

**RANCANG BANGUN SISTEM PEMBERHENTIAN
KERETA LISTRIK, BUKA – TUTUP PINTU KERETA
DAN STASIUN MENGGUNAKAN
PLC LG MASTER-K 120S DAN BASIC STAMP**

SKRIPSI

Oleh

M. ADY SETIAWAN

06 06 04 269 6



**DEPARTEMEN TEKNIK ELEKTRO
FAKULTAS TEKNIK UNIVERSITAS INDONESIA
GENAP 2007/2008**

**RANCANG BANGUN SISTEM PEMBERHENTIAN
KERETA LISTRIK, BUKA – TUTUP PINTU KERETA
DAN STASIUN MENGGUNAKAN
PLC LG MASTER-K 120S DAN BASIC STAMP**

SKRIPSI

Oleh

M. ADY SETIAWAN

06 06 04 269 6



**SKRIPSI INI DIAJUKAN UNTUK MELENGKAPI SEBAGIAN
PERSYARATAN MENJADI SARJANA TEKNIK**

**DEPARTEMEN TEKNIK ELEKTRO
FAKULTAS TEKNIK UNIVERSITAS INDONESIA
GENAP 2007/2008**

PERNYATAAN KEASLIAN SKRIPSI

Saya menyatakan dengan sesungguhnya bahwa skripsi dengan judul:

**RANCANG BANGUN SISTEM PEMBERHENTIAN KERETA LISTRIK,
BUKA – TUTUP PINTU KERETA DAN STASIUN MENGGUNAKAN
PLC LG MASTER-K 120S DAN BASIC STAMP**

yang dibuat untuk melengkapi sebagian persyaratan menjadi Sarjana Teknik pada Program Pendidikan Sarjana Teknik Ekstensi Departemen Teknik Elektro Fakultas Teknik Universitas Indonesia, sejauh yang saya ketahui bukan merupakan tiruan atau duplikasi dari skripsi yang sudah dipublikasikan dan atau pernah dipakai untuk mendapatkan gelar kesarjanaan di lingkungan Universitas Indonesia maupun di Perguruan Tinggi atau instansi manapun, kecuali bagian yang sumber informasinya dicantumkan sebagaimana mestinya.

Depok, 16 Juli 2008

M. Ady Setiawan

NPM. 06 06 04 269 6

PENGESAHAN

Skripsi dengan judul:

**RANCANG BANGUN SISTEM PEMBERHENTIAN KERETA LISTRIK,
BUKA – TUTUP PINTU KERETA DAN STASIUN MENGGUNAKAN
PLC LG MASTER-K 120S DAN BASIC STAMP**

dibuat untuk melengkapi sebagian persyaratan menjadi Sarjana Teknik pada Program Pendidikan Sarjana Ekstensi Departemen Teknik Elektro Fakultas Teknik Universitas Indonesia. Skripsi ini telah disetujui untuk diujikan pada sidang ujian skripsi pada tanggal 09 Juli 2008 dan dinyatakan memenuhi syarat/sah sebagai Skripsi pada Departemen Teknik Elektro Fakultas Teknik Universitas Indonesia.

Depok, 16 Juli 2008

Dosen Pembimbing,

Budi Sudiarto, S.T, M.T

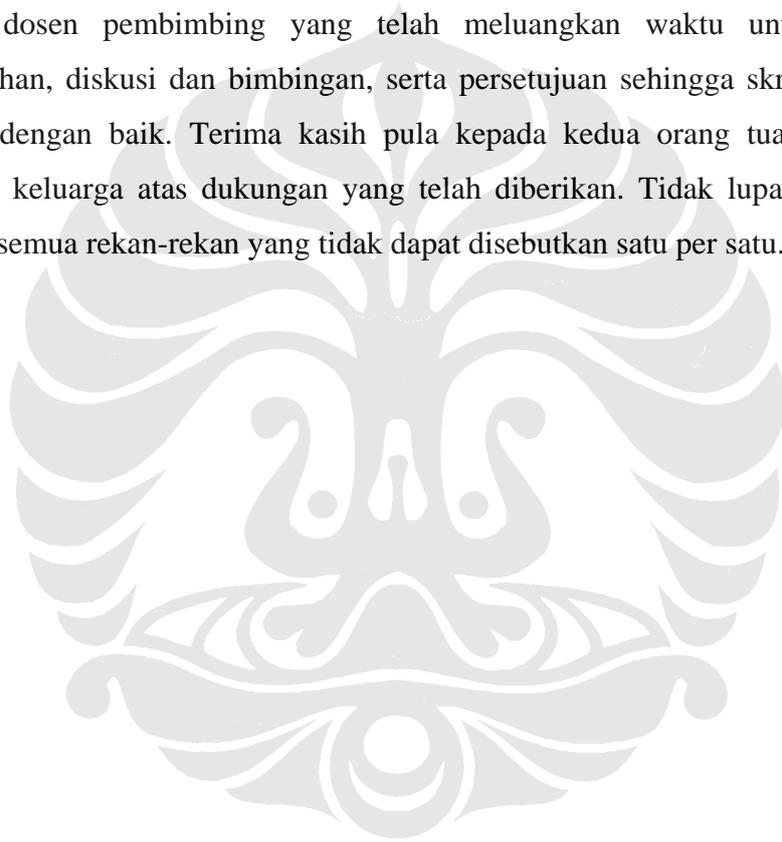
NIP. 040 705 0181

UCAPAN TERIMA KASIH

Puji dan syukur kehadirat **ALLAH SWT** yang telah melimpahkan rahmat dan hidayah-Nya. Tak lupa penulis juga mengucapkan terima kasih kepada:

Budi Sudiarto, S.T, M.T

selaku dosen pembimbing yang telah meluangkan waktu untuk memberi pengarahan, diskusi dan bimbingan, serta persetujuan sehingga skripsi ini dapat selesai dengan baik. Terima kasih pula kepada kedua orang tua dan seluruh anggota keluarga atas dukungan yang telah diberikan. Tidak lupa terima kasih kepada semua rekan-rekan yang tidak dapat disebutkan satu per satu.



ABSTRAK

M. ADY SETIAWAN NPM 06 06 04 269 6 Departemen Teknik Elektro	Dosen Pembimbing Budi Sudiarto, S.T, M.T
RANCANG BANGUN SISTEM PEMBERHENTIAN KERETA LISTRIK, BUKA – TUTUP PINTU KERETA DAN STASIUN MENGGUNAKAN PLC LG MASTER-K 120S DAN BASIC STAMP	
ABSTRAK <p>Kereta listrik umumnya membutuhkan masinis dalam menjalankan kereta dan membuka pintu kereta. Pada skripsi ini direalisasikan miniatur kereta listrik dapat berjalan secara otomatis serta buka-tutup pintu kereta dan stasiun secara otomatis menggunakan <i>Programmable Logic Control</i> (PLC) dan Basic Stamp. Dalam merealisasikan miniatur kereta listrik ini, terdapat 8 sub sistem yaitu Sensor dan Rangkaian Komparator, <i>Programmable Logic Control</i> (PLC), Rangkaian Kontrol dan <i>Driver</i> PWM, <i>Basic Stamp</i>, <i>Driver Motor Stepper</i>, <i>Driver Limit Switch</i>, LCD, dan <i>Power Supply</i>.</p> <p>Hasil yang dicapai dalam merealisasikan miniatur kereta listrik adalah mendapatkan suatu pemrograman ladder pada PLC yang dapat berinteraksi dengan Pbasic pada mikrokontroler Basic Stamp serta mendapatkan hasil pengukuran – pengukuran pada sub-sub sistem yang direalisasikan sehingga pada tahap selanjutnya dapat membantu kesempurnaan dalam pembuatan sistem transportasi menggunakan kereta listrik yang akan diaplikasikan di dalam kampus Universitas Indonesia. Diharapkan dengan adanya skripsi ini dapat memudahkan dalam merealisasikan ke kondisi nyata yang akan diperlihatkan dalam realita kehidupan masyarakat kampus Universitas Indonesia.</p>	
Kata kunci : Kereta Listrik, <i>Programmable Logic Controller</i> , <i>Basic Stamp</i> , <i>Pbasic</i> , <i>Ladder</i> , Miniatur	

ABSTRACT

M. ADY SETIAWAN NPM 06 06 04 269 6 Departemen Teknik Elektro	Dosen Pembimbing Budi Sudiarto, S.T, M.T
TRAMCAR CESSATION SYSTEM SIMILAR DESIGN, OPEN - CLOSE DOOR OF TRAIN AND STATION APPLIES PLC LG MASTER-K 120S AND BASIC STAMP	
ABSTRACT <p>Tramcar in general requires engine-driver in implementing tramcar and opens door of tramcar. At this final task realized tramcar miniature which can run automatically and open - close door of car and halting point automatically applies Programmable Logic Control (PLC) and Basic Stamp. In realizing this tramcar miniature, there is 8 system sub that is Sensor and Comparator Circuit, Programmable Logic Control (PLC), Control Circuit and Driver PWM, Basic Stamp, Driver Motor Stepper, Driver Limit Switch, LCD, and Power Supply.</p> <p>Result reached in realizing tramcar miniature is get a programming of ladder at PLC which interaction can with Pbasic at microcontroller Basic Stamp and gets result of gauging at system subs realized so that at phase hereinafter can assist perfection in making of transportation system applies tramcar which will be application in campus University of Indonesia. Expected with existence of this final project can facilitate in realizing to condition of reality which will be showed in reality life of campus public University of Indonesia</p>	
Keywords : Tramcar, <i>Programmable Logic Controller, Basic Stamp, Pbasic, Ladder, Miniature</i>	

DAFTAR ISI

	Halaman
PERNYATAAN KEASLIAN SKRIPSI	ii
PENGESAHAN	iii
UCAPAN TERIMA KASIH	iv
ABSTRAK	v
ABSTRACT	vi
DAFTAR ISI	vii
DAFTAR GAMBAR	x
DAFTAR TABEL	xii
DAFTAR SINGKATAN	xiii
DAFTAR ISTILAH	xiv
BAB 1 PENDAHULUAN	1
1.1 LATAR BELAKANG	1
1.2 PERUMUSAN MASALAH	2
1.3 TUJUAN PENULISAN	2
1.4 BATASAN MASALAH	2
1.5 METODOLOGI PENELITIAN	3
1.6 SISTEMATIKA PENULISAN	3
BAB 2 PLC, BASIC STAMP, PWM, MOTOR STEPPER, DAN MOTOR DC	5
2.1 <i>MIKROKONTROLER BASIC STAMP</i>	5
2.2 KOMPARATOR	7
2.3 <i>PROGRAMMABLE LOGIC CONTROL</i>	8
2.3.1 Bagian – bagian PLC	10
2.3.2 Prinsip Kerja	12
2.3.3 Metode Memprogram PLC	13
2.3.4 Pemilihan PLC	14
2.4 PEMROGRAMAN PLC	14
2.4.1 Instruksi PLC Master-K120S	14

2.4.2 Metode Y-Map	16
2.5 <i>MOTOR STEPPER</i>	22
2.6. <i>PULSE WIDTH MODULATION</i>	25
2.7 MOTOR DC	26
BAB 3 RANCANG BANGUN	30
3.1 SISTEM SECARA UMUM	30
3.1.1 Cara Kerja Sistem	31
3.1.2 Spesifikasi Sistem	32
3.2 RANCANG BANGUN PERANGKAT KERAS	33
3.2.1 Sensor dan Rangkaian Komparator	33
3.2.2 <i>Programmable Logic Control</i>	35
3.2.3 Rangkaian Kontrol dan <i>Driver PWM</i>	36
3.2.4 Mikrokontroler <i>Basic Stamp</i>	38
3.2.5 <i>Driver Motor Stepper</i>	41
3.2.6 <i>Driver Limit Switch</i>	43
3.2.7 LCD	44
3.2.8 <i>Power Supply</i>	46
3.2.9 Miniatur Kereta Listrik	47
3.2.10 Pintu Kereta dan Stasiun	49
3.3 RANCANG BANGUN PERANGKAT LUNAK	50
3.3.1 Spesifikasi Perangkat Lunak	50
3.3.2 Diagram Alir Perangkat Lunak	50
BAB 4 UJI COBA DAN ANALISIS	55
4.1 UJI COBA DAN ANALISIS PERANGKAT KERAS	55
4.1.1 Uji Coba Sensor dan Rangkaian Komparator	55
4.1.2 Uji Coba Rangkaian Kontrol dan Driver PWM	57
4.1.3 Uji Coba Rangkaian <i>Driver Motor Stepper</i>	58
4.1.4 Uji Coba <i>Driver Limit Switch</i>	60
4.2 UJI COBA SISTEM	61
4.2.1 Kondisi Kereta Berada Pada Stasiun 1, Pintu Kereta dan Stasiun Tertutup	61

4.2.2 Kondisi Kereta Berada Pada Stasiun 1, Pintu Kereta dan Stasiun Terbuka	62
4.2.3 Kondisi Kereta Berada Pada Stasiun 2, Pintu Kereta dan Stasiun Tertutup	62
4.2.4 Kondisi Kereta Berada Pada Stasiun 2, Pintu Kereta dan Stasiun Terbuka	63
4.2.5 Kondisi Kereta Berada Pada Stasiun 1, Pintu Kereta Terbuka dan Stasiun Tertutup	63
4.2.6 Kondisi Kereta Berada Pada Stasiun 2, Pintu Kereta Terbuka dan Stasiun Tertutup	64
4.3 SKENARIO PADA SISTEM	64
4.3.1 Skenario Pada Saat Pemberhentian	65
4.3.2 Skenario Pada Saat Pintu Kereta dan Stasiun Terbuka Bersamaan	66
4.3.2 Skenario Pada Saat Koneksi Antara Pintu Kereta Dan Pengendali	67
BAB 5 KESIMPULAN	68
DAFTAR ACUAN	69
DAFTAR PUSTAKA	70

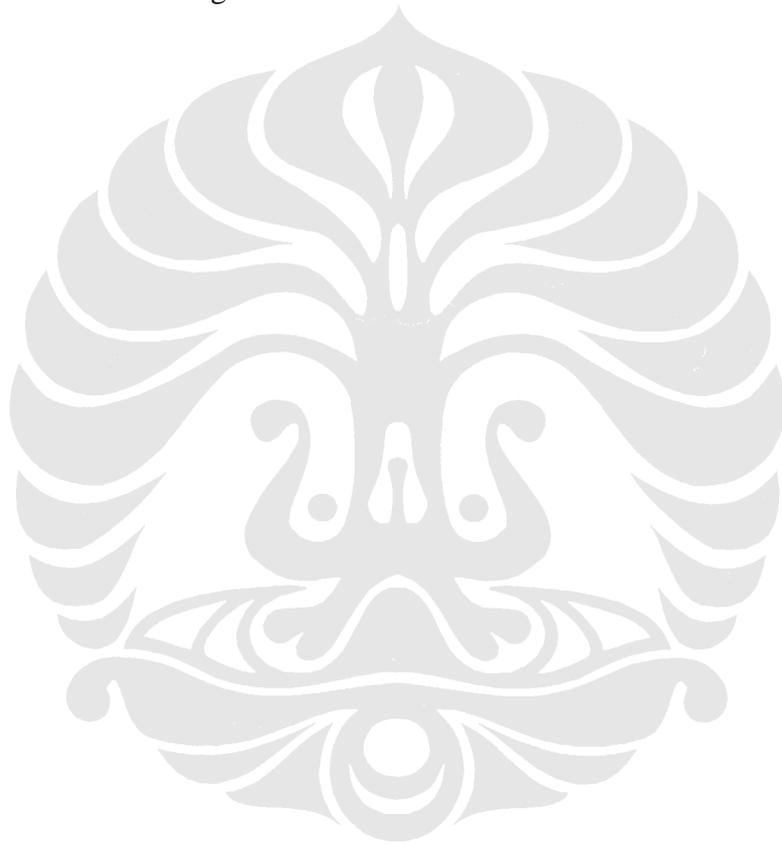
DAFTAR GAMBAR

	Halaman	
Gambar 2.1	Mikrokontroler	5
Gambar 2.2	Komparator	8
Gambar 2.3	Contoh bentuk fisik PLC LG MASTER-K 120S 30I/O	11
Gambar 2.4	Blok diagram dasar sistem kontrol PLC	12
Gambar 2.5	Blok diagram dasar sistem kontrol PLC	12
Gambar 2.6	Bagian-bagian program untuk <i>timer</i>	15
Gambar 2.7	<i>Timer</i> TON dengan <i>setting value</i> 10 detik	15
Gambar 2.8	<i>Self-holding relay</i> pada koil M0000	16
Gambar 2.9	Program utama kontrol proses metode Y-map SHR	19
Gambar 2.10	<i>Output</i> tipe (a)	20
Gambar 2.11	<i>Output</i> tipe (b)	20
Gambar 2.12	<i>Output</i> tipe (c)	21
Gambar 2.13	<i>Output</i> tipe (d)	21
Gambar 2.14	Ilustrasi sebuah kompas dengan elektromagnet	22
Gambar 2.15	Ilustrasi <i>motor stepper</i> dengan jarum kompas dengan elektromagnet	23
Gambar 2.16	Half Step Mode	24
Gambar 2.17	Bentuk fisik motor stepper disk drive 1,2"	24
Gambar 2.18	Prinsip Kerja <i>Rotary Encoder</i>	25
Gambar 2.19	Konstruksi dasar Motor DC	27
Gambar 2.20	Kurva Kecepatan Torsi pada Motor DC Magnet Permanen	28
Gambar 3.1	Blok Diagram Sistem	30
Gambar 3.2	Sensor dan Rangkaian Komparator	34
Gambar 3.3	Realisasi Rangkaian Sensor dan Komparator	34
Gambar 3.4	Bentuk fisik PLC LG	35
Gambar 3.5	Realisasi bentuk fisik PLC LG Master-K 120S	36

Gambar 3.6	Rangkaian Kontrol	37
Gambar 3.7	Realisasi Rangkaian Kontrol dan <i>Driver</i> PWM	38
Gambar 3.8	Alokasi Pin J1 Port MAINIO	39
Gambar 3.9	Alokasi Pin J2 Port AUXIO	39
Gambar 3.10	Realisasi Mikrokontroler <i>Basic stamp</i>	40
Gambar 3.11	Realisasi dari Rangkaian Pembagi Pin	40
Gambar 3.12	Rangkaian <i>Driver Motor Stepper</i>	41
Gambar 3.13	Realisasi Rangkaian <i>Driver Motor stepper</i>	42
Gambar 3.14	<i>Driver Limit Switch</i>	43
Gambar 3.15	Realisasi Rangkaian <i>Driver Limit Switch</i>	44
Gambar 3.16	LCD	45
Gambar 3.17	Realisasi Rangkaian LCD	45
Gambar 3.18	Rangkaian <i>Power Supply</i>	46
Gambar 3.19	Realisasi Rangkaian <i>Power supply</i>	47
Gambar 3.20	Dua Jalur Listrik yang Ditempelkan Pada Rel	48
Gambar 3.21	Miniatur Kereta Listrik	48
Gambar 3.22	Realisasi Pintu Kereta	49
Gambar 3.23	Realisasi Pintu Stasiun	49
Gambar 3.24	Diagram Alir Sistem	53
Gambar 4.1	Uji Coba Sensor dan Rangkaian Komparator	56
Gambar 4.2	Uji Coba Rangkaian Kontrol	57
Gambar 4.3	Uji Coba <i>Driver Motor Stepper</i>	59
Gambar 4.4	Uji Coba <i>Driver Limit Switch</i>	60
Gambar 4.5	Tata Letak Sensor – Sensor	66

DAFTAR TABEL

	Halaman	
Tabel II.1	Deskripsi kerja Y-map <i>self-holding relay</i> dengan 5 <i>output</i>	17
Tabel II.2	Tabel macam-macam tipe <i>output</i> dari metode Y-map SHR	18
Tabel II.3	<i>Full Step Mode</i>	23
Tabel II.4	<i>Half step Mode</i>	24
Tabel II.5	Perbandingan Motor	26



DAFTAR SINGKATAN

CPU	Central Processing Unit
DC	Direct Current
EPROM	Erasable Programmable Read-Only Memory
I/O	Input / Output
LED	Light Emmitting Diode
mA	mili ampere
NC	Normally Close
NO	Normally Open
PLC	Programmable Logic Controller
PROM	Programmable Read Only Memory
PWM	Pulse Width Modulation
SHR	Self Holding Relay

DAFTAR ISTILAH

Console. alat yang digunakan untuk memprogram PLC dengan metode offline.

KGLWIN. software yang digunakan dalam pemrograman PLC tipe MASTER-K.

Pbasic, merupakan bahasa yang digunakan dalam pembuatan program pada Basic Stamp

Ladder. merupakan salah satu metode pemrograman dalam PLC.

Normally Open. Switch dengan kondisi awal terbuka.

Normally Close. Switch dengan kondisi awal tertutup.

Relay. merupakan rangkaian yang bersifat elektronis sederhana dan tersusun oleh medan elektromagnet (kawat koil) dan kontak-kontak atau saklar.

Self Holding Relay. teknik dalam pemrograman PLC yang memiliki arti penguncian terhadap diri koil *relay*

BAB 1

PENDAHULUAN

1.1 LATAR BELAKANG

Universitas Indonesia atau sering disingkat dengan sebutan UI adalah salah satu perguruan tinggi negeri di Indonesia yang merupakan universitas tertua di Indonesia, yang resmi didirikan tahun 1946 dengan nama "*Universiteit van Indonesië*". Sebagai universitas tertua di Indonesia, ada hal yang sangat disayangkan dimana sejak dulu hingga sekarang perkembangan transportasi dalam lingkungan kampus UI hanya terdapat bus kuning UI. Dan seiring dengan berkembangnya zaman, UI mempunyai jumlah mahasiswa yang semakin bertambah hingga menjadi ribuan mahasiswa. Jika dilihat, armada bus kuning UI sebanyak 20 buah dengan jumlah mahasiswa yang setiap tahun mengalami peningkatan, hal ini adalah tidak sebanding. Hal ini memberikan dampak kepada mahasiswa yaitu dengan daya tampung armada bus yang dioperasikan tidak lagi mencukupi setiap tahun sehingga bus kuning selalu overload terutama pagi hari, jadwal bus kuning berhenti di setiap halte yang tidak jelas waktunya sehingga tidak jarang mahasiswa bisa menunggu bus antara 10 menit sampai 30 menit lamanya dan dampak-dampak lainnya. Hal ini memberikan masukan kepada Universitas Indonesia untuk memberikan transportasi tambahan di dalam kampus sehingga lebih dapat memberikan kemudahan dalam menunjang kegiatan kampus serta dapat memberikan polusi sehat. Hal yang tepat untuk itu dengan adanya transportasi kereta api.

Seperti yang kita ketahui bahwa kereta listrik merupakan salah satu transportasi massal yang dapat terjangkau oleh semua segmen masyarakat terkait dengan fungsinya untuk memfasilitasi pergerakan barang dan manusia. Jenis kereta api dari sistem tenaga penggerakannya terdiri dari kereta api uap, kereta api diesel, dan kereta api listrik. Kereta api yang paling tepat digunakan dalam kampus UI yaitu kereta listrik.

Kereta listrik pada umumnya membutuhkan masinis dalam menjalankan kereta dan membuka pintu kereta. Pada tugas akhir ini direalisasikan miniatur kereta listrik yang dapat berjalan secara otomatis serta buka-tutup pintu kereta dan stasiun secara otomatis menggunakan *Programmable Logic Control* dan *Basic Stamp*.

1.2 PERUMUSAN MASALAH

Dalam penyusunan pengerjaan tugas akhir ini terdapat permasalahan yang akan dihadapi, yaitu membuat program PLC dan *Basic Stamp* yang dapat saling berinteraksi. Dimana pada saat pemberhentian kereta listrik pada stasiun, maka PWM aktif, pintu stasiun dan pintu kereta api akan terbuka secara otomatis.

1.3 TUJUAN PENULISAN

Adapun tujuan pengerjaan tugas akhir ini adalah merealisasikan suatu sistem yang berfungsi untuk pengaturan pemberhentian kereta listrik, buka tutup pintu kereta dan stasiun. Selain itu, juga memperoleh program ladder dan bahasa Pbasic yang dapat mengendalikan pemberhentian kereta listrik dan dibukanya pintu pada stasiun dan kereta listrik dengan menggunakan PLC dan Mikrokontroler Basic Stamp.

1.4 BATASAN MASALAH

Dalam tugas akhir ini terdapat beberapa hal yang menjadi batasan yaitu :

1. PLC digunakan untuk menerima sinyal dari mikrokontroler *Basic Stamp* atau mengirimkan sinyal ke mikrokontroler *Basic Stamp*. Selain itu, PLC juga menerima sinyal dari rangkaian – rangkaian sensor yang diletakkan di tiap – tiap stasiun.
2. Mikrokontroler *Basic Stamp* digunakan sebagai pengendali pemberhentian kereta listrik, buka-tutup pintu kereta dan pintu stasiun.
3. Sensor yang digunakan adalah sensor cahaya photo-transistor dengan sumber cahaya yang berasal dari LED
4. Sensor yang digunakan hanya tiga buah sesuai dengan jumlah stasiun yang telah dirancang.

5. Pada pintu kereta dan pintu stasiun hanya menggunakan *motor stepper*, *belt* dan *gear* sebagai penggerakannya, tidak menggunakan pneumatik seperti pada kereta listrik umumnya.
6. Pada pintu kereta dan pintu stasiun tidak dilengkapi dengan pengaman.

1.5 METODOLOGI PENELITIAN

Dalam rangka mewujudkan tugas akhir ini, metodologi penelitian yang digunakan adalah sebagai berikut:

1. Metode kepustakaan, yaitu dengan membaca buku-buku referensi serta file-file yang berkaitan dengan tugas akhir.
2. Metode eksperimen dan rancang bangun, yaitu menerapkan program yang telah dibuat serta melakukan percobaan dengan BASIC Stamp dan PLC kemudian diaplikasikan terhadap miniatur *kereta listrik*.

1.6 SISTEMATIKA PENULISAN

Tugas akhir dibagi menjadi lima bab yang secara garis besar isinya dapat diuraikan sebagai berikut:

BAB I PENDAHULUAN

Memuat latar belakang masalah, batasan masalah, dan sistematika penulisan.

BAB II DASAR TEORI

Membahas tentang teori mikrokontroler *Basic Stamp*, PLC, komparator, motor stepper, PWM, dan motor DC. Pada bab ini juga terdiri dari metode dan teknik yang digunakan dalam pemrograman Basic Stamp dan PLC.

BAB III RANCANG BANGUN

Pada bab ini berisi penjelasan tentang perancangan perangkat keras dan perancangan perangkat lunak sistem.

BAB IV UJI COBA DAN ANALISIS

Pada bab ini berisi hasil pengujian dan analisis dari perangkat keras dan perangkat lunak serta integrasi sistem secara keseluruhan.

BAB V KESIMPULAN

Pada bab ini membahas kesimpulan yang didapat dari hasil penelitian.



BAB 2

PLC, *BASIC STAMP*, PWM, *MOTOR STEPPER*, DAN MOTOR DC

2.1 MIKROKONTROLER *BASIC STAMP*^[1]

Mikrokontroler merupakan sebuah sistem komputer yang seluruh atau sebagian besar elemennya dikemas dalam satu chip IC, sehingga disebut *single chip microcomputer*. Lebih lanjut, mikrokontroler merupakan sistem komputer yang mempunyai satu atau beberapa tugas yang sangat spesifik, berbeda dengan PC yang memiliki beragam fungsi. Perbedaan lainnya adalah perbandingan RAM dan ROM yang sangat berbeda antara komputer dengan mikrokontroler. Dalam mikrokontroler, ROM jauh lebih besar dibanding RAM, sedangkan dalam komputer atau PC RAM jauh lebih besar dibanding ROM.

Gambar 2.1 adalah salah satu contoh dari mikrokontroler. Mikrokontroler umumnya dikelompokkan dalam satu keluarga. Berikut ini adalah contoh-contoh keluarga mikrokontroler:

1. Keluarga MCS-51
2. Keluarga MC68HC05
3. Keluarga MC68HC11
4. Keluarga AVR
5. Keluarga PIC 8



Gambar 2.1 Mikrokontroller

Basic stamp adalah mikrokontroler dengan ukuran kecil, di khususkan untuk *basic interpreter* (PBASIC) digunakan untuk ROM, ini dibuat oleh parallax, inc. dan mikrokontroler ini telah cukup terkenal untuk hobi elektronika sejak awal 1990an.

Walaupun *Basic stamp* dalam bentuk DIP chip, tetapi dalam kenyataannya PCB yang kecil itu berisikan elemen-elemen yang diperlukan dalam sistem mikroprocessor :

Mikrokontroler ini berisikan CPU yang dibangun dari ROM yang berisikan *Basic Interpreter* dan beberapa *peripheral*

1. *Memory (serial Eeprom)*
2. *Clock*
3. *Power supply*
4. *Eksternal input dan output*

PBASIC, adalah bahasa pada *Basic stamp*, dimana dimasukkan fungsi mikrokontroler, termasuk PWM, komunikasi serial, I²C dan komunikasi 1 kabel, komunikasi dengan driver LCD, frekuensi gelombang sinus dan kemampuan untuk mendeteksi nilai rangkaian analog dengan rangkaian RC.

Basic stamp ini membutuhkan *power supply* sebesar 9 V, agar sistem dapat menyala dan untuk mendownload program ke *Basic stamp* dibutuhkan PC dengan basic minimum Pentium 1. Program yang telah di download akan tersimpan di memory dan akan bertahan hingga memory dihapus atau diprogram ulang.

Pada dasarnya terdapat 4 macam interpreter, *Basic stamp 1*, *basic stamp 2*, Javelin Stamp dan Spin Stamp. Pada *Basic stamp 2* terdapat 7 macam Sub yaitu :

- BS1
- BS2
- BS2e

BS2sx
BS2p24
BS2p40
BS2pe
BS2px

Sub-variant diatas memiliki memori tambahan, eksekusi program yang lebih cepat, penambahan perintah yang dikhususkan untuk PBASIC, ekstra I/O dan sebagainya dalam perbandingan pada *original* BS2 model, sementara BS1 dan BS2 menggunakan PIC, dan beberapa *Basic stamp 2* menggunakan SX processor.

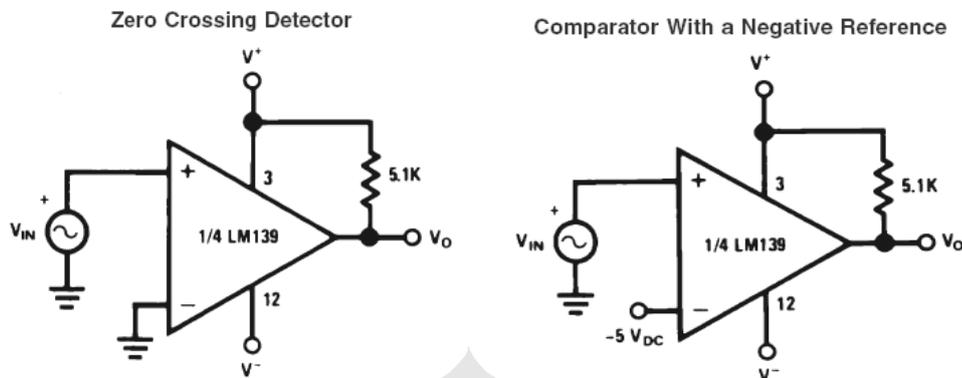
Variant yang ketiga adalah *Javelin Stamp*. Modul ini menggunakan bahasa pemrograman *java SUN microsystem's* sebagai pengganti dari parallax PBASIC. *Variant* yang keempat adalah *Spin Stamp*. Modul ini berdasarkan pada parallax propeller dan juga menggunakan bahasa pemrograman SPIN sebagai pengganti dari PBASIC.

Beberapa perusahaan membuat Virtual "*clones*" dari *Basic stamp* dengan penambahan beberapa *features*, seperti eksekusi program yang lebih cepat, *analog to digital converter* dan hardware yang berdasarkan pada PWM. walaupun menggunakan pin yang sama pada *Basic stamp* pemrograman memungkinkan para pengguna *Basic stamp* untuk menghubungkan produk mereka dengan desain yang dengan *Basic stamp*.

2.2 KOMPARATOR^[2]

Lm339 series terdiri dari empat *voltage comparator* yang presisi yang berdiri sendiri dengan spesifikasi *offset voltage* terendah max di 2 mV untuk keempat komparator tersebut. IC ini didesign secara spesifik untuk beroperasi pada single power supply dengan range voltage yang luas. IC ini mampu beroperasi dari power supply yang berubah dan pada keadaan arus power supply yang rendah. IC ini juga memiliki karakteristik yang unik bahwa input *common-*

mode voltage range terdapat ground, padahal beroperasi pada *single power supply voltage*. Gambar 2.2 merupakan contoh dari rangkaian komparator.



Gambar 2.2 Komparator

2.3 PROGRAMMBALE LOGIC CONTROLLER (PLC)^[3]

Programmable Logic Controller (PLC) merupakan perangkat kontrol elektronik (digital) yang dirancang salah satunya untuk keperluan pengontrolan mesin-mesin listrik. Semula PLC digunakan untuk menggantikan fungsi *relay* yang banyak digunakan pada saat itu. Spesifikasi yang banyak digunakan dan dibutuhkan antara lain :

- 1) *Rugged* (kasar dan keras) dan tahan terhadap *noise*
- 2) Disusun secara modular sehingga memudahkan untuk menambah/ dikurangi (untuk pengembangan dan perawatan)
- 3) Mempunyai sambungan dan level sinyal yang standar
- 4) Mudah diprogram dan diprogram ulang tanpa harus menambah perangkat kontrol.

PLC menggunakan memori yang dapat diisi oleh program yang dapat menerapkan fungsi-fungsi khusus dalam elektronika digital seperti logika, sekuensial/ urutan, pewaktuan (*timer*), Pencacahan (*counting*) dan lainnya, guna mengendalikan suatu proses analog/ digital dari suatu proses.

PLC merupakan piranti berbasis *microprocessor* dan dapat dianggap sebagai komputer yang dirancang untuk tujuan pengendalian tertentu. *Console* hanya diperlukan untuk memasukan program secara *online*, yang langsung diketikkan melalui terminal pemrograman genggam (*keypad*). Satu unit *console*

dapat menangani hingga beberapa unit PLC. PLC memberikan respon terhadap berbagai sensor yang dihubungkan ke *input*-nya, memutuskan proses apa yang dikerjakan berdasarkan instruksi yang telah diberikan dan diprogram ke dalam memorinya, dan memberikan ketetapan terhadap *output* yang diinginkan.

Aplikasi dari PLC sangatlah banyak untuk disebutkan satu per satu. Tetapi sebagai gambaran jenis peralatan yang sering digunakan adalah seperti Peralatan seperti motor listrik, pompa, katup, mesin pengemasan, mesin perakitan dan lainnya.

Untuk aplikasi-aplikasi tersebut, terdapat beberapa keuntungan yang dapat diperoleh dengan menggunakan PLC daripada menggunakan sistem kontrol konvensional. Pertama, mereka diprogram dengan bahasa yang sederhana, yang disebut logika tangga (*ladder logic*). Bahasa ini mengambil bentuk sebagai instruksi yang mudah dipelajari, yang dimasukkan secara langsung dari *tuts* berlabel pada *console*. Kedua, masukan dan keluaran PLC dapat secara langsung dihubungkan dengan berbagai piranti seperti saklar, *relay*, dan sensor, sedangkan untuk keluaran komputer diperlukan peralatan *interface* tambahan yaitu port masukan/ keluaran (*I/O Port*). Ketiga, PLC dapat digabungkan langsung dengan sebuah PROM (*Programmable Read Only Memory*), yang tetap akan menyimpan program meskipun daya listrik telah dimatikan. Sebaliknya, untuk menggabungkan sebuah PROM dengan komputer standar, perlu dilakukan beberapa modifikasi.

Selain keuntungan yang dijelaskan diatas, PLC juga dirancang dan dikonstruksi dengan kemampuan untuk tetap dapat bekerja pada lingkungan-lingkungan yang cukup berat dan kasar, dengan kata lain suatu lingkungan yang mempunyai kondisi temperatur yang cukup tinggi, kelembaban udara yang tinggi, pengaruh vibrasi serta kondisi *noise* dan kejutan-kejutan yang timbul karena kondisi kerja mesin-mesin atau peralatan listrik lainnya. Sehingga dengan memanfaatkan PLC sebagai alat untuk melaksanakan proses kontrol mesin-mesin listrik dapat diperoleh kelebihan/ keuntungan sebagai berikut :

- 1) Dapat bekerja dengan cukup aman, handal serta fleksibel. Yaitu dapat dihubungkan/ berkomunikasi dengan peralatan kontrol lainnya
- 2) Mudah untuk mengubah program atau rancangan dari rangkaian proses kontrol

- 3) Mengurangi/ menghemat pemakaian kawat/ kabel instalasi sitem kontrol
- 4) Tidak memerlukan tempat yang luas dalam penginstalasiannya walaupun kontrol yang ditangani cukup rumit.
- 5) Relatif mudah dalam pelaksanaan perawatan dan perbaikan.
- 6) Mudah dalam melaksanakan pengembangan dan perluasan sistem kontrol proses.

Untuk menetapkan penggolongan PLC, perlu terlebih dahulu dipilih kriterianya, salah satu kriteria yang penting adalah jumlah maksimum dari port masukan/ keluaran. Kriteria ini akan memberikan informasi mengenai kemampuan PLC, karena makin banyak I/O *port* yang dapat dikendalikan maka jumlah memori yang diperlukan juga makin besar, sehingga diperlukan pula CPU yang lebih cepat. Dengan menggunakan kriteria ini maka akan diperoleh penggolongan PLC sebagai berikut ^[1] :

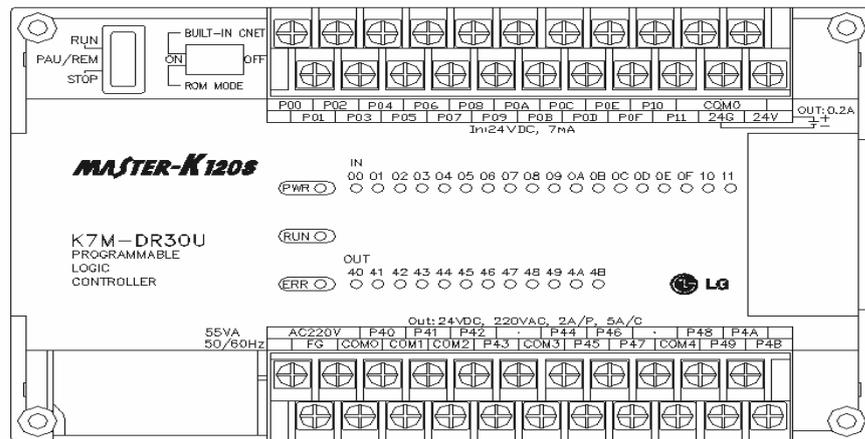
- 1) Skala mikro (15 hingga 64 *port* I/O)
- 2) Skala kecil (64 hingga 128 *port* I/O)
- 3) Skala menengah (128 hingga 512 *port* I/O)
- 4) Skala besar (lebih dari 512 *port* I/O)

PLC setidaknya terdiri dari unit kontrol, memori program, rangkaian *input*, rangkaian *output* dan unit *power supply*.

2.3.1 Bagian bagian PLC

PLC memiliki tiga bagian utama, yaitu pusat pengolahan data, modul *input* dan modul *output*, selain itu juga PLC memiliki EPROM yang akan menyimpan data data/ hasil pemrograman.

Gambar 2.3 merupakan bentuk fisik dari PLC dari “LG Master-K 120S 30 I/O” adalah:



Sumber : www.lgis.com

Gambar 2.3. Contoh bentuk fisik PLC “LG Master-K 120S 30 I/O”

a) Pusat Pengolahan Data (CPU)

CPU dapat dianggap sebagai otak dari PLC. Program dipanggil dari unit memori dan diproses CPU. Pemrosesan dapat disebut sebagai menjalankan program. Apa yang sebenarnya terjadi adalah bahwa program tersebut di-scan, ini berarti bahwa program diperiksa dari awal hingga akhir dan diinformasikan baru dimasukkan. Ini sering disebut “waktu *scan*” PLC, walaupun sebenarnya lebih berkaitan dengan waktu pengoperasian program. *Scan* dari program umumnya memakan waktu beberapa milidetik, tetapi hal ini tergantung pada panjang program dan kerumitannya. Setelah *scan* yang satu selesai, *scan* berikutnya akan dimulai dengan segera. PLC membutuhkan tempat untuk menyimpan program kerja, fasilitas ini disebut unit memori atau memori saja. Peralatan memori dapat memiliki penyimpanan data jangka pendek atau jangka panjang.

b) Modul *Input* PLC

Input adalah peralatan luar yang terhubung pada modul masukan yang akan memberikan sinyal kepada PLC, bahwa sesuatu sedang terjadi diluar PLC. Masukan bisa berupa sakelar, sensor dan berbagai macam transduser. Pada PLC “LG Master-K 120S 30 I/O” masukan diberikan Kode I dengan jumlah *Input* sebanyak 18 buah P00 –P11 dan P0A-P0F, dan kontak untuk yang tersedia untuk *input* adalah NO dan NC.

c) Modul *Output* PLC

Output PLC adalah Relay atau Transistor, yang dihubungkan dengan peralatan luar. Melalui relay atau transistor ini nantinya akan menggerakkan atau mengaktifkan peralatan luar, seperti lampu, *display*, relay, kontaktor. Untuk PLC “LG Master-K 120S 30 I/O” *Output* diberikan Notasi Q dengan jumlah sebanyak 12 buah P40 – P49 dan P4A-P4B.

PLC bekerja dengan menerima data dari peralatan *input* yang merupakan saklar-saklar, tombol-tombol, sensor-sensor dan lain sebagainya. Perubahan yang terjadi pada peralatan *input* akan memberikan sinyal pada PLC, kondisi *input* tersebut akan diolah oleh PLC selanjutnya perintah-perintah dari *input* akan ditransfer oleh PLC ke *outputnya* yang kemudian dapat digunakan untuk menggerakkan mesin-mesin.

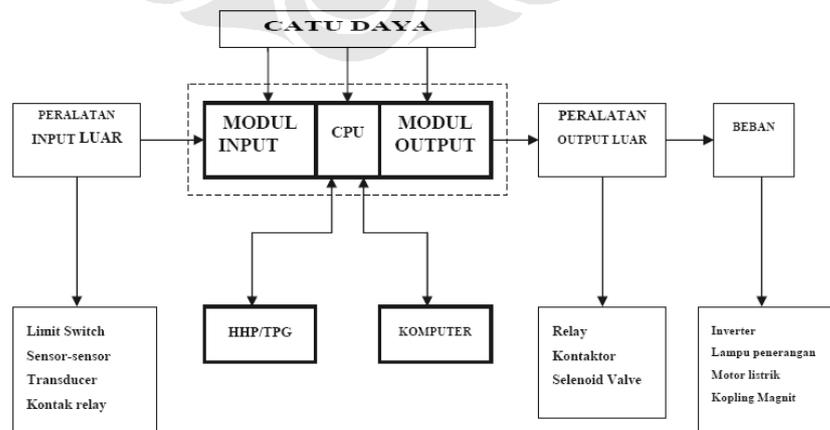
2.3.2 Prinsip Kerja

Dari definisi diatas didapat gambaran bahwa prinsip kerja PLC tetap memenuhi kriteria dari blok diagram dasar dari sistem kontrol seperti pada Gambar 2.4.



Gambar 2.4. Blok diagram dasar sistem kontrol PLC

Dari blok diagram dasar diatas, prinsip kerja PLC dikembangkan seperti yang terlihat pada Gambar 2.5.



Gambar 2.5. Blok diagram lengkap sistem kontrol PLC

2.3.3 Metode Memprogram PLC

Pemrograman adalah penulisan serangkaian perintah yang memberikan instruksi kepada PLC untuk melaksanakan tugas yang telah ditentukan. Untuk memprogram PLC dapat dilakukan dengan Cara :

1. *Instruction List*

Sistem pemrograman ini bersifat tekstual. Singkatan-singkatan khusus yang disebut mnemonics digunakan untuk mengidentifikasi perintah-perintah berbeda yang sedang dijalankan. Tipe ini dapat juga disebut sebagai programmer 'on line' untuk menulis program, unit pemrograman (*console*) ini harus dihubungkan pada PLC. Contoh: LD, OR, AND, OUT.

2. *Diagram Ladder*

Alternatif lain dari program format instruksi adalah sistem grafis yang disebut dengan *ladder*. Sistem ini menggunakan gambar grafis atau ikon-ikon untuk mewakili perintah-perintah. Pemrograman ladder pada umumnya digunakan pada perlengkapan *portable* (komputer) yang digunakan untuk pemrograman. Perlengkapan ini memungkinkan pemrograman *offline*. Pemrograman *offline* berarti bahwa perlengkapan pemrograman tidak harus dihubungkan dengan PLC. Setelah program ditulis, PLC yang digunakan dihubungkan pada komputer dan program ditransfer ke PLC. Yang unik ladder dari satu salah produsen PLC tidak dapat langsung dimasukkan ke produsen PLC yang lain. Jadi kita harus ladder lagi dengan program pembawanya masing-masing.

3. *Sequential Function Chart (SFC)*

Perangkat lunak pemrograman terbaru dinamakan SFC (*Sequential Function Chart*). Perangkat lunak ini juga dirancang untuk digunakan pada komputer yang kompatibel dengan IBM. Dalam SFC, grafis yang ditampilkan mewakili alur logika suatu tugas, jadi program hanya berisi diagram alur dari respon yang diinginkan

2.3.4 Pemilihan PLC

Dalam pengerjaan tugas akhir ini terdapat beberapa langkah yang harus diperhatikan dalam pemilihan PLC, yaitu :

- 1) Menentukan *Input* dan *Output* PLC (dalam hal ini telah ditentukan sebanyak 18 *Input* dan 12 *Output*)
- 2) Menentukan sebuah deskripsi kerja secara lengkap dan sistematis
- 3) Merubah kedalam bentuk *flowchart* untuk memudahkan pemrograman
- 4) Membuat program dalam bentuk diagram tangga (*ladder diagram*)
- 5) Memasukkan program kedalam CPU dan menyimpannya
- 6) Menguji program yang telah dibuat
- 7) Memperbaiki program yang telah dibuat
- 8) Menguji pada rangkaian yang sebenarnya dengan cara menghubungkan modul I/O pada peralatan yang sebenarnya, dalam hal ini miniatur *tram*
- 9) Menyimpan program yang telah sesuai kedalam memori PLC.

2.4 PEMROGRAMAN PLC

Untuk membentuk satu rangkaian secara logis diperlukan bermacam-macam instruksi pada PLC. Sebelum memasukan program yang telah telah dibuat PLC harus dalam kondisi *Stop*, jika PLC dalam kondisi *Run* program tidak akan bisa masuk ke dalam memori PLC. Jika didalam memori PLC masih terdapat program yang tersimpan, program tersebut akan dihapus dengan sendirinya ketika dimasukan program yang baru, atau bisa juga menghapusnya secara manual sebelum memasukan program yang baru.

Pemrograman PLC Master-K menggunakan *software* KGLWIN didalamnya terdapat suatu metoda khusus yang sederhana disebut *ladder diagram*. Dasar dari pemrograman *ladder* adalah pemahaman mengenai fungsi *relay*, kontak-kontak seperti *normaly open* (NO) dan *normaly close* (NC).

2.4.1 Instruksi PLC Master-K 120S^[4]

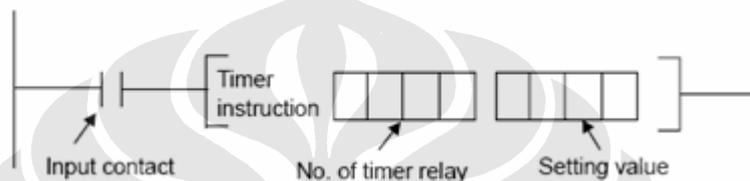
Master-K 120S memiliki instruksi program yang terdapat pada *software* KGLWIN seperti *auxiliary relay*, fungsi waktu, *counter* dan sebagainya. Berikut ini adalah beberapa instruksi yang digunakan pada pemrograman sistem panggilan otomatis *tram*.

a) *Input*

PLC akan menerima sinyal *input* baik digital (ON/OFF) maupun analog (berdasarkan nilai mV). *Input* pada tugas akhir ini sebanyak 18 *point* dan dinotasikan dengan P00-P11, P0A-P0F.

b) *Timer Relay*

Tipe PLC Master-K memiliki *timer* dengan spesifikasi 100msec 10 msec dan 1 msec. Penomoran *timer* seperti yang ditunjukkan pada Gambar 2.6 berdasarkan spesifikasi yang dimilikinya adalah T00-T191 (100msec), T192-T250 (10msec) dan T251-255 (1msec).



Gambar 2.6. Bagian-bagian program untuk *timer*

Metode *timer* memiliki beberapa variasi sesuai dengan instruksi *timer* yang tersedia (TON, TOFF, TMR, TMON, TRTG). Maksimum *setting value* pada *timer* adalah hFFFF untuk heksadesimal dan 65535 untuk desimal. Gambar 2.7 adalah contoh penggunaan *timer* TON.



Gambar 2.7. *Timer* TON dengan *setting value* 10 detik

a) *Auxiliary relay/ Internal relay (M)*

Auxiliary relay adalah salah satu fasilitas yang tersedia pada PLC, dimana *internal relay* ini sangat membantu dalam pemrograman. *Internal relay* pada PLC Master-K dinotasikan dengan M dan berjumlah M00-M191.

b) *Output*

PLC akan memberikan sinyal *output* baik digital (ON/OFF) maupun analog (berdasarkan nilai mV). *Output* pada tugas akhir ini sebanyak 12 *point* dan dinotasikan dengan P40-P49, P4A-P4B.

2.4.2 Metode Y-Map (*self-holding relay*) ^[5]

Metode pemrograman Y-Map adalah salah satu teori dalam perancangan program *ladder* dalam PLC, dimana metode ini ditujukan untuk membantu *mapping* pemrograman bahasa *ladder*. Pada dasarnya teori ini ditujukan untuk merancang deskripsi kerja suatu program yang memiliki cara kerja yang berurutan atau *sequential* serta memiliki siklus. Salah satu jenis metode yang sering digunakan adalah metode *self-holding relay* (SHR). *Self-holding relay* sendiri adalah suatu teknik dalam pemrograman PLC yang memiliki arti penguncian terhadap diri koil *relay*, maksudnya kontak yang digunakan pada koil *relay* dimanfaatkan untuk mengunci koil ketika koil mendapatkan sinyal tanpa terpengaruh lamanya waktu pemberian sinyal.



Gambar 2.8. *Self-holding relay* pada koil M0000

Metode SHR seperti yang ditunjukkan pada Gambar 2.8 memungkinkan mengontrol berapapun jumlah *output* (beban) hanya dengan menggunakan satu buah saklar (*input*). Berikut ini adalah contoh dasar pemrograman ladder dengan metode *self-holding relay*, dimana jumlah *output* yang dikontrol oleh sebuah saklar sebanyak 5 buah. Simbol '1' digunakan untuk menyatakan kondisi ON dan simbol '0' untuk kondisi OFF.

Metode Y-map dapat dijadikan dasar pemrogram *ladder* dan dapat dikembangkan pada aplikasi dengan deskripsi kerja berurutan serta memiliki siklus pengulangan (*looping*).

Tabel II.1 menjelaskan deskripsi kerja dari Y-map, bahwa kondisi ON pada *output* berurutan dengan dikontrol oleh satu *input* yaitu M. *Input* hanya memiliki kondisi ON dan OFF yang dinyatakan pada kolom status.

Tabel II.1. Deskripsi kerja Y-map *self-holding relay* dengan 5 output

Input (M)		Output (Y)				
Kondisi	status	Y1	Y2	Y3	Y4	Y5
1	1	1				
2	0	1				
3	1	1	1			
4	0	1	1			
5	1	1	1	1		
6	0	1	1	1		
7	1	1	1	1	1	
8	0	1	1	1	1	
9	1	1	1	1	1	1
10	0	1	1	1	1	1
11	1	0	0	0	0	0
12	0	0	0	0	0	0

Pada Tabel II.2 digambarkan deskripsi kerja beberapa jenis *output* dengan menggunakan metode SHR. Deskripsi tersebut dapat diwujudkan dengan memprogram *ladder* sesuai metode SHR, yaitu program dibagi menjadi dua bagian, dimana bagian pertama merupakan *main program* sedangkan bagian yang satunya merupakan program untuk *Output*. Pembagian program menjadi dua bagian tersebut dimaksudkan untuk mempermudah user dalam mengevaluasi program apabila terdapat kesalahan didalamnya. Program *ladder* lengkap untuk metode SHR dengan 5 *output* diberikan pada Gambar 2.9.

Tabel II.2. Tabel macam-macam tipe *output* dari metode Y-map SHR

Output (Y)				
Y1	Y2	Y3	Y4	Y5
1				
1				
1	1			
1	1	1		
1	1	1	1	
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
0	0	0	0	0
0	0	0	0	0

Output (Y)				
Y1	Y2	Y3	Y4	Y5
1				
1				
0	1			
0	1			
	0	1		
	0	1		
		0	1	
		0	1	
			0	1
			0	1
				0
				0

(a)

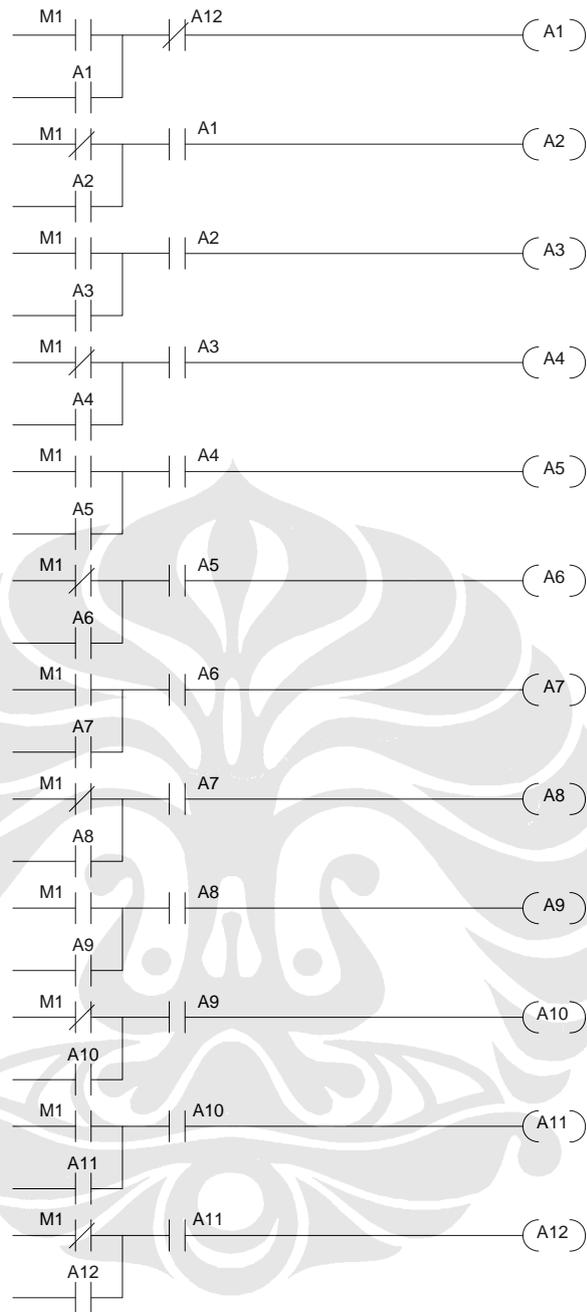
(b)

Output (Y)				
Y1	Y2	Y3	Y4	Y5
		1		
		1		
	1		1	
	1		1	
1				1
1				1
	1		1	
	1		1	
		1		
		1		
		0		
		0		

Output (Y)				
Y1	Y2	Y3	Y4	Y5
1				1
1				1
	1		1	
	1		1	
		1		
		1		
	1		1	
	1		1	
1				1
1				1
0				0
0				0

(c)

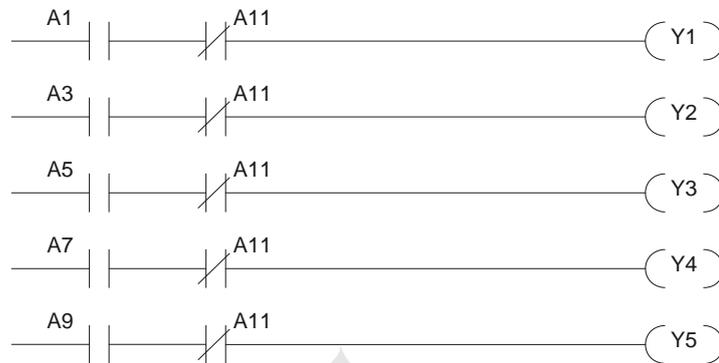
(d)



Gambar 2.9 Program utama kontrol proses metode Y-map SHR

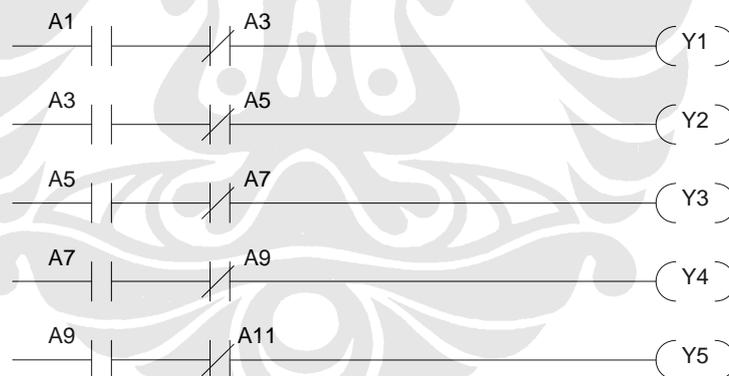
Program pada Gambar 2.9 merupakan program utama dari keseluruhan program dengan metode Y-map SHR. Program ini berisikan kontrol yang akan mempengaruhi kondisi *output*.

Program pada Gambar 2.10 merupakan program untuk *output* tipe (a), dimana cara kerja *output* adalah ON secara berurutan atau satu per satu lalu kemudian akan OFF secara serentak.



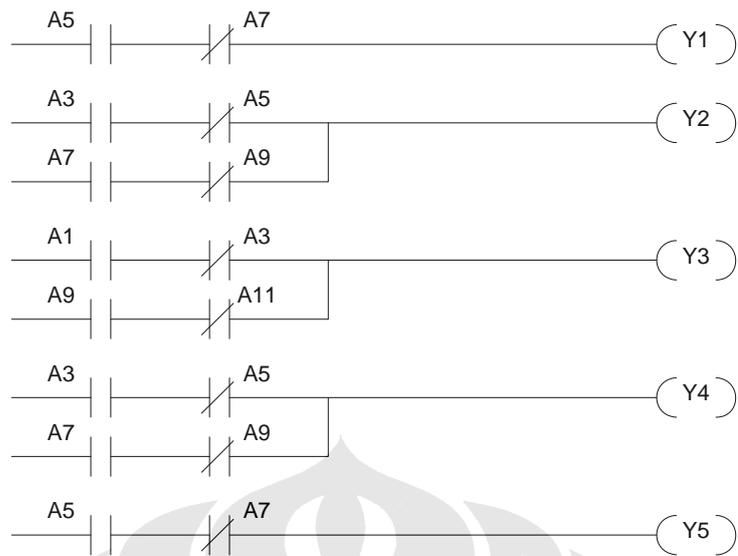
Gambar 2.10 *Output* tipe (a)

Program pada Gambar 2.11 merupakan program untuk *output* tipe (b), dimana cara kerja *output* hampir sama dengan *output* tipe (a) ON secara berurutan yang membedakan ketika *output* selanjutnya yang ON maka *output* sebelumnya akan OFF.



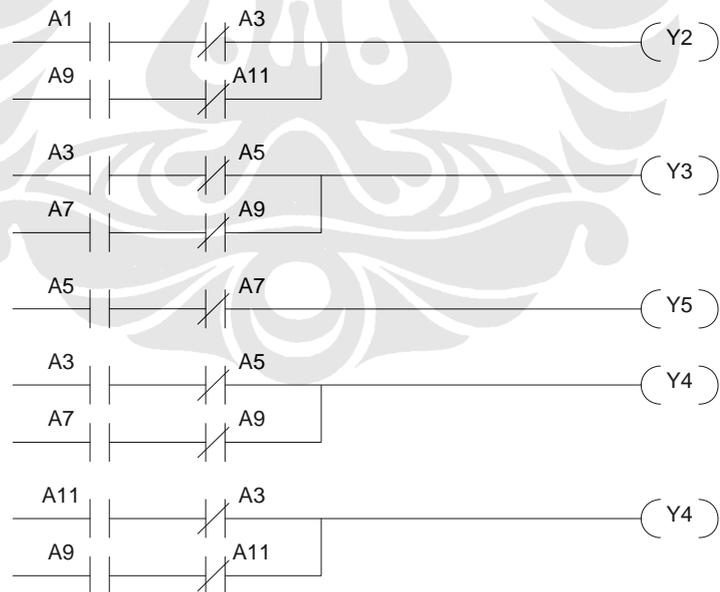
Gambar 2.11 *Output* tipe (b)

Sementara untuk program pada Gambar 2.12 merupakan program untuk *output* tipe (c), cara kerja *output* adalah sesuai dengan tabel dimana *output* akan aktif dari terminal *input* paling kiri dan paling kanan menuju ke tengah.



Gambar 2.12 *Output* tipe (c)

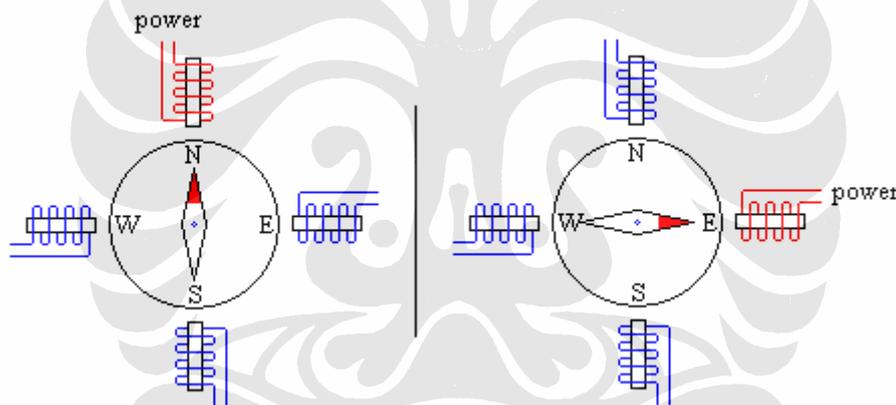
Program pada Gambar 2.13 merupakan program untuk *output* tipe (d), cara kerja *output* adalah sesuai dengan tabel dimana *output* akan aktif dari terminal *input* paling tengah menuju terminal *input* paling kanan dan kiri.



Gambar 2.13 *Output* tipe (d)

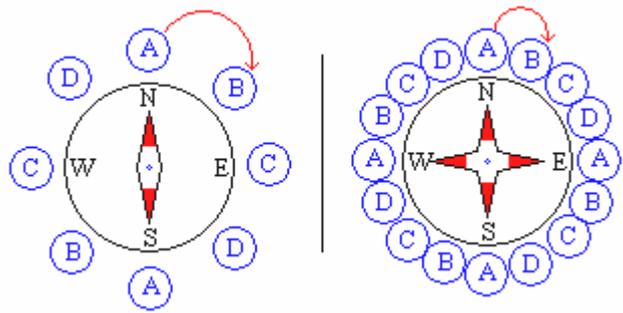
2.5 MOTOR STEPPER^[6]

Pada Motor DC biasa, akan berputar dan berputar terus selama *power supply* ada. Tidak ada rangkaian cerdas tertentu yang diperlukan untuk mengendalikan motor tersebut, kecuali hanya memperlambat putaran atau membalik putaran, dengan menerapkan polaritas balik. *Motor stepper* adalah sangat berbeda. Jika diberikan power pada motor ini, maka motor ini akan berada dalam keadaan diam, agar motor dapat berputar, anda harus merubah sinyal yang masuk ke motor. Sebagai ilustrasi, dapat dibayangkan sebuah kompas dengan elektromagnet disekitarnya. Sebagaimana digambarkan pada gambar dibawah, apabila *power* yang diberikan pada elektromagnet diganti, maka akan merubah posisi jarum dari kompas. Pada Gambar 2.14 merupakan ilustrasi *motor stepper* dengan sebuah kompas



Gambar 2.14. Ilustrasi sebuah kompas dengan elektromagnet

Dengan empat buah elektromagnet maka gerakan akan melompat secara kasar. Sekarang bayangkan susunan yang sama dengan 100 elektromagnet yang mengitari kompas. Dengan mengatur energi yang mengalir pada setiap elektromagnet dalam berurutan, maka jarum akan memerlukan sebanyak 100 langkah untuk melakukan satu kali putaran. Tetapi dengan pengaturan 100 elektromagnet secara individu, akan memerlukan elektronika yang kompleks. Gambar 2.15 merupakan ilustrasi *motor stepper* dengan jarum kompas dengan elektromagnet.



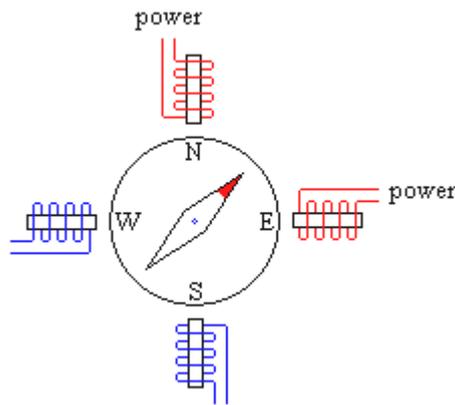
Gambar 2.15 Ilustrasi *motor stepper* dengan jarum kompas dengan elektromagnet

Pada ilustrasi tersebut, huruf-huruf yang melingkar mewakili elektromagnet. Semua magnet dengan huruf yang sama berada dalam keadaan koneksi. Ketika anda memberi arus pada rangkaian tersebut, maka semua elektromagnet dengan huruf yang sama akan on pada saat itu, untuk menggerakkan kompas, maka elektromagnet berikutnya harus dialiri arus, sehingga akan menimbulkan gerakan. Tabel II.3 merupakan tabel *Full Step Mode*

Table II.3 *Full Step Mode*

A	B	C	D	Komentar
1	0	0	0	<i>Take a step clock wise</i>
0	1	0	0	<i>another step clock wise</i>
0	0	1	0	<i>another step clock wise</i>
0	0	0	1	<i>another step clock wise</i>
0	0	0	1	<i>No step take</i>
0	0	1	0	<i>Take a step back</i>

Dengan menghidupkan dua koil pada waktu yang bersamaan maka motor akan berada dalam posisi diantaranya. Gambar 2.16 adalah gambar *Half Step Mode* dan tabel II.4 merupakan tabel *Half Step Mode*



Gambar 2.16 *Half Step Mode*

Tabel II.4 Half Step Mode

A	B	C	D	Komentar
1	0	0	0	<i>Take A Step Clock Wise</i>
1	1	0	0	<i>Half a step clock wise</i>
0	1	0	0	<i>The complete full step clock wise</i>
0	1	1	0	<i>another half step clock wise</i>
0	0	1	0	<i>The complete full step clock wise</i>
0	0	1	1	<i>another half step clock wise</i>
0	0	0	1	<i>The complete full step clock wise</i>
1	0	0	1	<i>another half step clock wise</i>
1	0	0	0	<i>Start Position</i>

Gambar 2.17 merupakan contoh dari *motor stepper* yang biasa digunakan pada disk drive.

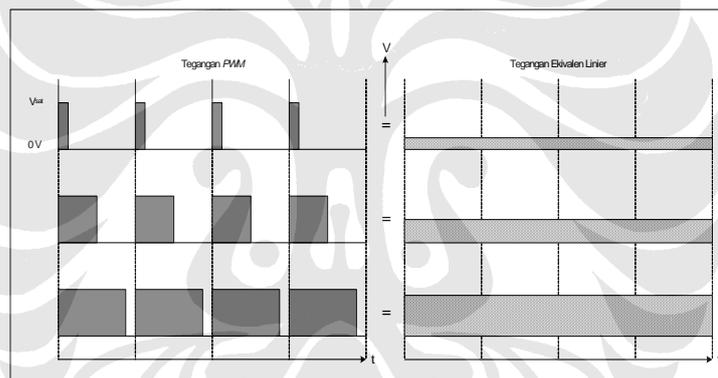


Gambar 2.17 Bentuk fisik *motor stepper* disk drive 1,2"

2.6 PWM (*Pulse With Modulation*)^[7]

Pada dasarnya putaran motor dc secara umum berbanding lurus dengan tegangan *supply* pada terminalnya. Untuk mendapatkan tegangan rendah maka diberikan tegangan rendah. Untuk putaran tinggi maka tegangan harus tinggi. Dengan demikian, masalah yang harus diselesaikan pada rangkaian pengemudi motor adalah bagaimana membuat tegangan output dapat bervariasi (dapat diatur mulai dari 0 hingga tegangan maksimum secara linier). Untuk membuat tegangan tersebut dapat bervariasi maka digunakan teknik *PWM (Pulse With Modulation)*.

PWM (Pulse With Modulation) adalah suatu teknik manipulasi dalam pengendalian motor (atau perangkat elektronik berarus besar lainnya) yang menggunakan prinsip *cut-off* dan *saturasi*. Transistor atau komponen *switching* didisain bekerja dengan karakteristik mirip linier. Gambar 2.18 mengilustrasikan sinyal *PWM* versus tegangan ekuivalen liniernya.



Gambar 2.18 Prinsip Kerja *Rotary Encoder*

Lebar pulsa dinyatakan dalam *Duty Cycle*. Misal *duty-cycle* 10 %, berarti lebar pulsa adalah 1/10 bagian dari satu perioda penuh. Dalam Gambar 2.18 nampak bahwa makin sempit pulsa *PWM*, tegangan ekuivalen liniernya makin kecil. Jika *duty-cycle* 100 % maka tegangan ekuivalen liniernya sama dengan tegangan maksimumnya pada motor dikurangi tegangan saturasi pada kolektor emiter.

Dalam aplikasi untuk driver motor *dc* secara umum, frekuensi *PWM* dapat ditentukan mulai dari 60 hingga 2000 *Hz* sesuai dengan kemampuan *switching* komponen. Pembangkitan pulsa *PWM* dapat diperoleh melalui berbagai rangkaian timer yang bisa didapat dipasaran. Misalnya yaitu rangkaian driver motor *dc* yang

terdapat pada IC L293D dengan kemampuan daya hingga 60 W dan Input PWM yaitu dengan memanfaatkan saluran *enable*.

2.7 MOTOR DC^[8]

Motor adalah *actuator* bagi kereta yang merupakan syarat mutlak bagi kereta untuk dapat bergerak atau berpindah dari satu tempat ke tempat lainnya dalam rangka mewujudkan fungsi atau tujuan pembuatan miniatur kereta listrik.

Dikenal beberapa jenis motor yang sering digunakan untuk membangun sebuah robot, yaitu motor DC, Stepper, motor servo dan tidak menutup kemungkinan penggunaan motor AC pada kereta. Setiap jenis motor memiliki kekurangan dan kelebihan masing-masing. Hal ini ditunjukkan pada tabel II.5.

Tabel II.5. Perbandingan motor

Jenis motor	keuntungan	Kerugian
motor DC	banyak tersedia, banyak variasi, <i>power</i> besar, mudah dalam <i>interface</i>	terlalu cepat, dibutuhkan <i>gearbox</i> , arus sangat besar, <i>complex</i> kontrol PWM
motor servo	sudah terdapat <i>gearbox</i> , variasi banyak, <i>power</i> yang dibutuhkan tidak banyak, <i>speed medium</i> , mudah dalam <i>interface</i> , baik dalam kontrol posisi	kemampuan mengangkat beban berat rendah, kontrol kecepatan kurang
<i>motor stepper</i>	<i>precise speed control</i> , variasi banyak, mudah dalam <i>interface</i>	<i>not very powerful</i> , <i>complex control</i> , sukar dalam penempatan roda

Motor DC terdiri dari beberapa jenis, namun yang sangat umum digunakan adalah tipe magnet permanen (*Permanent Magnet DC Motor-PMDC*).

Motor DC tipe ini menggunakan magnet permanen untuk menghasilkan fluks magnet yang diperlukan, sehingga fluks magnet bersifat konstan dan tidak memerlukan adanya eksitasi dari luar.

2.7.1 Dasar Cara Kerja Motor DC

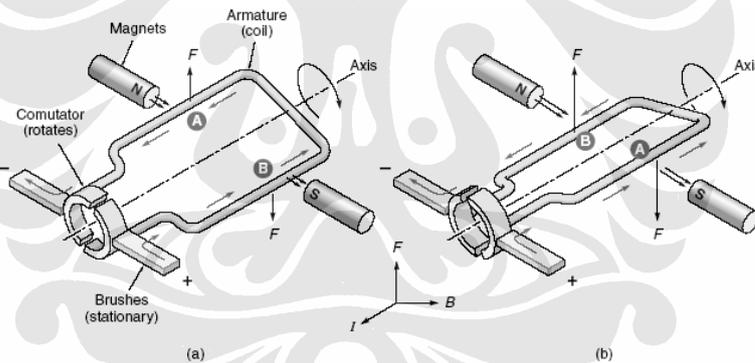
Pada dasarnya motor DC bekerja berdasarkan prinsip gaya *Lorentz* pada kawat berarus. Bila kawat sepanjang l dialiri arus listrik sebesar I dan terletak di dalam medan magnet B , maka kawat tersebut akan mengalami gaya *Lorentz* yang besarnya :

$$F_L = B I l \sin \alpha \dots\dots\dots(2.1)$$

Dengan α = sudut antara medan magnet B dan arus I .

Bila desain kontruksi diatur sedemikian rupa, maka gaya *Lorentz* yang timbul dapat dimanfaatkan untuk menghasilkan gerakan rotasi yang kontinyu.

Konstruksi dasar untuk sebuah motor DC konvensional ditunjukkan pada Gambar 2.19.

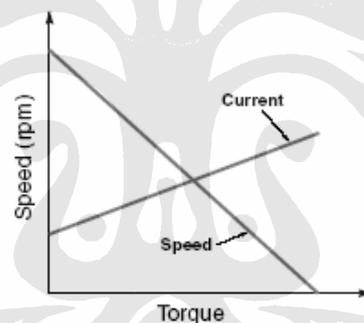


Gambar 2.19 Kontruksi Dasar Motor DC

Arus yang mengalir pada simpal kawat akan melewati medan magnet sehingga menimbulkan gaya *Lorentz*. Pada titik A gaya ini akan mengarah ke atas, sementara pada titik B gaya akan mengarah ke bawah (karena arah arus yang berbeda). Karena adanya gaya ini, maka kawat beserta komutatornya akan berputar ke arah kanan (Gambar 2.19a). Setelah kawat berputar 90° maka sekarang titik A yang akan mengalami gaya ke bawah, dan titik B mengalami gaya ke atas sehingga kawat akan berputar lagi ke arah kanan (Gambar 2.19b) dan sistem membentuk putaran lengkap 360° . Selama arus tetap mengalir, hal ini akan terjadi berulang-ulang sehingga menghasilkan gerakan rotasi yang kontinyu.

Pada dasarnya pada motor terdapat bagian yang bergerak (*rotor*) serta bagian yang tidak bergerak (*stator*). Pada gambar 2.20 magnet permanen merupakan bagian *stator*, sementara simpal kawat yang berputar merupakan bagian *rotor*. Dalam prakteknya bagian *rotor* tidak hanya terdiri dari satu buah simpal/*loop*, tapi disusun dari beberapa buah *loop*. Dengan konstruksi *multi loop* seperti ini maka total gaya yang timbul akan lebih besar, sehingga menghasilkan torsi yang lebih besar.

Karena fluks magnet pada motor DC tipe magnet permanen bersifat konstan, maka motor DC tipe ini memiliki kurva kecepatan terhadap torsi yang bersifat linier. Hal ini sangat menguntungkan ditinjau dari sisi kendali, karena akan mempermudah proses kendali serta komputasi yang diperlukan pada sistem kontrol *digital*. Gambar 2.20 merupakan Kurva Kecepatan-Torsi pada Motor DC Magnet Permanen



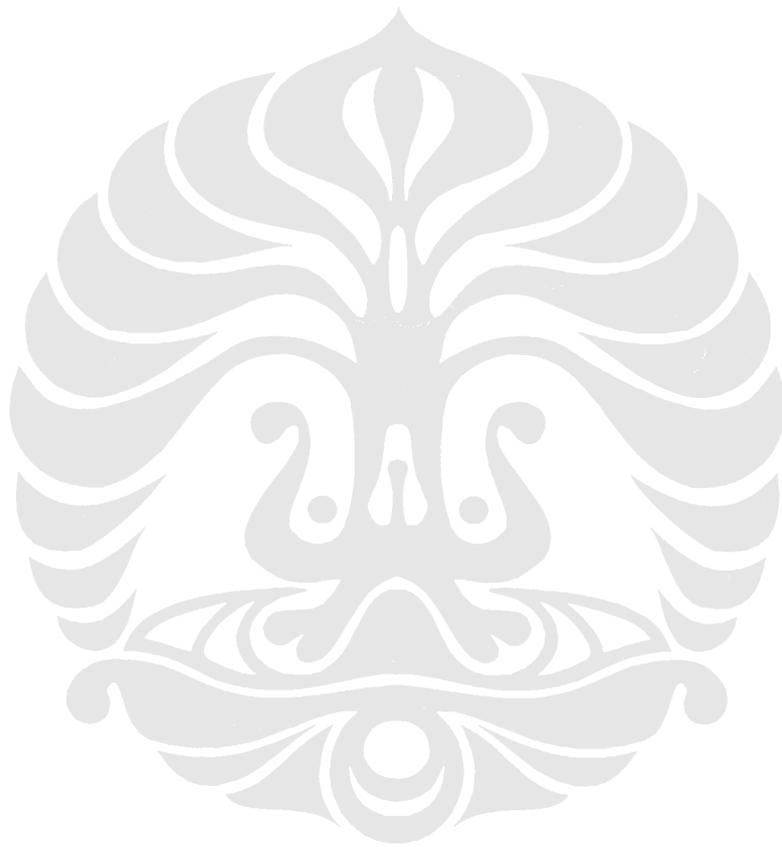
Gambar 2.20 Kurva Kecepatan-Torsi pada Motor DC Magnet Permanen

Sesuai dengan namanya, motor DC adalah sebuah motor yang diberi sumber tegangan DC (*Direct Current*) atau arus searah. Untuk itu, jika ingin merubah putaran motor ini dapat dilakukan dengan mengubah polaritas tegangan yang diberikan.

Motor ini memiliki tegangan kerja yang bervariasi mulai dari 3 Volt, 6 Volt, 12 Volt hingga 24 Volt. Jika motor DC diberi tegangan yang lebih rendah dari tegangan kerjanya maka kinerja motor ini tidak akan maksimal. Hal ini akan terlihat dari putaran motor yang lambat. Sebaliknya jika motor ini diberi tegangan yang lebih besar dari tegangan kerjanya maka motor ini akan berputar sangat cepat hingga mencapai putaran maksimumnya. Jika melewati tegangan

maksimum maka motor ini akan rusak karena kumparan yang ada dalam motor akan terbakar.

Untuk mengatur kecepatan dan torsi yang dihasilkan oleh motor selain mengubah tegangan juga dapat dilakukan dengan cara memasang susunan roda gigi (*gear*) pada motor tersebut.

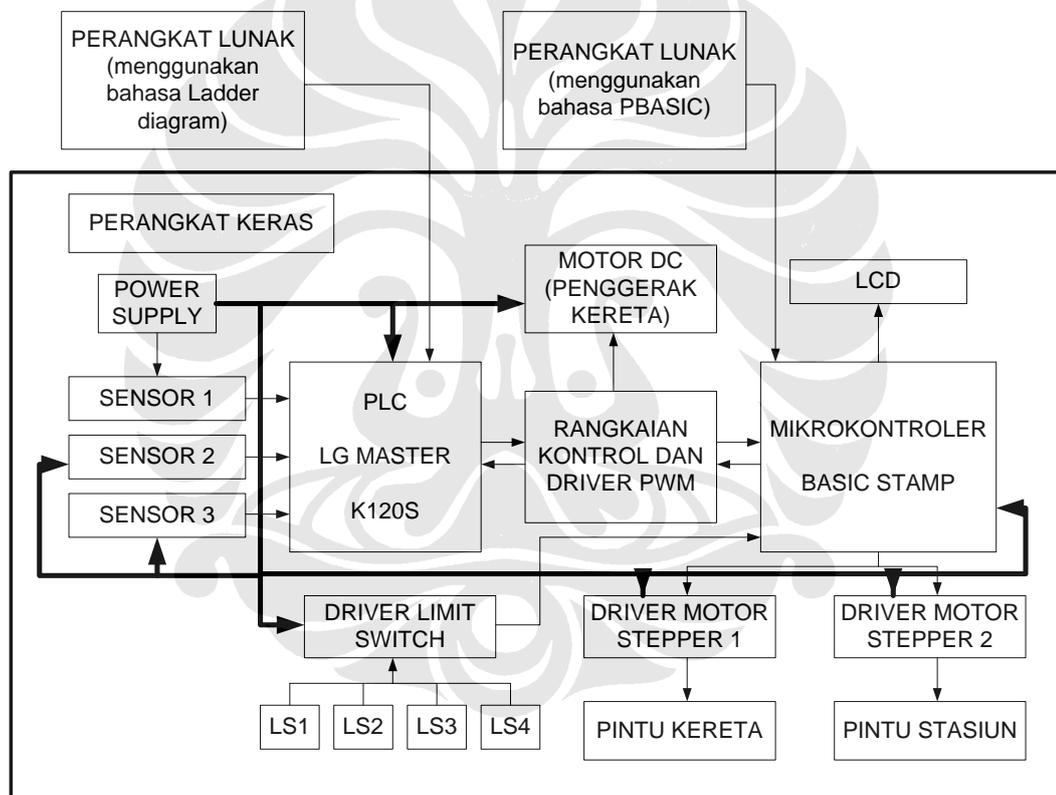


BAB 3

RANCANG BANGUN

3.1 SISTEM SECARA UMUM

Pada bab ini akan dibahas rancang bangun dari perangkat keras dan perangkat lunak sistem pemberhentian kereta listrik, buka-tutup pintu kereta dan pintu stasiun dengan menggunakan PLC LG Master K-120S dan Mikrokontroler *BASIC STAMP*. Untuk blok diagram secara umum, dapat dilihat pada Gambar 3.1.



Gambar 3.1 Blok Diagram Sistem

3.1.1 Cara Kerja Sistem

Cara kerja sistem secara umum berdasarkan pada blok diagram Gambar 3.1 adalah sebagai berikut:

1. Pada saat pertama mikrokontroler *BASIC STAMP* akan melakukan inisialisasi kondisi awal. Apabila keadaan awal pintu kereta dan pintu stasiun terbuka, maka kedua pintu tersebut akan tertutup sampai terkena limit switch 2 dan 4. LCD digunakan untuk menampilkan proses kerja yang sedang dilakukan mikrokontroler.
2. Kereta pada pertama kali harus berada pada stasiun 1 (terkena sensor 1). Apabila kereta tidak berada pada stasiun 1 maka PLC akan memberikan sinyal rendah ke mikrokontroler untuk menjalankan kereta sampai pada stasiun 1.
3. Pada saat sampai di stasiun 1 (terkena sensor 1 dan PLC mengirimkan sinyal tinggi ke mikrokontroler), PWM aktif, pintu kereta akan terbuka sampai terkena LS1 kemudian dilanjutkan dengan pintu stasiun sampai terkena LS3. Setelah 10 detik, pintu stasiun akan tertutup kembali sampai terkena LS4 kemudian dilanjutkan dengan pintu kereta sampai terkena LS2. Setelah itu mikrokontroler mengirimkan sinyal ke PLC untuk memberitahukan bahwa seluruh pintu sudah tertutup dan kereta harus berangkat ke stasiun berikutnya. PLC memberikan sinyal rendah ke mikrokontroler agar menjalankan kereta sampai menuju stasiun 2 (sampai terkena sensor 2).
4. Pada saat sampai di stasiun 2 (terkena sensor 2 dan PLC mengirimkan sinyal tinggi ke mikrokontroler), PWM aktif, pintu kereta akan terbuka sampai terkena LS1 kemudian dilanjutkan dengan pintu stasiun sampai terkena LS3. Setelah 10 detik, pintu stasiun akan tertutup kembali sampai terkena LS4 kemudian dilanjutkan dengan pintu kereta sampai terkena LS2. Setelah itu mikrokontroler mengirimkan sinyal ke PLC untuk memberitahukan bahwa seluruh pintu sudah tertutup dan kereta harus berangkat ke stasiun berikutnya. PLC memberikan sinyal rendah ke mikrokontroler agar menjalankan kereta sampai menuju stasiun 3 (sampai terkena sensor 3).

5. Pada saat sampai stasiun 3 (terkena sensor 3 dan PLC mengirimkan sinyal tinggi ke mikrokontroler), maka hal yang terjadi adalah sama seperti yang dialami oleh stasiun 1 dan stasiun 2. Begitu seterusnya sampai kembali lagi ke stasiun 1.

3.1.2 Spesifikasi Sistem

Spesifikasi sistem pada tugas akhir ini adalah sebagai berikut :

1. *Power supply* yang digunakan adalah 24V, 12V, 9V, dan 5V
2. Seluruh sensor menggunakan *photo-transistor* yang dikombinasikan dengan rangkaian komparator. *Output* dari rangkaian ini dimanfaatkan untuk memberikan *input* ke PLC
3. PLC yang digunakan adalah PLC LG Master K-120S yang mempunyai 30 I/O
4. Mikrokontroler menggunakan *BASIC STAMP BS2P40*
5. LCD yang digunakan adalah LCD 2 X 16.
6. Penggerak Kereta (Motor DC) menggunakan motor DC 5V
7. *Driver motor stepper* yang digunakan pada pintu kereta dan pintu stasiun adalah ULN2003.
8. Pintu kereta dan Pintu Stasiun menggunakan *motor stepper* sebagai penggerakannya.
9. *Driver PWM* menggunakan komponen regulator 7805 dan mosfet IRF2305
10. *Driver kontrol* digunakan untuk komunikasi antara PLC dan mikrokontroler. PLC mengirimkan logik rendah atau tinggi sedangkan mikro memberikan supply 24 V ke PLC
11. Miniatur kereta menggunakan 2 jalur listrik pada rel-relnya. Jalur – jalur listrik ini berfungsi sebagai penggerak kereta. Pada jalur listrik bagian dalam dihubungkan dengan *ground* sedangkan bagian luar dihubungkan dengan tegangan 5V yang berasal dari rangkaian *driver PWM*.

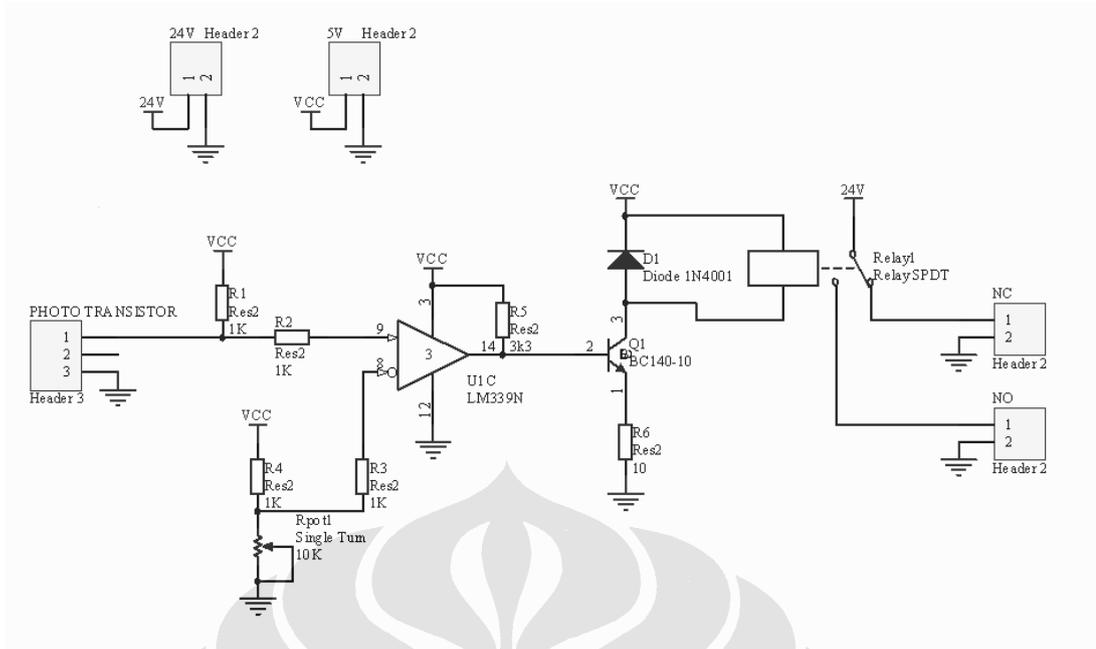
3.2 RANCANG BANGUN PERANGKAT KERAS

Perangkat keras sistem pemberhentian kereta listrik, buka-tutup pintu kereta dan pintu stasiun dengan menggunakan PLC LG Master K-120S dan Mikrokontroler *BASIC STAMP* terbagi menjadi delapan buah subsistem. Spesifikasi dari subsistem – subsistem tersebut adalah sebagai berikut :

3.2.1 Sensor dan Rangkaian Komparator

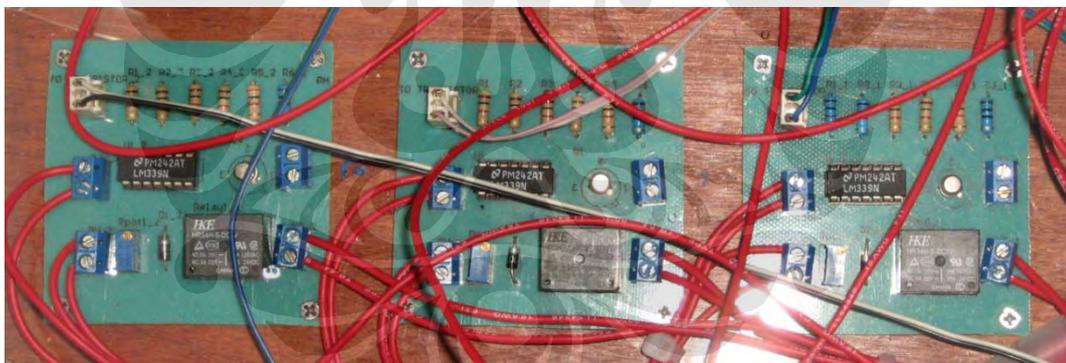
Pada Gambar 3.2 dapat dilihat bahwa apabila photo transistor tidak terkena cahaya maka tegangan pada *photo-transistor* akan tinggi hampir mendekati VCC dan masuk ke dalam kaki positif pada komparator. Kemudian nilai positif yang tinggi akan dibandingkan dengan tegangan masukan dari kaki negatif dari komparator. *Output* dari komparator menghasilkan nilai positif, maka tegangan *output* komparator sama dengan tegangan VCC. *Output* dari komparator akan mengaktifkan transistor, sehingga arus akan mengalir dari kolektor ke emiter yang menyebabkan relay menjadi *Normally Open*.

Begitu juga sebaliknya, apabila photo transistor terkena cahaya maka tegangan pada *photo-transistor* akan rendah dan masuk ke dalam kaki positif pada komparator. Kemudian nilai positif yang rendah dibandingkan dengan tegangan masukan dari kaki negatif dari komparator. *Output* dari komparator menghasilkan nilai negatif, maka tegangan *output* komparator sama dengan tegangan *Ground*. *Output* tersebut tidak akan mengaktifkan transistor NPN ini karena nilainya lebih negatif sehingga relay menjadi *Normally Close*.



Gambar 3.2. Sensor dan Rangkaian Komparator

Gambar 3.3 merupakan realisasi dari tiga buah rangkaian sensor dan komparator yang masing – masing dihubungkan dengan PLC melalui P08, P09 dan P0A.

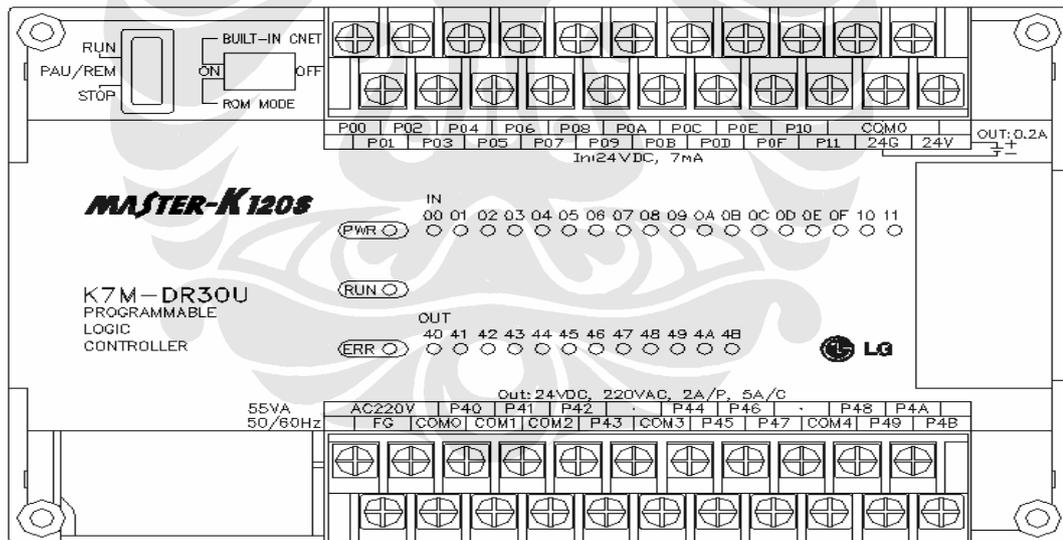


Gambar 3.3 Realisasi Rangkaian Sensor dan Komparator

3.2.2 Programmable Logic Control

Programmable Logic Control (PLC) disini digunakan sebagai *input* dari sensor 1, 2, dan 3 pada masing-masing stasiun dan *input* yang berasal dari mikrokontroller. Keluaran dari sensor 1 dan rangkaian komparator masuk kedalam *input* PLC yaitu P08. Begitu juga pada sensor 2 dan 3, masing-masing masuk kedalam *input* PLC yaitu P09 dan P0A. Untuk *input* yang diberikan dari mikrokontroller terlebih dahulu masuk ke dalam rangkaian kontrol. *Output* dari rangkaian ini masuk ke dalam *input* PLC yaitu P0B. Sedangkan com pada *input* plc diberikan *ground*.

Output dari PLC yaitu P41 dan P42 digunakan untuk mengirimkan sinyal ke mikrokontroller tetapi terlebih dahulu masuk ke dalam rangkaian kontrol. Pada Gambar 3.4 dapat dilihat lebih jelas *port-port* yang tersedia di dalam PLC LG Master-K 120S yaitu terdiri dari 18 *input* dan 12 *output* dan Gambar 3.5 merupakan realisasi dari bentuk PLC LG Master-K 120S.



Gambar 3.4 Contoh bentuk fisik PLC “LG Master-K 120S 30 I/O”

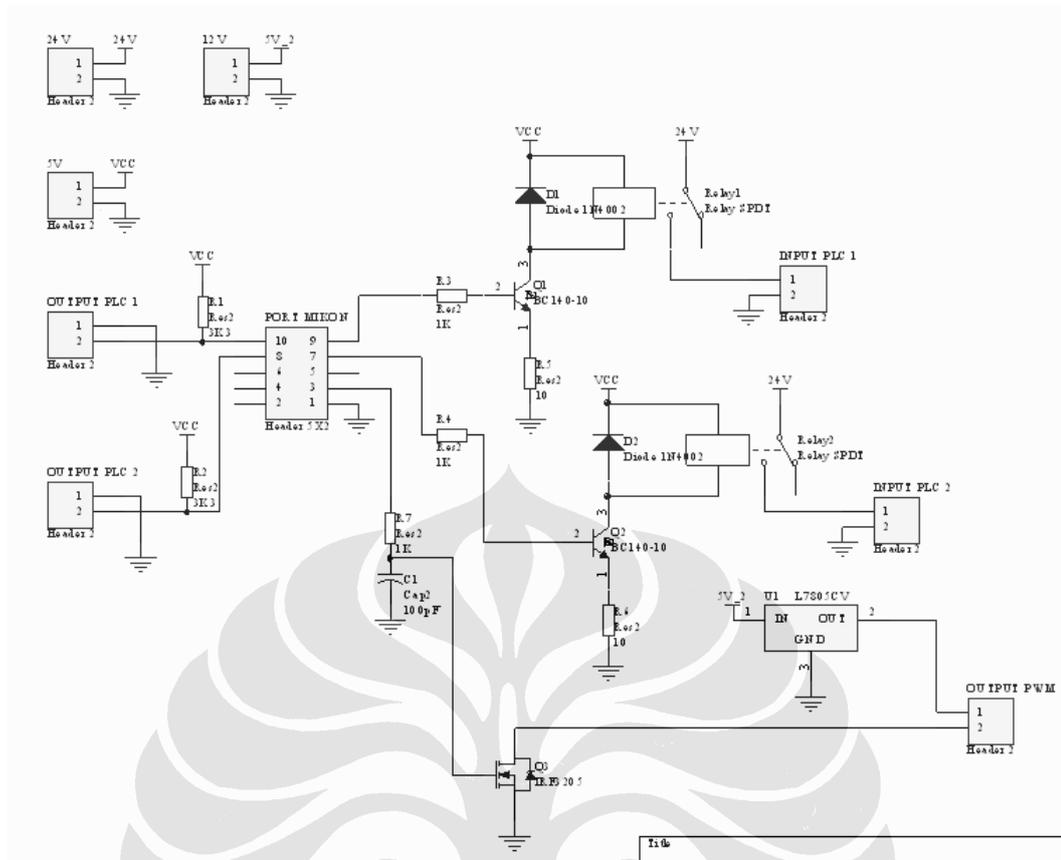


Gambar 3.5 Realisasi bentuk fisik PLC LG Master-K 120S

3.2.3 Rangkaian Kontrol dan *Driver* PWM

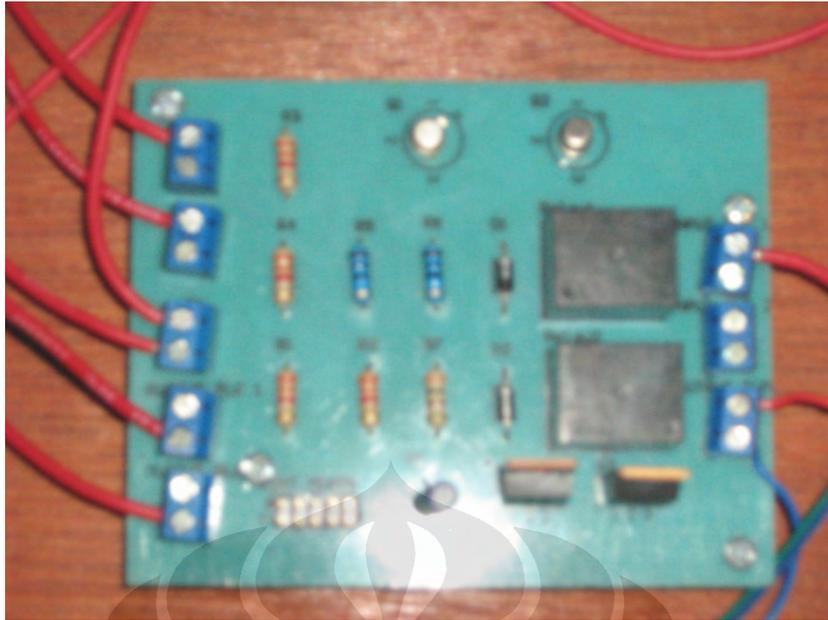
Rangkaian kontrol digunakan untuk menghubungkan PLC dan mikrokontroler. *Output* dari PLC masuk ke dalam rangkaian kontrol agar *output* tersebut dapat digunakan sebagai *input* pada mikrokontroler. Pada rangkaian ini terdapat dua buah *output* PLC yang dapat masuk. *Output* PLC yang masuk adalah P41 dan P42. Apabila *output* PLC adalah *Normally Open* maka rangkaian kontrol akan memberikan logik “0” ke mikrokontroler. Begitu juga sebaliknya apabila *output* PLC adalah *Normally Close* maka rangkaian kontrol akan memberikan logik “1” ke mikrokontroler

Selain itu, rangkaian ini dapat memberikan *input* ke PLC apabila *output* dari mikrokontroler diberikan logik 1. *Output* dari mikrokontroler ini akan mengaktifkan transistor NPN sehingga arus akan mengalir dari kolektor ke emiter yang menyebabkan relay menjadi *Normally Open* yang siap digunakan sebagai *input* PLC. Pada rangkaian ini terdapat dua buah *input* yang dapat masuk ke dalam PLC. Untuk lebih jelasnya, dapat dilihat pada Gambar 3.6.



Gambar 3.6 Rangkaian Kontrol

Pada AUXIO *port* 0 dari mikrokontroler terhubung dengan rangkaian *driver* PWM. Rangkaian ini digunakan untuk mengatur kecepatan kereta listrik pada saat pemberhentian. Komponen yang digunakan adalah mosfet IRF3205. Hal ini disebabkan mosfet memiliki kecepatan pensaklaran yang sangat tinggi. Untuk tegangan motor DC (penggerak kereta) digunakan tegangan 9V yang diregulasikan menjadi 5V dengan komponen regulator 7805. Gambar 3.7 merupakan realisasi dari rangkaian kontrol dan *driver* PWM

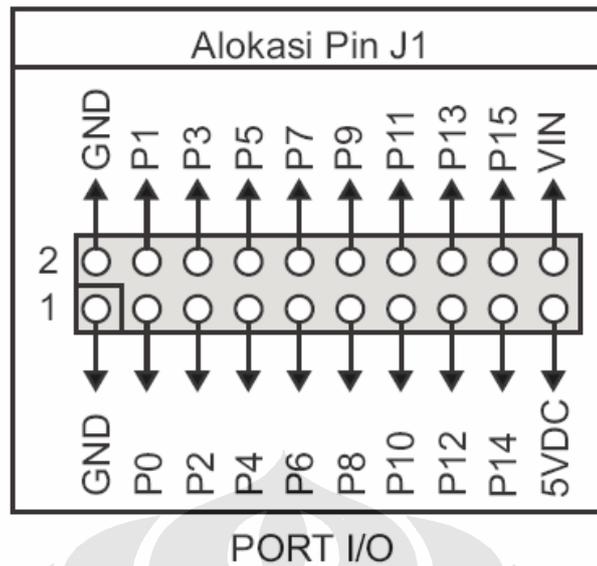


Gambar 3.7 Realisasi Rangkaian Kontrol dan *Driver* PWM

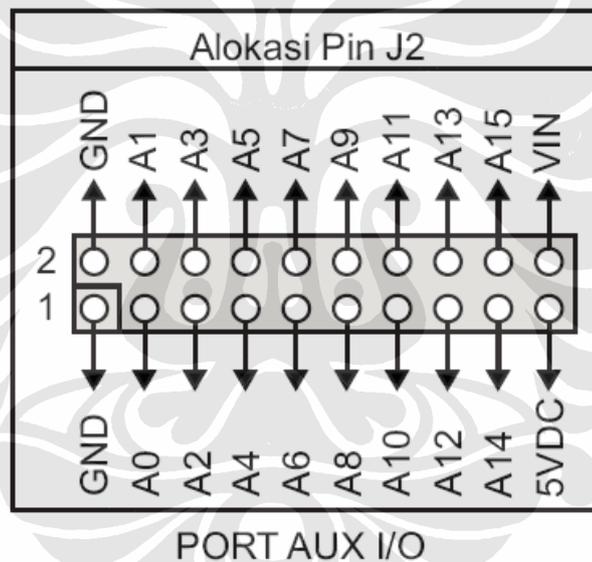
3.2.4 Mikrokontroler *BASIC STAMP*

Mikrokontroler *BASIC STAMP* digunakan untuk menampilkan tampilan menggunakan LCD pada AUXIO port 8 – port 15, memberikan sequence ke *driver motor stepper* pada MAINIO port 0 – port 7, menerima *input* dari *driver* Limit Switch pada MAINIO port 8 – port 11, dan terhubung dengan rangkaian kontrol pada AUXIO port 4 – port 7 dan *driver* PWM pada AUXIO port 0.

Pada Gambar 3.8 dan 3.9 merupakan alokasi pin-pin dari mikrokontroler. Masing-masing pin terhubung dengan yang telah disebutkan sebelumnya. Pin – pin akan menghasilkan *output* logik “0” atau logik “1” ataupun *input* yang diberikan logik “0” atau logok “1” tergantung dari program yang dibuat.

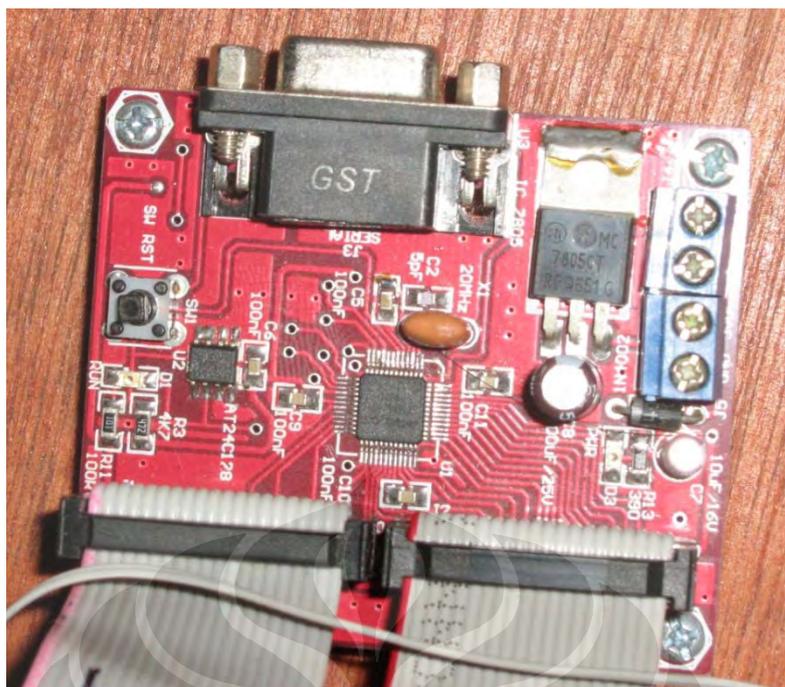


Gambar 3.8 Alokasi Pin J1 *Port MAINIO*

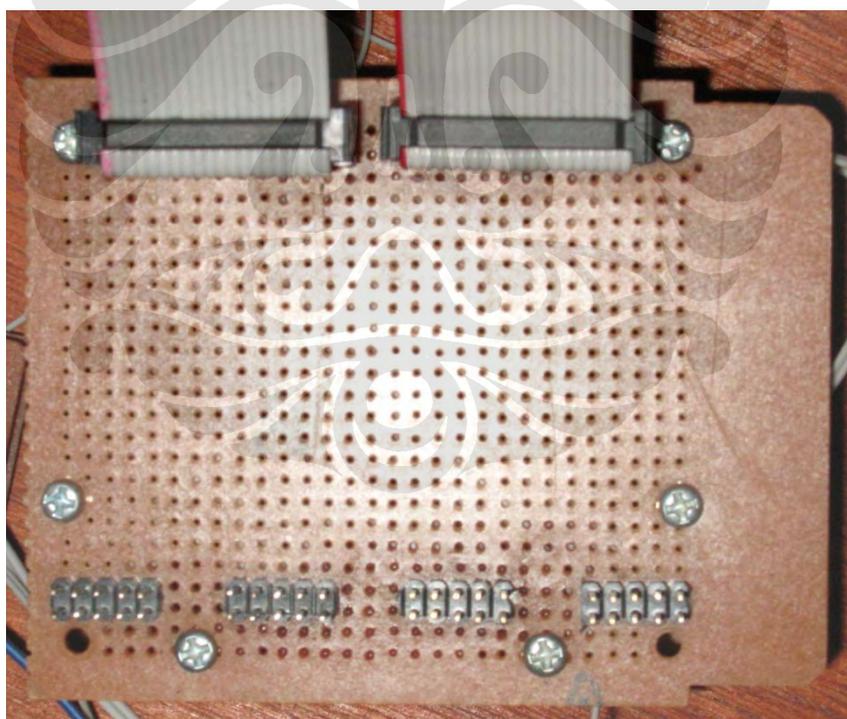


Gambar 3.9 Alokasi Pin J2 *Port AUXIO*

Gambar 3.10 merupakan realisasi dari mikrokontroler *basic stamp* yang terdiri dari 36 I/O. *Port* pertama disebut dengan MAINIO terdiri dari 20 pin dan yang kedua disebut dengan AUXIO terdiri dari 20 pin. Pada Gambar 3.11 merupakan rangkaian pemisah pin sehingga MAINIO dibagi menjadi 2 bagian yaitu MAINIO *port* 0 – *port* 7 dan MAINIO *port* 8 – *port* 15 sedangkan AUXIO dibagi menjadi 2 bagian yaitu AUXIO *port* 0 - *port* 7 dan AUXIO *port* 8 – *port* 15.



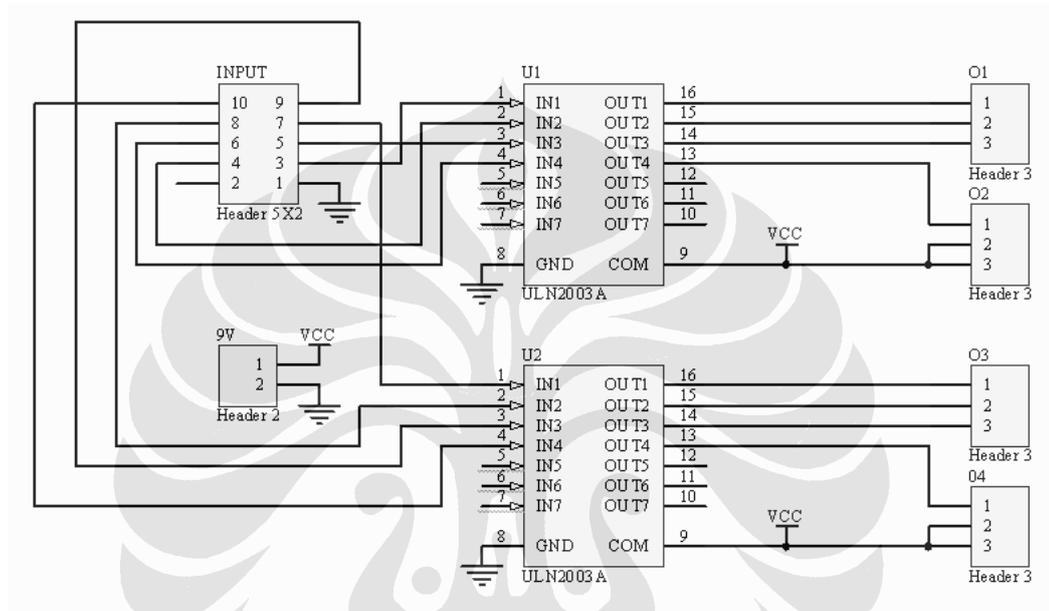
Gambar 3.10 Realisasi Mikrokontroler *Basic stamp*



Gambar 3.11 Realisasi dari Rangkaian Pembagi Pin

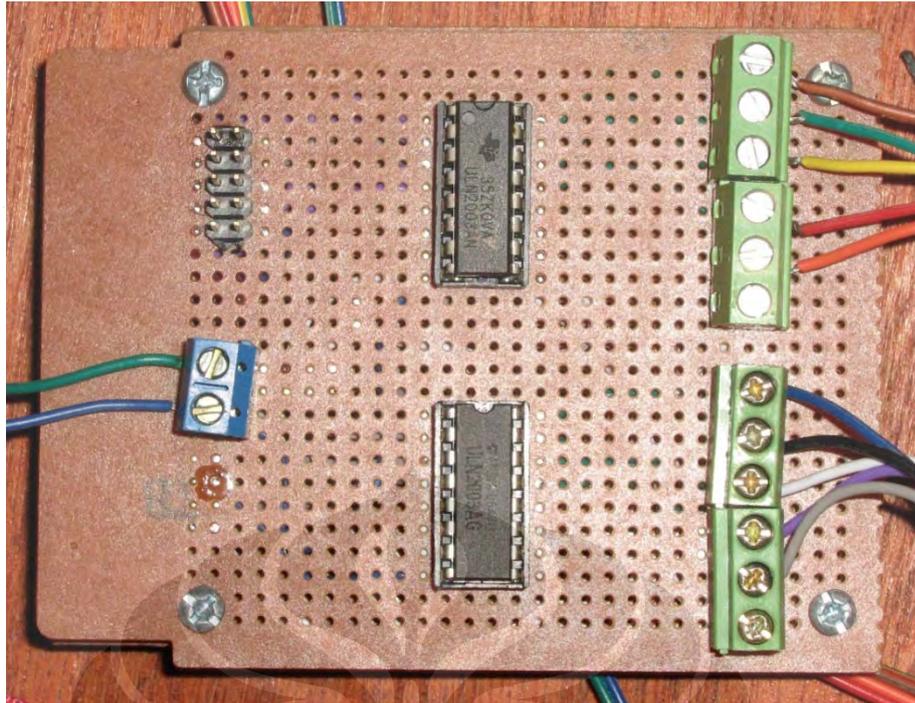
3.2.5 Driver Motor stepper

Dapat dilihat pada Gambar 3.12, *output* pertama yang dihasilkan mikrokontroler pada MAINIO *port* 0 – *port* 3 masuk ke dalam IC ULN2003 secara berurutan. Begitu juga pada MAINIO *port* 4 – *port* 7. IC disini digunakan untuk mendrive arus agar dapat memberikan pulsa-pulsa elektrik yang dapat menggerakkan *motor stepper*.



Gambar 3.12. Rangkaian *Driver Motor stepper*

Dalam menggerakkan *motor stepper*, pulsa-pulsa yang digunakan harus diperhatikan agar tidak terjadi kesalahan dalam melakukan putaran. Pada tabel III.1 dan tabel III.2 akan dijelaskan *sequence* yang harus diberikan dalam menggerakkan *motor stepper*. Gambar 3.13 merupakan realisasi dari rangkaian *driver motor stepper* yang berfungsi untuk menggerakkan buka tutup pintu kereta dan stasiun.



Gambar 3.13 Realisasi Rangkaian *Driver Motor stepper*

Tabel III.1 *Full Step Mode*

A	B	C	D	Komentar
1	0	0	0	<i>Take a step clock wise</i>
0	1	0	0	<i>another step clock wise</i>
0	0	1	0	<i>another step clock wise</i>
0	0	0	1	<i>another step clock wise</i>
0	0	0	1	<i>No step take</i>
0	0	1	0	<i>Take a step back</i>

Tabel 3.2 *Half Step Mode*

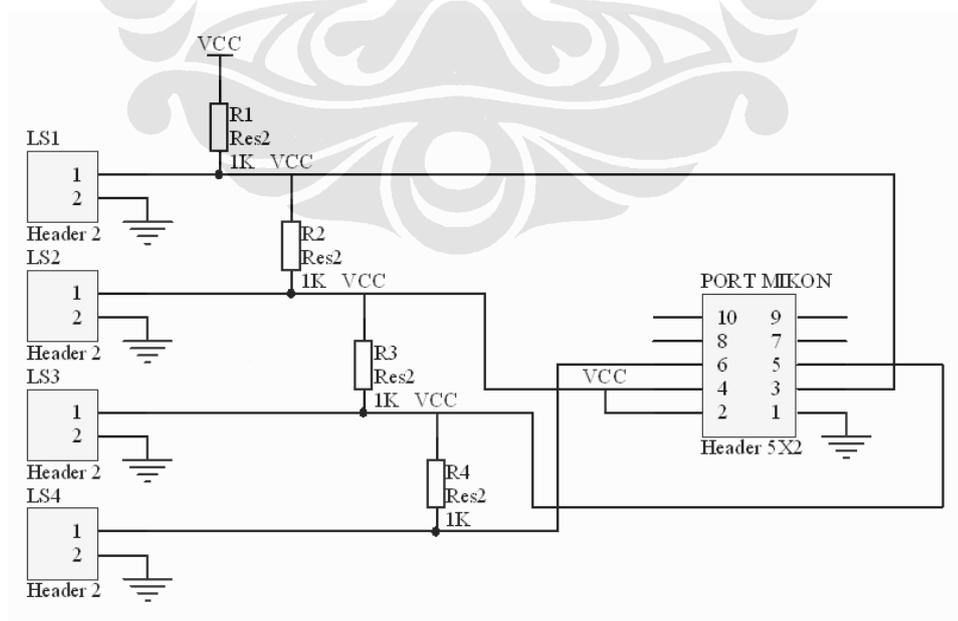
A	B	C	D	Komentar
1	0	0	0	<i>Take A Step Clock Wise</i>
1	1	0	0	<i>Half a step clock wise</i>
0	1	0	0	<i>The complete full step clock wise</i>
0	1	1	0	<i>another half step clock wise</i>
0	0	1	0	<i>The complete full step clock wise</i>

0	0	1	1	<i>another half step clock wise</i>
0	0	0	1	<i>The complete full step clock wise</i>
1	0	0	1	<i>another half step clock wise</i>
1	0	0	0	<i>Start Position</i>

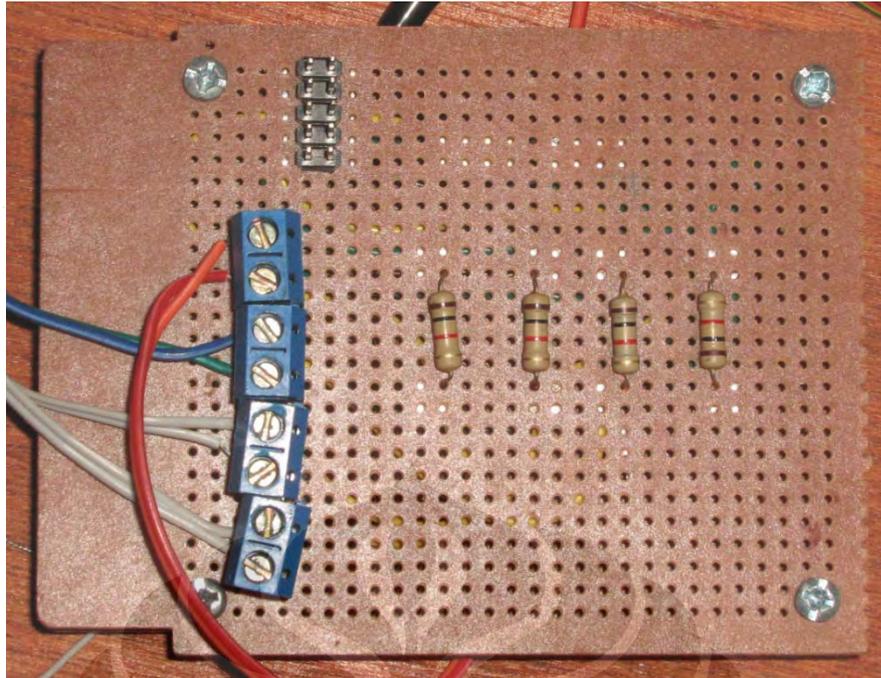
3.2.6 Driver Limit Switch

Driver limit switch digunakan untuk memberikan *input* kepada mikrokontroler agar mikrokontroler memberikan perintah kepada *motor stepper* berhenti. *Limit switch* pertama digunakan untuk memberhentikan pintu kereta pada saat terbuka, sedangkan *limit switch* kedua digunakan untuk memberhentikan pintu kereta pada saat tertutup. *Limit switch* ketiga digunakan untuk memberhentikan pintu stasiun pada saat terbuka, sedangkan *limit switch* keempat digunakan untuk memberhentikan pintu stasiun pada saat tertutup.

Pada Gambar 3.14 dapat dilihat rangkaian *driver limit switch* dengan menggunakan komponen resistor. Apabila *limit switch* tertekan maka *driver* mengirimkan logik “0” ke mikrokontroler. Sebaliknya apabila tidak tertekan, maka *driver* mengirimkan logik “1” ke mikrokontroler. Gambar 3.15 merupakan realisasi dari rangkaian *driver limit switch*.



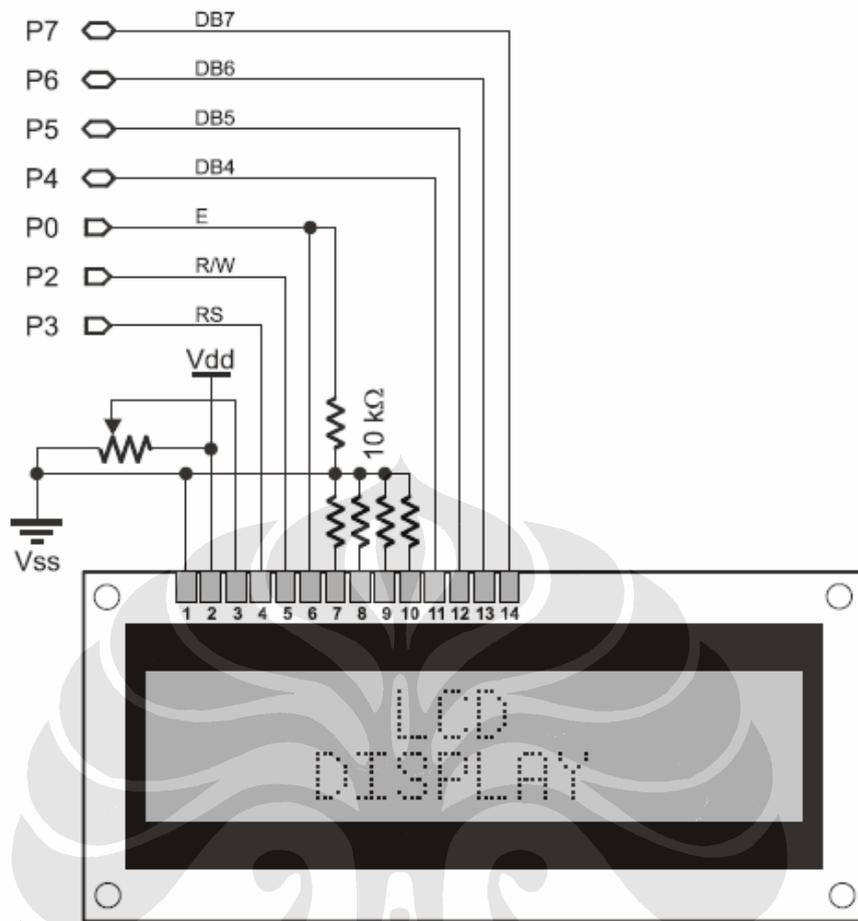
Gambar 3.14 *Driver Limit Switch*



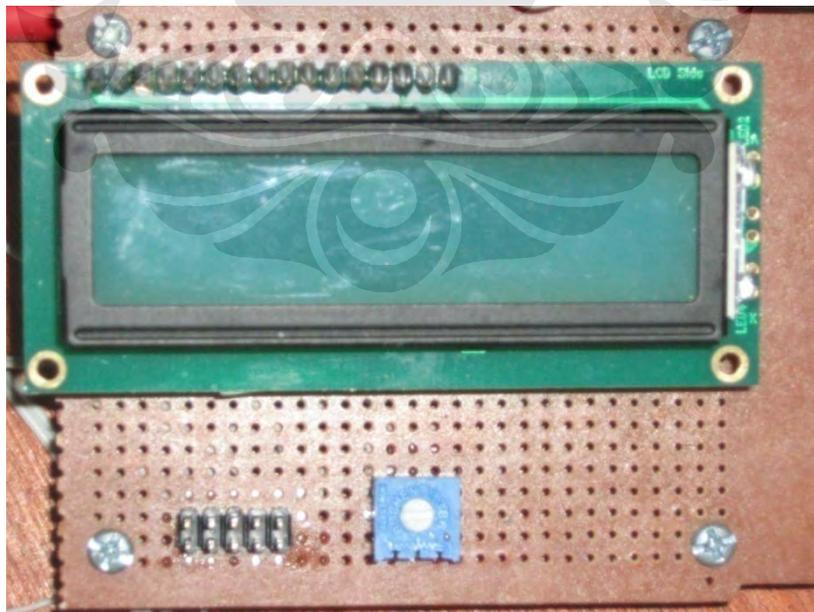
Gambar 3.15 Realisasi Rangkaian *Driver* Limit Switch

3.2.7 LCD

LCD digunakan untuk menampilkan data yang diberikan oleh mikrokontroler melalui program yang dibuat. LCD akan menampilkan judul, nama, dan segala bentuk kerja yang sedang dilakukan oleh mikrokontroler. LCD diberikan *input* melalui AUXIO *port* 8 – *port* 15 seperti pada Gambar 3.16. Gambar 3.17 merupakan realisasi dari rangkaian LCD.



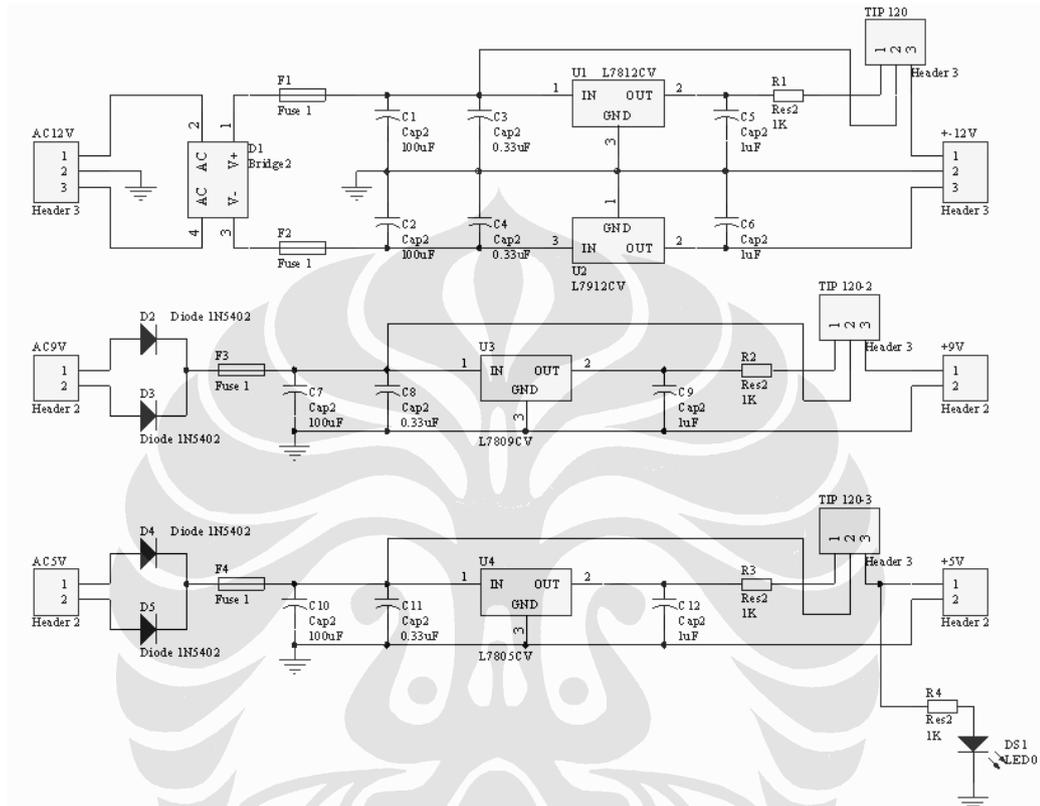
Gambar 3.16 LCD



Gambar 3.17 Realisasi Rangkaian LCD

3.2.8 Power Supply

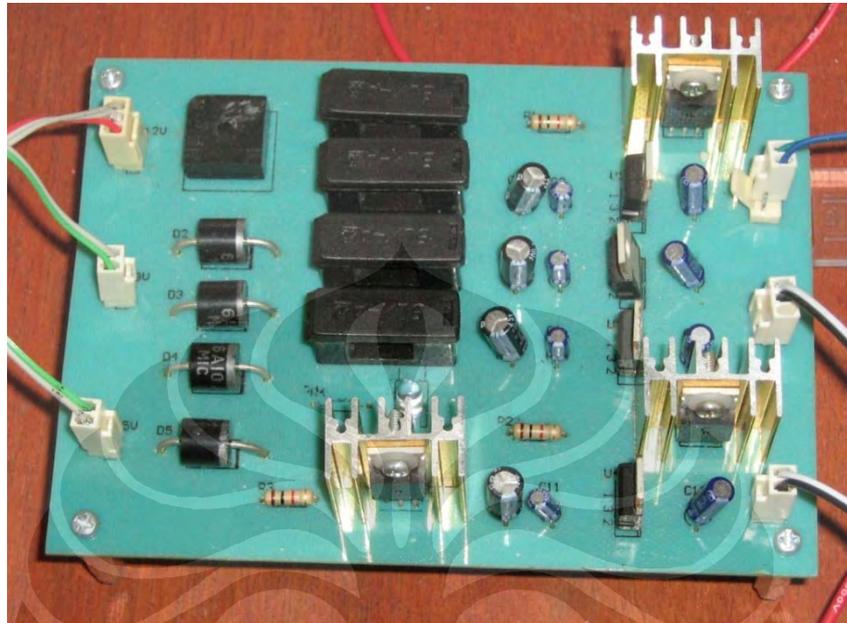
Power supply yang digunakan adalah *power supply* DC dengan tegangan 24V, 12V, 9V, dan 5V. Tegangan Supply 24V diambil langsung dari PLC, sedangkan tegangan supply 12V, 9V, dan 5V diambil dari rangkaian *power supply*. Gambar 3.18 merupakan gambar rangkaian *power supply* yang dirancang.



Gambar 3.18 Rangkaian *Power supply*

Untuk menghasilkan *power supply* +12V, *output* 12V, CT, dan 12V dari trafo masuk ke dalam dioda bridge. Dioda ini berfungsi untuk menyearahkan tegangan. *Output* dari V+ diberikan *fuse* agar dapat megamankan rangkaian apabila terjadi kelebihan beban. Tegangan V+ masuk ke regulator 7812 untuk diregulasi agar *output* dari *power supply* benar-benar 12V. *Output* dari regulator masuk ke dalam basis transistor NPN TIP 120. TIP 120 digunakan untuk memperkuat arus sampai 3A. Begitu juga pada +9V dan +5V, masing-masing menggunakan dua buah dioda untuk menyearahkan tegangan. Kemudian *output* dari dioda-dioda ini diberikan *fuse* dan masuk ke regulator. Untuk menghasilkan tegangan +9V, regulator yang digunakan adalah 7809. Sedangkan untuk

menghasilkan tegangan +5V, regulator yang digunakan adalah 7805. Masing-masing *output* dari regulator diberikan penguat arus yaitu TIP 120. Gambar 3.19 merupakan realisasi dari rangkaian *power supply*.



Gambar 3.19 Realisasi Rangkaian *Power supply*

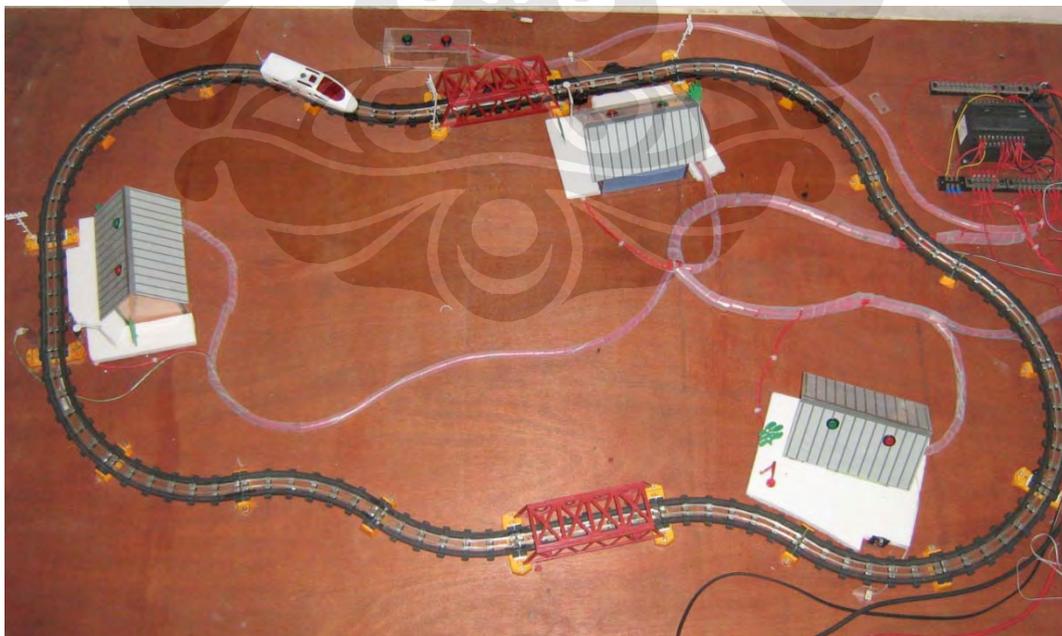
3.2.9 Miniatur Kereta Listrik

Miniatur kereta listrik terbuat dari bahan plastik yang terdiri dari 20 rel dimana masing – masing rel ditempelkan dua jalur listrik. Masing – masing jalur listrik ini terbuat dari 5 buah kawat tembaga yang disatukan dengan menggunakan timah sehingga permukaan kawat menjadi lebar. Dua buah jalur listrik ini ditempelkan dengan menggunakan permukaan papan tembaga yang sebelumnya telah direkatkan pada sekat – sekat rel. Untuk lebih jelasnya dapat dilihat seperti pada Gambar 3.20



Gambar 3.20 Dua Jalur Listrik yang Ditempelkan Pada Rel

Miniatur kereta listrik ditempelkan dengan satu lembar papan tripleks yang telah dipernis sebelumnya. Pada papan tripleks juga ditempelkan perangkat - perangkat keras lainnya yang mendukung kerja dari sistem ini. Pada Gambar 3.21 merupakan keseluruhan dari miniatur kereta listrik.



Gambar 3.21 Miniatur Kereta Listrik

3.2.10 Pintu Kereta dan Stasiun

Pintu kereta dan stasiun dirancang dengan menggunakan kerangka alumunium yang berukuran 40cm x 10 cm x 26 cm. Pintu kereta dan stasiun menggunakan bahan acrylig. Pintu kereta dan stasiun digerakkan oleh *motor stepper* unipolar yang masing – masing ditempelkan pada kerangka pintu kereta dan stasiun. *Motor stepper* ini ditempelkan gear dan belt agar dapat membuka dan menutupnya pintu kereta dan stasiun. Untuk lebih jelasnya dapat dilihat pada Gambar 3.22 untuk pintu kereta dan Gambar 3.23 untuk pintu stasiun.



Gambar 3.22 Realisasi Pintu Kereta



Gambar 3.23 Realisasi Pintu Stasiun

3.3 RANCANG BANGUN PERANGKAT LUNAK

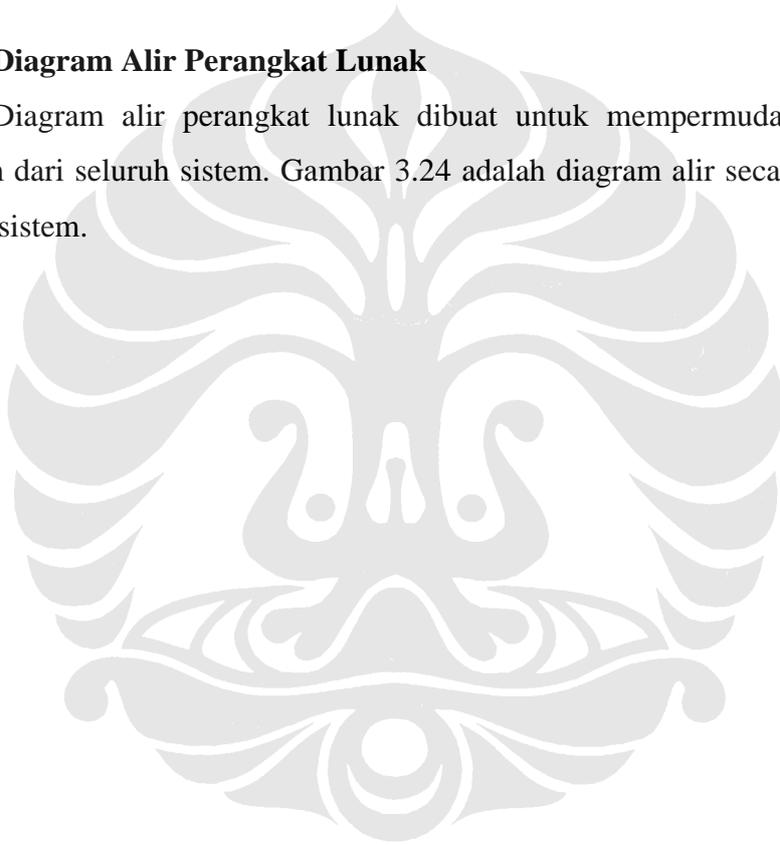
3.3.1 Spesifikasi Perangkat Lunak

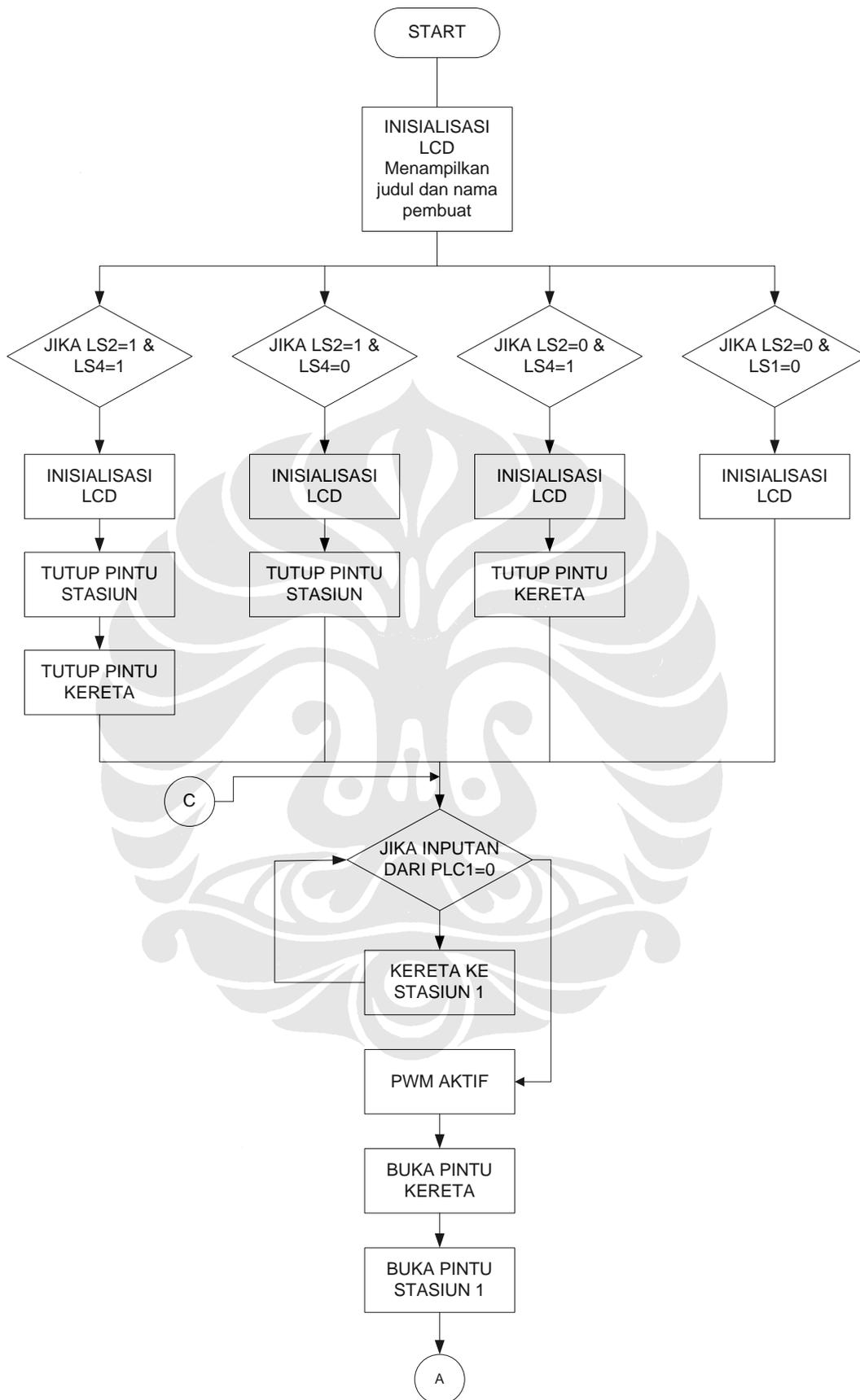
Spesifikasi perangkat lunak yang akan dirancang adalah sebagai berikut:

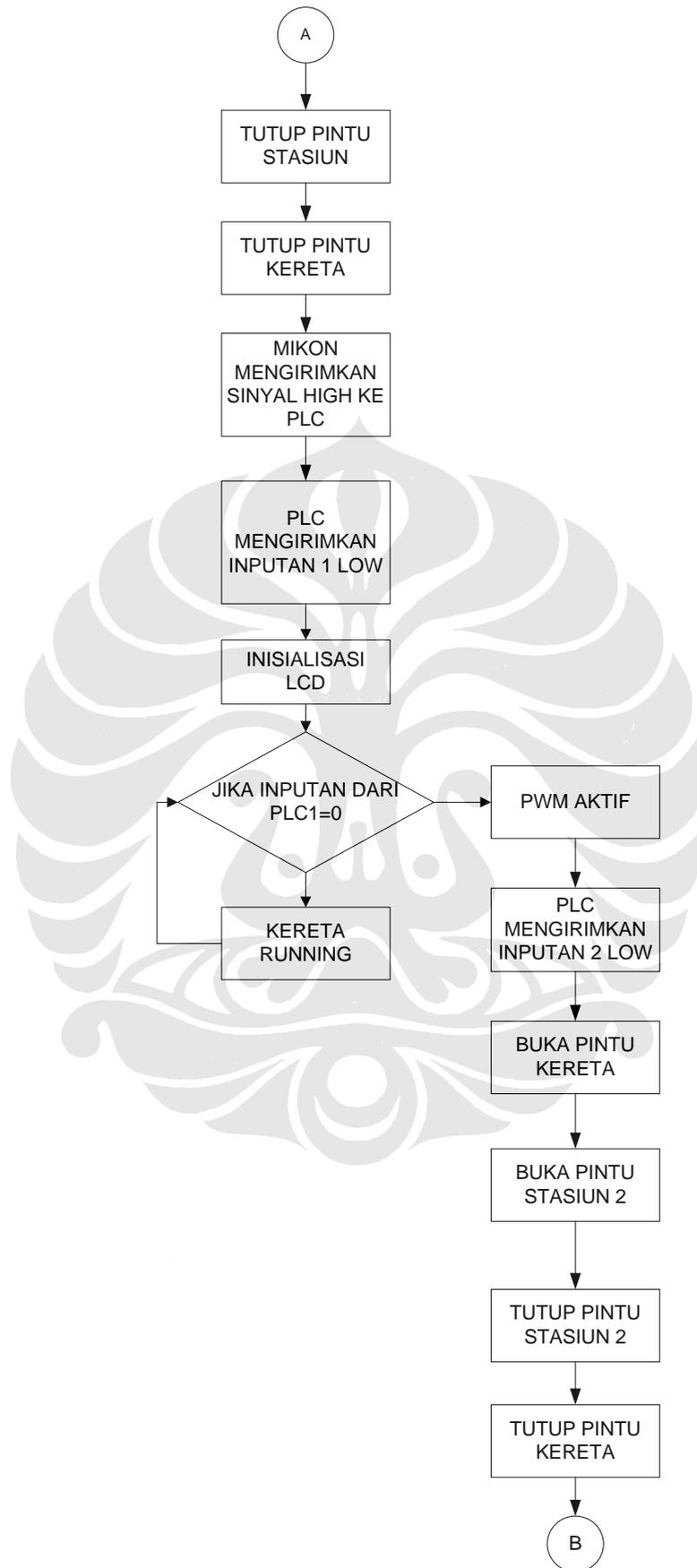
1. Perangkat lunak yang dirancang dibuat dengan bahasa ladder diagram pada PLC dan PBASIC pada *BASIC STAMP*
2. Software yang digunakan untuk meuliskan program adalah KGL_WIN pada PLC LG MASTER-K 120S dan *BASIC STAMP* EDITOR v2.4 pada *BASIC STAMP*

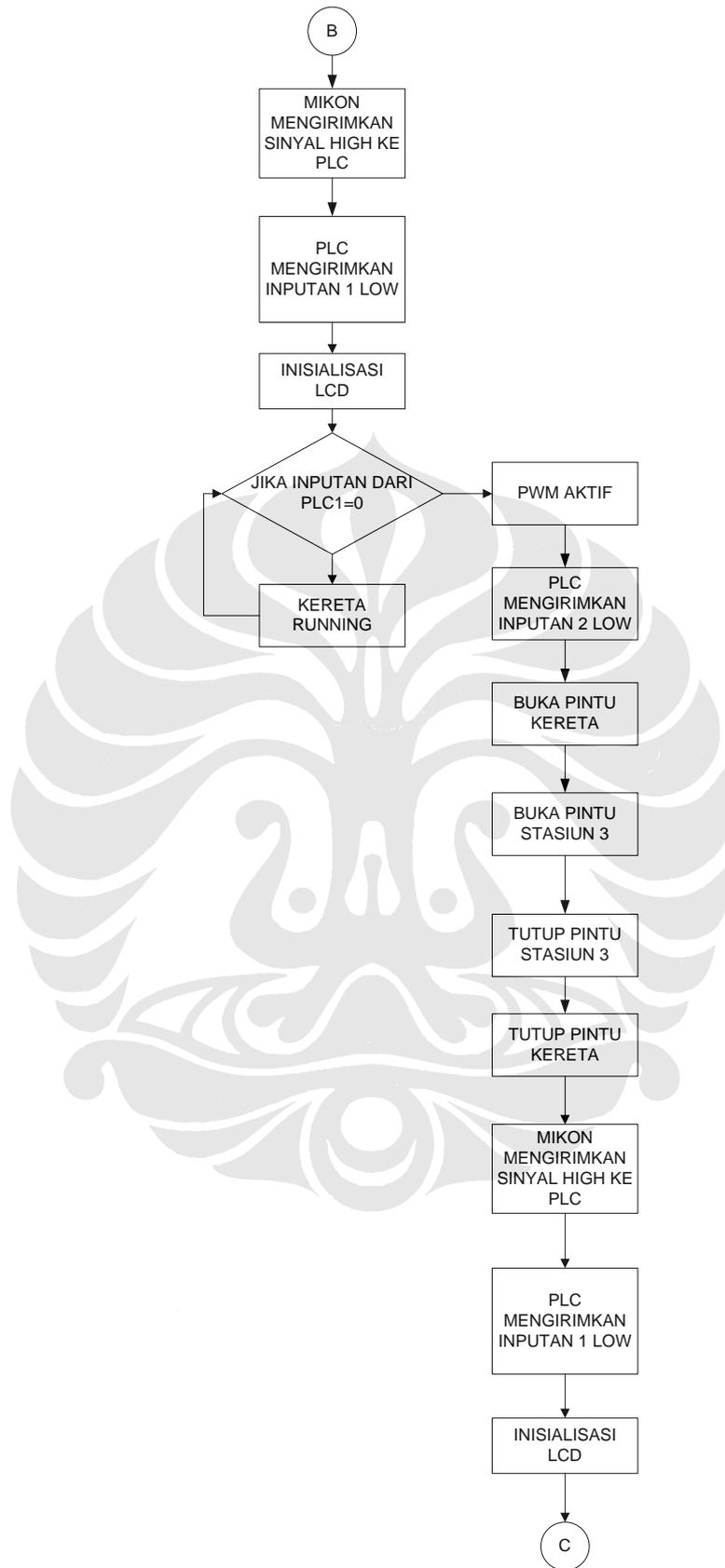
3.3.2 Diagram Alir Perangkat Lunak

Diagram alir perangkat lunak dibuat untuk mempermudah pembuatan program dari seluruh sistem. Gambar 3.24 adalah diagram alir secara umum dari seluruh sistem.









Gambar 3.24 Diagram Alir dari Sistem

Pada saat pertama kali sistem dijalankan maka sistem akan menginisialisasi keadaan awal pintu kereta dan pintu stasiun. Apabila pintu kereta dan pintu stasiun terbuka maka mikrokontroler akan menutup pintu stasiun kemudian dilanjutkan dengan pintu kereta. Begitu juga apabila salah satu pintu masih terbuka maka mikrokontroler akan menutup pintu tersebut. Apabila kedua pintu dalam posisi tertutup maka program akan mengecek keberadaan kereta.

Pada kondisi awal, kereta harus berada pada stasiun 1. Apabila kereta tidak berada pada stasiun 1, maka kereta akan berjalan sampai menemukan stasiun 1. Setelah kereta terkena sensor 1 pada stasiun 1 maka *output* PLC 1 memberikan sinyal high ke mikrokontroler sehingga PWM aktif kemudian *output* PLC 2 mengirimkan sinyal ke mikrokontroler untuk membuka pintu stasiun 1 dan dilanjutkan pintu kereta. Setelah 10 detik, pintu kereta tertutup dan dilanjutkan dengan pintu stasiun 1. Mikrokontroler mengirimkan sinyal ke PLC sebagai tanda bahwa pintu sudah tertutup. *Output* PLC 1 kembali mengirimkan sinyal *LOW* ke mikrokontroler agar kereta berjalan menuju ke stasiun 2.

Pada saat terkena sensor 2 pada stasiun 2 maka *output* maka *output* PLC 1 memberikan sinyal high kembali ke mikrokontroler sehingga PWM aktif kemudian *output* PLC 2 mengirimkan sinyal ke mikrokontroler untuk membuka pintu stasiun 2 dan dilanjutkan pintu kereta. Setelah 10 detik, pintu kereta tertutup dan dilanjutkan dengan pintu stasiun 2. Mikrokontroler mengirimkan kembali sinyal ke PLC sebagai tanda bahwa pintu sudah tertutup. *Output* PLC 1 kembali mengirimkan sinyal *LOW* ke mikrokontroler agar kereta berjalan menuju ke stasiun 3. Hal ini juga sama terjadi pada pemberhentian di stasiun 3. Sistem ini akan berulang terus sampai listrik dimatikan.

BAB 4

UJI COBA DAN ANALISIS

Sebelum melakukan pengujian pada sistem secara keseluruhan, pengujian yang dilakukan terlebih dahulu adalah pengujian terhadap perangkat – perangkat keras yang telah dirancang pada bab sebelumnya

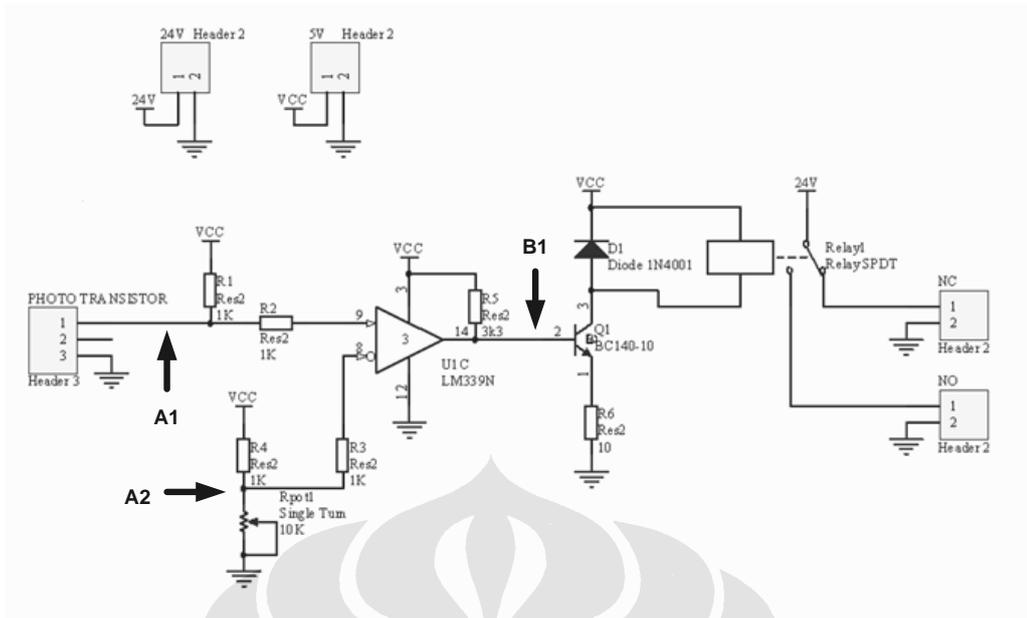
4.1 UJI COBA DAN ANALISIS PERANGKAT KERAS

4.1.1 Uji Coba Sensor dan Rangkaian Komparator

Dapat dilihat pada Gambar 4.1, pengujian yang dilakukan adalah mengukur tegangan pada A1, A2, dan B1. Pada saat *phototransistor* tidak terkena cahaya dari LED maka tegangan yang dihasilkan pada A1 terhadap *ground* adalah 3,75V. Tegangan tersebut masuk kedalam kaki positif dari komparator. Tegangan pada A2 diatur sedemikian rupa menggunakan *multiturn* agar tegangan yang dihasilkan lebih kecil daripada tegangan A1. Tegangan A2 yang diukur adalah 2,6V. Tegangan A2 masuk ke dalam kaki negatif komparator. Komparator akan membandingkan antara tegangan A1 dan A2. Jadi,

$$\text{Output Komparator} = A1 - A2 = 3.75V - 2,6V = 1.15V$$

Output komparator pada perhitungan menghasilkan nilai tegangan yang positif. Pada pengukuran, tegangan B1 adalah 1,2V. Tegangan B1 masuk ke dalam kaki basis transistor sehingga menyebabkan transistor konduksi. Arus akan mengalir dari kolektor ke emiter yang menyebabkan *relay* menjadi kontak. Com dari *relay* diberikan tegangan 24V sehingga tegangan pada *Normally Open* adalah 24V. Tegangan ini digunakan untuk memberikan sinyal tinggi kepada PLC bahwa kereta berada di stasiun karena sensor tidak terkena cahaya (terhalangi oleh kereta). Selain itu, sinyal tinggi ini juga dimanfaatkan untuk menjalankan PWM serta buka – tutup pintu kereta dan stasiun.



Gambar 4.1 Uji Coba Sensor dan Rangkaian Komparator

Begitu juga sebaliknya, pada saat *phototransistor* terkena cahaya dari LED maka tegangan yang dihasilkan pada A1 terhadap *ground* adalah 1,07V. Tegangan tersebut masuk kedalam kaki positif dari komparator. Tegangan pada A2 telah diset sebelumnya. Tegangan A2 masuk ke dalam kaki negatif komparator. Komparator akan membandingkan antara tegangan A1 dan A2. Jadi,

$$\text{Output Komparator} = A1 - A2 = 1,07V - 2,6V = - 1,53 V$$

Output komparator pada perhitungan menghasilkan nilai tegangan yang negatif maka tegangan pada B1 akan mendekati 0V. Tegangan B1 tidak cukup untuk mengkonduksikan transistor NPN ini. Sehingga kontak *relay* akan menuju ke Normally Close. *Output* dari *Normally Open* menjadi tidak memiliki tegangan sehingga PLC diberikan sinyal rendah. Sinyal rendah ini dimanfaatkan untuk menandakan bahwa kereta tidak berada pada stasiun.

Pada sistem pemberhentian dan buka-tutup pintu kereta dan stasiun terdapat 3 buah rangkaian sensor dan komparator. Masing-masing mewakili stasiun 1, 2, dan 3. Pada pengujian, ketiga rangkaian ini bekerja sesuai dengan yang diinginkan.

Pada saat *output* PLC2 (P42) ON maka C2 menjadi 0V sehingga AUXIO *port* 5 (kaki 8) pada mikrokontroler mendapatkan logik “0”. Logik “0” pada *port* ini dimanfaatkan mikrokontroler untuk mengaktifkan pintu kereta dan pintu stasiun.

Output pada mikrokontroler AUXIO dan *port* 6 (kaki9) diberikan logik “1” (tegangan pada D1 sebesar 5V) apabila pintu kereta dan pintu stasiun sudah tertutup. *Output* ini akan mengaktifkan transistor NPN sehingga arus mengalir dari kolektor ke emiter menyebabkan *relay* menjadi kontak. *Output* dari mikrokontroler ini dikirimkan ke *port* PLC P0B. *Output* ini memberitahukan kepada PLC bahwa pintu kereta dan stasiun sudah tertutup. Sehingga PLC mengaktifkan P41 menjadi ON sehingga C1 menjadi 0V kembali. Logik “0” pada *input port* mikrokontroler ini menyebabkan *output* pada AUXIO *port* 0 (kaki3) menjadi logik “1” sehingga tegangan pada D3 menjadi 5V. Tegangan ini mengaktifkan kembali Mosfet sehingga kereta berjalan kembali menuju ke stasiun berikutnya. Rangkaian ini bekerja sesuai dengan yang diinginkan.

4.1.3 Driver Motor stepper

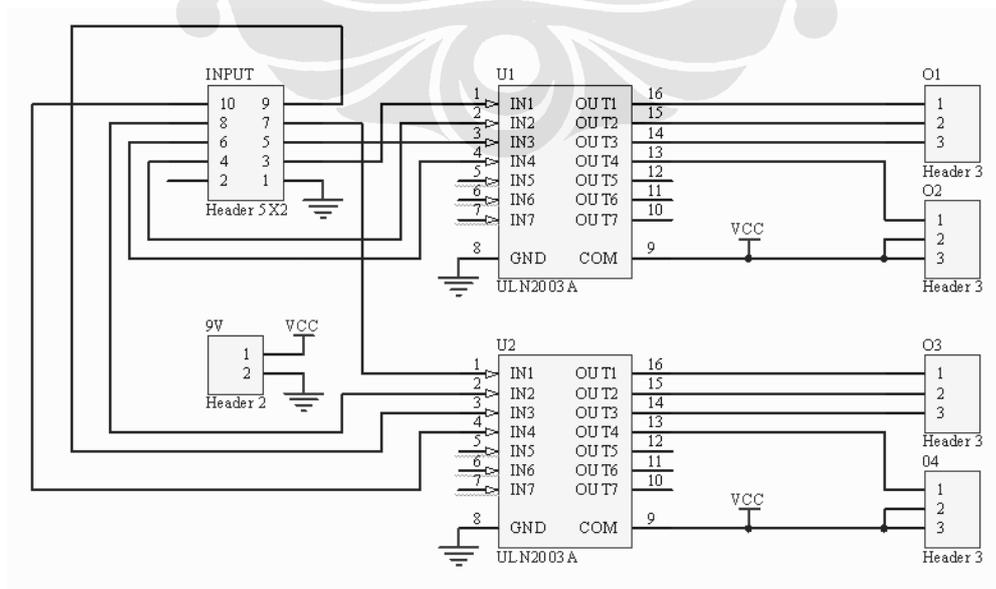
Uji coba *driver motor stepper* dilakukan dengan memberikan pulsa-pulsa elektrik ke MAINIO *port* 0 – *port* 3 (kaki 3 – 6) digunakan untuk menggerakkan *motor stepper* buka – tutup pintu stasiun dan MAINIO *port* 4 – *port* 7 (kaki 7 – 10) digunakan untuk menggerakkan *motor stepper* buka – tutup pintu kereta. Sebagai pengujian *driver*, diambil satu contoh program yaitu program untuk membuka pintu kereta.

```

DO
MAINIO
OUTB=%1001
PAUSE 3
OUTB=%0101
PAUSE 3
OUTB=%0110
PAUSE 3
OUTB=%1010
PAUSE 3
BUTTON LIMIT3,0,255,255,a,1,END3
LOOP

```

OUTB pada program memberikan *output* pada mikrokontroler untuk memberikan logik-logik kepada *input* dari *driver*. Misal OUTB % 1001 maka *output* MAINIO port 4 berlogik “1”, port 5 berlogik “0”, port 6 berlogik 0, port 7 berlogik “1”. Logik-logik ini digunakan untuk menjalankan *motor stepper* menyebabkan pintu kereta terbuka sampai terkena *Limit switch* 3. Pada Gambar 4.3 dapat dilihat *port – port* yang masuk ke dalam *driver motor stepper*. *Motor stepper* yang digunakan untuk menggerakkan pintu kereta dan stasiun bergerak lambat karena berukuran kecil dengan catu daya 5VDC. Rangkaian ini bekerja sesuai dengan yang diinginkan.

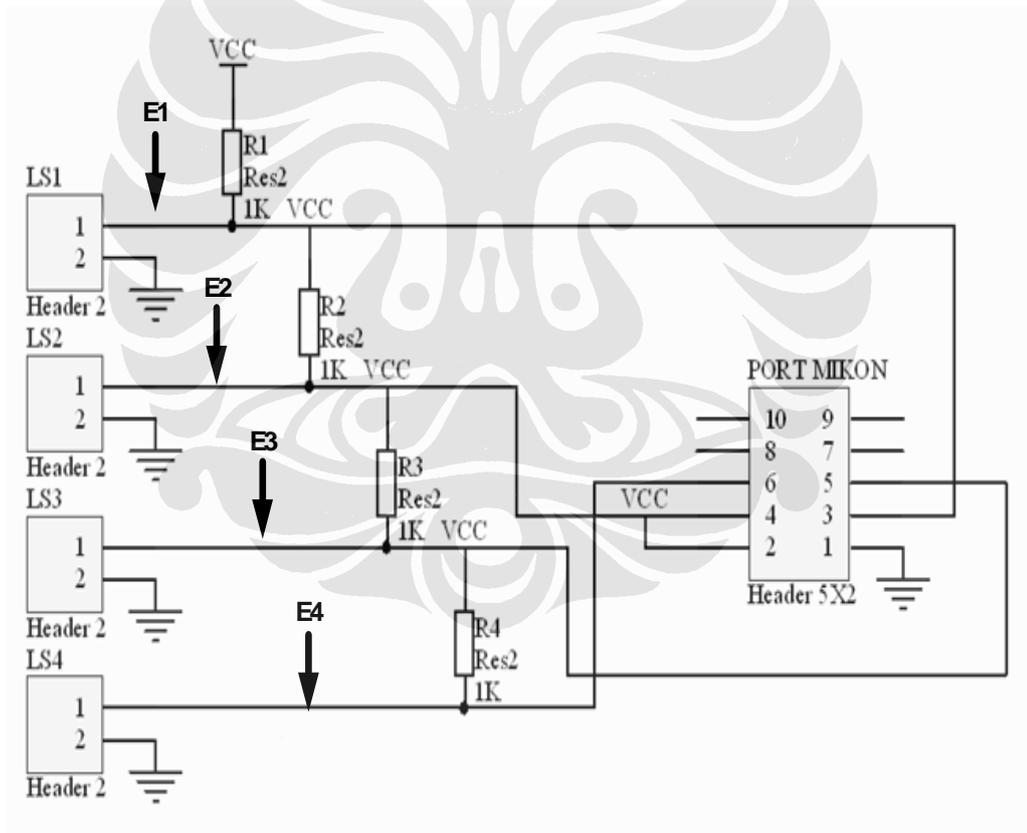


Gambar 4.3 Uji Coba *Driver Motor stepper*

4.1.4 Driver Limit switch

Pada Gambar 4.4 uji coba dilakukan dengan mengukur tegangan pada E1, E2, E3, dan E4. Apabila LS1 ON (terkena penghalang pada pintu) maka tegangan pada E1 adalah 0V sehingga MAINIO port 8 (kaki 3) menerima *input* berlogik “0”. Logik pada *port* ini digunakan untuk memberhentikan *motor stepper* pada saat pintu stasiun terbuka.

Begitu juga pada *Limit switch* lainnya, apabila LS2, LS3, dan LS4 ON maka tegangan pada E2, E3, E4 adalah 0V sehingga MAINIO port 9, 10, 11, dan 12 (kaki 4, 5, dan 6) menerima *input* berlogik “0”. Logik pada *port – port* ini masing – masing digunakan secara berurutan untuk memberhentikan *motor stepper* pada saat pintu stasiun tertutup, pintu kereta terbuka, pintu kereta tertutup. Rangkaian ini bekerja sesuai dengan yang diinginkan.



Gambar 4.4 Uji Coba Driver Limit switch

4.2 UJI COBA SISTEM

Uji coba sistem dilakukan dengan beberapa kondisi awal. Kondisi-kondisi tersebut yaitu:

4.2.1 Kondisi Kereta Berada Pada Stasiun 1, Pintu Kereta dan Stasiun Tertutup

Pada kondisi awal ini, maka pertama kali yang dilakukan adalah mikrokontroler akan mengirimkan sinyal ke *port* PLC P0B yang menyatakan bahwa kedua pintu pada keadaan tertutup. Kemudian sinyal tersebut digunakan PLC sebagai *input* (mengaktifkan M01) sehingga PLC mengirimkan logik “1” yang menyatakan bahwa kereta berada pada stasiun 1 melalui *port* P41 dan juga mengirimkan logik “0” melalui *port* P42 yang digunakan oleh mikrokontroler untuk membuka pintu kereta dan stasiun. Pertama, pintu kereta terbuka sampai terkena *limit switch 1* dan dilanjutkan dengan pintu stasiun terbuka sampai terkena *limit switch 3*. Setelah kedua pintu terbuka selama 10 detik maka pintu stasiun kembali tertutup sampai terkena *limit switch 4* dan dilanjutkan dengan pintu kereta tertutup sampai terkena *limit switch 2*. Kemudian, mikrokontroler akan mengirimkan sinyal ke *port* PLC P0B yang menyatakan bahwa kedua pintu sudah tertutup sehingga PLC mengirimkan logik “0” ke mikrokontroler melalui *port* P41 agar mikrokontroler menjalankan kereta berhenti sampai stasiun 2. Ketika kereta terkena sensor 2 pada stasiun 2 maka PWM pada *port* mikrokontroler aktif sehingga menyebabkan kereta berhenti secara halus. *Port* PLC P42 mengirimkan logik “0” ke mikrokontroler sehingga mikrokontroler membuka pintu kereta kemudian dilanjutkan dengan pintu stasiun. Setelah 10 detik, pintu stasiun kembali tertutup dan dilanjutkan dengan pintu kereta tertutup. Mikrokontroler mengirimkan sinyal ke *port* PLC P0B sehingga mengaktifkan P41 yang menyebabkan PLC kembali mengirimkan logik “0” ke mikrokontroler agar mikrokontroler menjalankan kereta ke stasiun 3. Sama halnya seperti pada stasiun 2, ketika kereta terkena sensor 3 maka *port* PLC P41 mengirimkan logik “1” sehingga PWM aktif, dan *port* PLC P42 mengirimkan logik “0” menyebabkan pintu kereta terbuka dan dilanjutkan dengan pintu stasiun. Setelah 10 detik, pintu stasiun tertutup dan dilanjutkan dengan pintu kereta. Mikrokontroler mengirimkan

sinyal ke *port* PLC P0B menandakan bahwa kedua pintu sudah tertutup sehingga *port* PLC P41 mengirimkan logik “0” kembali ke mikrokontroler sehingga mikrokontroler menjalankan kereta kembali ke stasiun 1. Hal ini akan terus berulang sampai mikrokontroler dinonaktifkan.

4.2.2 Kondisi Kereta Berada Pada Stasiun 1, Pintu Kereta dan Stasiun Terbuka

Pada kondisi awal ini, maka pertama kali yang dilakukan adalah mikrokontroler akan menutup pintu stasiun dan dilanjutkan dengan pintu kereta. Setelah keduanya tertutup maka mikrokontroler mengirimkan sinyal ke PLC yang menandakan bahwa kedua pintu sudah tertutup. Kemudian mikrokontroler akan memeriksa keberadaan kereta dengan memeriksa masukan yang diberikan dari *port* PLC P41 logik “0” atau “1”. Karena kereta berada pada stasiun 1 maka pintu stasiun kembali terbuka dan dilanjutkan dengan pintu kereta. Setelah 10 detik, maka pintu stasiun tertutup dan dilanjutkan dengan pintu kereta. Kemudian mikrokontroler mengirimkan sinyal ke *port* PLC P0B yang menandakan bahwa kedua pintu sudah tertutup dan PLC kembali mengirimkan logik “0” melalui *port* P41 ke mikrokontroler sehingga mikrokontroler menjalankan kereta ke stasiun 2. Kemudian hal selanjutnya akan sama terjadi seperti pada kondisi pertama di atas.

4.2.3 Kondisi Kereta Berada Pada Stasiun 2, Pintu Kereta dan Stasiun Tertutup

Pada kondisi awal ini, maka pertama kali yang dilakukan adalah mikrokontroler akan mengirimkan sinyal ke *port* PLC P0B yang menyatakan bahwa kedua pintu pada keadaan tertutup. Karena kereta tidak terkena sensor 1 pada stasiun 1, maka *port* PLC P41 mengirimkan logik “0” ke mikrokontroler agar mikrokontroler menjalankan kereta sampai pada stasiun 1 melewati stasiun 3. Pada stasiun 3, kereta tidak akan berhenti walaupun terkena sensor 3. Setelah terkena sensor pada sensor 1 pada stasiun 1 maka PWM aktif dan PLC mengirimkan logik “0” melalui *port* P42 agar mikrokontroler membuka pintu kereta dan dilanjutkan dengan pintu stasiun. Setelah 10 detik, maka pintu stasiun tertutup dan dilanjutkan dengan pintu kereta. Mikrokontroler mengirimkan sinyal

ke PLC bahwa mikrokontroler sudah menutup kedua pintu sehingga PLC mengirimkan kembali logik “0” ke mikrokontroler agar menjalankan kereta ke stasiun 2. Kemudian hal selanjutnya akan sama terjadi seperti pada kondisi pertama di atas.

4.2.4 Kondisi Kereta Berada Pada Stasiun 2, Pintu Kereta dan Stasiun Terbuka

Pada kondisi awal ini, maka pertama kali yang dilakukan adalah mikrokontroler akan menutup pintu stasiun dan dilanjutkan dengan pintu kereta. Setelah keduanya tertutup maka mikrokontroler mengirimkan sinyal ke PLC yang menandakan bahwa kedua pintu sudah tertutup. Setelah itu PLC memeriksa apakah kereta berada pada stasiun 1 atau tidak. Karena sensor 1 pada stasiun 1 tidak terhalangi maka kereta berada di stasiun lainnya sehingga PLC mengirimkan logik “0” melalui *port* P41 ke mikrokontroler agar menjalankan kereta sampai menuju ke stasiun 1 (sampai terkena sensor 1). Ketika terkena sensor 1 pada stasiun 1 maka PWM aktif dan *port* PLC P42 mengirimkan logik “0” ke mikrokontroler agar mikrokontroler membuka pintu kereta dan dilanjutkan dengan pintu stasiun. Setelah 10 detik, pintu stasiun kembali tertutup dan dilanjutkan dengan pintu kereta. Kemudian mikrokontroler mengirimkan sinyal ke *port* PLC P0B yang menandakan bahwa kedua pintu sudah tertutup. PLC akan mengirimkan logik “0” ke mikrokontroler melalui *port* P41 agar mikrokontroler menjalankan kereta menuju stasiun 2. Kemudian hal selanjutnya akan sama terjadi seperti pada kondisi pertama di atas.

4.2.5 Kondisi Kereta Berada Pada Stasiun 1, Pintu Kereta Terbuka dan Stasiun Tertutup

Pada kondisi awal ini, maka pertama kali yang dilakukan adalah mikrokontroler akan menutup pintu kereta dan mengirimkan sinyal ke *port* PLC P0B yang menandakan bahwa pintu kereta sudah tertutup. Setelah itu PLC memeriksa apakah kereta berada pada stasiun 1 atau tidak. Karena sensor 1 terhalangi oleh kereta maka kereta berada pada stasiun 1. Kemudian PLC pada *port* P42 mengirimkan logik “0” ke mikrokontroler agar mikrokontroler membuka

pintu kereta dan dilanjutkan dengan pintu stasiun. Setelah 10 detik, pintu stasiun kembali tertutup dan dilanjutkan dengan pintu kereta. Kemudian mikrokontroler mengirimkan sinyal ke *port* PLC P0B yang menandakan bahwa kedua pintu sudah tertutup. PLC akan mengirimkan logik “0” ke mikrokontroler melalui *port* P41 agar mikrokontroler menjalankan kereta menuju stasiun 2. Kemudian hal selanjutnya akan sama terjadi seperti pada kondisi pertama di atas.

4.2.6 Kondisi Kereta Berada Pada Stasiun 2, Pintu Kereta Terbuka dan Stasiun Tertutup

Pada kondisi awal ini, maka pertama kali yang dilakukan adalah mikrokontroler akan menutup pintu kereta dan mengirimkan sinyal ke *port* PLC P0B yang menandakan bahwa pintu kereta sudah tertutup. Setelah itu PLC memeriksa apakah kereta berada pada stasiun 1 atau tidak.

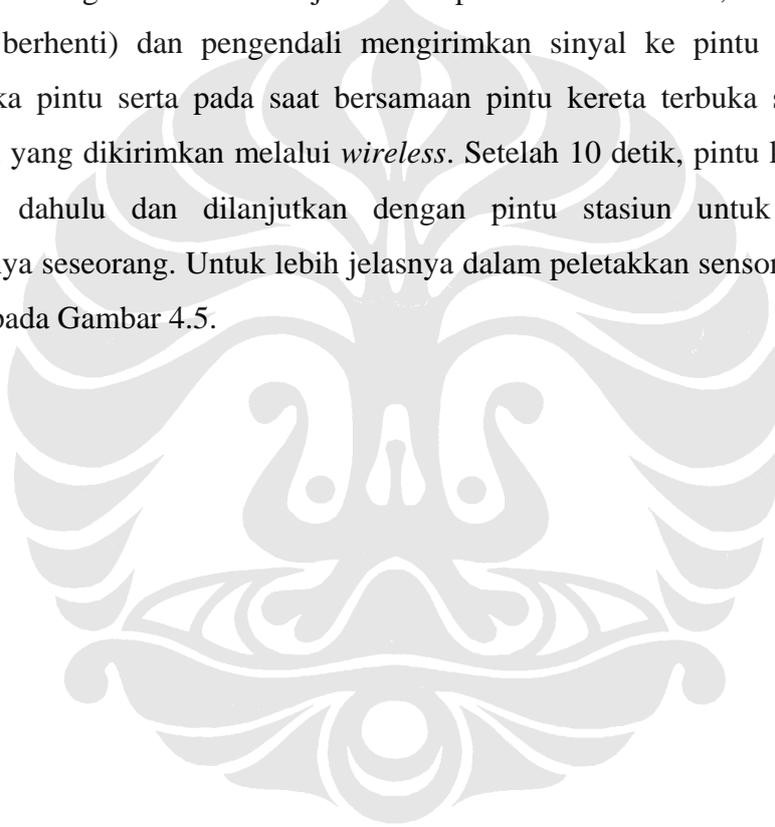
Karena sensor 1 pada stasiun 1 tidak terhalangi maka kereta berada di stasiun lainnya sehingga PLC mengirimkan logik “0” melalui *port* P41 ke mikrokontroler agar menjalankan kereta sampai menuju ke stasiun 1 (sampai terkena sensor 1). Ketika terkena sensor 1 pada stasiun 1 maka PWM aktif dan *port* PLC P42 mengirimkan logik “0” ke mikrokontroler agar mikrokontroler membuka pintu kereta dan dilanjutkan dengan pintu stasiun. Setelah 10 detik, pintu stasiun kembali tertutup dan dilanjutkan dengan pintu kereta. Kemudian mikrokontroler mengirimkan sinyal ke *port* PLC P0B yang menandakan bahwa kedua pintu sudah tertutup. PLC akan mengirimkan logik “0” ke mikrokontroler melalui *port* P41 agar mikrokontroler menjalankan kereta menuju stasiun 2. Kemudian hal selanjutnya akan sama terjadi seperti pada kondisi pertama di atas.

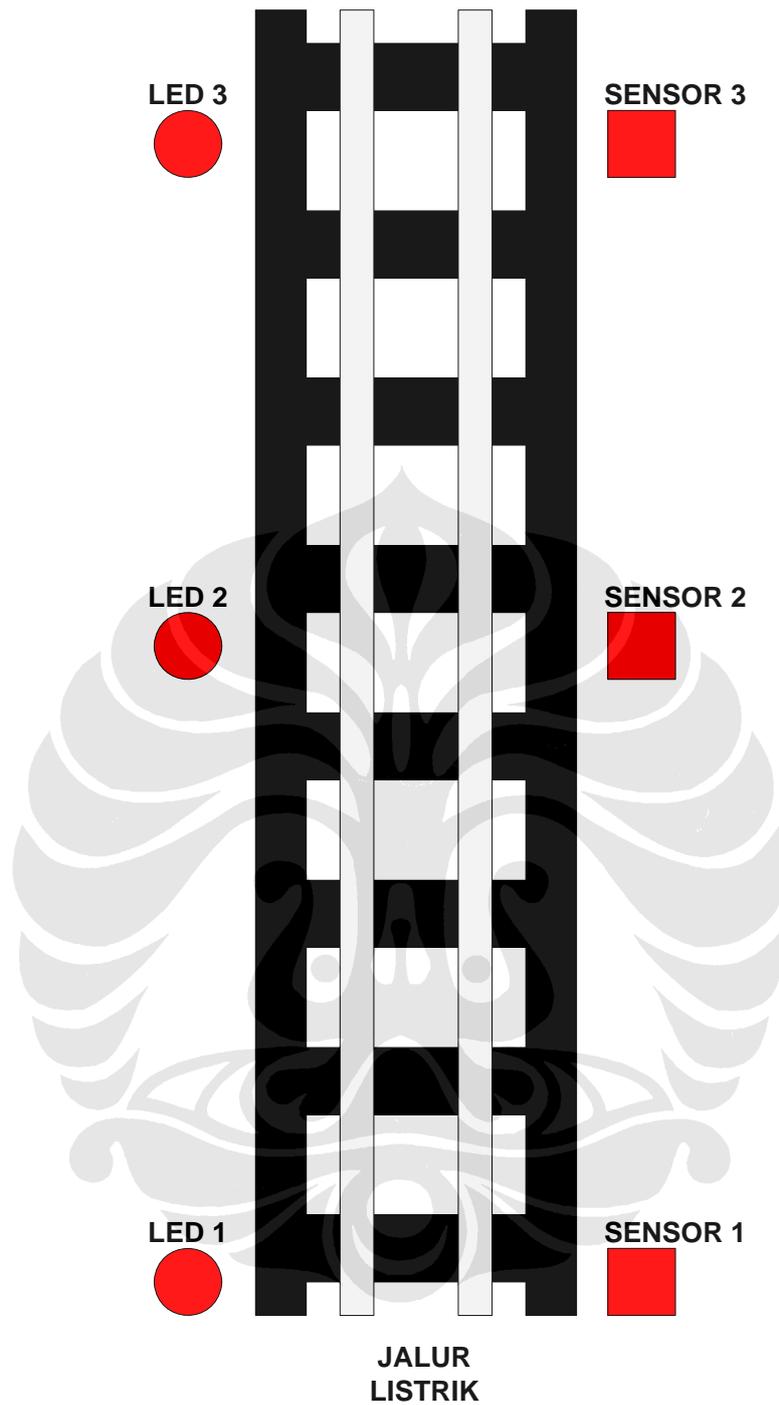
4.3 SKENARIO PADA SISTEM

Di bawah ini merupakan beberapa skenario yang dapat terjadi apabila menginginkan kereta melakukan pemberhentian secara halus dan berhenti tepat pada stasiun, buka pintu kereta dan stasiun secara bersamaan, serta koneksi antara pintu kereta dan pengendali.

4.3.1 Skenario Pada Saat Pemberhentian

Pada saat melakukan pemberhentian, kereta akan melewati sensor 1 yang digunakan untuk mengaktifkan PWM agar mengurangi kecepatan kereta dan mengirimkan sinyal ke pengendali agar mengaktifkan *wireless* yang nantinya digunakan untuk membuka dan menutupnya pintu kereta. Selanjutnya, kereta akan melewati sensor 2 yang digunakan untuk menurunkan kembali kecepatan kereta dan mengirimkan sinyal ke pengendali menandakan kereta akan sampai di stasiun agar pengendali bersiap – siap untuk mengirimkan sinyal ke pintu kereta melalui teknologi *wireless*. Dilanjutkan sampai terkena sensor 3, PWM tidak aktif (kereta berhenti) dan pengendali mengirimkan sinyal ke pintu stasiun untuk membuka pintu serta pada saat bersamaan pintu kereta terbuka sesuai dengan perintah yang dikirimkan melalui *wireless*. Setelah 10 detik, pintu kereta tertutup terlebih dahulu dan dilanjutkan dengan pintu stasiun untuk menghindari terjepitnya seseorang. Untuk lebih jelasnya dalam peletakkan sensor, dapat dilihat seperti pada Gambar 4.5.





Gambar 4.5 Tata Letak Sensor – Sensor

4.3.2 Skenario Pada Saat Pintu Kereta dan Stasiun Terbuka Bersamaan

Mengacu pada Gambar 4.5, pada saat kereta terkena sensor 3 maka kereta akan berhenti dan mengirimkan sinyal ke pengendali. Kemudian pengendali mengirimkan sinyal ke aktuator pintu kereta dan stasiun, pada pengiriman sinyal

pada pintu kereta menggunakan teknologi *wireless* sedangkan pada pintu stasiun hanya menggunakan kabel data. Untuk itu, diperlukan perhitungan waktu (*delay*) antara pengiriman sinyal ke pintu kereta dan pintu stasiun. Waktu pada pengiriman sinyal pada pintu kereta dapat berbeda dengan pintu stasiun disebabkan pengiriman sinyal pada pintu kereta butuh waktu yang lebih lama karena menggunakan teknologi *wireless*. Dengan perbedaan waktu seperti ini, hal yang dapat dilakukan adalah menambahkan waktu (*delay*) pada pengiriman sinyal ke pintu stasiun sampai *delay* tersebut sama seperti pada pengiriman sinyal pada pintu stasiun sehingga antara pintu kereta dan stasiun mendapatkan sinyal secara bersamaan menyebabkan pintu kereta dan stasiun terbuka bersamaan.

4.3.3 Skenario Pada Saat Koneksi Antara Pintu Kereta Dan Pengendali

Mengacu pada Gambar 4.5, pada saat kereta terkena sensor 1 maka pengendali mengaktifkan *wireless* yang mana sinyal pada kereta akan terhubung dengan pengendali sehingga pengendali dapat mengontrol subsistem yang terdapat pada kereta. Pada saat kereta terkena sensor 2, maka pengendali bersiap – siap untuk mengirimkan sinyal ke stasiun dan kereta dengan mengirimkan *delay* agar nantinya digunakan untuk membuka pintu kereta dan stasiun secara bersamaan. Pada saat kereta terkena sensor 3, maka pengendali mengirimkan sinyal ke kereta dan stasiun sehingga pintu kereta dan stasiun terbuka secara bersamaan. Setelah 10 detik, maka pintu kereta kembali tertutup dan dilanjutkan dengan pintu stasiun. Dengan menutupnya pintu stasiun, pengendali menonaktifkan *wireless* kemudian kereta berjalan menuju ke stasiun selanjutnya.

BAB 5

KESIMPULAN

Berdasarkan hasil uji coba dan analisis dapat disimpulkan:

1. Kereta berhenti secara tidak halus disebabkan kecilnya tegangan yang dikontrol yaitu sebesar 5 VDC dan mekanik kereta yang kasar.
2. Pintu kereta dan stasiun bergerak lambat disebabkan *motor stepper* yang digunakan berukuran kecil dengan catu daya 12VDC dan perbandingan gear antara gear pada *motor stepper* dengan pintu adalah 2:1½.
3. Rangkaian sensor pada tiap stasiun bekerja sesuai dengan yang diinginkan dengan memberikan *input* kepada PLC sebesar 24 VDC apabila terhalangi kereta dan tidak memberikan *input* kepada PLC apabila tidak terhalangi kereta.
4. Driver *motor stepper* bekerja sesuai dengan yang diinginkan dengan mengirimkan pulsa-pulsa elektrik dengan *sequence* yang telah diberikan oleh program sehingga dapat menggerakkan *motor stepper* unipolar dengan catu daya 12VDC.
5. Rangkaian kontrol bekerja sesuai dengan yang diinginkan dengan mengirimkan logik “0” atau “1” ke mikrokontroler dan 24VDC ke PLC.
6. Program *ladder* PLC dan Pbasic pada mikrokontroler *BASIC STAMP* bekerja sesuai dengan yang diinginkan dengan kondisi-kondisi yang telah ditentukan.

DAFTAR ACUAN

- [1] Martin Hebel, “Electronic Systems Technologies College of Applied Sciences and Arts Southern Illinois University Carbondale”, Ch. 7 – 22
- [2] Floyd, “Electronic Devices”, Fourth Edition, Prentice Hall, New Jersey, Ch. 740 -741
- [3] Permadi Hudoyo, “Pemrograman PLC Pada Modul Latih Inverter berbasis PLC.” Tugas Akhir, Program Diploma Teknik Listrik PNJ, Depok, 2005, hal. 8 – 10.
- [4] LG Industrial System, LG Programmable Logic Controller Instruction & Programming. (User’s Manual LG-PLC MASTER-K, 2003), Ch. 2-7 – 2-9.
- [5] Achmad Fauzi, Modul Praktikum Laboratorium PLC (Teknik Listrik, Politeknik Negeri Jakarta, 2004), hal. 9 – 13.
- [6] Hemdayani, “Perancangan dan Realisasi Line Follower Robot berbasis mikrokontroler AT89S52, hal 38 - 39.
- [7] Hemdayani, “Perancangan dan Realisasi Line Follower Robot berbasis mikrokontroler AT89S52, hal 28 - 32.

DAFTAR PUSTAKA

Fauzi, Achmad. *Modul Praktikum Laboratorium PLC* (Teknik Listrik, Politeknik Negeri Jakarta, 2004), hal. 9-13.

Floyd, "Electronic Devices", Fourth Edition, Prentice Hall, New Jersey, Ch. 740 - 741

<http://en.wikipedia.org/wiki/Tram>.

Hudoyo, Permadi. "Pemrograman PLC Pada Modul Latih Inverter Berbasis PLC." Tugas Akhir, Program Diploma Teknik Listrik PNJ, Depok, 2005, hal. 8.

Isnandar, Dadang. "Kontrol Motor Induksi Pada Modul Latih Inverter Berbasis PLC." Tugas Akhir, Program Diploma Teknik Listrik PNJ, Depok, 2005, hal. 5-13.

LG Industrial System, *LG Programmable Logic Controller Instruction & Programming*. (User's Manual LG-PLC MASTER-K, 2003), Ch. 2-7 – 2-9.

LG Industrial System, *MASTER-K 120S Programmable Logic Controller*. (Catalogs LG-PLC MASTER-K 120S, 2003), hal. 8-12.

Sujatmoko, MN (2000). *Dasar-dasar Control Component dan SYSMAC*. Diakses 25 Mei 2008 dari wordpress.

<http://ninafkoe.files.wordpress.com/2008/04/basic-plc.pdf>