ulang di beberapa bagian dalam diagram tangga yang bersangkutan.

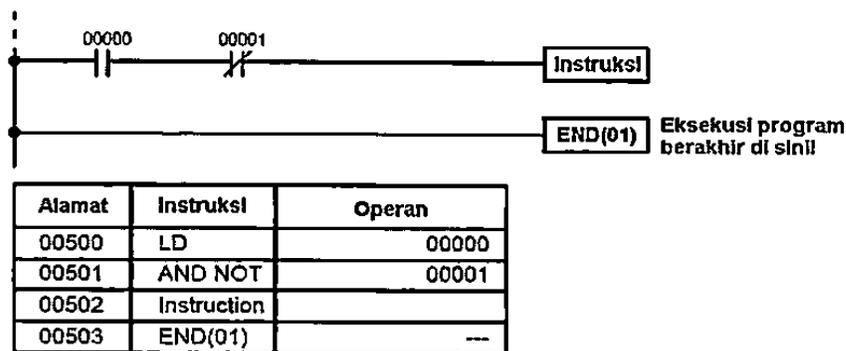
4. Instruksi END

Instruksi END merupakan instruksi terakhir yang harus dituliskan atau digambarkan dalam diagram tangga. CPU pada PLC akan mengerjakan semua instruksi dalam program dari awal (baris pertama) sampai ditemui instruksi END yang pertama, sebelum kembali lagi mengerjakan instruksi dalam program dari awal (artinya instruksi- instruksi yang ada di bawah instruksi END akan diabaikan).

Instruksi END tidak memerlukan operan dan tidak boleh diawali

dan any suatu kondisi seperti pada instruksi lainnya

Suatu diagram tangga atau program PLC harus diakhiri dengan instruksi END, jika tidak maka program tidak dijalankan sama sekali. Angka yang dituliskan pada instruksi END pada kode mnemonik merupakan kode fungsinya.



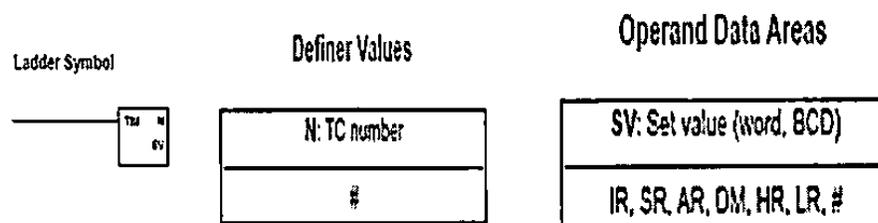
Gambar 2.21 Contoh Penggunaan Instruksi END, Tabel 2.15 Area

Operan END

5. Instruksi TIMER dan COUNTER

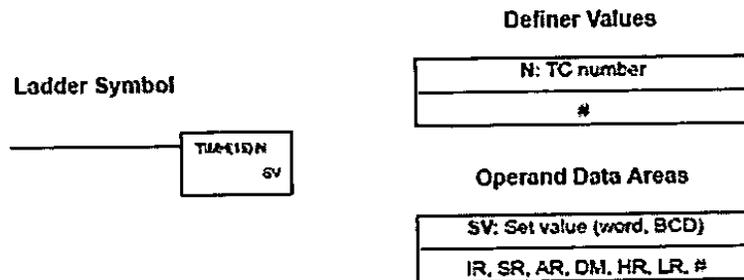
a. Instruksi TIMER (TIM)

Instruksi TIM dapat digunakan sebagai timer (pewaktu) ON-delay pada rangkaian relai. Instruksi TIM membutuhkan angka timer (N), dan nilai set (SV) antara 0000 sampai 9999 (artinya 000,0 sampai 999,9 detik).



Gambar 2.22 Simbol Tangga dan Area Data Operan Dari Instruksi

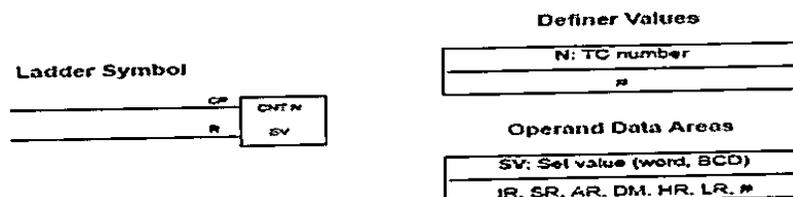
Instruksi untuk Timer Kecepatan Tinggi (High speed Timer) adalah TIMH. TIMH membutuhkan angka timer (N), dan nilai set (SV) antara 0000 sampai 9999 (artinya 00,00 sampai 99,99 detik).



Gambar 2.23 Simbol Tangga dan Area Data Operan Dari Instruksi
TIMH

b. Instruksi COUNTER (CNT)

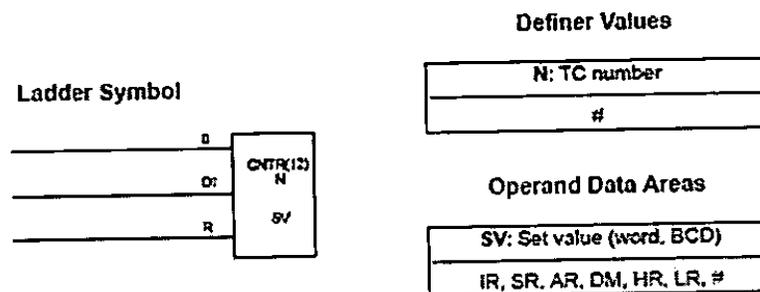
CNT yang digunakan di sini adalah counter penurunan yang di set awal. Penurunan satu hitungan setiap kali saat sebuah sinyal berubah dari OFF ke ON. Counter harus diprogram dengan input hitung, input reset, angka counter, dan nilai set (SV) Nilai set ini adalah 0000 sampai 9999.



Gambar 2.24 Simbol Tangga dan Area Data Operan Dari Instruksi
COUNTER (CNT)

Instruksi untuk Reversible Counter adalah CNTR. CNTR(12)

untuk menghitung antara nol dan SV [set value] tergantung dari dua masukan, apakah increment pada input (II) atau decrement pada input (DI). Nilai saat itu (PV) akan naik satu jika input II mendapat masukan ON kemudian OFF, dan akan turun satu jika input DI mendapat masukan ON kemudian OFF. Jika keduanya mendapatkan masukan bersama maka nilai saat itu tidak berubah. Jika nilai saat itu 0000 dan mendapat masukan turun maka nilai saat itu akan berubah ke nilai set value SV dan Completion Flag akan ON, jika nilai saat itu melebihi SV maka PV akan menjadi 0000. CNTR(12) akan reset [nilai saat itu menjadi nol] jika input reset R berubah dari OFF ke ON. PV tidak akan naik/turun selama R ON kemudian akan menghitung lagi jika R OFF. Nilai PV untuk CNTR(12) tidak akan reset pada saat program interlock atau oleh karena gangguan supply.



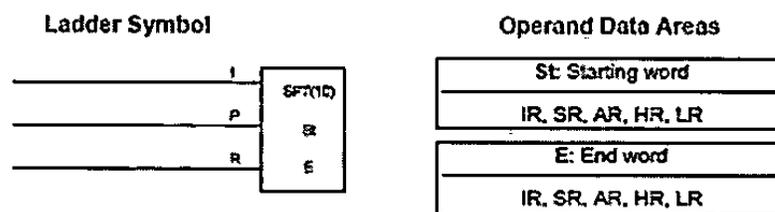
Gambar 2.25 Simbol Tangga dan Area Data Operan Dari Instruksi

CNTR

6. Instruksi SHIFT REGISTER (SFT(10))

Instruksi SFT akan menggeser 1 bit dari suatu 16 data bit pada kanal yang ditentukan. Meskipun instruksi ini menggeser data dalam

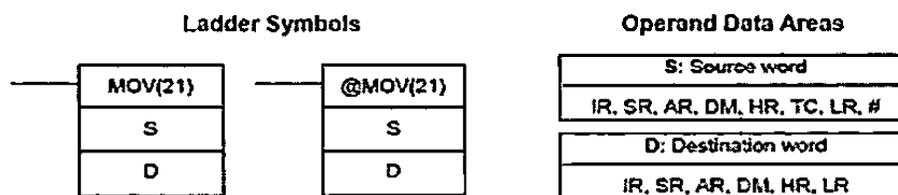
spesifikasikan sebagai data. Jika input reset diaktifkan, 16 bit tersebut semuanya akan direset secara bersamaan. Apabila Holding Relay Area digunakan, data tersebut akan ditahan sewaktu terjadi gangguan daya.



Gambar 2.26 Simbol tangga dan area data operan dari instruksi SFT

7. Instruksi Pemindahan Data : MOVE (MOV(21))

Instruksi MOV berfungsi untuk mengkopi isi dari S (source word) ke D (destination word).



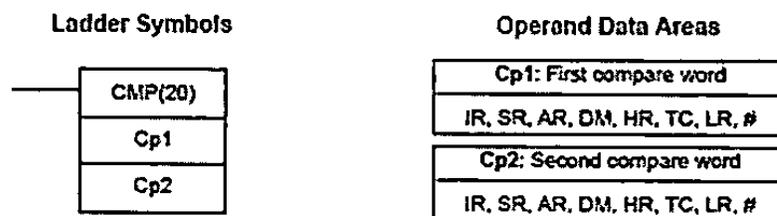
Gambar 2.27 Simbol Tangga dan Area Data Operan Dari Instruksi

MOV

8. Instruksi Perbandingan: COMPARE (CMP(20))

Instruksi CMP berfungsi untuk membandingkan Cp1 dan Cp2

1. Hasilnya disimpan di flag CF (Carry Then), EQ (Equal), dan LT



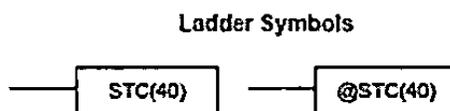
Gambar 2.28 Simbol Tangga dan Area Data Operan Dari Instruksi

CMP

9. Instruksi Perhitungan BCD

a. Set Carry – STC(40)

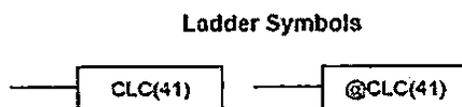
Instruksi STC(40) menyebabkan ON pada CY (SR 25504).



Gambar 2.29 Simbol tangga dari instruksi STC

b. Clear Carry – CLC(41)

Instruksi CLC(41) menyebabkan OFF pada CY (SR 25504).

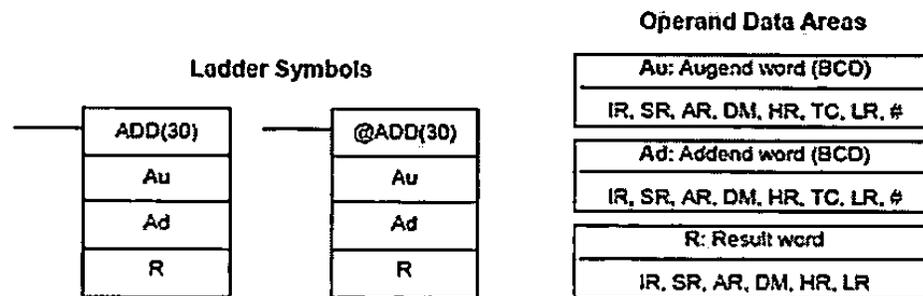


Gambar 2.30 Simbol Tangga Dari Instruksi CLC

c. BCD ADD – ADD(30)

Instruksi ADD(30) menjumlahkan isi Au, Ad dan CY dan

hasilnya disimpan di B. CY diatur jika hasilnya lebih dari 0000

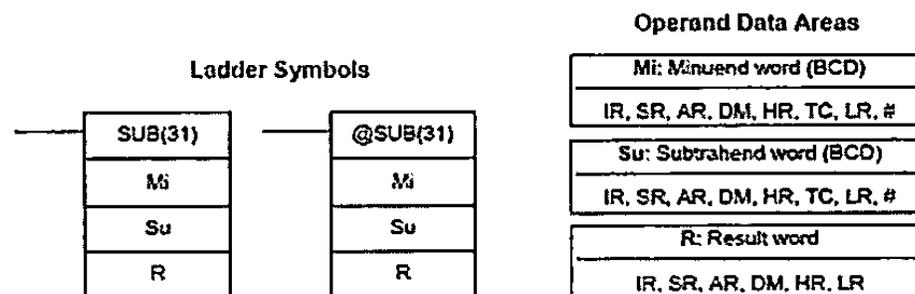


Gambar 2.31 Simbol Tangga dan Area Data Operan Dari Instruksi

ADD

d. BCD SUBTRACT – SUB(31)

Instruksi SUB(31) mengurangi isi dari Su dan CY dari Mi dan hasilnya ditempatkan di R. Jika hasilnya negatif maka CY di set dan komplemen 10 dari hasil sebenarnya disimpan di R. Untuk merubah komplemen 10 ke hasil sebenarnya kurangkan isi R dengan nol.

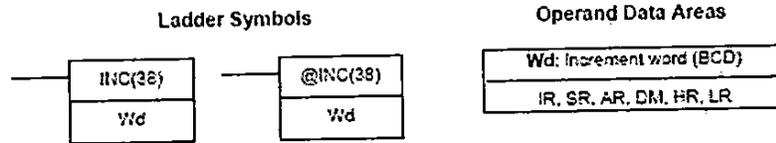


Gambar 2.32 Simbol Tangga dan Area Data Operan Dari Instruksi

SUB

10. Instruksi Increment/Decrement

a. BCD Increment – INC(38)

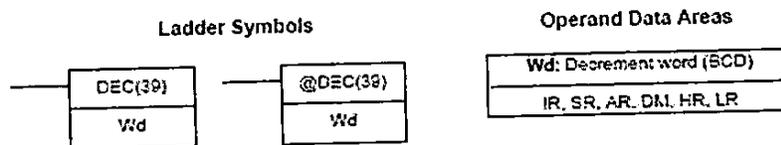


Gambar 2.33 Simbol Tangga dan Area Data Operan Dari Instruksi

DEC(38)

b. BCD Decrement – DEC(39)

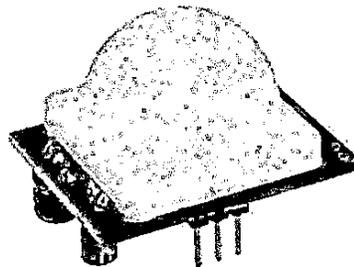
Instruksi DEC bertujuan menurunkan satu isi Wd, tanpa berakibat pada CY. DEC(39) cara kerjanya sama dengan INC(38) tetapi pada DEC(39) turunkan satu.



Gambar 2.34 Simbol Tangga dan Area Data Operan Dari Instruksi

DEC(39)

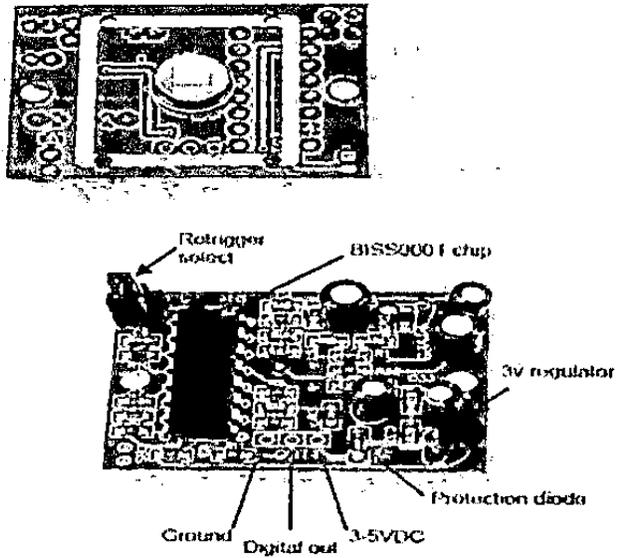
2.2.3 Sensor PIR (*Passive Infra Red*)



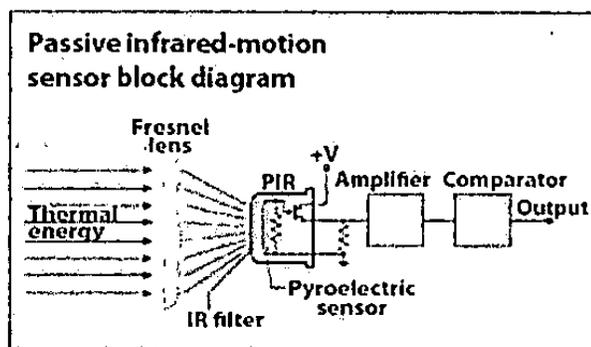
Gambar 2.35 Sensor PIR

Sensor PIR (*Passive Infra Red*) adalah sensor yang digunakan untuk mendeteksi adanya pancaran sinar infra merah. Sensor PIR bersifat pasif, artinya sensor ini tidak memancarkan sinar infra merah tetapi hanya menerima radiasi sinar infra merah dari luar. Sensor ini biasanya digunakan dalam perancangan detektor gerakan berbasis PIR. Karena semua benda memancarkan energi radiasi, sebuah gerakan akan terdeteksi ketika sumber infra merah dengan suhu tertentu (misal : manusia) melewati sumber infra merah yang lain dengan suhu yang berbeda (misal : dinding), maka sensor akan membandingkan pancaran infra merah yang diterima setiap satuan waktu, sehingga jika ada pergerakan maka akan terjadi perubahan pembacaan pada sensor.

Sensor PIR terdiri dari beberapa bagian yaitu :



Gambar 2.36 Bagian-bagian Sensor PIR



Gambar 2.37 Block Diagram Sensor PIR

2.2.3.1 Prinsip Kerja Pembacaan Sensor PIR

Pancaran infra merah masuk melalui lensa Fresnel dan mengenai sensor pyroelektrik, karena sinar infra merah mengandung energi panas maka sensor pyroelektrik akan menghasilkan arus listrik. Sensor pyroelektrik terbuat dari bahan *galium nitrida* (GaN), *cesium nitrat* (CsNo₃) dan *litium tantalate* (LiTaO₃). Arus listrik inilah yang akan menimbulkan tegangan dan dibaca secara analog oleh sensor. Kemudian sinyal ini akan dikuatkan oleh penguat dan dibandingkan oleh komparator dengan tegangan referensi tertentu (keluaran berupa sinyal 1-bit). Jadi sensor PIR hanya akan mengeluarkan logika 0 dan 1, 0 saat sensor tidak mendeteksi adanya pancaran infra merah dan 1 saat sensor mendeteksi infra merah. Sensor PIR didesain dan dirancang hanya mendeteksi pancaran infra merah dengan panjang gelombang 8-14 mikrometer. Diluar panjang gelombang tersebut sensor tidak akan mendeteksinya. Untuk manusia sendiri memiliki suhu badan yang dapat menghasilkan pancaran infra merah dengan panjang gelombang antara 9-10 mikrometer (nilai standar 9,4 mikrometer), panjang gelombang tersebut dapat terdeteksi oleh sensor PIR. (*Secara umum sensor PIR memang dirancang untuk mendeteksi manusia*)

2.2.3.2 Jarak pancar sensor PIR

Sensor PIR memiliki jangkauan jarak yang bervariasi, tergantung karakteristik sensor. Proses penginderaan sensor PIR dapat digambarkan sebagai berikut:

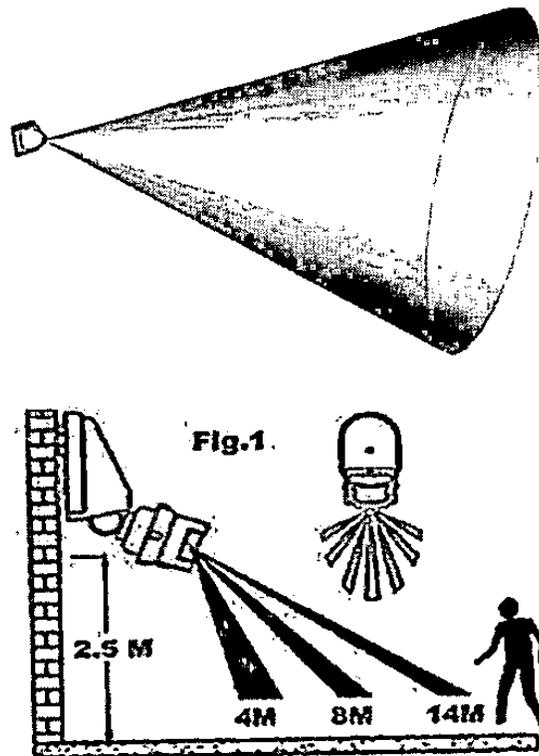


Fig. 1. Range Sensor PIR

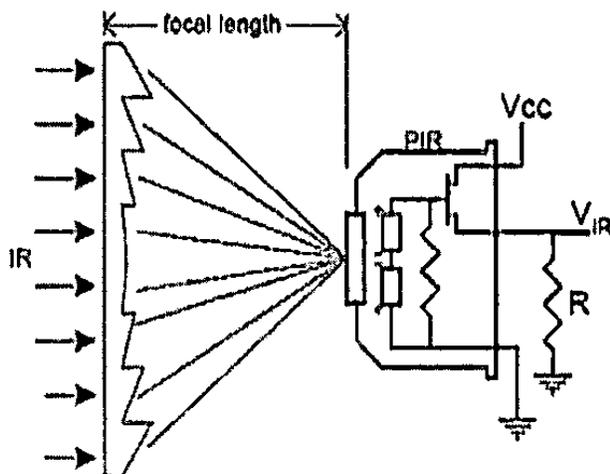
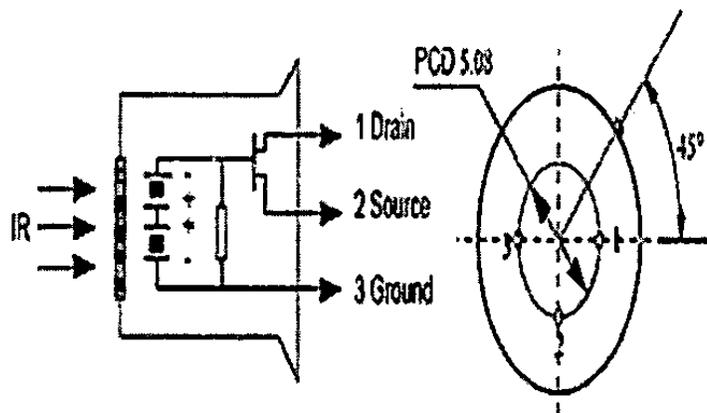
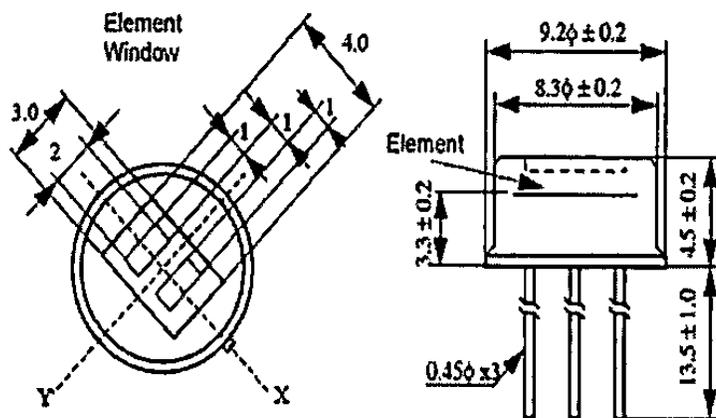
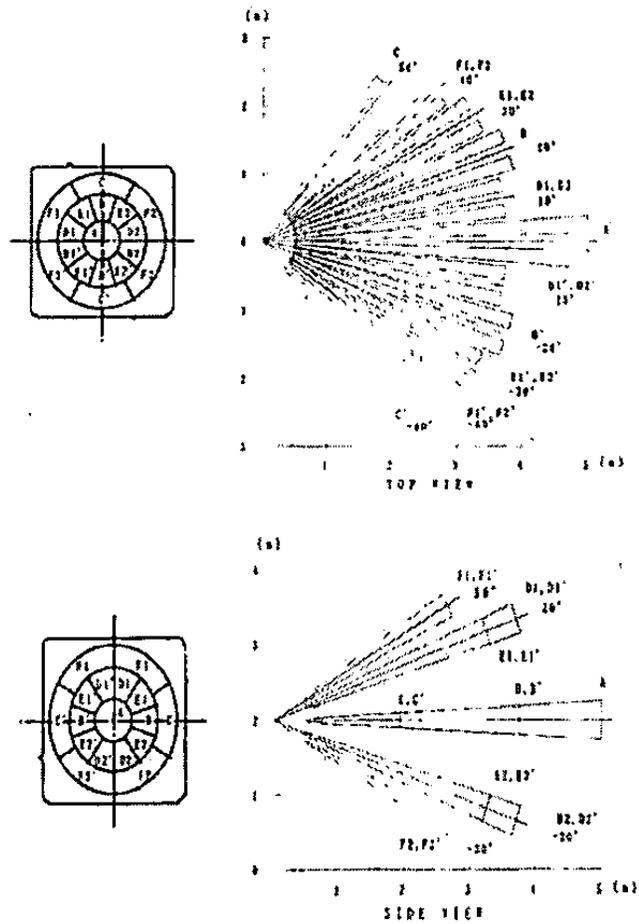


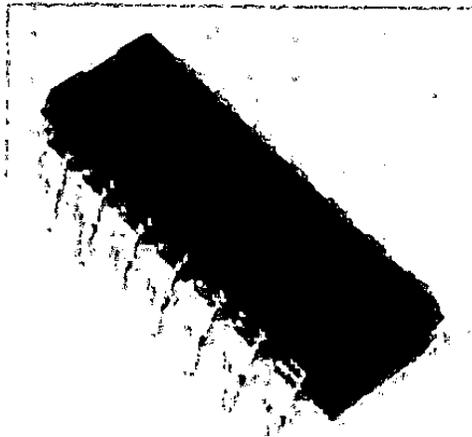
Figure 2-20 Element Sensor PIR



Gambar 2.40 Rangkaian-rangkaian Jarak Deteksi Sensor PIR

Pada umumnya sensor PIR memiliki jangkauan pembacaan efektif hingga 5 meter, dan sensor ini sangat efektif digunakan sebagai human

2.2.4 Integrated Circuit (IC)



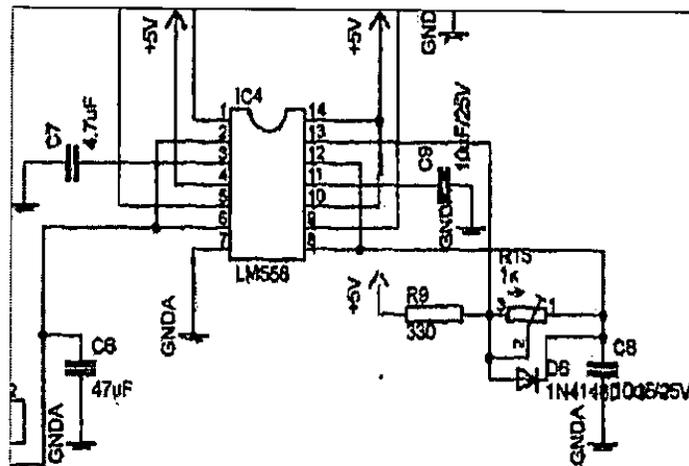
Gambar 2.41 Jenis-jenis IC

Integrated Circuit (IC) adalah suatu komponen elektronika yang dibuat dari bahan semi konduktor, dimana IC merupakan gabungan dari beberapa komponen seperti Resistor, Kapasitor, Dioda dan Transistor yang telah terintegrasi menjadi sebuah rangkaian berbentuk chip kecil dan mempunyai kaki banyak, IC digunakan untuk beberapa keperluan pembuatan peralatan elektronika agar mudah dirangkai menjadi peralatan yang berukuran relatif kecil dan efisien.

Dalam perancangan alat ini ada tiga jenis IC yang digunakan yaitu :

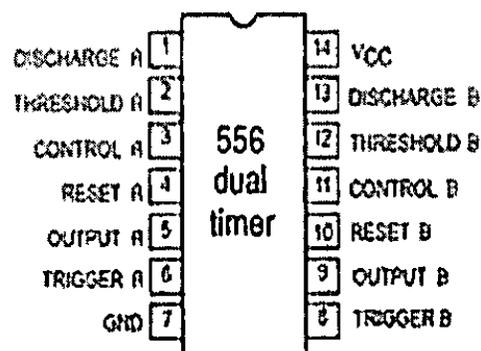
2.2.4.1 IC LM556

Dalam rangkaian ini digunakan beberapa jenis IC (Integrated Circuit) diantaranya yaitu IC LM556 yang berfungsi sebagai pembangkit PWM dan pengatur kecepatan pompa



Gambar 2.42 Rangkaian IC LM324

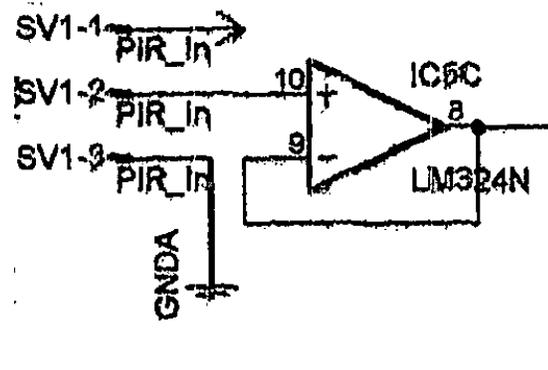
IC LM556 adalah kontroler sangat stabil yang mampu menghasilkan waktu yang akurat penundaan atau osilasi. IC LM556 ini adalah merupakan dual dari IC LM555. Timing disediakan oleh resistor eksternal dan kapasitor untuk setiap fungsi waktu. Kedua timer beroperasi secara independen satu sama lain hanya berbagi V dan CC pada letaknya. Rangkaian dapat dipicu dan reset pada jatuh gelombang. Struktur Output dapat tenggelam atau sumber 200mA. Pemakaian IC ini di tujukan agar aliran air dan sabun stabil, sehingga tidak terjadi lonjakan aliran air dan sabun yang melalui selang begitu kencang atau deras.



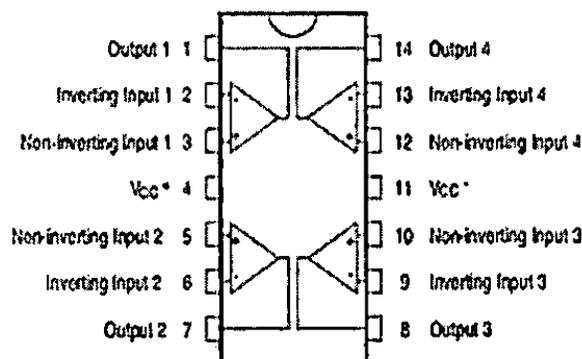
Gambar 2.43 Pin-out diagram IC LM556

2.2.4.2 IC LM324N OP-Amp

Dalam rangkaian ini juga di gunakan IC LM324N OP-Amp yang berfungsi untuk signal conditioning sensor. Perangkat ini terdiri dari empat independen frekuensi gain tinggi penguat operasional kompensasi yang dirancang khusus untuk beroperasi dari pasokan tunggal atas berbagai tegangan. Operasi dari pasokan split juga mungkin ketika perbedaan antara dua pasokan adalah 3 V sampai 30V. Aplikasi termasuk amplifier transduser, amplifikasi dc blok, dan semua rangkaian penguat operasional konvensional yang sekarang dapat lebih mudah di implementasikan dalam sistem tunggal pasokan tegangan. Berikut adalah gambar rangkaiannya.



Gambar 2.44 Rangkaian IC LM 324N Op-Amp



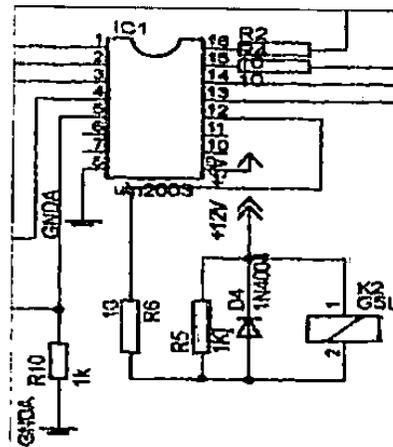
Gambar 2.45 Pin-out diagram IC LM324N

2.2.4.3 IC ULN2003

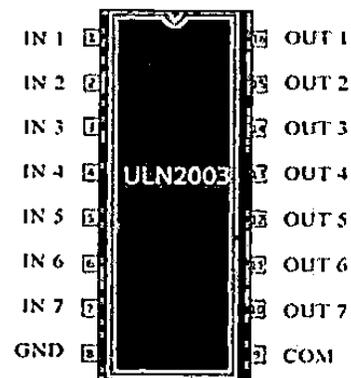
Kemudian pada rangkaian alat mekanik juga di tambahkan komponen IC ULN2003 yang berfungsi untuk penguat arus relay. ULN2003 adalah tegangan tinggi monolitik dan dapat di sebut juga arus tinggi Darlington transistor array. IC ini terdiri dari tujuh Darlington NPN pasangan yang memiliki output tegangan tinggi dengan bagian umumnya adalah katoda dioda dan penjepit untuk pengalih beban induktif. Jumlah nilai dari pasangan Darlington tunggal adalah 500mA. Aplikasi pada IC ini menyertakan driver relay, driver palu, lampu driver, driver display (LED gas discharge), driverline, dan buffer logika. ULN2003 memiliki $2.7k\Omega$ (kilo-ohm) seri resistor dasar untuk masing-masing pasangan Darlington untuk operasi langsung dengan TTL atau CMOS 5V DIP-16 perangkat. Dan memiliki fitur-fitur sebagai berikut :

- Tingkatan arus kolektor (output Single) 500mA
- Tinggi tegangan output: 50V
- Input compatibale dengan berbagai pesof logika
- Aplikasi driver Relay

Dipilih adalah rangkaian...

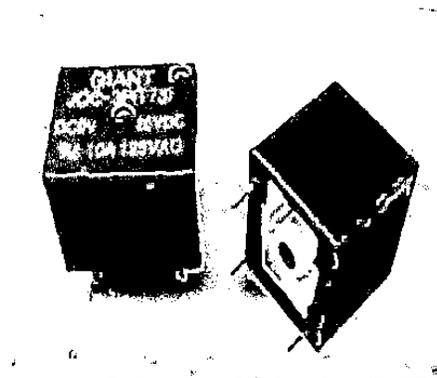


Gambar 2.46 Rangkaian IC ULN 2003



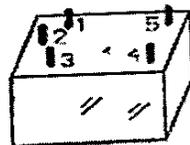
Gambar 2.47 Pin-out diagram ULN2003

2.2.5 Geganti / Relay

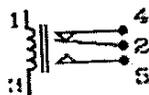


Gambar 2.48 Jenis-jenis Geganti / Relay

Geganti/relay adalah merupakan peranti pasif yang berfungsi sebagai suis elektronik. Geganti/relay juga merupakan suis yang digerakan secara elektronik. Bagian elektronik yang menggunakan arus kecil dapat melalui bagian yang menggunakan arus yang tinggi dengan menggunakan relay sebagai suis. Relay dapat melewati bagian AC dan DC. Pada dasarnya relay memiliki 5 kaki yang biasanya di gunakan dalam sebuah rangkaian mekanik. Duah buah kaki biasanya di hubungkan ke bagian elektronik karena pada bagian dalamnya merupakan kumparan elektromagnet untuk menggerakan suis. Kaki tengah adalah common, sisanya adalah NC dan NO. Dimana NC adalah *Normally Closed* sedangkan NO adalah *Normally Open*. Untuk tujuan menghidupkan kedua bagian maka di gunakan C dan NO. Sedangkan untuk mematikan kedua bagian maka C dan NC yang di gunakan.

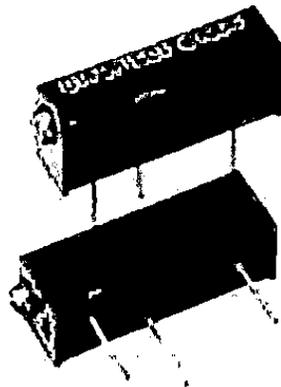


1,3 = RELAY COIL
 2 = COMMON
 4 = NORMAL. CLOSE
 5 = NORMAL OPEN



Gambar 2.49 Diagram Geganti / Relay

2.2.6 Trimmer Potentiometer



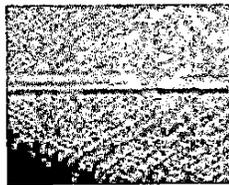
Gambar 2.50 Trimmer Potentiometer 3006P

Trimmer potentiometer jenis 3006P yang akan digunakan adalah potentiometer yang cara mengubah nilai tahanannya dengan cara mentrim dengan menggunakan obeng trim. Biasanya digunakan untuk mengatur besaran arus pada rangkaian oscilator, rangkaian driver, atau pada penyetelan keseimbangan putih (*white balance*).

2.2.7 Selang Air dan Sabun

Selang air dan sabun digunakan sebagai penghubung dari lubang penampungan air dan sabun (seperti yang sudah dibahas di atas). Digunakan selang air ini, karena bahannya yang lentur dan dapat di bentuk sesuai keinginan dan ukuran panjang selang sesuai dengan kemampuan atau kekuatan pompa untuk mengalirkan air dan sabun. Ukuran panjang selang ini dibutuhkan karena kemampuan kekuatan pompa air dan sabun dapat sesuai dengan yang diinginkan. Hal ini juga

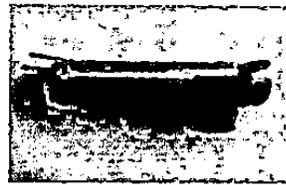
terjadi karena kemampuan atau kekuatan alat pompa memiliki ketebatasan sesuai dengan *power supply*. Supaya tidak terjadi terhambatnya aliran air dan sabun menuju ke kran. Dalam hal ini digunakan selang transparan yang lentur untuk mempermudah mengatur kemana air dan sabun akan mengalir sesuai letak yang diinginkan. Dan salah satu alasan yang penting karena selang ini berwarna transparan, adalah untuk mempermudah proses pengamatan aliran air dan sabun saat melakukan percobaan yang cukup sering dilakukan.



Gambar 2.51 Selang Transparan

2.2.8 Pompa Air dan Sabun

Alat kran air dan sabun ini membutuhkan sarana untuk mengalirkan air dan sabun, air dan sabun dialirkan masuk menuju ke kran, air dan sabun tersebut diambil dari wadah tempat penampung air dan sabun. Dari hasil percobaan yang sering dilakukan, dibutuhkan air dan sabun yang cukup di tabung penampung. Hal ini disebabkan karena apabila jumlah air dan sabun tidak mencukupi maka pompa tidak dapat mengalirkan air dan sabun. Pompa yang digunakan adalah pompa wiper untuk kendaraan roda 4 dengan kekuatan pompa diberikan (Input) power supply 12v, 1A.



Gambar 2.52 Domo Wings Air dan Sekur