

**PERANCANGAN DAN PEMBUATAN ROBOT DENGAN
PENGENDALIAN MOTOR DC MENGGUNAKAN
RELATIVE / INCREMENTAL ROTARY ENCODER
BERBASIS ATMEGA 16**

SKRIPSI



**Disusun Oleh :
Bisma Pranadipta
06.12.213**

**JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK ELEKTRONIKA
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
2011**

LEMBAR PERSETUJUAN

PERANCANGAN DAN PEMBUATAN ROBOT DENGAN PENGENDALIAN MOTOR DC MENGGUNAKAN *RELATIVE / INCREMENTAL ROTARY ENCODER* BERBASIS ATMEGA 16

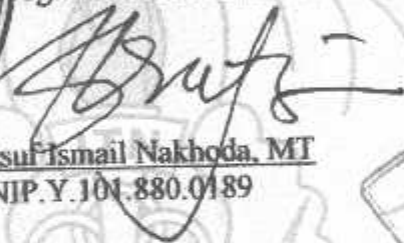
SKRIPSI

Disusun dan diajukan untuk melengkapi dan memenuhi persyaratan
guna mencapai gelar Sarjana Teknik

Disusun Oleh :
Bisma Pranadipta
NIM. 06.12.213



Mengetahui,
Ketua Jurusan Program Studi Teknik Elektro S-1



Ir. Yusuf Ismail Nakhoda, MT
NIP.Y.101.880.0189

Diperiksa dan Disetujui,

Dosen Pembimbing I

Dosen Pembimbing II


Dr. Eng. Aryuanto Soetedjo, ST, MT
NIP.P.103.080.0417


I Komang Somawirata, ST, MT
NIP.P.103.010.0361

**JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK ELEKTRONIKA
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
2011**



PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAN TEKNOLOGI NASIONAL MALANG
INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
FAKULTAS TEKNIK SIPIL DAN PERENCANAAN
PROGRAM PASCASARJANA MAGISTER TEKNIK

NI (PERSERO) MALANG
BANK NIAGA MALANG
Kampus I : Jl. Bendungan Sigura-gura No. 2 Telp. (0341) 551431 (Hunting), Fax. (0341) 553015 Malang 65145
Kampus II : Jl. Raya Karanglo, Km 2 Telp. (0341) 417636 Fax. (0341) 417634 Malang

BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI

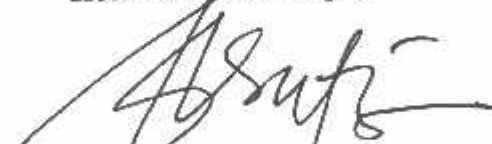
Nama : Bisma Pranadipta
NIM : 06.12.213
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Elektronika
Judul Skripsi : PERANCANGAN DAN PEMBUATAN ROBOT DENGAN
PENGENDALIAN MOTOR DC MENGGUNAKAN *RELATIVE /*
INCREMENTAL ROTARY ENCODER BERBASIS *ATMEGA 16*

Dipertahankan dihadapan Majelis Penguji Skripsi Jenjang Strata Satu (S-1) pada :

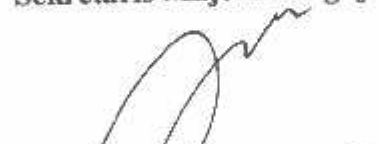
Hari : Kamis
Tanggal : 17 Februari 2011
Dengan Nilai : 82,75 (A) *✓*

Panitia Ujian Skripsi,

Ketua Majelis Penguji

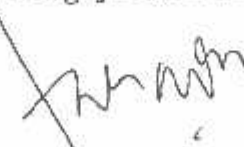

Ir. Yusuf Ismail Nakhoda, MT
NIP.Y.101.880.0189

Sekretaris Majelis Penguji



Dr. Eng. Aryuanto Soetedjo, ST, MT
NIP.Y.103.080.0417

Anggota Penguji,

Penguji Pertama


Irmalia Suryani Faradisa, ST, MT
NIP.P.103.000.0365

Penguji Kedua


Ir. Eko Nurcahyo
NIP.Y.103.870.0172

ABSTRAK

PERANCANGAN DAN PEMBUATAN ROBOT DENGAN PENGENDALIAN MOTOR DC MENGGUNAKAN *RELATIVE / INCREMENTAL ROTARY ENCODER* BERBASIS *ATMEGA 16*

Bisma Pranadipta, NIM 0612213

Dosen Pembimbing : Dr. Eng. Aryuanto Soetedjo, ST, MT
I Komang Somawirata, ST, MT

Robot merupakan suatu peralatan yang bergerak tanpa bantuan manusia, robot juga dapat diartikan sebagai perangkat otomatis. Robot banyak digunakan untuk kebutuhan masyarakat dan juga untuk kebutuhan-kebutuhan industry, seperti memindahkan barang dari satu tempat ke tempat lainnya secara otomatis. Robot yang bergerak, menempuh jarak yang ditentukan, dibutuhkan piranti yang memiliki tingkat akurasi yang tinggi dalam mengendalikan motor DC yang digunakan.

Rotary encoder merupakan salah satu piranti yang memiliki tingkat keakuratan tinggi. *Rotary encoder* merupakan suatu *optointerrupter* mendeteksi adanya perpotongan pada cahaya infra merah yang dipancarkan ke penerimanya. Perpotongan ini ditimbulkan oleh adanya celah-celah pada *encoder*.

Optointerrupter pada *rotary encoder* menimbulkan pulsa pada setiap pergerakan robot yang kemudian akan di eksekusi oleh mikrokontroller *ATMEGA 16*. Robot yang diperintahkan untuk bergerak akan mengalami *error* apabila kecepatan gerak motor DC terlalu cepat. Untuk mengatasi masalah tersebut maka kecepatan putar motor harus diperlambat menggunakan *PWM (Pulse Width Modulation)* agar motor dapat langsung berhenti saat aliran arus dihentikan.

Kata kunci : *Rotary encoder, ATMEGA 16, Pulse Width Modulation, Motor DC*

KATA PENGANTAR

Puji syukur ke hadirat Tuhan Yang Maha Kuasa atas segala limpahan berkat dan Rahmat-Nya sehingga Skripsi yang berjudul Perancangan dan Pembuatan Robot dengan Pengendalian Motor DC Menggunakan Relative / Incremental Rotary Encoder Berbasis ATMEGA 16 dapat di selesaikan.

Penelitian ini dibuat untuk memenuhi salah satu syarat dalam memperoleh gelar sarjana teknik. Ucapan terima kasih sebesar-besarnya kami sampaikan pada:

1. Bapak Prof. Dr. Ir. Abraham Lomi, MSEE selaku rektor ITN Malang.
2. Bapak Ir. H. Sidik Noertjahjono, MT selaku Dekan Fakultas Teknologi Industri ITN Malang.
3. Bapak Ir. Yusuf Ismail Nakhoda, MT selaku Ketua Jurusan Teknik Elektro S-1 ITN Malang.
4. Bapak Dr. Eng. Aryunto Soetedjo, ST, MT selaku dosen pembimbing I.
5. Bapak I Komang Somawirata ST, MT selaku dosen pembimbing II.
6. Semua pihak yang telah membantu penulis dalam menyelesaikan skripsi ini.

Penulis menyadari bahwa skripsi ini masih jauh dari sempurna. Untuk itu kritik dan saran dari pembaca sangat penulis harapkan untuk perbaikan skripsi ini.

Malang, Februari 2011

Penulis

DAFTAR ISI

Lembar Persetujuan	
Abstrak	
Kata Pengantar	iii
Daftar Isi	iv
Daftar Gambar	vii
Daftar Tabel.....	x
Daftar Lampiran	xi
BAB I PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah.....	2
1.3. Batasan Masalah	2
1.4. Tujuan	2
1.5. Metodologi Penulisan.....	2
1.6. Sistematika Penulisan	2
BAB II LANDASAN TEORI	4
2.1. Mikrokontroler ATMEGA 16.....	5
2.1.1. Arsitektur Mikrokontroler ATMEGA 16.....	5
2.1.2. Konfigurasi Port ATMEGA 16.....	6
2.1.3. Timer	8
2.2. H-Bridbge	8
2.2.1. L298	8
2.3. Rotary Encoder Relative	10
2.4. LED	12
2.4.1. LED Infra Red.....	12
2.4.2. LED Superbright.....	13
2.5. Photodioda	14
2.6. Op-Amp.....	15
2.6.1. Comparator.....	15
2.7. Schmitt Trigger	16
BAB III PERANCANGAN ALAT.....	18
3.1. Perancangan Perangkat Keras (Hardware).....	18

3.1.1. Perancangan Minimum Sytem ATMEGA16.....	20
3.1.1.1. Rangkaian Clock Generator.....	20
3.1.1.2. Alokasi Penggunaan Port Mikrokontroller ATMEGA 16	21
3.1.2. Perancangan Rangkaian Sensor	23
3.1.3. Perancangan Rangkaian Comparator	24
3.1.4. Perancangan Rangkaian Schmitt Trigger	25
3.1.5. Perancangan Rangkaian Push Button	26
3.1.6. Perancangan Driver L298	27
3.1.7. Perancangan Rangkaian Rotary Encoder	27
3.1.8. Perancangan Rangkaian LCD	28
3.2. Perancangan Perangkat Lunak (Software).....	30
3.2.1. Program Simulasi.....	30
3.2.2. Program Real.....	31
BAB IV PENGUJIAN ALAT.....	32
4.1. Pengujian Minimum System ATMEGA 16.....	32
4.1.1. Tujuan	32
4.1.2. Rangkaian Pengujian Minimum System.....	32
4.1.3. Peralatan Yang Digunakan	33
4.1.4. Langkah – Langkah Pengujian.....	33
4.1.5. Hasil Pengujian.....	33
4.1.5.1. Pengujian Pada Port A.....	33
4.1.5.2. Pengujian Pada Port B.....	34
4.1.5.3. Pengujian Pada Port C.....	35
4.1.5.4. Pengujian Pada Port D.....	37
4.2. Pengujian Rangkaian Rotary Encoder	38
4.2.1. Tujuan	38
4.2.2. Rangkaian Pengujian Rotary Encoder	38
4.2.3. Peralatan Yang Digunakan	38
4.2.4. Langkah – Langkah Pengujian	39
4.2.5. Hasil Pengujian Rangkaian Rotary Encoder.....	39
4.3. Pengujian Rangkaian Sensor	41
4.3.1. Tujuan	41
4.3.2. Rangkaian Pengujian Sensor	41

4.3.3. Peralatan Yang Digunakan	41
4.3.4. Langkah – Langkah Pengujian.....	41
4.3.5. Hasil Pengujian Rangkaian Sensor	42
4.4. Pengujian Rangkaian Comparator	43
4.4.1. Tujuan	43
4.4.2. Rangkaian Pengujian Comparator	44
4.4.3. Peralatan Yang Digunakan	44
4.4.4. Langkah-Langkah Pengujian	44
4.4.5. Hasil Pengujian Rangkaian Comparator.....	45
4.5. Pengujian Rangkaian Driver Motor DC.....	49
4.5.1. Tujuan	49
4.5.2. Rangkaian Pengujian Driver Motor DC.....	49
4.5.3. Peralatan Yang Diperlukan.....	49
4.5.4. Langkah-Langkah Pengujian Rangkaian Driver Motor.....	49
4.5.5. Hasil Pengujian Rangkaian Driver Motor DC	50
4.6. Pengujian Rangkaian LCD (Liquid Crstal Display)	54
4.6.1. Tujuan	54
4.6.2. Rangkaian Pengujian LCD	54
4.6.3. Peralatan Yang Diperlukan.....	55
4.6.4. Langkah-Langkah Pengujian Rangkaian LCD	55
4.6.5. Hasil Pengujian Rangkaian LCD	55
4.7. Pengujian Keseluruhan Sistem	55
4.7.1. Hasil Pengujian Rotary Encoder Dan Sensor	55
4.7.2. Hasil Pengujian RPM Pada Motor DC	56
BAB V PENUTUP	59
5.1. Kesimpulan	59
5.2. Saran	59
DAFTAR PUSTAKA	60
LAMPIRAN.....	

Gambar 4.3 : Tegangan Keluaran Mikrokontroler PortA Kondisi Low	34
Gambar 4.4 : Tegangan Keluaran Mikrokontroler PortB Kondisi High	35
Gambar 4.5 : Tegangan Keluaran Mikrokontroler PortB Kondisi Low	35
Gambar 4.6 : Tegangan Keluaran Mikrokontroler PortC Kondisi High	36
Gambar 4.7 : Tegangan Keluaran Mikrokontroler PortC Kondisi Low	36
Gambar 4.8 : Tegangan Keluaran Mikrokontroler PortD Kondisi High	37
Gambar 4.9 : Tegangan Keluaran Mikrokontroler PortD Kondisi Low	37
Gambar 4.10 : Rangkaian Mikrokontroler ATMEGA 16.....	38
Gambar 4.11 : Rangkaian Pengujian Rotary Encoder	38
Gambar 4.12 : Tegangan Keluaran Rotary Encoder Pada Saat Tidak Terhalang.....	39
Gambar 4.13 : Tegangan Keluaran Rotary Encoder Pada Saat Terhalang	40
Gambar 4.14 : Pulsa Keluaran Rotary Encoder	40
Gambar 4.15 : Rotary Encoder	40
Gambar 4.16 : Rangkaian Pengujian Sensor.....	41
Gambar 4.17 : Tegangan Keluaran Sensor Saat Terkena Warna Putih	42
Gambar 4.18 : Tegangan Keluaran Sensor Saat Terkena Warna Hijau.....	43
Gambar 4.19 : Rangkaian Sensor.....	43
Gambar 4.20 : Rangkaian Pengujian Comparator	44
Gambar 4.21 : Tegangan Masukan V1 Pada Comparator Saat Sensor Terkena Warna Putih.....	45
Gambar 4.22 : Tegangan Masukan V2 Pada Comparator	46
Gambar 4.23 : Tegangan Keluaran Vo Pada Comparator	46
Gambar 4.24 : Tegangan Masukan Vin Schmitt Trigger Pada Saat Kondisi Low	46
Gambar 4.25 : Tegangan Keluaran Vout Pada Schmitt Trigger	47
Gambar 4.26 : Tegangan Masukan V1 Pada Comparator Saat Sensor Terkena Warna Hijau	47
Gambar 4.27 : Tegangan Masukan V2 Pada Comparator	47
Gambar 4.28 : Tegangan Keluaran Vo Pada Comparator	48
Gambar 4.29 : Tegangan Masukan Vin Schmitt Trigger Pada Saat Kondisi High.....	48
Gambar 4.30 : Tegangan Keluaran Vout Pada Schmitt Trigger	48

Gambar 4.31 : Pengujian Rangkaian Driver Motor DC.....	49
Gambar 4.32 : Enable PWM Digital Dengan Input Duty Cycle 96,078 %	50
Gambar 4.33 : Tegangan Enable PWM Digital Dengan Input Duty Cycle 96,078 %	51
Gambar 4.34 : Enable PWM Digital Dengan Input Duty Cycle 75,294 %	51
Gambar 4.35 : Tegangan Enable PWM Digital Dengan Input Duty Cycle 75,294 %	52
Gambar 4.36 : Enable PWM Digital Dengan Input Duty Cycle 50,19 %	52
Gambar 4.37 : Tegangan Enable PWM Digital Dengan Input Duty Cycle 50,19 %	53
Gambar 4.38 : Enable PWM Digital Dengan Input Duty Cycle 25,09%	53
Gambar 4.39 : Tegangan Enable PWM Digital dengan Input Duty Cycle 25,09%	54
Gambar 4.40 : Pengujian Rangkaian LCD.....	54
Gambar 4.41 : Pengujian LCD.....	55
Gambar 4.42 : RPM Motor Kiri Pada Saat Tanpa Beban.....	57
Gambar 4.43 : RPM Motor Kiri Pada Saat Beban 7Kg	57
Gambar 4.44 : RPM Motor Kanan Pada Saat Tanpa Beban.....	57
Gambar 4.45 : RPM Motor Kanan Pada Saat Beban 7Kg	58



BAB I

PENDAHULUAN

1.1. Latar Belakang

Kemajuan ilmu pengetahuan dan teknologi telah mendorong manusia untuk berusaha mengatasi segala permasalahan yang timbul di sekitarnya. Keterbatasannya untuk terjun langsung melakukan proses pengendalian karena jauh, banyaknya *plant* yang dikendalikan dan membutuhkan waktu yang lama, lambat laun telah dapat diatasi dengan ditemukannya teknologi-teknologi baru, salah satunya adalah sistem gerak pada robot yang memungkinkan manusia lebih mudah mengangkut barang atau memindahkan barang ke tempat lain dalam bidang industri serta dapat juga digunakan sebagai alat transportasi otomatis.

Robot dapat diartikan suatu peralatan yang dioperasikan dengan atau tanpa bantuan manusia atau dengan kata lain bahwa robot merupakan suatu perangkat otomatis. Saat ini robot banyak digunakan dalam kehidupan masyarakat. Bahkan dalam beberapa tahun belakang ini diadakan suatu kontes atau perlombaan robot yang diadakan setiap tahun di Indonesia, yang bertujuan untuk mengenal maupun memperluas ilmu pengetahuan khususnya dalam bidang ilmu robotika. Saat ini, komponen yang banyak digunakan untuk pengukuran gerakan motor pada robot tersebut adalah *encoder*. *Encoder* secara umum dapat dikategorikan ke dalam *optical (photoelectric)*, *magnetic encoder*, dan tipe kontak mekanik. *Photoelectric encoder* memiliki tingkat akurasi yang tinggi, handal, dan relatif murah, mudah dalam aplikasinya. Berdasarkan atas hal tersebut penulis berusaha untuk membuat suatu robot dengan menggunakan metode *Rotary Encoder* sebagai kendali motor DC dari robot berbasis *mikrokontroller ATMEGA16*.

1.2. Rumusan Masalah

Berdasarkan latar belakang diatas, permasalahan yang akan dibahas pada laporan akhir ini adalah :

1. Bagaimana merencanakan membuat *hardware* dan *software* dari robot berbasis *mikrokontroller ATMEGA 16*.

2. Bagaimana mengendalikan kecepatan motor DC dengan menggunakan *Rotary Encoder*.

1.3. Batasan Masalah

Penulis mengambil batasan masalah sebagai berikut :

1. Pembangkitan pulsa *PWM* digital diperoleh secara program
2. Desain robot beroda.
3. Jalur robot sudah ditentukan

1.4. Tujuan

Tujuan dari penulisan skripsi ini adalah :

Untuk mengetahui keakuratan *Rotary Encoder* mengendalikan motor DC dalam menempuh jarak yang ditentukan, dan mengatasi nilai error (e) pada jarak (s).

1.5. Metodologi Penulisan

Metodologi penulisan yang dipakai dalam pembuatan skripsi ini.

1. Studi Literatur
Pengumpulan bahan sebagai referensi yang berkaitan dengan metode *Rotary Encoder*
2. Perancangan dan Pembuatan Alat
3. Perancangan *hardware* dan pemrograman
4. Pelaksanaan uji coba alat, dan pengambilan dan analisa data
5. Menarik Kesimpulan

1.6. Sistematika Penulisan

Penulisan skripsi ini terbagi dalam lima bab dengan sistematika sebagai berikut:

BAB I PENDAHULUAN

Berisi latar belakang tujuan, permasalahan, batasan masalah, metodologi, dan sistematika penulisan.

BAB II LANDASAN TEORI

Membahas teori-teori dasar penunjang, perancangan dan pembuatan alat.

BAB III PERANCANGAN DAN PEMBUATAN ALAT

Membahas tentang perancangan alat baik perangkat keras maupun perangkat lunak, serta cara kerja blok diagram.

BAB IV PENGUJIAN ALAT

Mencakup pembahasan tentang proses pengujian alat yang terdiri dari peralatan yang digunakan, langkah kerja dan analisa hasil pengujian.

BAB V PENUTUP

Penulis menarik kesimpulan dari apa yang telah diuraikan pada bab-bab sebelumnya dan mengemukakan saran saran yang mungkin akan bermanfaat bagi laporan akhir ini.



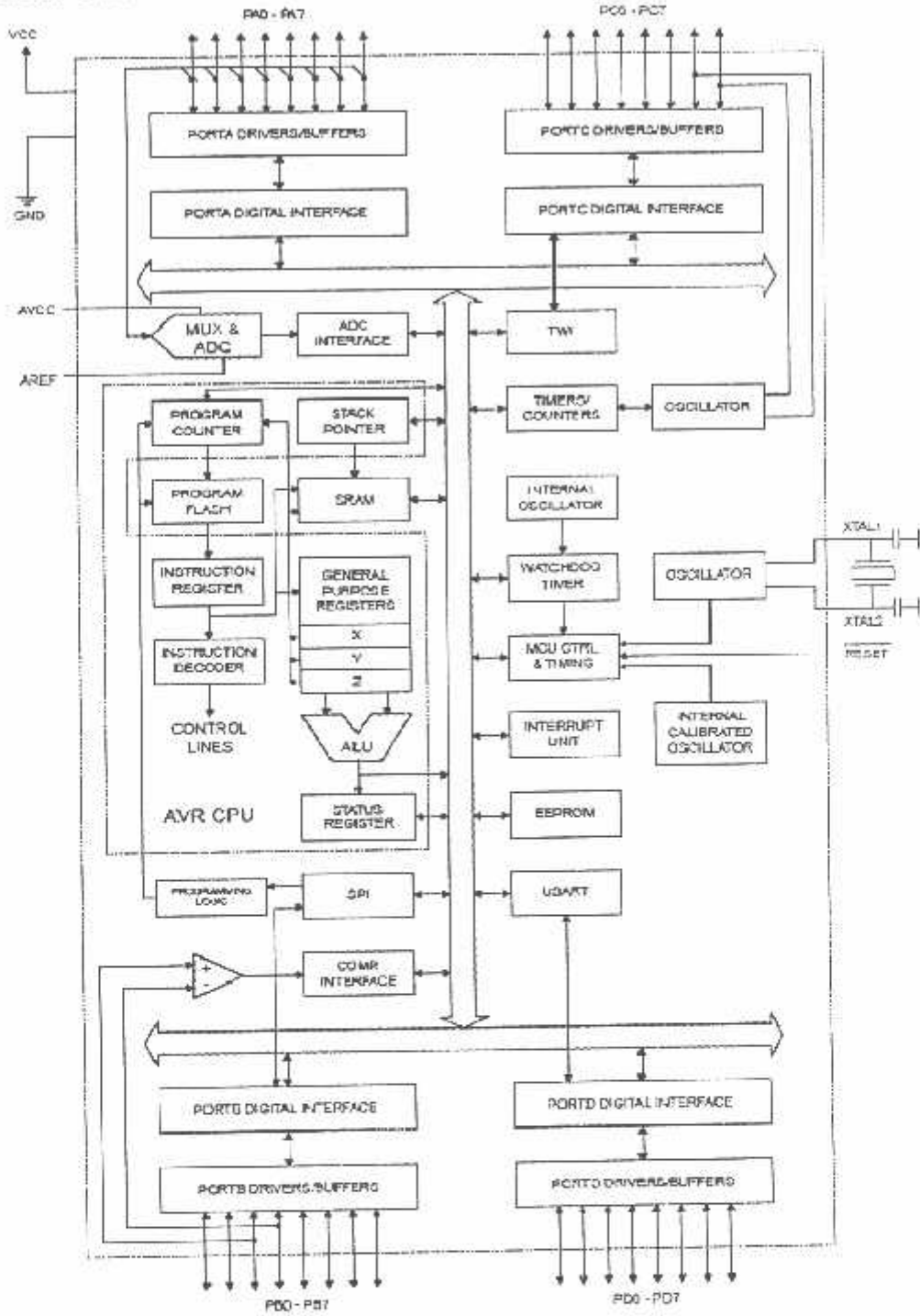
BAB II

LANDASAN TEORI

Landasan teori sangat membantu untuk dapat memahami suatu sistem, selain daripada itu, dapat juga dijadikan sebagai bahan literatur, teori dasar, serta referensi yang berguna sebagai acuan, rujukan perhitungan dan landasan acuan di dalam merencanakan suatu sistem. Dengan pertimbangan hal-hal tersebut, maka landasan teori merupakan bagian yang harus dipahami untuk pembahasan selanjutnya. Pengetahuan yang mendukung perencanaan dan realisasi alat meliputi tentang dasar – dasar teori yang menunjang dalam perencanaan dan pembuatan alat tersebut. Uraian teori dalam bab ini meliputi *Mikrokontroler ATMEGA16, L298, Rotary Encoder, LED Superbright, Photodiode, Comparator, Schmitt Trigger*.

2.1. Mikrokontroler ATMEGA 16

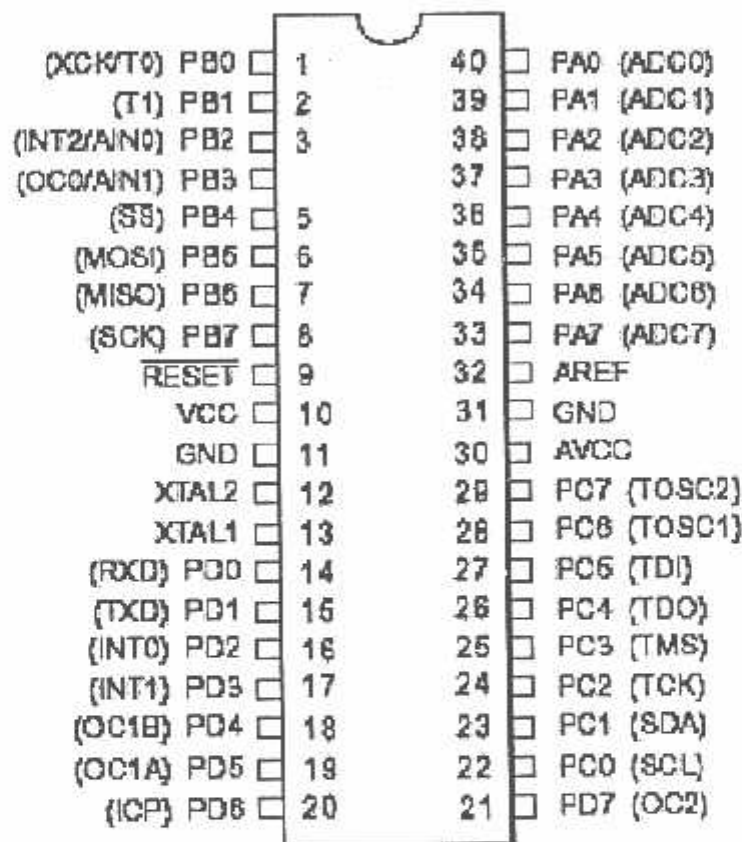
2.1.1. Arsitektur Mikrokontroler ATMEGA 16



Gambar 2.1. Blok Diagram Mikrokontroler ATMEGA 16

ATMEGA 16 (tergolong dalam keluarga AVR) merupakan seri mikrokontroler CMOS 8-bit buatan Atmel, berbasis arsitektur RISC (*Reduced Instruction Set Computer*). Hampir semua instruksi dieksekusi dalam satu siklus

clock. AVR mempunyai 32 *register general-purpose*, *timer/counter* fleksibel dengan mode *compare*, *interrupt internal* dan *eksternal*, *serial UART*, *programmable Watchdog Timer*, dan mode *power saving*. Mempunyai *ADC* dan *PWM internal*. AVR juga mempunyai *In-System Programmable Flash on-chip* yang mengijinkan memori program untuk diprogram ulang dalam sistem menggunakan hubungan serial SPI. ATMEGA 16 adalah *mikrokontroler CMOS 8-bit* daya-rendah berbasis arsitektur *RISC* yang ditingkatkan. Kebanyakan instruksi dikerjakan pada satu siklus clock, Atmega 16 mempunyai *throughput* mendekati 1 *MIPS per MHz* membuat disainer sistem untuk mengoptimasi konsumsi daya versus kecepatan proses.



Gambar 2.2. Konfigurasi Pin ATMEGA 16

2.1.2. Konfigurasi Port ATMEGA 16

ATMEGA 16 mempunyai empat buah *port* yang bernama *PortA*, *PortB*, *PortC*, dan *PortD*. Keempat port tersebut merupakan jalur *bi-directional* dengan pilihan internal *pull-up*. Tiap port mempunyai tiga buah *register* bit, yaitu *DDxn*, *PORTxn*, dan *PINxn*. Huruf 'x' mewakili nama huruf dari *port* sedangkan huruf 'n' mewakili nomor bit. Bit *DDxn* terdapat pada I/O address *DDRx*, bit *PORTxn*

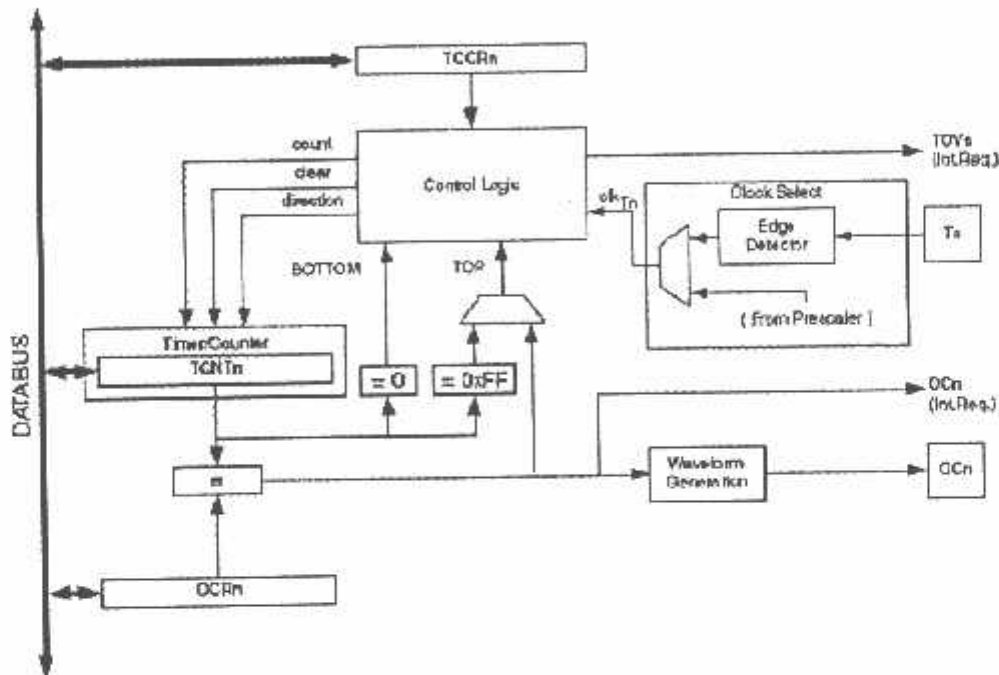
terdapat pada I/O address PORTx, dan bit PINxn terdapat pada I/O address PINx. Bit DDxn dalam register DDRx (*Data Direction Register*) menentukan arah pin. Bila DDxn diset 1 maka Px berfungsi sebagai pin *output*. Bila DDxn diset 0 maka Px berfungsi sebagai pin *input*. Bila PORTxn diset 1 pada saat pin terkonfigurasi sebagai pin input, maka *resistor pull-up* akan diaktifkan. Untuk mematikan *resistor pull-up*, PORTxn harus diset 0 atau pin dikonfigurasi sebagai pin *output*. Pin port adalah *tri-state* setelah kondisi *reset*. Bila PORTxn diset 1 pada saat pin terkonfigurasi sebagai pin *output* maka pin port akan berlogika 1. Dan bila PORTxn diset 0 pada saat pin terkonfigurasi sebagai pin *output* maka pin port akan berlogika 0. Saat mengubah kondisi port dari kondisi *tri-state* (DDxn=0, PORTxn=0) ke kondisi *output high* (DDxn=1, PORTxn=1) maka harus ada kondisi peralihan apakah itu kondisi *pull-up enabled* (DDxn=0, PORTxn=1) atau kondisi *output low* (DDxn=1, PORTxn=0). Biasanya, kondisi *pull-up enabled* dapat diterima sepenuhnya, selama lingkungan impedansi tinggi tidak memperhatikan perbedaan antara sebuah *strong high driver* dengan sebuah *pull-up*. Jika ini bukan suatu masalah, maka bit *PUD* pada register *SFIOR* dapat diset 1 untuk mematikan semua *pull-up* dalam semua port. Peralihan dari kondisi input dengan *pull-up* ke kondisi *output low* juga menimbulkan masalah yang sama. Kita harus menggunakan kondisi *tri-state* (DDxn=0, PORTxn=0) atau kondisi *output high* (DDxn=1, PORTxn=0) sebagai kondisi transisi. Berikut penjelasan dari masing-masing port:

- Port A
Port A menyediakan *input* analog untuk *ADC*, Port A juga bisa digunakan sebagai jalur *bi-directional I/O* jika fitur *ADC* tidak digunakan
 - Port B
Port B menyediakan fitur *bi-directional I/O* dan beberapa fungsi pin khusus, antara lain *timer/counter*, komparator analog, dan *SPI*
 - Port C
Port C menyediakan fitur *bi-directional I/O* dan beberapa fungsi pin khusus, antara lain : *I²C*, komparator analog, dan *Timer Oscillator*
 - Port D
Port D menyediakan fitur *bi-directional I/O* dan beberapa fungsi pin khusus, antara lain : komparator analog, interupsi eksternal, dan komunikasi *serial*
-

2.1.3. Timer

Timer atau *counter* adalah tujuan umum *single channel*, modul 8 bit *timer/counter*. Beberapa fasilitas *channel* dari *timer counter* antara lain:

- *Counter channel* tunggal
- Pengosongan data *timer* sesuai dengan data pembanding
- Behas *glitch*, tahap yang tepat untuk *Pulse Width Modulator (PWM)*
- Pembangkit *frekuensi*
- *Event counter external*



Gambar 2.3. Blok Diagram *Timer/Counter*

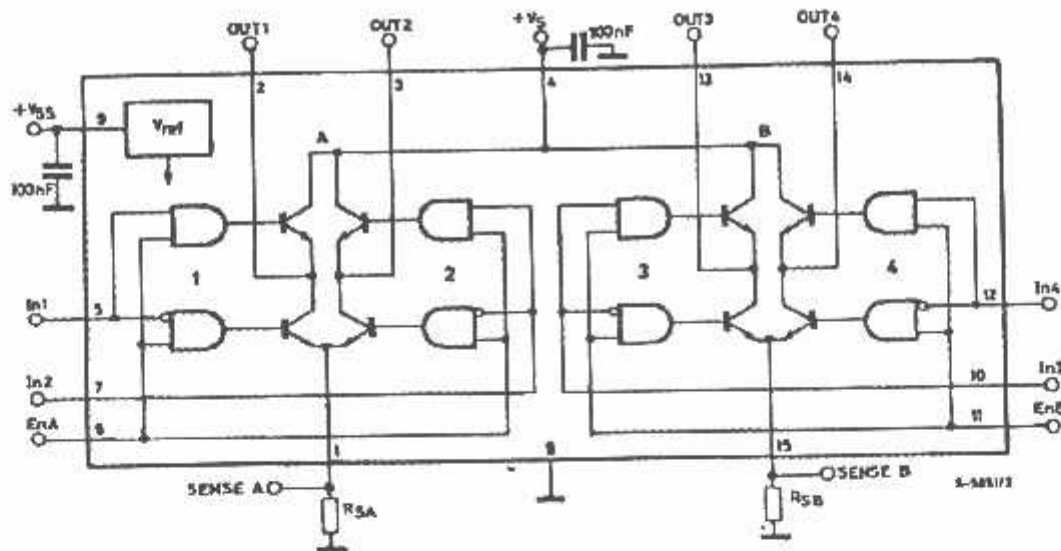
2.2. H-Bridge

Rangkaian *H-bridge* adalah rangkaian untuk mendrive motor yang menghasilkan keluaran dengan putaran motor yang dapat berputar searah jarum jam (*clock wise*), maupun berlawanan dengan arah jarum jam (*counter clock wise*). Untuk *H-bridge* ini digunakan *L298 (dual full bridge driver)*, yang berisi dua rangkaian *H-bridge*. *H-bridge L298* ini digunakan sebagai *driver* untuk menggerakkan motor sampai dengan arus 3A.

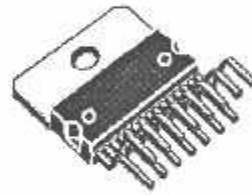
2.2.1. L298

IC *L298* adalah IC *dual full bridge driver* yang beroperasi sampai tegangan 46V dan arus DC 4A, yang di desain untuk menerima *logic level TTL* standar. IC ini

berfungsi untuk mendrive induktif load seperti relay, selenoida, motor DC dan motor stepper.



Gambar 2.4. Rangkaian *H-Bridge* pada *L298*



Gambar 2.5. Bentuk Fisik *L298*

L298 mempunyai 2 *output power stage* yaitu A dan B. *output power stage* adalah sebuah konfigurasi *bridge*, dimana *output*-nya dapat men-drive sebuah induktif load secara mode umum atau diffensial tergantung pada input state-nya. Arus yang mengalir keluar dari rangkaian *bridge* keluar melalui pin *output sense*, dimana *sense output* dihubungkan dengan resistor *external* R_{SA} dan R_{SB} . Dengan adanya resistor R_{SA} dan R_{SB} dapat diketahui intensitas arusnya.

Setiap *H-bridge* di drive oleh 2 AND gate yang masing-masing inputnya adalah In_1 , In_2 , EnA dan In_3 , In_4 , EnB . Inputan In akan men-set *bridge* bila inputan En *high*, sebaliknya bila inputan En *low* maka *bridge* tidak aktif.

2.3. Rotary Encoder Relative

Teori dasar :

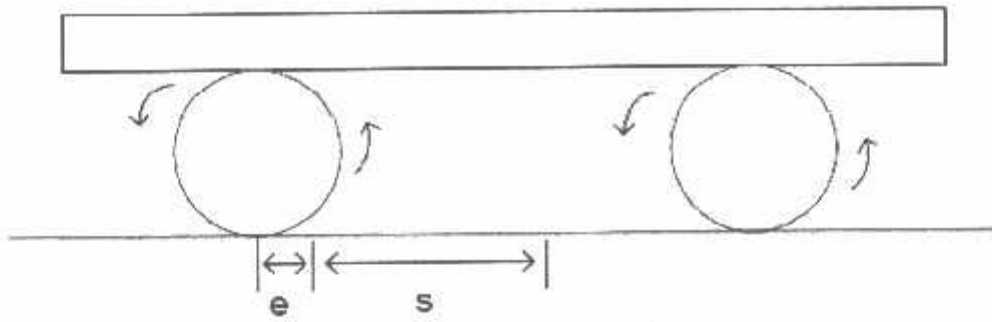
Untuk menggerakkan robot baik maju atau mundur dengan jarak tertentu biasanya dilakukan dengan menghitung waktu. Dengan kecepatan dan durasi tertentu maka akan dapat ditentukan bahwa robot telah bergerak dengan jarak tertentu melalui persamaan $s = v \times t$. Namun kecepatan motor pada robot tidaklah selalu stabil seiring dengan perubahan kapasitas *battery* sebagai sumber daya. Pada saat kapasitas *battery* menurun maka kecepatan putar motor akan berkurang sehingga perhitungan jarak dengan menghitung nilai t (durasi) tidaklah akurat. Untuk perhitungan jarak yang lebih presisi dapat dilakukan dengan menghitung gerakan roda yaitu dengan menghitung seberapa jauh roda robot berputar. Proses ini dapat dilakukan dengan menggunakan piringan *encoder* dan *opto interrupter*. *encoder* disini memiliki 36 celah maka gerakan satu putaran roda dapat dihitung dengan adanya 113 pulsa pada *opto interrupter*. Dengan digunakan roda diameter 8,4 cm maka dengan menggunakan persamaan berikut akan diperoleh jarak dari gerakan robot untuk setiap putaran roda yaitu dengan menghitung keliling roda.

$$\begin{aligned} S &= \pi \times d \\ &= 3,14 \times 8,4 = 26,376 \text{ cm} \end{aligned}$$

Dengan adanya 36 pulsa untuk setiap putaran maka untuk setiap pulsa yang ditimbulkan oleh *optointerrupter* akan diperoleh jarak sebesar $26,376 / 113 \text{ cm} = 0,233 \text{ cm}$ atau 2,33 mm. *Opto interrupter* adalah sebuah sensor yang mendeteksi adanya perpotongan pada cahaya infra merah yang dipancarkan ke penerimanya. Perpotongan ini ditimbulkan oleh adanya celah-celah pada *encoder*.

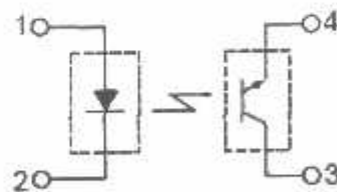
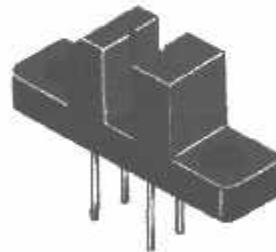
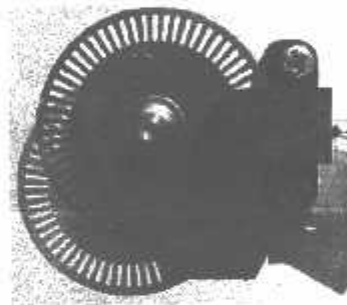
Cara kerja :

Motor DC yang mempunyai kecepatan tinggi sebagai penggerak. Motor tersebut tidak langsung berhenti walaupun arus yang mengalir telah dihentikan oleh karena itu akan terjadi *error* seperti yang tampak pada gambar dibawah.



Gambar 2.6. nilai error (e) pada jarak (s)

Robot yang diperintahkan untuk bergerak sejauh ' s ' akan mengalami error ' e ' apabila kecepatan gerak motor terlalu cepat. Untuk mengatasi masalah tersebut maka kecepatan putar motor harus diperlambat agar motor dapat langsung berhenti saat aliran arus dihentikan. Pengaturan kecepatan motor ini tergantung dari beban yang dibawa oleh robot. Untuk beban yang cukup besar, otomatis gerakan robot akan lebih lambat daripada beban yang ringan.



Gambar 2.7. *Relative Rotary Encoder*

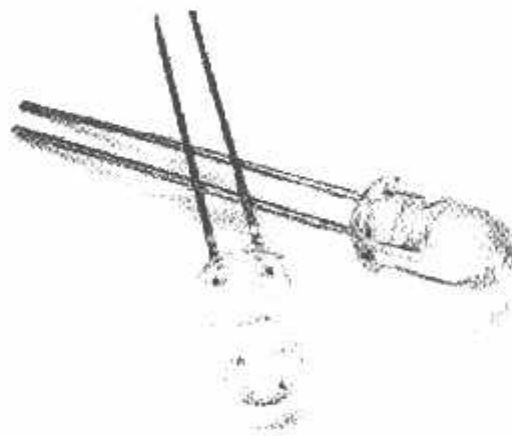
2.4. LED

2.4.1 LED *Infra Red*

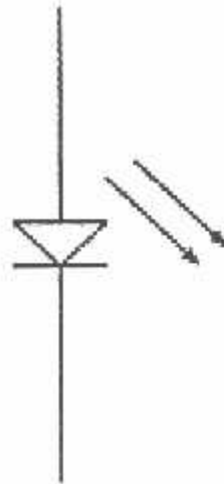
Spektrum cahaya termasuk gelombang *elektromagnetik* dan dapat di kelompokkan dalam beberapa macam, yaitu cahaya tampak dan cahaya tak tampak. *Infra red* termasuk golongan cahaya tak tampak oleh mata kita, memiliki panjang gelombang kurang lebih $0,7-100 \times 10^{-6}$ m. Gelombang elektromagnetik dapat dibagi dalam beberapa daerah, yaitu gelombang radio, sinar x, sinar *gama*, sinar *infra red*, dan sinar *ultraviolet*.

Sifat dari *infra red* tidak berbeda jauh dengan cahaya tampak, di antaranya yaitu dapat memancarkan cahaya yang merupakan satu garis lurus dan dapat menembus benda – benda transparan serta dapat dipantulkan oleh benda yang memantulkan cahaya, misalnya cermin. Spektrum *infra red* banyak digunakan untuk keperluan *elektromagnetik*, terutama untuk alat yang menggunakan *remote control*.

Cahaya *infra red* dapat dibangkitkan melalui *difusi* pada *dioda semi konduktor* yang disebut *LED*. Sedangkan *dioda* sendiri juga banyak jenisnya termasuk yang *12epr* memancarkan cahaya saat dialiri arus *forward* atau arus masuk *elektron* dari pita *konduksi* yang melewati *junction* dan jatuh ke dalam *hole* pita *valensi*, sehingga *elektron* memancarkan energi. Pada *dioda* energi di pancarkan melalui panas dan *dioda* yang tidak memancarkan cahaya contohnya *dioda zener* maupun *dioda biasa*. Lambang dari *LED Infra Red* dapat dilihat pada gambar di bawah ini :



Gambar 2.8. *LED Infrared*



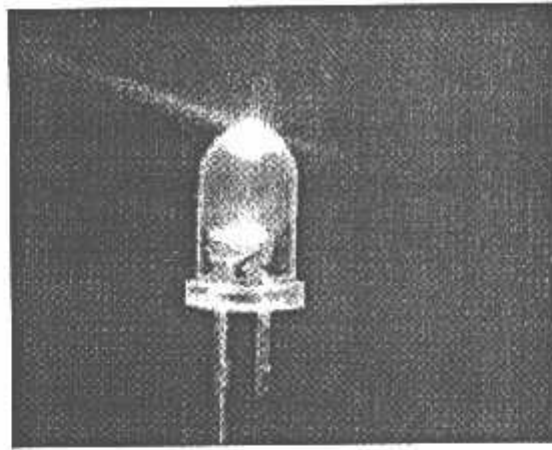
Gambar 2.9. Simbol dari *LED* Infra Red

Keuntungan dalam menggunakan *Spektrum* Infra Red adalah sebagai berikut:

- *Infra Red* mempunyai panjang gelombang yang kecil pada daerah *frekuensi* tertentu dengan panjang gelombang 7 – 100 (μ m).
- Pemancar *Infra Red* membutuhkan sumber tegangan yang kecil.
- Tidak mengganggu pemakaian udara untuk kepentingan umum, sehingga tidak memerlukan ijin khusus.
- *Spektrum Infra Red* merupakan cahaya yang tidak tampak, sehingga dapat digunakan untuk keperluan rahasia.

2.4.2. *LED Superbight*

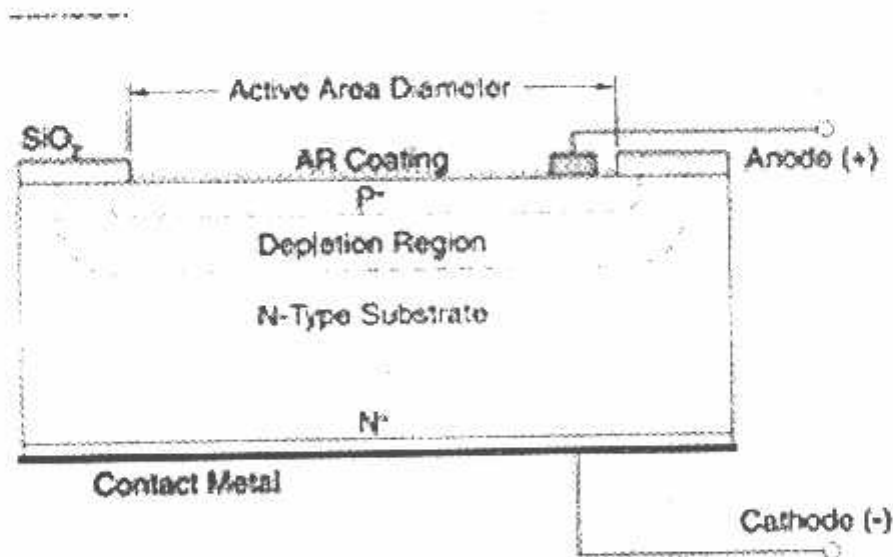
LED Superbight berfungsi sama halnya dengan *Infrared*, bedanya dalam hal ini *Infrared* merupakan cahaya tak tampak, sedangkan *Led Superbight* merupakan cahaya yang tampak. *Led Superbight* berfungsi sebagai pengirim cahaya ke garis untuk dibaca sensor. Kerjanya ketika sumber tegangan masuk pada *battery on*, maka arus masuk sehingga *led superbright* menyala dengan terang yang kemudian dibiaskan pada *photodiode*.



Gambar 2.10. *Led Superbright*

2.5. *Photodiode*

Photodiode mempunyai daerah *depleksi* dengan medan listrik yang tinggi yang menyebabkan pemisahan pasangan *elektron hole* secara *photogeneration*. Untuk operasi kerja yang tinggi, daerah *depleksi* harus dijaga ketipisannya untuk mereduksi waktu transit. Dengan kata lain peningkatan *efisiensi kuantum* (jumlah pasangan *elektron* dihasilkan per-*photon* yang timbul). Lapisan *depleksi* harus cukup tebal untuk menyerap sebagian besar cahaya. Gambar di bawah ini menunjukkan struktur dari *photodiode* :

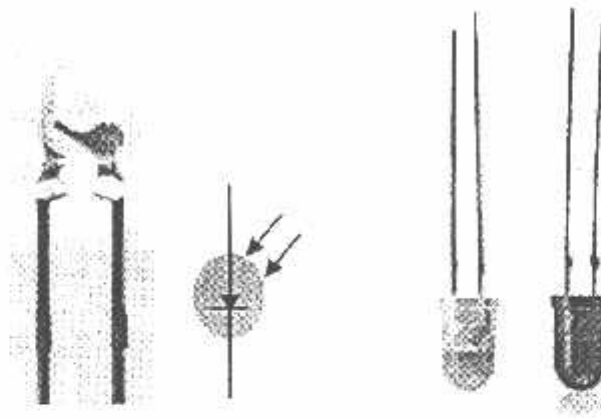


Gambar 2.11. Struktur dari *Photodiode*

Photodiode dapat dioperasikan *photovoltaic*, *photodiode* dalam keadaan tidak terbias dan terhubung pada impedansi beban sama seperti *solarsel* yang akan diuraikan.

Selanjutnya bagaimanapun juga *desain* piranti ini berbeda secara dasarnya. Pada *photodiode* hanya *range* panjang gelombang yang sempit terpusat pada panjang gelombang sinyal optik merupakan hal yang penting.

Bentuk fisik *photodiode* adalah seperti ditunjukkan pada gambar di bawah ini yang mempunyai jenis warna yang bermacam – macam. Berikut ini ditunjukkan *photodiode* yang memilih warna gelap dan warna terang :



Gambar 2.12. Bentuk Fisik *Photodiode*

Photodiode sulit untuk memperkecil *kapasitansi junction* dan satu hal yang terpenting dari pada *photodiode* adalah *efisiensi kuantum*. P-I-N *photodiode* adalah salah satu *photodetector* yang paling umum oleh karena daerah *depleksi* yang lebar yang dapat mengoptimalkan *efisiensi kuantum* dan respon frekuensinya.

Penyerapan cahaya pada *semikonduktor* menghasilkan pasangan *elektron-hole*, pasangan *elektron-hole* menghasilkan daerah *depleksi* atau panjang *difusi* dari pasangan *elektron-hole* akan dipisahkan oleh medan listrik dimana arus mengalir kedalam rangkaian *external* sebagai *carrier drift* melalui lapisan *depleksi*.

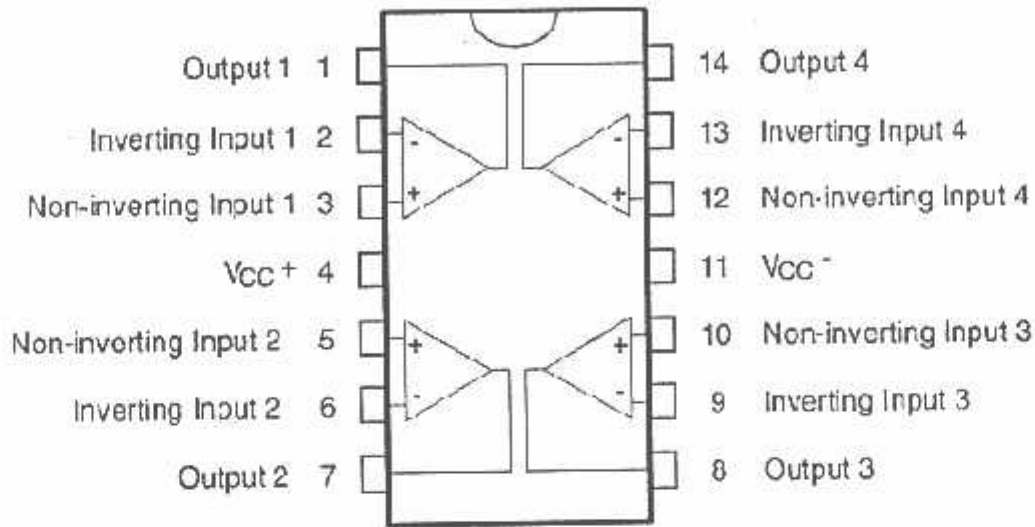
Ketebalan lapisan yang lebar dioptimalisasikan untuk panjang gelombang sinyal optik dan *frekuensi* modulasi.

2.6. Op-Amp

2.6.1. Comparator

Comparator adalah suatu rangkaian yang membandingkan tegangan *input* dari rangkaian *photodiode* untuk membedakan gelap terang jalur robot sehingga robot bisa bergerak mengikuti garis. Pada rangkaian ini digunakan IC LM 324.

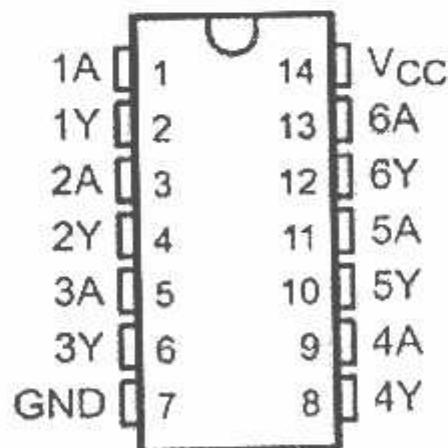
IC LM324 merupakan *IC Operational Amplifier*, *IC* ini mempunyai 4 buah *op-amp* yang berfungsi sebagai *komparator*. *IC* ini mempunyai tegangan kerja antara +5 V sampai +15V untuk +Vcc dan -5V sampai -15V untuk -Vcc. Adapun definisi dari masing-masing pin *IC LM 324* adalah sebagai berikut :



Gambar 2.13. Konfigurasi Pin *LM 324*

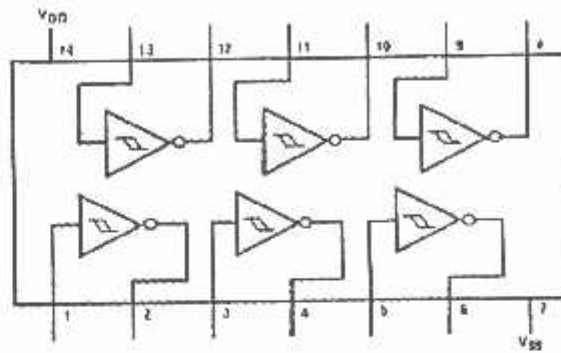
2.7. Schmitt Trigger

IC 74LS14 merupakan *IC TTL* yang berfungsi sebagai penguat tegangan dan menegaskan masukan sinyal yang berupa sinyal bergerigi atau *sinus* menjadi sinyal kotak yang tegas. *IC* ini mempunyai 6 gerbang *inverter* yang masing-masing dengan *input* Schmitt trigger. Gambar konfigurasi pin-pinnya adalah sebagai berikut:



Gambar 2.14. Konfigurasi Pin *74LS14*

Akibat adanya *inversi*, maka histeresis *IC TTL 74LS14* menjadi terbalik. Bila masukkan melampaui 1,9V maka keluaran berpindah ke keadaan rendah. Bila masukkan kurang dari 1V maka keluaran berpindah kembali ke keadaan tinggi, dengan kata lain, keluaran dari *inverter schmitt trigger* ialah kebalikan dari masukannya. *Schmitt trigger* dengan gerbang *NOT* yang dikemas dalam bentuk *IC TTL 74LS14* diperlihatkan dalam Gambar 2.15.



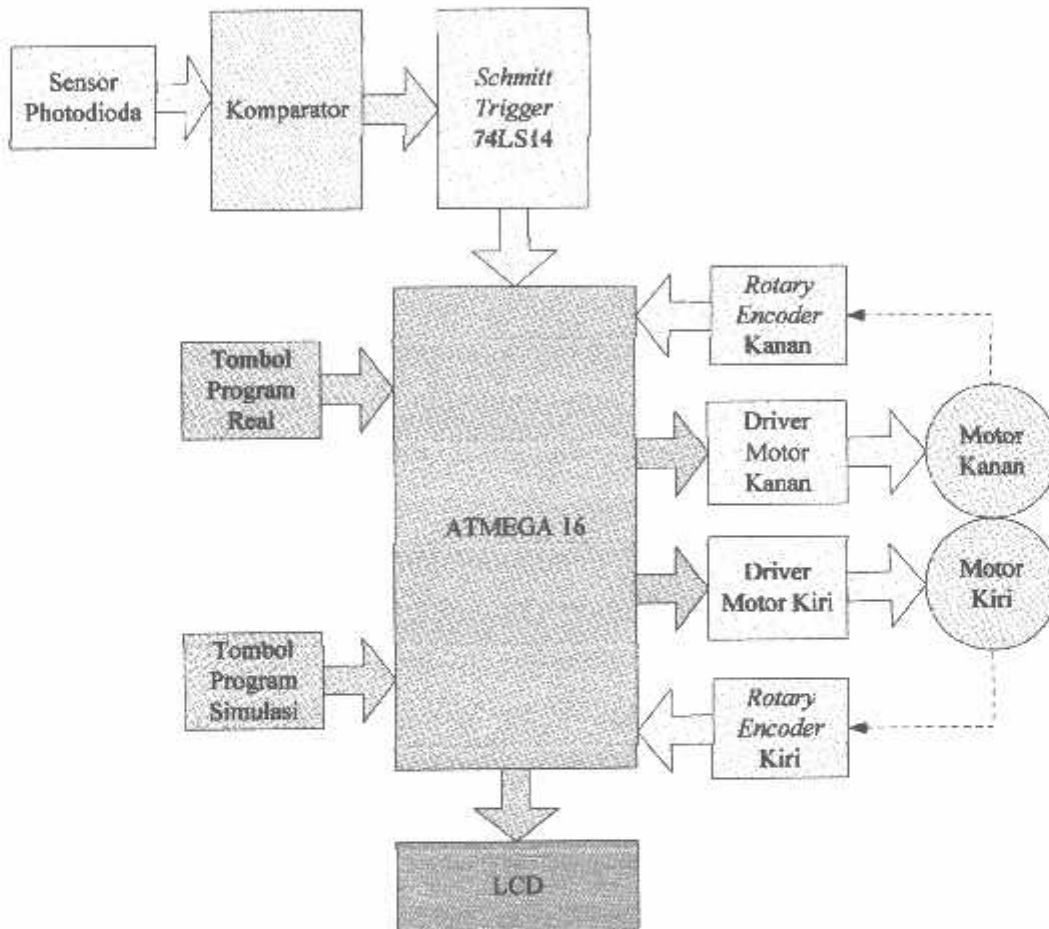
Gambar 2.15. *Inverter Schmitt Trigger IC 74LS14*



BAB III PERANCANGAN ALAT

Pada bab ini akan dibahas mengenai perancangan dan pembuatan perangkat keras (*Hardware*) dan perangkat lunak (*software*) yang berfungsi sebagai sistem dari robot itu sendiri

3.1. Perancangan Perangkat Keras (*Hardware*)



Gambar 3.1. Blok Diagram Sistem

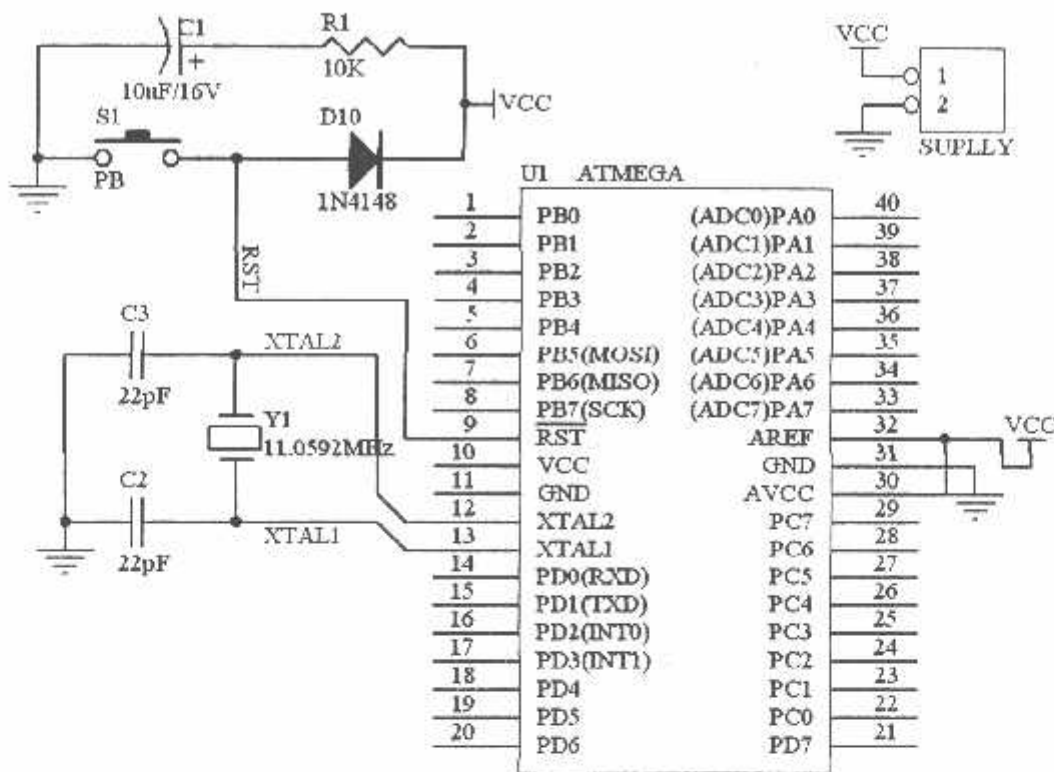
Keterangan Blok Diagram Sistem :

➤ Rotary Encoder Kanan

Untuk mengkonversi posisi angular (sudut) dari shaft (lubang) atau roda pada motor kanan ke dalam kode digital

- *Rotary Encoder Kiri*
Untuk mengkonversi posisi anguler (sudut) dari shaft (lubang) atau roda pada motor kiri ke dalam kode digital
 - *Sensor Photodiode*
Sensor yang berfungsi untuk mendeteksi garis putih. Sensor ini berfungsi untuk untuk robot untuk berbelok
 - *Op-Amp*
Op-Amp disini berfungsi sebagai *comparator*. Rangkaian ini bertujuan untuk membandingkan tegangan referensi dengan tegangan masukan dari sensor garis
 - *Schmitt Trigger 74LS14*
Rangkaian ini berfungsi untuk membalik *logic* yang dihasilkan oleh pengkondisi sinyal.
 - *Tombol Pilih Program Real*
Tombol untuk menjalankan program utama
 - *Tombol Pilih Program Simulasi*
Tombol untuk menjalankan program simulasi
 - *Mikrokontroler ATMEGA16*
Berfungsi sebagai pengontrol utama dari sistem robot
 - *Driver Motor Kanan*
Berfungsi sebagai penguat arus untuk menggerakkan motor kanan sebagai penggerak roda robot bagian kanan
 - *Driver Motor Kiri*
Berfungsi sebagai penguat arus untuk menggerakkan motor kiri sebagai penggerak roda robot bagian kiri
 - *LCD*
Sebagai penampil untuk simulasi system
-

3.1.1. Perancangan *Minimum System* ATMEGA16



Gambar 3.2. Perancangan *Minimum System* ATMEGA 16

3.1.1.1. Rangkaian *Clock Generator*

Mikrokontroler Atmega16 memiliki *osilator internal* (*on chip oscillator*) yang dapat digunakan sebagai sumber *clock* bagi CPU. Untuk menggunakan *osilator internal* diperlukan sebuah kristal antara pin *XTAL1* dan *XTAL2* serta kapasitor ke *ground* seperti Gambar 3.2. Untuk kristal dari *minimum system* ini menggunakan frekuensi 11.0592 MHz, sedangkan untuk kapasitor menggunakan 22 pF sampai 22 pF. Nilai kapasitor ini diperoleh dari tabel *datasheet* tentang penggunaan kapasitor untuk rangkaian *osilator/sistem clock* pada Atmega 16. Penggunaan *cristal* 11,0592 MHz ini bertujuan agar perhitungan *baudrate* tidak mengalami *error* yang disebabkan karena selisih perhitungan. Perhitungan *baudrate* pada AVR ATmega16 dengan menggunakan kristal 11,0592 MHz, dimana *baudrate* yang diinginkan ialah 38400 bps, maka nilai pada *UBRR* (*USART Baud Rate Register*) dapat ditentukan dengan perhitungan :

$$UBRR = (f_{osc} / 16 \cdot \text{Baud}) - 1$$

$$UBRR = (11059200 / 16 \cdot 38400) - 1$$

$$UBRR = (11059200 / 614400) - 1$$

$$UBRR = 18 - 1$$

$$= 17 = 11 \text{ MHz}$$

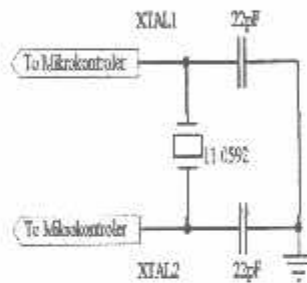
Dimana :

UBRR : *USART Baud Rate Register*.

Fosc : Kristal yang digunakan.

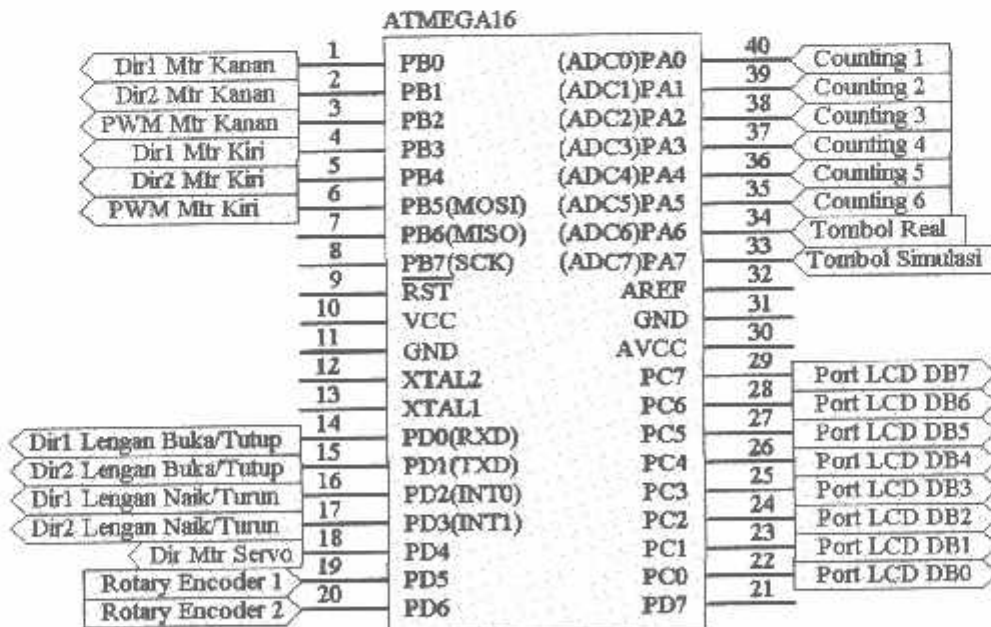
Baud : *baudrate* yang diinginkan.

Penggunaan kristal 11,0592 MHz memungkinkan hasil perhitungan *baudrate* tidak sisa dan *error* dari selisih perhitungan tidak ada.



Gambar 3.3. Perancangan Rangkaian *Clock Generator*

3.1.1.2. Alokasi Penggunaan *Port Mikrokontroler ATMEGA 16*

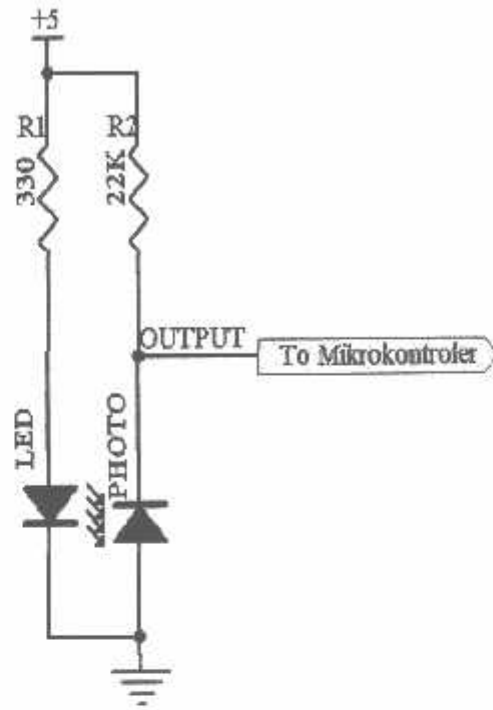


Gambar 3.4. Alokasi Penggunaan *Port Mikrokontroler*

Keterangan :

- Port A0-A5 digunakan sebagai input dari outputan sensor photodiode yang akan diolah untuk mengindikasikan motor kanan dan kiri untuk berbelok
 - Port A6 digunakan sebagai input tombol simulasi, dimana akan memanggil program real jika ditekan
 - Port A7 digunakan sebagai input tombol real, dimana akan memanggil program simulasi jika ditekan
 - Port B0-B2 digunakan sebagai output untuk data yang akan dikirim ke *driver* motor kanan
 - Port B3-B5 digunakan sebagai output untuk data yang akan dikirim ke *driver* motor kiri
 - Port C0-C7 digunakan sebagai output untuk disambungkan ke port DB0-DB7 pada *LCD* untuk menampilkan tulisan yang diinginkan
 - Port D0-D1 digunakan sebagai output untuk data yang akan dikirim ke *driver* motor lengan buka/tutup
 - Port D2-D3 digunakan sebagai output untuk data yang akan dikirim ke *driver* motor lengan naik/turun
 - Port D4 digunakan sebagai output untuk menggerakkan motor servo
 - PD5, dimana PD5 digunakan sebagai input dari pulsa yang dihasilkan oleh *rotary encoder* kanan yang selanjutnya akan diolah oleh mikrokontroler untuk mengatur jarak tempuh dari robot
 - PD6, dimana PD6 digunakan sebagai input dari pulsa yang dihasilkan oleh *rotary encoder* kiri yang selanjutnya akan diolah oleh mikrokontroler untuk mengatur jarak tempuh dari robot
-

3.1.2. Perancangan Rangkaian Sensor Photodioda



Gambar 3.5. Rangkaian Sensor Photodioda

Untuk mencari R1 pada rangkaian sensor diatas adalah :

$$\text{Dik : } VCC = +5V$$

$$I_{LED} = 20 \text{ mA} = 0,02 \text{ A}$$

$$V_R = 2V$$

$$\text{Jadi, } R1 = \frac{VCC - V_R}{I}$$

$$= \frac{5 - 2}{0,02}$$

$$= \frac{3}{0,02}$$

$$= 150\Omega = 330\Omega$$

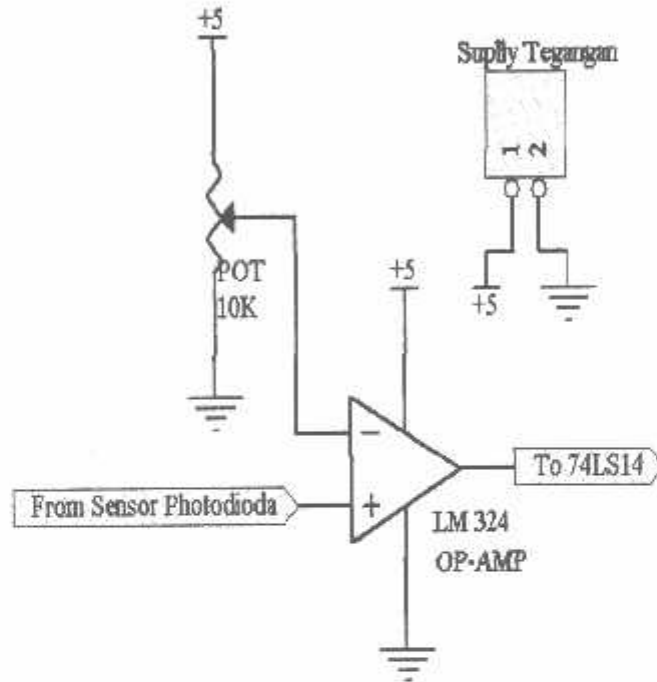
Untuk mencari R2 pada rangkaian sensor diatas adalah :

$$\text{Dik : } VCC = +5V$$

$$I = 100\mu\text{A} = 100 \times 10^{-6} \text{ A}$$

$$\begin{aligned}
 \text{Jadi, } R1 &= \frac{VCC - VD}{I} \\
 &= \frac{5 - 0,7}{100 \times 10^{-6}} \\
 &= \frac{4,3}{100 \times 10^{-6}} \\
 &= 43 \text{K}\Omega = 22 \text{K}\Omega
 \end{aligned}$$

3.1.3. Perancangan Rangkaian *Comparator*



Gambar 3.6. Rangkaian *Comparator*

Pada perancangan rangkaian *Op-Amp*, *Op-Amp* difungsikan sebagai *comparator*. Cara kerja dari rangkaian *Comparator* ini adalah LED akan memancarkan cahaya ke lantai dan *photodiode* akan menerima cahaya yang dipantulkan oleh lantai tersebut. Intensitas cahaya yang diterima oleh *photodiode* akan mempengaruhi nilai *resistansi*-nya. Lantai dan garis memantulkan cahaya dengan intensitas yang berbeda. Garis akan memantulkan cahaya dengan intensitas yang lebih tinggi daripada lantai, sehingga nilai *resistansi*-nya akan berbeda. Semakin besar intensitas cahaya yang diterima oleh *photodiode*, maka nilai

resistansi-nya akan semakin kecil dan nilai tegangan *output*-nya akan semakin kecil pula.

Perbedaan nilai tegangan *output* dari *photodiode* saat menerima cahaya pantulan dari lantai dan garis akan dideteksi oleh rangkaian komparator. Tegangan referensi dapat diatur dengan memutar variabel resistor. Untuk dapat membedakan antara garis dan lantai, nilai tegangan referensi diatur sehingga memiliki nilai diantara nilai tegangan *output* dari *photodiode* saat menerima pantulan cahaya dari lantai dan garis. Pada saat *photodiode* menerima pantulan cahaya dari lantai, nilai tegangan *output*-nya akan lebih kecil dari tegangan referensi, sehingga *output* dari komparator akan bernilai "0". Sedangkan saat *photodiode* menerima pantulan cahaya dari garis, nilai tegangan *output*-nya akan lebih besar dari tegangan referensi, sehingga *output* dari komparator bernilai "1".

Rumus untuk mencari nilai tegangan output :

$$\text{Dik : } A_{OL} = 200.000$$

$$V_{CC} = +5V$$

$$\text{Misal, } V_1 = 3V$$

$$V_2 = 2V$$

$$\text{Jadi, } V_o = A_{OL} \times E_d$$

$$= 200.000 \times (V_1 - V_2)$$

$$= 200.000 \times (3 - 2)$$

$$= 200.000 \times 1$$

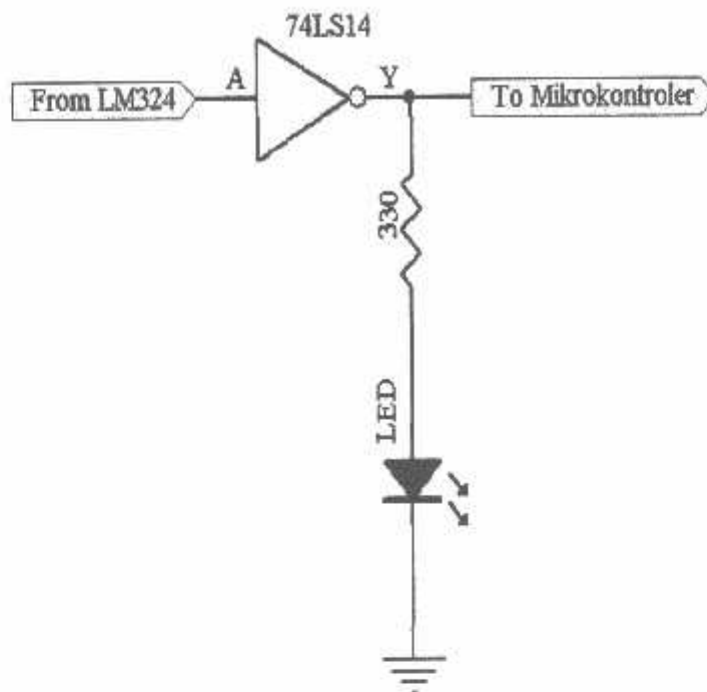
$$= 200.000 > V_{CC}$$

$$V_o \approx V_{sat} = (V_{CC} - 2)$$

$$V_o = 3V$$

3.1.4. Perancangan Rangkaian *Schmitt Trigger*

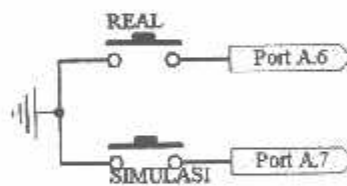
Karena level tegangan keluaran dari rangkaian pengkondisi sinyal masih dalam kondisi ambang, hal ini menyebabkan *mikrokontroler* tidak dapat melakukan instruksi dengan baik, karena input dari pengkondisi sinyal tidak sepenuhnya high, atau tidak sepenuhnya low, maka untuk mengatasi hal ini, digunakan rangkaian *schmitt trigger* dengan menggunakan IC 74LS14 yang bertujuan untuk menegaskan tegangan yang dianggap ambang oleh *mikrokontroler*.



Gambar 3.7. Rangkaian *Schmitt Trigger* 74LS14

3.1.5. Perancangan Rangkaian *Push Button*

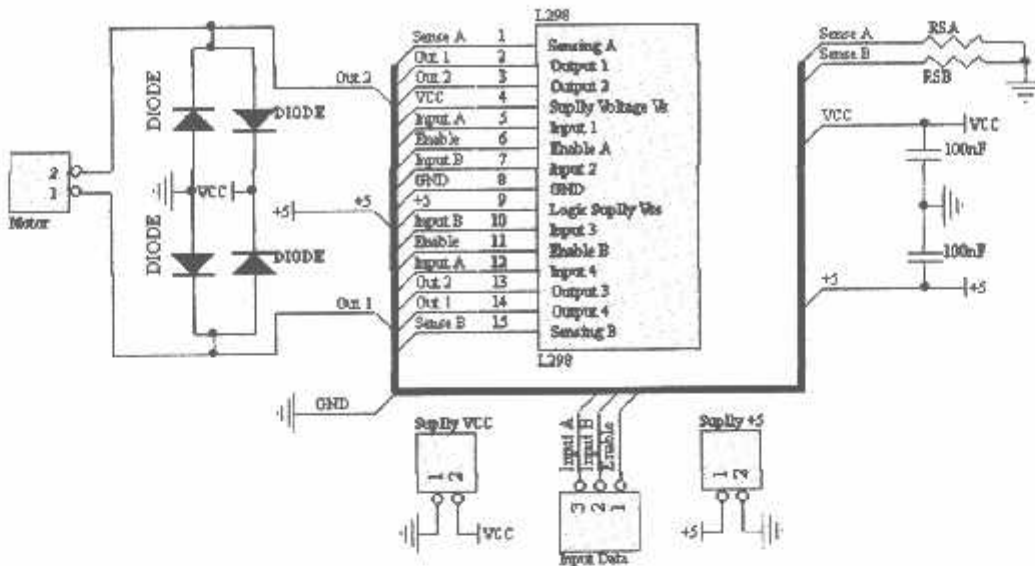
Tombol *push button* digunakan untuk memilih program, apakah untuk program yang sesungguhnya atau untuk *display* simulasi, hal ini bertujuan agar data yang ditampilkan oleh *LCD* bisa terlihat dengan jelas.



Gambar 3.8. Rangkaian *Push Button*

Cara kerjanya, pada saat tombol dipencet, maka port akan terhubung dengan ground, sehingga akan memanggil program real / simulasi yang telah diprogram pada mikrokontroler, dan selanjutnya akan ditampilkan ke *LCD*.

3.1.6. Perancangan Driver L298



Gambar 3.9. Rangkaian Driver L298

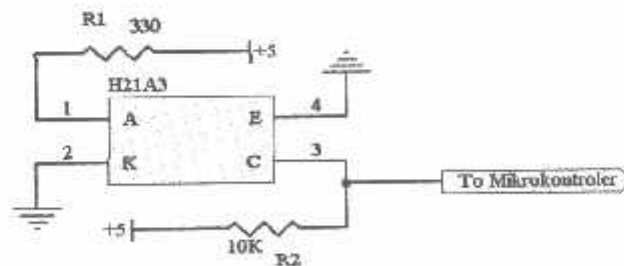
Cara kerja :

Pada saat input A diberi logika "1", input B diberi logika "0" dan enable diberi logika 1, maka motor akan berputar ke kanan. Sebaliknya, pada saat input A diberi logika "0", input B diberi logika "1" dan enable diberi logika "1", maka motor akan berputar ke kiri.

Catatan :

- Berapa pun logika yang di beri pada inputan A dan B, jika enable berlogika "0" maka motor tidak akan berputar
- Pada rangkaian diatas, H-bridge L298 di rancang secara paralel agar di dapat arus maksimum yaitu 4 ampere

3.1.7. Perancangan Rangkaian Rotary Encoder



Gambar 3.10. Rangkaian Rotary Encoder

Untuk memperoleh jarak dari gerakan robot untuk setiap putaran roda yaitu dengan rumus :

Dik : $d = \text{diameter roda} = 8,4 \text{ cm}$

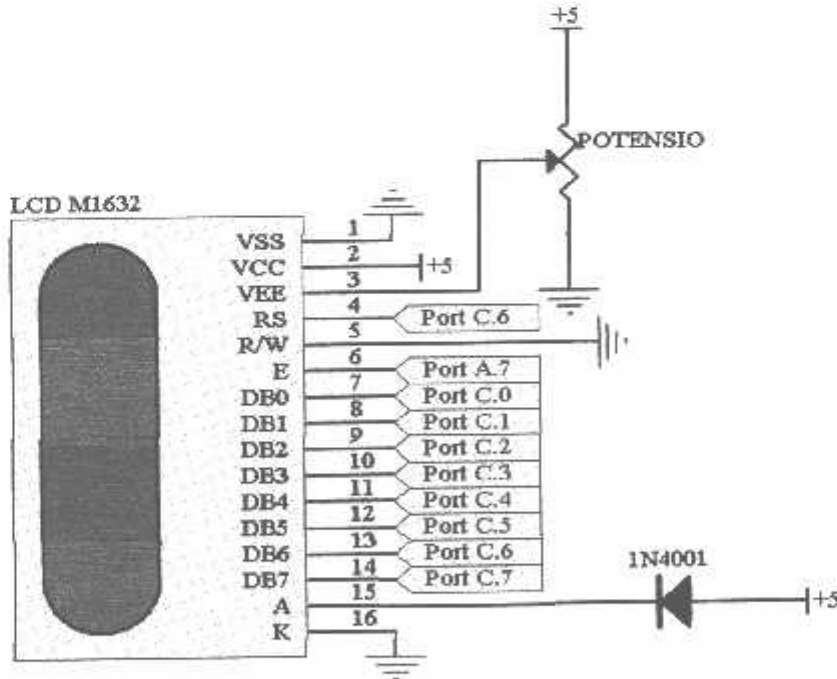
$$\pi = 3,14$$

jumlah lubang/putaran rotary = 36

- $S = \pi \times d$
 $= 3,14 \times 8,4$
 $= 26,376 \text{ cm}$
- Jumlah lubang/putaran roda = $26,376 \text{ cm} / 8,4 \text{ cm} \times (36 \text{ lubang})$
 $= 113 \text{ lubang}$
- Jarak roda/putaran rotary = $26,376 / 3,14 = 8,4 \text{ cm}$
- Resolusi = $S / 113 \text{ lubang}$
 $= 26,376 / 113$
 $= 0,233 \text{ cm} = 2,33 \text{ mm}$

Dengan adanya 36 pulsa untuk setiap putaran rotary maka resolusi untuk setiap pulsa yang ditimbulkan oleh interrupter akan diperoleh jarak roda sebesar $0,233 \text{ cm} = 2,33 \text{ mm}$.

3.1.8. Perancangan Rangkaian LCD



Gambar 3.11. Rangkaian LCD

Keterangan :

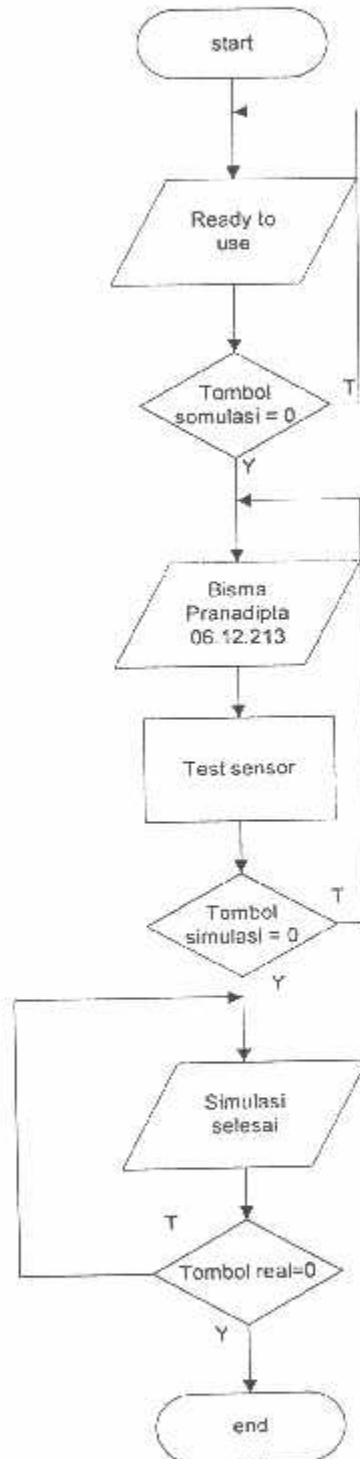
- Kaki 1 (VSS)
Terhubung pada *ground*
 - Kaki 2 (VCC)
Terhubung pada +5
 - Kaki 3 (VEE)
Tegangan pengatur kontras *LCD*. Terhubung pada kaki tengah pada potensio
 - Kaki 4 (RS)
Register *select*, kaki pemilih register yang akan di akses. Untuk akses ke register data, logika dari kaki ini adalah 1, dan untuk akses ke register perintah, logika dari kaki ini adalah 0
 - Kaki 5 (R/W)
Logika pada kaki ini menunjukkan bahwa modul *LCD* sedang pada mode pembacaan dan logika 0 menunjukkan bahwa modul *LCD* sedang pada mode penulisan. Untuk aplikasi yang tidak memerlukan pembacaan data pada modul *LCD*, kaki ini dapat dihubungkan langsung ke *ground*
 - Kaki 6 (E)
Enable clock LCD, kaki ini mengaktifkan *clock LCD*. Logika 1 pada kaki ini diberikan pada saat penulisan atau pembacaan data
 - Kaki 7-14 (D0-D7)
Data bus, kedelapan kaki modul *LCD* ini adalah bagian dimana aliran data sebanyak 4 bit atau 8 bit mengalir saat proses penulisan atau pembacaan data
 - Kaki 15 (Anoda)
Tegangan positif dari *backlight* modul *LCD* sekitar 4,5 volt. Nilai tegangan 4,5 volt didapat dengan rumus:

$$\begin{aligned} \text{Dik : } V_d &= 0,7 \text{ volt} \\ V_{cc} &= 5 \text{ volt} \\ \text{Jadi, } V_p &= V_{cc} - V_d \\ &= 5 - 0,7 \\ &= 4,3 \text{ volt} \end{aligned}$$
 - Kaki 16 (Katoda)
Tegangan negative *backlight* modul *LCD* sebesar 0 volt
-

3.2. Perancangan Perangkat Lunak (*Software*)

Pemrograman *mikrokontroller* menggunakan bahasa *C*, dengan *compiler* menggunakan *codevision*, untuk meng-*upload* program pada *mikrokontroller* menggunakan program *SPI Programmer* dengan komunikasi *serial*.

3.2.1. Flowchart Program Simulasi



Gambar 3.12. Flowchart Program Simulasi



BAB IV

PENGUJIAN ALAT

Pada bab ini membahas cara pengujian dan analisa dari alat yang dirancang, sehingga dapat diketahui apakah alat tersebut dapat bekerja sesuai dengan yang telah direncanakan. Dalam rangka pengujian alat tersebut diuraikan percobaan yang dilakukan untuk mengetahui respon dari keseluruhan alat yang sudah dirancang.

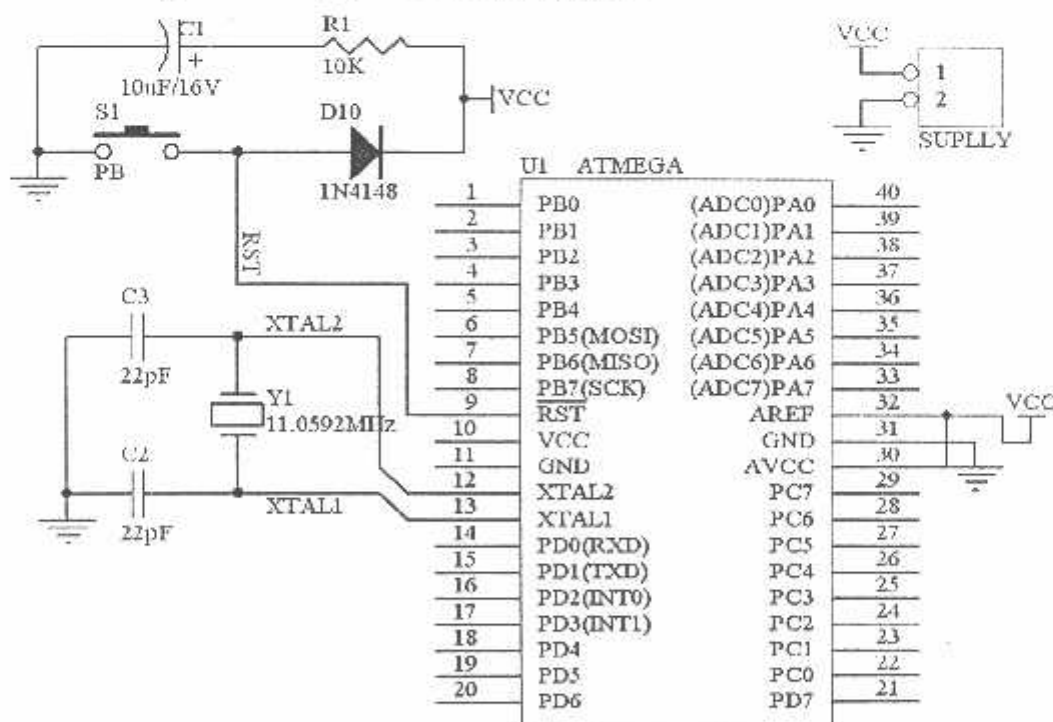
Untuk mengetahui kemampuan alat dan sistem kerja sesuai dengan program yang telah dibuat maka dilakukan pengujian pada alat dan sistem kerja alat dengan prosedur pengujian per blok pada setiap sistem.

4.1. Pengujian *Minimum System* ATMEGA 16

4.1.1. Tujuan

Tujuan dari pengujian rangkaian *Minimum System* ATMEGA 16 adalah untuk menguji data yang telah diprogram pada *mikrokontroler* bisa dieksekusi sesuai dengan alur program.

4.1.2. Rangkaian Pengujian *Minimum System*



Gambar 4.1. Rangkaian Pengujian *Minimum System* ATMEGA 16

4.1.3. Peralatan Yang Digunakan

- Suplly 5V
- Rangkaian Pengujian
- *Multimeter Digital*

4.1.4. Langkah – Langkah Pengujian

- 1) Menghubungkan suplly 5 volt dengan rangkaian *minimum system*
- 2) Memprogram *port mikrokontroller* dengan memberikan *logic "0"* dan menampilkannya pada *led* yang terhubung dengan *port mikrokontroller*
- 3) Mengukur tegangan keluaran dari *port keluaran mikrokontroller*
- 4) Mencatat hasil pengamatan yang dilakukan.

4.1.5. Hasil Pengujian

4.1.5.1. Pengujian Pada *Port A*

Tabel 4.1. Hasil Pengujian Pada *Port A Minimum System*

<i>Port</i>	<i>Logic Masukan (Bit)</i>	Tegangan Keluaran (<i>Volt</i>)	<i>LED Indikator</i>
A.0	1	4.97	Mati
A.1	1	4.98	Mati
A.2	1	4.98	Mati
A.3	1	4.94	Mati
A.4	0	0.20	Nyala
A.5	0	0.20	Nyala
A.6	0	0.22	Nyala
A.7	0	0.21	Nyala

Pada saat *Minimum System* di program dengan *logic masukan "1"*, *led indicator* mati. Sebaliknya pada saat *Minimum System* di program dengan *logic masukan "0"*, *led indicator* nyala. Jadi *Minimum System* yang digunakan disini adalah aktif *low*.



Gambar 4.2. Tegangan Keluaran *Mikrokontroller PortA* Kondisi *High*



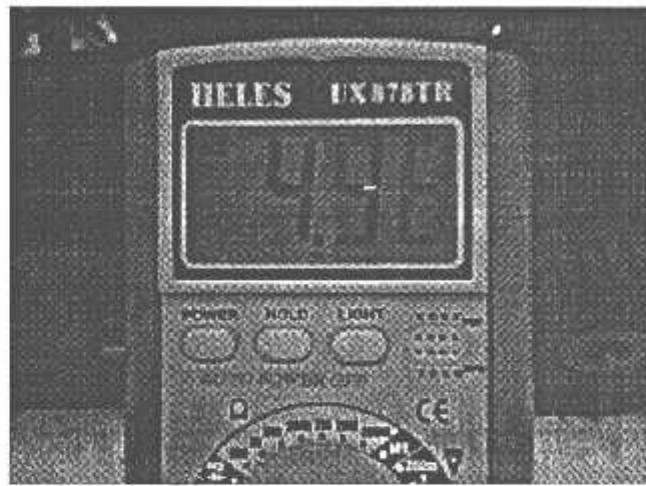
Gambar 4.3. Tegangan Keluaran *Mikrokontroller PortA* Kondisi *Low*

4.1.5.2. Pengujian Pada *Port B*

Tabel 4.2. Hasil Pengujian Pada *Port B Minimum System*

<i>Port</i>	<i>Logic Masukan (Bit)</i>	Tegangan Keluaran (<i>Volt</i>)	<i>LED Indikator</i>
B.0	1	4.94	Mati
B.1	1	4.97	Mati
B.2	1	4.98	Mati
B.3	1	4.95	Mati
B.4	0	0.21	Nyala

B.5	0	0.22	Nyala
B.6	0	0.20	Nyala
B.7	0	0.22	Nyala



Gambar 4.4. Tegangan Keluaran *Mikrokontroller PortB* Kondisi *High*



Gambar 4.5. Tegangan Keluaran *Mikrokontroller PortB* Kondisi *Low*

4.1.5.3. Pengujian Pada *Port C*

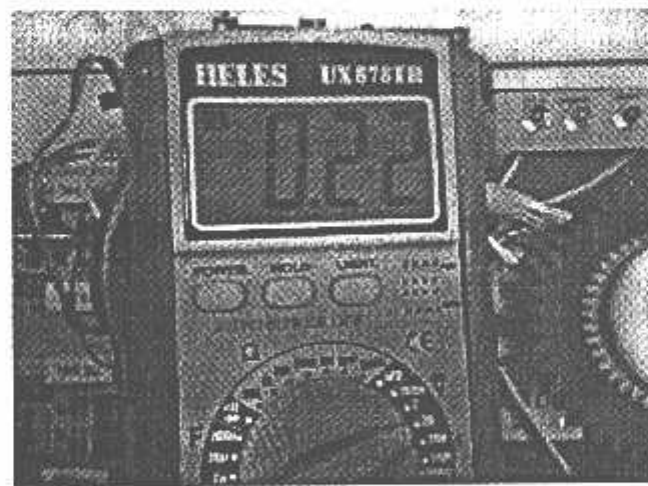
Tabel 4.3. Hasil Pengujian Pada *Port C Minimum System*

<i>Port</i>	<i>Logic Masukan (Bit)</i>	<i>Tegangan Keluaran (Volt)</i>	<i>LED</i>
C.0	1	4.97	Mati
C.1	1	4.97	Mati

C.2	1	4.98	Mati
C.3	1	4.97	Mati
C.4	0	0.21	Nyala
C.5	0	0.22	Nyala
C.6	0	0.22	Nyala
C.7	0	0.21	Nyala



Gambar 4.6. Tegangan Keluaran *Mikrokontroller PortC* Kondisi *High*



Gambar 4.7. Tegangan Keluaran *Mikrokontroller PortC* Kondisi *Low*

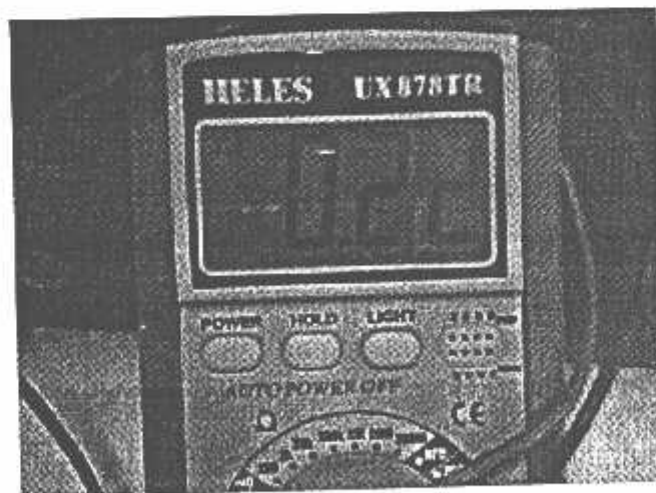
4.1.5.4. Pengujian Pada Port D

Tabel 4.4. Hasil Pengujian Pada Port D Minimum System

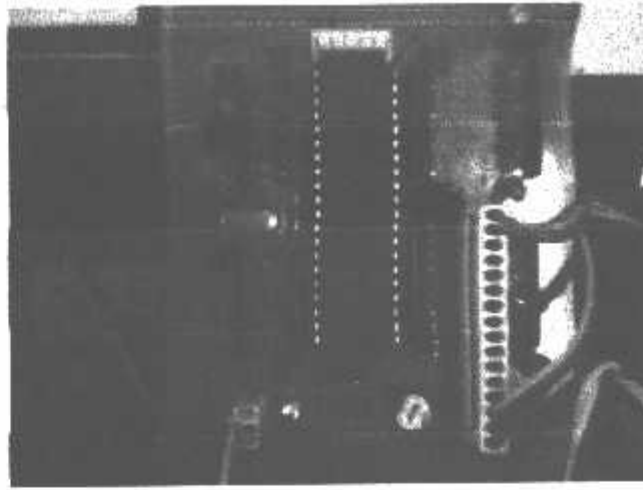
Port	Logic Masukan (Bit)	Tegangan Keluaran (Volt)	LED
D.0	1	4.99	Mati
D.1	1	4.98	Mati
D.2	1	4.98	Mati
D.3	1	4.96	Mati
D.4	0	0.22	Nyala
D.5	0	0.21	Nyala
D.6	0	0.21	Nyala
D.7	0	0.22	Nyala



Gambar 4.8. Tegangan Keluaran Mikrokontroller PortD Kondisi High



Gambar 4.9. Tegangan Keluaran Mikrokontroller PortD Kondisi Low



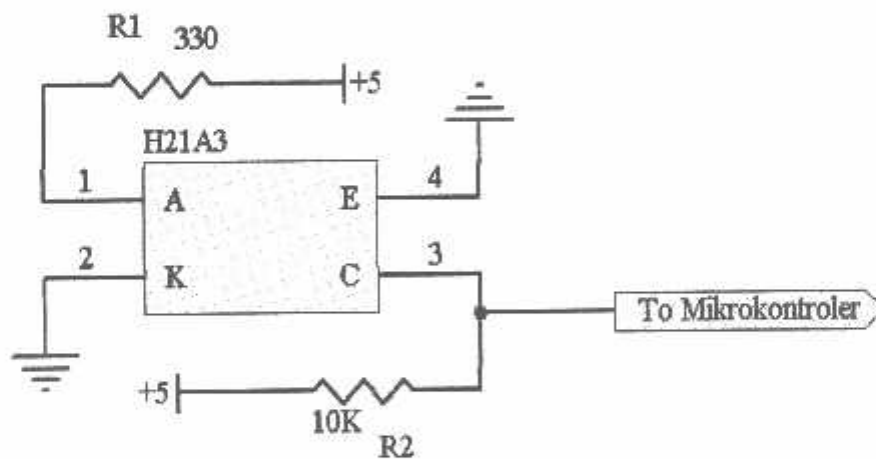
Gambar 4.10. Rangkaian *Mikrokontroller ATMEGA 16*

4.2. Pengujian Rangkaian *Rotary Encoder*

4.2.1. Tujuan

Pengujian ini dilakukan untuk mengetahui pulsa dan tegangan keluaran yang dihasilkan oleh *Rotary Encoder*.

4.2.2. Rangkaian Pengujian *Rotary Encoder*



Gambar 4.11. Rangkaian Pengujian *Rotary Encoder*

4.2.3. Peralatan Yang Digunakan

- *Multimeter digital*
- *Oscilloscope*
- *Suplly 5 volt*

4.2.4. Langkah – Langkah Pengujian

- 1) Menghubungkan rangkaian *rotary encoder* dengan suply 5 Volt
- 2) Hubungkan kabel input data pada *oscilloscope* dengan data keluaran *rotary encoder*
- 3) Mengukur tegangan keluaran *rotary encoder* pada saat terhalang dan pada saat tidak terhalang
- 4) Melihat pulsa keluaran *rotary encoder* pada *oscilloscope*
- 5) Mencatat hasil pengukuran tegangan keluaran *rotary encoder*

4.2.5. Hasil Pengujian Rangkaian *Rotary Encoder*

Tabel 4.5. Hasil Pengujian *Rotary Encoder*

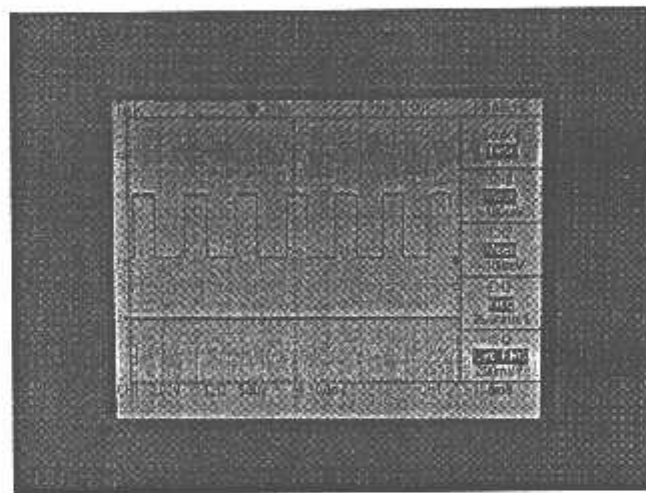
<i>Rotary Encoder</i>	Terhalang (Volt)	Tidak Terhalang (Volt)
Kanan	4,91	0.53
Kiri	4.92	0.54



Gambar 4.12. Tegangan Keluaran *Rotary Encoder* Pada Saat Tidak Terhalang



Gambar 4.13. Tegangan Keluaran *Rotary Encoder* Pada Saat Terhalang



Gambar 4.14. Pulsa Keluaran *Rotary Encoder*



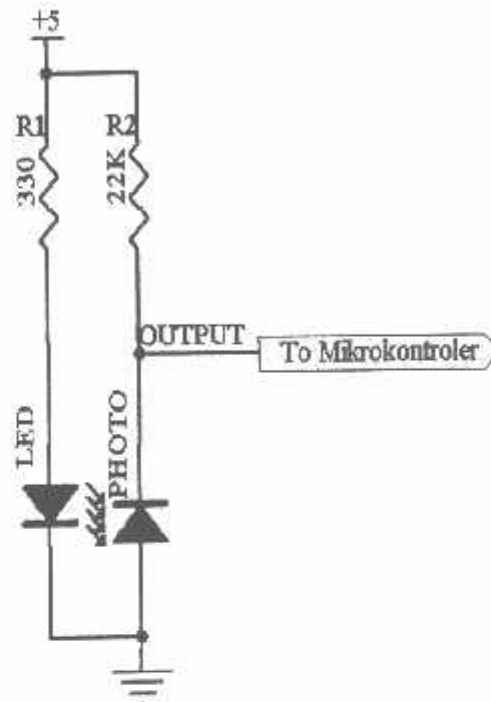
Gambar 4.15. *Rotary Encoder*

4.3. Pengujian Rangkaian Sensor Photodiode

4.3.1. Tujuan

Pengujian ini dilakukan untuk mengetahui tegangan keluaran yang dihasilkan pada saat sensor Photodiode terkena warna putih dan pada saat terkena warna hijau. Yang selanjutnya akan dibandingkan oleh *comparator*.

4.3.2. Rangkaian Pengujian Sensor Photodiode



Gambar 4.16. Rangkaian Pengujian Sensor Photodiode

4.3.3. Peralatan Yang Digunakan

- *Suplly* 5 volt
- *Multimeter digital*
- Arena berwarna hijau dan putih sebagai obyek pembanding warna gelap dan terang

4.3.4. Langkah – Langkah Pengujian

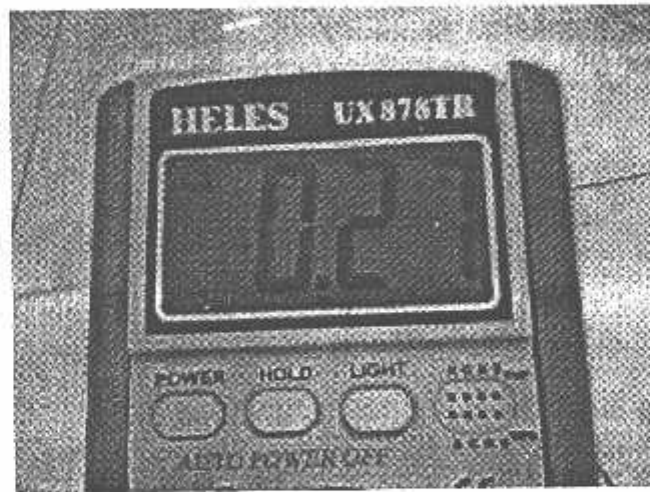
- 1) Hubungkan rangkaian sensor photodiode dengan *suplly* 5 volt
- 2) Mengukur tegangan keluaran sensor photodiode saat sensor photodiode terkena warna putih

- 4) Mencatat hasil pengukuran tegangan keluaran *sensor*

4.3.5. Hasil Pengujian Rangkaian *Sensor*

Tabel 4.6. Hasil Pengujian Rangkaian *Sensor*

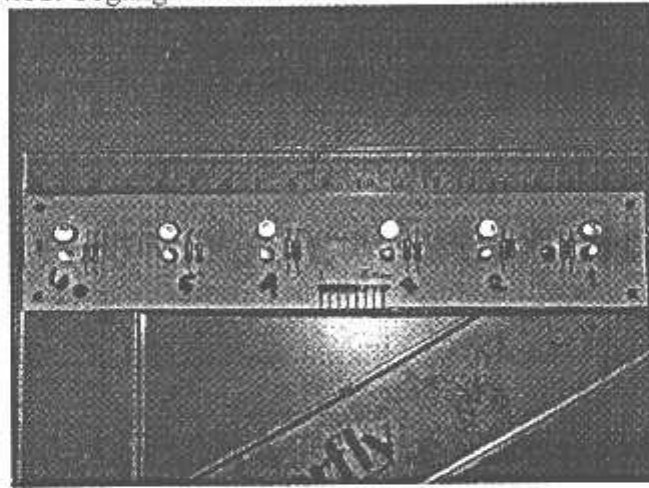
Sensor	V_{out} Sensor Terkena Warna Putih (Volt)	V_{out} Sensor Terkena Warna Hijau (Volt)
1	0.27	2.55
2	0.22	2.55
3	0.24	2.58
4	0.22	2.65
5	0.26	2.59
6	0.23	2.53



Gambar 4.17. Tegangan Keluaran *Sensor* Saat Terkena Warna Putih



Gambar 4.18. Tegangan Keluaran *Sensor* Saat Terkena Warna Hijau



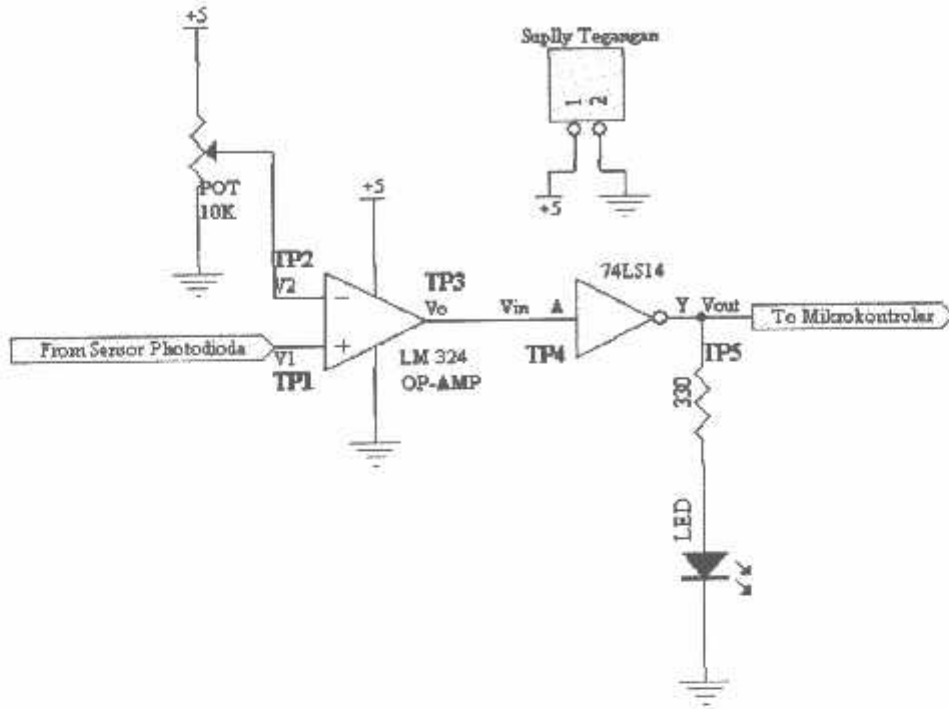
Gambar 4.19. Rangkaian *Sensor*

4.4. Pengujian Rangkaian *Comparator*

4.4.1. Tujuan

Pengujian ini dilakukan untuk mengetahui perbandingan level tegangan saat *sensor* terkena warna putih dan saat *sensor* terkena warna hijau.

4.4.2. Rangkaian Pengujian *Comparator*



Gambar 4.20. Rangkaian Pengujian *Comparator*

4.4.3. Peralatan Yang Digunakan

- *Multimeter Digital*
- *Obeng Trimmer*
- Kabel untuk menghubungkan keluaran sensor photodiode dengan input comparator

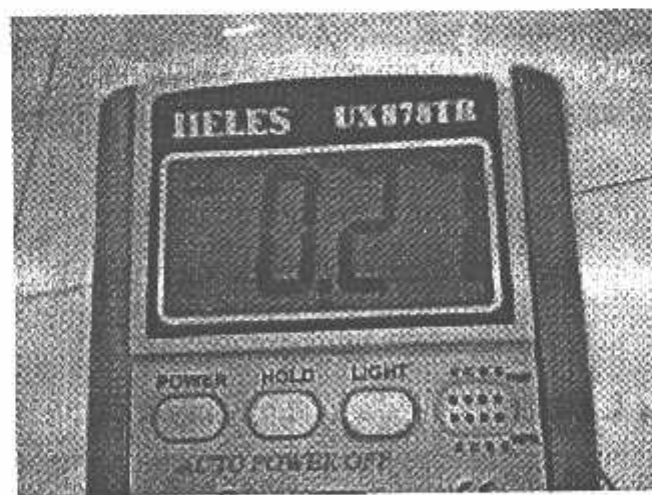
4.4.4. Langkah-Langkah Pengujian

- 1) Menghubungkan rangkaian *comparator* dengan *suplly 5 Volt*
- 2) Menghubungkan input *comparator* dengan output dari *sensor*
- 3) Men-*setting variable resistor* dengan obeng *trimmer*, bergerak memutar hingga indikator led menyala, kemudian memutar sedikit hingga indikator mati, kemudian memutar perlahan hingga indikator led menyala kembali
- 4) Mengukur tegangan pada TP1, TP2, TP3, TP4 dan TP5 pada saat *sensor* terkena warna putih dan terkena warna hijau
- 5) Mencatat hasil pengukuran tegangan keluaran *comparator*

4.4.5. Hasil Pengujian Rangkaian *Comparator*

Tabel 4.7. Hasil Pengujian Rangkaian *Comparator*

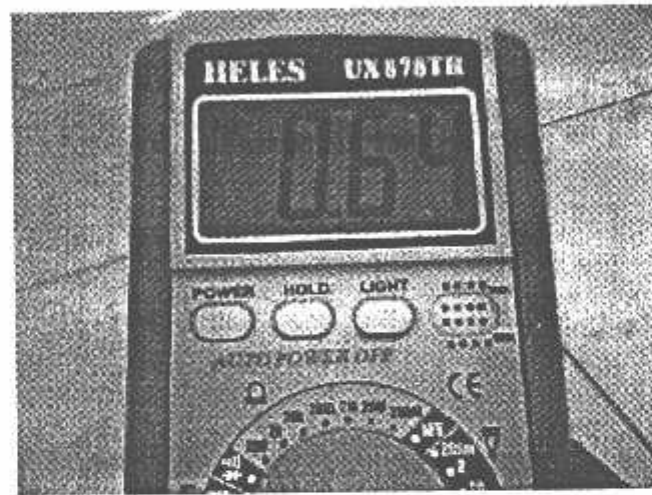
<i>Comparator</i>	<i>Input Comparator V1/TP1 (Volt)</i>	<i>Input Comparator V2/TP2 (Volt)</i>	<i>Output Comparator Vo/TP3 (Volt)</i>	<i>Input Smitch Trigger Vin/TP4 (Volt)</i>	<i>Output Smitch Trigger Vout/TP5 (Volt)</i>
1	2.55	1.27	3.62	3.62	0.66
	0.27	2.75	0.64	0.64	4.92
2	2.55	1.27	3.66	3.66	0.67
	0.22	2.75	0.65	0.65	4.93
3	2.58	1.25	3.66	3.66	0.64
	0.24	2.66	0.64	0.64	4.92
4	2.65	1.25	3.66	3.66	0.67
	0.22	2.70	0.65	0.65	4.93
5	2.59	1.27	3.62	3.62	0.66
	0.26	2.65	0.67	0.67	4.92
6	2.53	1.26	3.66	3.66	0.64
	0.23	2.76	0.65	0.65	4.92



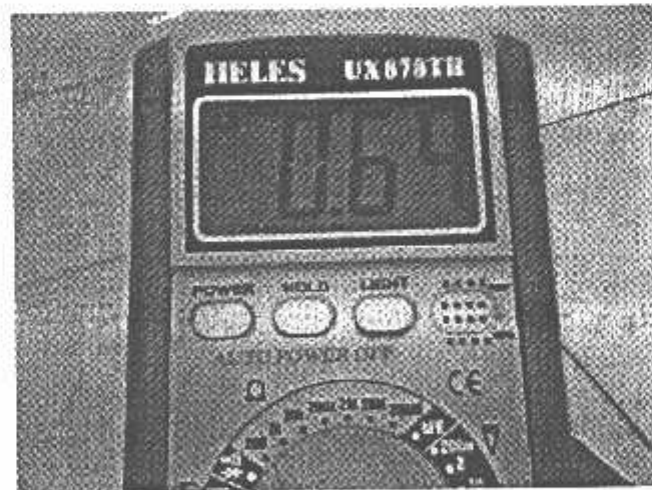
Gambar 4.21. Tegangan Masukan *V1* Pada *Comparator* Saat Sensor Terkena Warna Putih



Gambar 4.22. Tegangan Masukan V_2 Pada *Comparator*



Gambar 4.23. Tegangan Keluaran V_o Pada *Comparator*



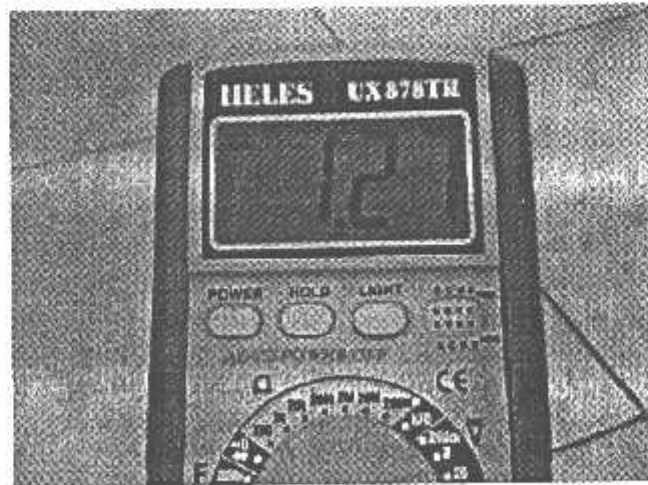
Gambar 4.24. Tegangan Masukan V_{in} Schmitt Trigger Pada Saat Kondisi *Low*



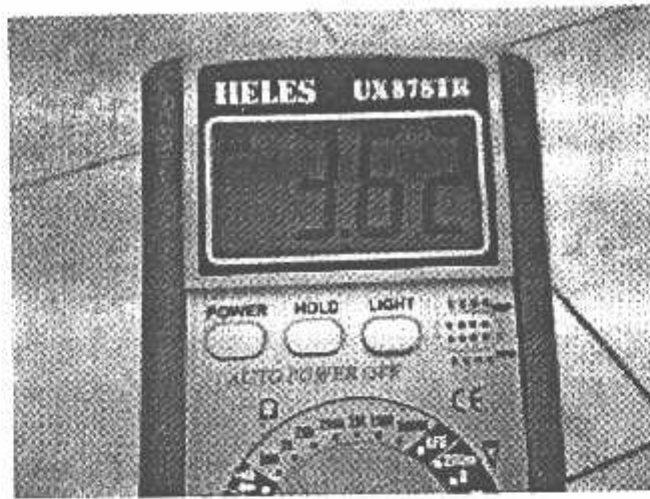
Gambar 4.25. Tegangan Keluaran V_{out} Pada Schmitt Trigger



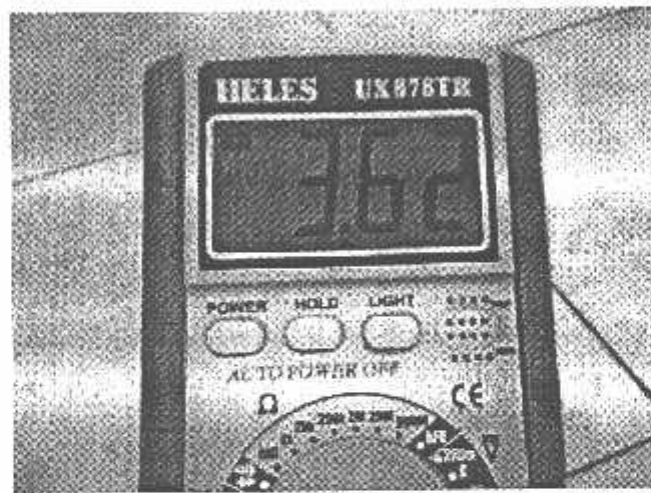
Gambar 4.26. Tegangan Masukan V_1 Pada Comparator Saat Sensor
Terkena Warna Hijau



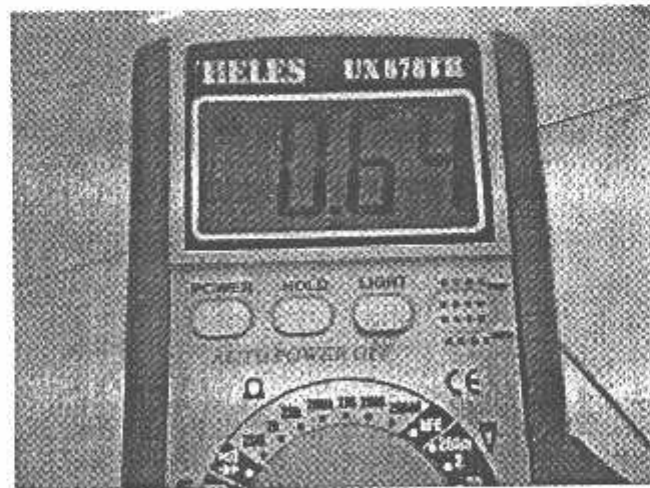
Gambar 4.27. Tegangan Masukan V_2 Pada Comparator



Gambar 4.28. Tegangan Keluaran V_o Pada Comparator



Gambar 4.29. Tegangan Masukan V_{in} Schmitt Trigger Pada Saat Kondisi *High*



Gambar 4.30. Tegangan Keluaran V_{out} Pada Schmitt Trigger

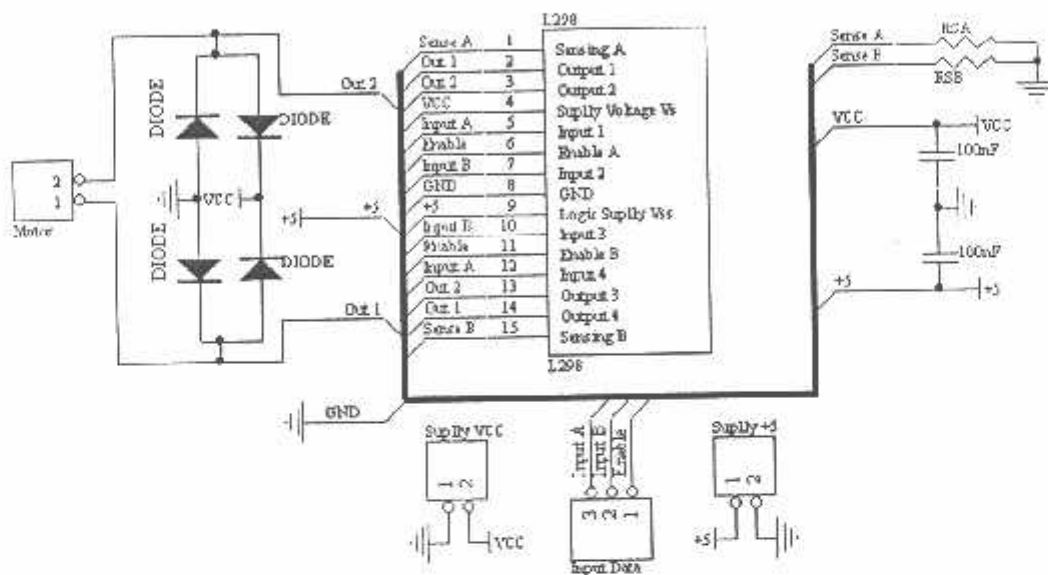
4.5. Pengujian Rangkaian *Driver Motor DC*

4.5.1. Tujuan

Tujuan dari pengujian rangkaian *driver motor* ini adalah untuk mengetahui apakah arah dari motor DC sudah bisa dikendalikan secara program.

Pengujian ini bertujuan juga untuk menganalisa data *pulse width modulation (PWM)* keluaran dari *mikrokontroler*.

4.5.2. Rangkaian Pengujian *Driver Motor DC*



Gambar 4.31. Pengujian Rangkaian *Driver Motor DC*

4.5.3. Peralatan Yang Diperlukan

- Suplly 18 Volt
- Suplly 5 Volt
- Multimeter Digital
- Motor DC
- Oscilloscope

4.5.4. Langkah-Langkah Pengujian Rangkaian *Driver Motor*

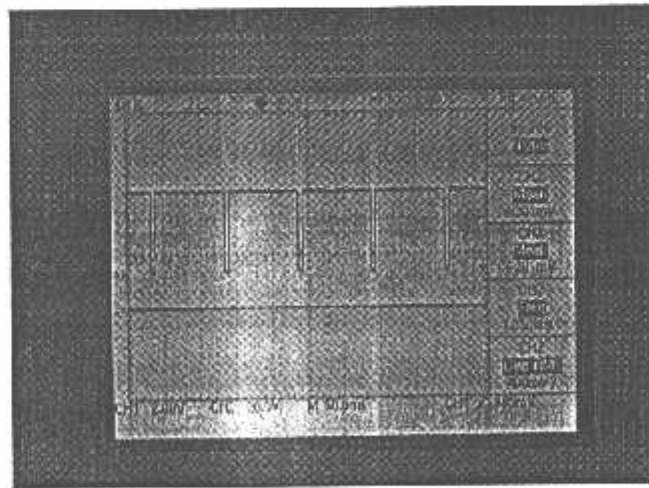
- 1) Menghubungkan rangkaian *driver motor dc* dengan suplly 5 Volt
- 2) Menghubungkan motor dengan suplly 18 Volt
- 3) Mengukur tegangan pada keluaran rangkaian *driver motor dc*
- 4) Menganalisa data *PWM* dengan menampilkannya pada *oscilloscope*

4.5.5. Hasil Pengujian Rangkaian *Driver Motor DC*

Tabel 4.8. Hasil Pengukuran Pada Rangkaian *Driver Motor DC*

Motor DC	<i>Direction (Bit)</i>	<i>Duty Cycle (%)</i>	Arah Putaran Motor	$V_{OUT\ PWM}$ (Volt)
Kanan	1 0	96	Maju	4,46
Kanan	1 0	75	Maju	3,46
Kanan	1 0	50	Maju	2,37
Kanan	1 0	25	Maju	1,26
Kiri	0 1	96	Mundur	4,46
Kiri	0 1	75	Mundur	3,46
Kiri	0 1	50	Mundur	2,37
Kiri	1 0	25	Mundur	1,26

$$\begin{aligned}
 \text{Perhitungan } \textit{duty cycle} &= t_{\text{high}} / (t_{\text{high}} + t_{\text{low}}) \times 100\% \\
 &= 245 / (245 + 10) \times 100\% \\
 &= 96\%
 \end{aligned}$$

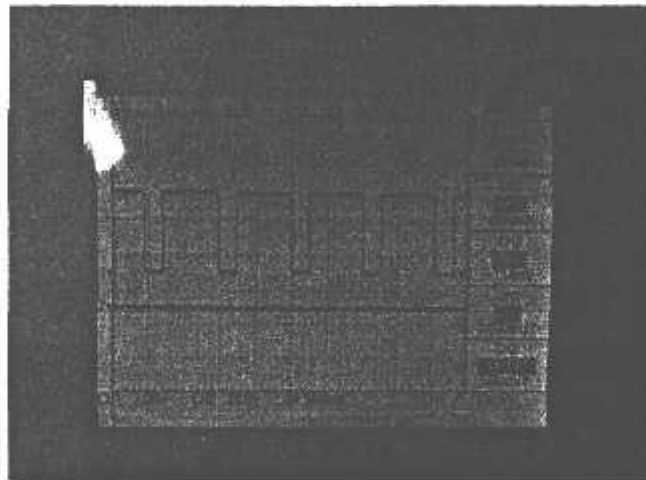


Gambar 4.32. *Enable PWM Digital Dengan Input Duty Cycle 96,078 %*



Gambar 4.33. Tegangan *Enable PWM Digital* Dengan *Input Duty Cycle* 96,078 %

$$\begin{aligned}\text{Perhitungan } \textit{duty cycle} &= t_{\text{high}} / (t_{\text{high}} + t_{\text{low}}) \times 100\% \\ &= 192 / (228 + 27) \times 100\% \\ &= 75\%\end{aligned}$$

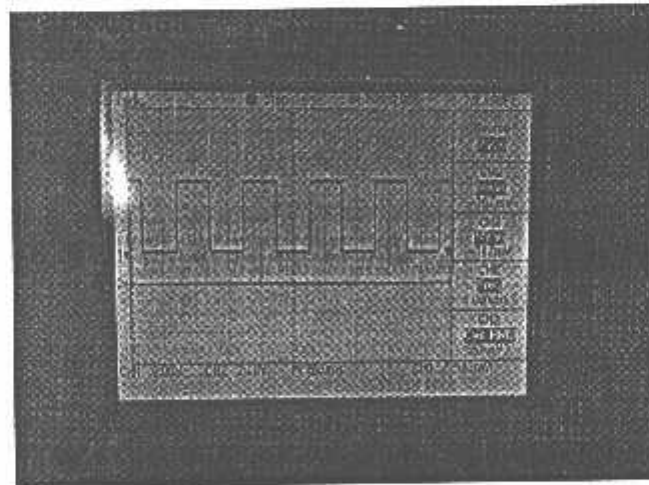


Gambar 4.34. *Enable PWM Digital* Dengan *Input Duty Cycle* 75,294 %



Gambar 4.35. Tegangan *Enable PWM Digital* Dengan *Input Duty Cycle* 75,294 %

$$\begin{aligned}
 \text{Perhitungan } \textit{duty cycle} &= t_{\text{high}} / (t_{\text{high}} + t_{\text{low}}) \times 100\% \\
 &= 128 / (128 + 127) \times 100\% \\
 &= 50 \%
 \end{aligned}$$

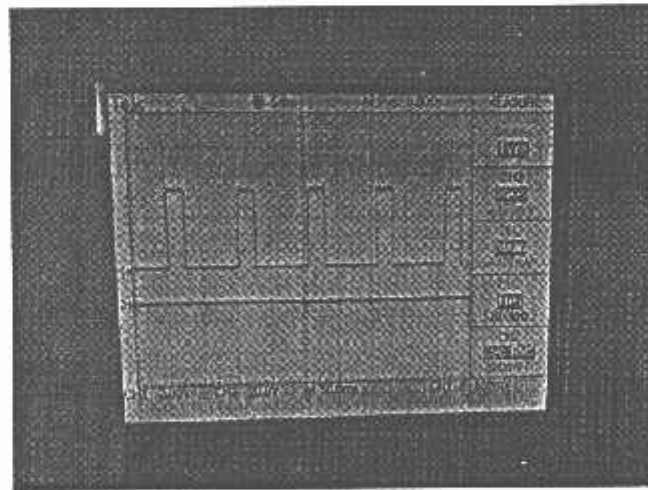


Gambar 4.36. *Enable PWM Digital* Dengan *Input Duty Cycle* 50,19 %



Gambar 4.37. Tegangan *Enable PWM Digital* Dengan *Input Duty Cycle* 50,19 %

$$\begin{aligned}
 \text{Perhitungan } \textit{duty cycle} &= t_{\text{high}} / (t_{\text{high}} + t_{\text{low}}) \times 100\% \\
 &= 64 / (64 + 191) \times 100\% \\
 &= 25 \%
 \end{aligned}$$



Gambar 4.38. *Enable PWM Digital* Dengan *Input Duty Cycle* 25,09%



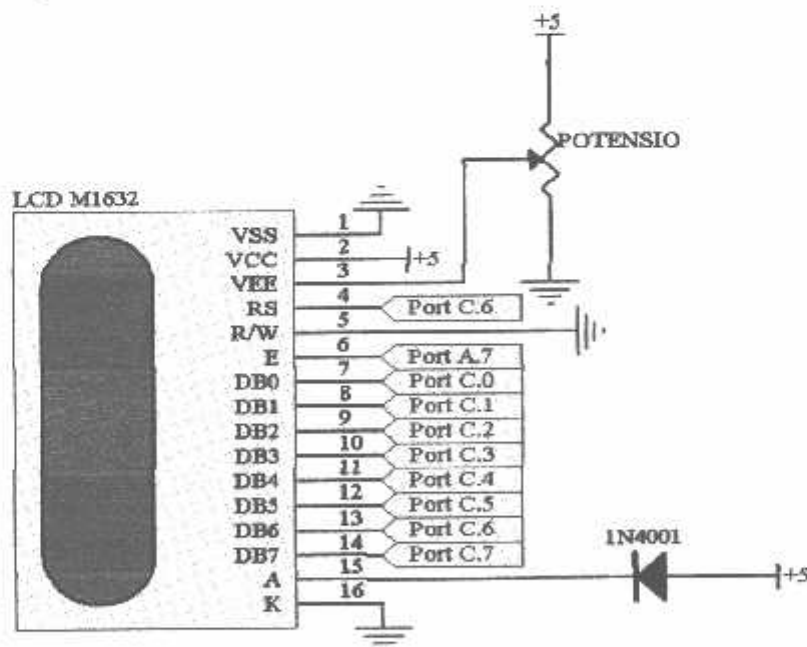
Gambar 4.39. Tegangan *Enable PWM Digital* dengan *Input Duty Cycle 25,09%*

4.6. Pengujian Rangkaian *LCD (Liquid Crstal Display)*

4.6.1. Tujuan

Tujuan dari pengujian rangkaian *LCD* adalah untuk menguji apakah *LCD* dapat menampilkan sesuai yang telah diprogram pada *mikrokontroller*

4.6.2. Rangkaian Pengujian *LCD*



Gambar 4.40. Pengujian Rangkaian *LCD*

4.6.3. Peralatan Yang Diperlukan

- Suply 5 Volt

4.6.4. Langkah-Langkah Pengujian Rangkaian *LCD*

- 1) Menghubungkan rangkaian *LCD* dengan suply 5 Volt
- 2) Menampilkan tulisan “Bisma Pranadipta”, dan pada baris selanjutnya menampilkan “06.12.213”

4.6.5. Hasil Pengujian Rangkaian *LCD*



Gambar 4.41. Pengujian *LCD*

4.7. Pengujian Keseluruhan Sistem

Pada pengujian keseluruhan sistem, terdapat 2 pengujian, yaitu:

- Pengujian rotary encoder dan sensor
- Pengujian *RPM* pada motor DC

4.7.1. Hasil Pengujian Rotary Encoder Dan Sensor

Pada pengujian disini memiliki 2 tahap, yaitu:

- Pengujian pergerakan robot hanya menggunakan sensor counting:
 Pada pengujian ini robot tidak dapat berjalan sesuai program dikarenakan bentuk mekanik dari robot itu sendiri, selain itu intensitas cahaya pada setiap ruangan yang berbeda-beda, sehingga mengganggu kinerja dari sensor tersebut.

- Pengujian pergerakan robot hanya menggunakan rotary encoder:

Pada pengujian ini robot dapat berjalan sesuai program, jarak yang ditempuh tidak sesuai yang diinginkan, hal ini disebabkan karena kelembaman dari motor DC tersebut.

Table 4.9. Hasil Pengujian Pada Rotary Encoder

Rotary Encoder	Counting	Jarak Yang Ditempuh Sesuai Perhitungan (cm)	Jarak Yang Ditempuh Sesuai Pengujian (cm)	Error (cm)
Kanan	216	50	60	10
Kanan	432	100	110	10
Kiri	216	50	60	10
Kiri	432	100	110	10

4.7.2. Hasil Pengujian RPM Pada Motor DC

Pada pengujian ini, membahas hasil pengamatan dimana *RPM* pada motor pada saat terbebani (beban robot = 7kg) dan pada saat tanpa beban.

Tabel 4-10 Hasil Pengujian *RPM* Pada Motor DC

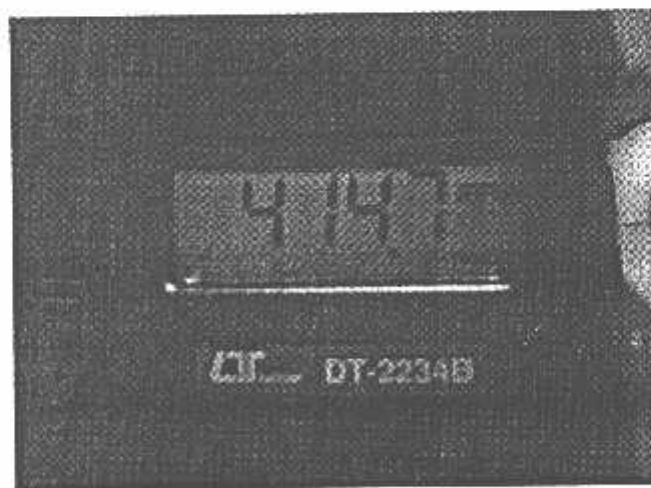
Motor	Beban (Kg)	<i>RPM</i>
Kiri	0	412,1
Kiri	7	347,2
Kanan	0	414,7
Kanan	7	336,3



Gambar 4.42. RPM Motor Kiri Pada Saat Tanpa Beban



Gambar 4.43. RPM Motor Kiri Pada Saat Beban 7Kg



Gambar 4.44. RPM Motor Kanan Pada Saat Tanpa Beban



Gambar 4.45. *RPM* Motor Kanan Pada Saat Beban 7Kg



BAB V

PENUTUP

Pada bab ini penulis akan menyampaikan kesimpulan dan saran setelah melakukan pengujian pada sistem yang telah dibuat

5.1. Kesimpulan

Setelah melakukan perencanaan, pembuatan, dan pengujian, dapat diambil beberapa kesimpulan sebagai berikut :

- *Minimum System* yang digunakan disini adalah aktif *low*.
- *Rotary encoder* sangat efektif untuk menempuh jarak yang diinginkan disebabkan karena sensor *optointerrupter* yang digunakan tidak terpengaruh oleh gangguan cahaya dari luar, sehingga memiliki tingkat keakuratan yang sangat baik.
- Penggunaan sensor photodioda yang difungsikan sebagai *indicator* untuk berbelok, sangat rentan terpengaruh oleh cahaya luar, sehingga diperlukan kalibrasi manual agar robot dapat berjalan dengan baik.
- *Output* dari *comparator* menghasilkan tegangan ambang yang dapat menyebabkan mikrokontroler tidak dapat memproses *output comparator* dengan baik, sehingga digunakan *Schmitt trigger* yang berfungsi sebagai penguat tegangan dan menegaskan masukan sinyal yang berupa sinyal bergerigi atau *sinus* menjadi sinyal kotak yang tegas.
- *LCD* yang digunakan sebagai tampilan program simulasi, dimana sangat penting kegunaannya untuk penulis mengecek apakah sensor photodioda dan *rotary encoder* sudah bekerja dengan baik sebelum robot dijalankan.
- Pergerakan robot lebih halus dengan menggunakan metode *Pulse Width Modulation* sebagai pengendali kecepatan, dimana *pulse width modulation* dibangkitkan oleh program
- Pada nilai *Duty Cycle 25%*, robot tidak dapat bergerak, hal ini dikarenakan oleh beban dari robot tersebut

5.2. Saran

Alat yang dibuat ini masih memiliki keterbatasan, nantinya diharapkan dapat dikembangkan untuk mengatasi keterbatasan itu. Sehingga mendapatkan alat yang diharapkan dapat mendekati alat yang ideal.

- Dalam penggunaan *rotary encoder*, robot bergerak sejauh 's' akan mengalami error 'e' apabila kecepatan gerak motor terlalu cepat, jadi dibutuhkan *PWM* untuk mengurangi nilai *error*. Dimana kecepatan putar motor harus diperlambat agar motor dapat langsung berhenti saat aliran arus dihentikan
 - Bagi peneliti selanjutnya diharapkan bisa menambahkan metode *close loop* dengan menggunakan metode *PID*
 - Masih besarnya beban robot, oleh karena itu sebaiknya pembuatan mekanik robot lebih efisien dan efektif
-



DAFTAR PUSTAKA

- *Datasheet ATMEGA 16*
 - *Datasheet L298*
 - *Datasheet LM324*
 - *Datasheet 74LS14*
 - <http://www.scribd.com>
 - <http://www.robotindonesia.com>
 - www.paralax.com
 - <http://www.fischl.de/ushasp/>
 - http://elkabook.blogspot.com/2010_04_01_archive.html
 - Tim Lab Mikroprosesor. 2006. “ Pemrograman mikrokontroler AT89S51 dengan C/C++ dan Assembler”. Yogyakarta. Andi
 - Heri Andrianto.2008. *Pemrograman Mikrokontroler AVR ATMega 16 Menggunakan Bahasa C (CodeVision AVR)*. Bandung: Informatika Bandung.
-



LAMPIRAN





FORMULIR BIMBINGAN SKRIPSI

Nama : Bisma Pranadipta
Nim : 06.12.213
Masa Bimbingan : 8 Oktober 2010 s/d 8 April 2011
Judul Skripsi : PERANCANGAN DAN PEMBUATAN ROBOT
DENGAN PENGENDALIAN MOTOR DC MENGGUNAKAN
RELATIVE / INCREMENTAL ROTARY ENCODER BERBASIS
ATMEGA 16

NO	TANGGAL	URAIAN	PARAF PEMBIMBING
1	6-12-2010	Bab I, II	
2	20-12-2010	Bab I, II	
3	11-1-2011	Bab III	
4	14-1-2011	Blok Diagram Bab III	
5	19-1-2011	Gambar Rangkaian Bab III Hitam Putih	
6	24-1-2011	Rumus Perhitungan Pada Bab III	
7	27-1-2011	Keterangan Kaki-kaki Pada LCD	
8	1-2-2011	Bab IV	
9	5-2-2011	Pengujian Komparator dan Schmitt Trigger pada Bab IV Jadi Satu	
10	9-2-2011	Pengujian Keseluruhan Bab IV	

Malang,

Dosen Pembimbing I

Dr. Eng. Aryuanto Soetedjo, ST, MT
NIP.P.103.080.0417

Form S-4B



FORMULIR BIMBINGAN SKRIPSI

Nama : Bisma Pranadipta
Nim : 06.12.213
Masa Bimbingan : 8 Oktober 2010 s/d 8 April 2011
Judul Skripsi : PERANCANGAN DAN PEMBUATAN ROBOT
DENGAN PENGENDALIAN MOTOR DC MENGGUNAKAN
RELATIVE / INCREMENTAL ROTARY ENCODER BERBASIS
ATMEGA 16

NO	TANGGAL	URAIAN	PARAF PEMBIMBING
1	6-12-2010	Bab I, II	
2	20-12-2010	Bab I, II	
3	11-1-2011	Bab III	
4	14-1-2011	Blok Diagram Bab III	
5	19-1-2011	Gambar Rangkaian Bab III Hitam Putih	
6	24-1-2011	Rumus Perhitungan Pada Bab III	
7	27-1-2011	Keterangan Kaki-kaki Pada LCD	
8	1-2-2011	Bab IV	
9	5-2-2011	Pengujian Komparator dan Schmitt Trigger pada Bab IV Jadi Satu	
10	9-2-2011	Pengujian Keseluruhan Bab IV	

Malang,

Dosen Pembimbing II

I Komang Somawirata, ST, MT
NIP.Y.103.010.0361

Form S-4B



FORMULIR PERBAIKAN SKRIPSI

Dalam pelaksanaan ujian skripsi jenjang program Strata satu (S-1) Jurusan Teknik Elektro Konsentrasi Teknik Elektronika, maka perlu adanya perbaikan skripsi untuk mahasiswa :

Nama : Bisma Pranadipta
NIM : 06.12.213
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Elektronika
Masa Bimbingan : 8 Oktober 2010 s/d 8 April 2011
Judul Skripsi : PERANCANGAN DAN PEMBUATAN ROBOT DENGAN PENGENDALIAN MOTOR DC MENGGUNAKAN *RELATIVE INCREMENTAL ROTARY ENCODER* BERBASIS *ATMEGA 16*

Tanggal	Uraian	Paraf
Penguji Pertama 17 Februari 2011	<ul style="list-style-type: none">Abstrak terdiri 3 paragraf : masalah, pemecahan, hasilGambar rangkaian input / output disesuaikanSensor counting diganti sensor photodiode	
Penguji Kedua 17 Februari 2011		

Mengetahui,

Dosen Pembimbing I

Dr. Eng. Arnyanto Soetedjo, ST, MT
NIP. Y. 103.080.0417

Dosen Pembimbing II

I Komang Somawirata, ST, MT
NIP. P. 103.010.0361

Dosen Penguji,

Penguji Pertama

Irmalia Suryani Faradisa, ST, MT
NIP. P. 103.000.0365



Penguji Kedua

Ir. Eko Nurcahyo
NIP. Y. 103.870.0172



LEMBAR PENGAJUAN JUDUL SKRIPSI JURUSAN TEKNIK ELEKTRO S-1

Konsentrasi : Teknik Energi Listrik/Teknik Elektronika/Teknik Komputer & Informatika*)

1.	Nama Mahasiswa : Bisma Pranadipta	Nim : 06.12.213
2.	Waktu Pengajuan	Tanggal : Bulan : Tahun :
3.	Spesifikasi Judul (berilah tanda silang)**)	
	a. Sistem Tenaga Elektrik b. Energi & Konversi Energi c. Tegangan Tinggi & Pengukuran d. Sistem Kendali Industri	e. Elektronika & Komponen f. Elektronika Digital & Komputer g. Elektronika Komunikasi h. lainnya.....
4.	Konsultasikan judul sesuai materi bidang ilmu kepada Dosen*) <i>Dr. Arsyanto, ST, MT</i>	Ketua Jurusan  Ir. Yusuf Ismail Makhoda, MT NIP. Y. 101.890.0189
5.	Judul yang diajukan mahasiswa :	DEKODIFIKASI DAN PEMBUATAN PUSK... DENGAN... PENDEKIDALAN... MUDA DI... MENGENAL... RELATIVE/ M... ELEMENTAL POTAPY ENCODER BERBASIS ATMEKAL
6.	Perubahan judul yang disetujui Dosen sesuai materi bidang ilmu
Catatan :		
7.	Persetujuan judul skripsi yang dikonsultasikan kepada Dosen materi Bidang ilmu	Disetujui 30/8/20 Dosen 

Perhatian :

1. Formulir pengajuan ini harap dikembalikan kepada jurusan paling lambat satu minggu setelah disetujui kelompok dosen keahlian dengan dilampirkan proposal skripsi beserta persyaratan skripsi sesuai form S-1
2. Keterangan : *) Coret yang tidak perlu
 **) dilingkari a, b, c, atau g sesuai dengan keahlian

Lampiran : 1 (satu) berkas
Pembimbing Skripsi

Kepada : Yth. Bapak Dr. Eng. Aryuanto Soetedjo, ST, MT
Dosen Institut Teknologi Nasional
M A L A N G

Yang bertanda tangan dibawah ini :

Nama : Bisma Pranadipta
Nim : 06.12.213
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Elektronika

Dengan ini mengajukan permohonan, kiranya Bapak/Ibu bersedia menjadi Dosen Pembimbing utama/pendamping *), untuk penyusunan Skripsi dengan judul (proposal terlampir) :

**PERANCANGAN DAN PEMBUATAN
ROBOT DENGAN PENGENDALIAN MOTOR DC MENGGUNAKAN
RELATIVE / INCREMENTAL ROTARY ENCODER
BERBASIS *ATMEGA16***

Adapun tugas tersebut sebagai salah satu syarat untuk menempuh Ujian Akhir Sarjana Teknik.
Demikian permohonan kami dan atas kesediaan Bapak kami ucapkan terima kasih.

Malang,

Mengetahui,
Ketua Jurusan Teknik Elektro

Ir. Yusuf Ismail Nakhoda, MT
NIP.Y.101.880.0189

Hormat kami



Bisma Pranadipta
NIM. 06.12.213

*) Coret yang tidak perlu

Lampiran : 1 (satu) berkas
Pembimbing Skripsi

Kepada : Yth. Bapak I Komang Somawirata, ST, MT
Dosen Institut Teknologi Nasional
MALANG

Yang bertanda tangan dibawah ini :

Nama : Bisma Pranadipta
Nim : 06.12.213
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Elektronika


Dengan ini mengajukan permohonan, kiranya Bapak/Ibu bersedia menjadi Dosen Pembimbing utama/pendamping *), untuk penyusunan Skripsi dengan judul (proposal terlampir) :

PERANCANGAN DAN PEMBUATAN
ROBOT DENGAN PENGENDALIAN MOTOR DC MENGGUNAKAN
RELATIVE / INCREMENTAL ROTARY ENCODER
BERBASIS ATMEGA16

Adapun tugas tersebut sebagai salah satu syarat untuk menempuh Ujian Akhir Sarjana Teknik.
Demikian permohonan kami dan atas kesediaan Bapak kami ucapkan terima kasih.

Malang,

Mengetahui,
Ketua Jurusan Teknik Elektro


Ir. Yusuf Ismail Nakhoda, MT
NIP.Y.101.880.0189

Hormat kami


Bisma Pranadipta
NIM. 06.12.213

*) Coret yang tidak perlu

Form S-3 a

PERNYATAAN KESEDIAAN DALAM PEMBIMBINGAN SKRIPSI

Sesuai permohonan dari mahasiswa/i :

Nama : Bisma Pranadipta
Nim : 06.12.213
Semester : IX (Sembilan)
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Elektronika


Dengan ini menyatakan bersedia / tidak bersedia *) Membimbing Skripsi dari mahasiswa tersebut, dengan judul :

**PERANCANGAN DAN PEMBUATAN
ROBOT DENGAN PENGENDALIAN MOTOR DC MENGGUNAKAN
RELATIVE / INCREMENTAL ROTARY ENCODER
BERBASIS ATMEGA16**

Demikian surat pernyataan ini kami buat agar dapat dipergunakan seperlunya.

Malang, 20/9/20

Kami yang Membuat pernyataan,


Dr. Eng. Aryuanto Soetedjo, ST, MT
NIP.Y. 1030800417

Catatan :

Setelah disetujui agar formulir ini diserahkan mahasiswa/i yang bersangkutan kepada Jurusan untuk diproses lebih lanjut.

*) Coret yang tidak perlu

Form S-3 b

PERNYATAAN KESEDIAAN DALAM PEMBIMBINGAN SKRIPSI

Sesuai permohonan dari mahasiswa/i :

Nama : Bisma Pranadipta
Nim : 06.12.213
Semester : IX (Sembilan)
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Elektronika

Dengan ini menyatakan bersedia / tidak bersedia *) Membimbing Skripsi dari mahasiswa tersebut, dengan judul :

**PERANCANGAN DAN PEMBUATAN
ROBOT DENGAN PENGENDALIAN MOTOR DC MENGGUNAKAN
RELATIVE / INCREMENTAL ROTARY ENCODER
BERBASIS ATMEGA16**

Demikian surat pernyataan ini kami buat agar dapat dipergunakan seperlunya.

Malang,

Kami yang membuat pernyataan,



I Komang Somawirata, ST, MT
NIP.Y. 103.010.0361

Catatan :

Setelah disetujui agar formulir ini diserahkan mahasiswa/i yang bersangkutan kepada Jurusan untuk diproses lebih lanjut.

*) Coret yang tidak perlu

Form S-3 b



BERITA ACARA SEMINAR PROPOSAL SKRIPSI JURUSAN TEKNIK ELEKTRO S-1

Konsentrasi: Teknik Energi Listrik/Teknik Elektronika/Teknik Komputer & Informatika*)

1.	Nama Mahasiswa: <u>BISMA PRANADIPTA</u>	Nim: <u>0612213</u>	
2.	Keterangan	Tanggal	
	Pelaksanaan	<u>08-10-2010</u>	
Tempat			
Ruang:			
Spesifikasi Judul (berilah tanda silang)**)			
3.	a. Sistem Tenaga Elektrik	<input checked="" type="checkbox"/> e. Elektronika & Komponen	
	b. Energi & Konversi Energi	<input type="checkbox"/> f. Elektronika Digital & Komputer	
	c. Tegangan Tinggi & Pengukuran	<input type="checkbox"/> g. Elektronika Komunikasi	
	d. Sistem Kendali Industri	<input type="checkbox"/> h. lainnya	
4.	Judul Proposal yang diseminarkan Mahasiswa	<u>PERANCANGAN DAN PEMBUATAN ROBOT DENGAN PENGEENDALIAN MOTOR DC MENGGUNAKAN RELATIVE / INCREMENTAL ROTARY ENCODER BERBAHAI ATMGA 16</u>	
5.	Perubahan judul yang diusulkan oleh Kelompok Dosen Keahlian	
6.	Catatan:	
	Catatan:	
Persetujuan Judul Skripsi			
7.	Disetujui, Dosen Keahlian	Disetujui, Dosen Keahlian II	
	Mengetahui, Ketua Jurusan	Disetujui, Calon Dosen Pembimbing ybs	
	 <u>Ir. Yusuf Ismail Nakhoda, MT</u> NIP. Y. 1018600189	Pembimbing I	Pembimbing II

Perhatian

1. Keterangan: *) Coret yang tidak perlu
- **) dilingkari a, b, c, atau g sesuai bidang keahlian

Form S-3c



PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAN TEKNOLOGI NASIONAL MALANG
INSTITUT TEKNOLOGI NASIONAL MALANG

FAKULTAS TEKNOLOGI INDUSTRI
FAKULTAS TEKNIK SIPIL DAN PERENCANAAN
PROGRAM PASCASARJANA MAGISTER TEKNIK

RSEROJ MALANG
IAGA MALANG

Kampus I : Jl. Bendungan Sigura-gura No. 2 Telp. (0341) 551431 (Hunting), Fax. (0341) 553015 Malang 65145
Kampus II : Jl. Raya Karanglo, Km 2 Telp. (0341) 417636 Fax. (0341) 417634 Malang

Malang, 25 Oktober 2010

Nomor : ITN- 971/7/TA /2010
Lampiran :
Perihal : Bimbingan Skripsi

Kepada : Yth. Sdr. DR. ARYUANTO S, ST, MT
Dosen Pembimbing
Jurusan Teknik Elektro S-1
di
Malang

Dengan hormat,
Sesuai dengan permohonan dan persetujuan dalam proposal skripsi
untuk mahasiswa:

Nama : RISMA PRANADIPTA
Nim : 06 12 213
Fakultas : Teknologi Industri
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Elektronika S-1

Maka dengan ini pembimbingan tersebut kami serahkan sepenuhnya
kepada Saudara/i selama masa waktu 6 (enam) bulan, terhitung mulai
tanggal:

08 OKTOBER 2010 s/d 08 APRIL 2011

Sebagai satu syarat untuk menempuh Ujian sarjana.
Demikian atas perhatian serta kerjasama yang baik kami ucapkan
terima kasih

Ketua Jurusan
Teknik Elektro S-1

Dr. Yusuf Smail Nakhoda, MT
NIP. Y. 1018800189

Tindakan:

1. Mahasiswa yang Bersangkutan
2. Arsip

Form S-4a



PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAN TEKNOLOGI NASIONAL MALANG
INSTITUT TEKNOLOGI NASIONAL MALANG

FAKULTAS TEKNOLOGI INDUSTRI
FAKULTAS TEKNIK SIPIL DAN PERENCANAAN
PROGRAM PASCASARJANA MAGISTER TEKNIK

BERSERIK MALANG
JALAN MALANG

Kampus I : Jl. Bendungan Sigura-gura No. 2 Telp. (0341) 551431 (Hunting), Fax. (0341) 553015 Malang 65145
Kampus II : Jl. Raya Karanglo, Km 2 Telp. (0341) 417636 Fax. (0341) 417634 Malang

Malang, 25 Oktober 2010

Nomor : ITN- 970/7/TA /2010
Lampiran :
Perihal : Bimbingan Skripsi

Kepada : Yth. Sdr. I KOMANG SOMAWIRATA, ST, MT
Dosen Pembimbing
Jurusan Teknik Elektro S-1
di
Malang

Dengan hormat,
Sesuai dengan permohonan dan persetujuan dalam proposal skripsi
untuk mahasiswa:

Nama : BISMA PRANALIPTA
Nim : 06 12 213
Fakultas : Teknologi Industri
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Elektronika S-I

Maka dengan ini pembimbingan tersebut kami serahkan sepenuhnya
kepada Saudara/i selama masa waktu 6 (enam) bulan, terhitung mulai
tanggal:

08 OKTOBER 2010 s/d 08 APRIL 2011

Sebagai satu syarat untuk menempuh Ujian sarjana.
Demikian atas perhatian serta kerjasama yang baik kami ucapkan
terima kasih


Ketua Jurusan
Teknik Elektro S-1
M. Yusuf Ismail Nakhoda, MT
NIP. Y. 1018800189

Tindakan:

1. Mahasiswa yang Bersangkutan
2. Arsip

Form S-4a

LISTING PROGRAM

```
/*  
This program was produced by the  
CodeWizardAVR V1.24.8d Professional  
Automatic Program Generator  
© Copyright 1998-2006 Pavel Haiduc, HP InfoTech s.r.l.  
http://www.hpinfotech.com
```

```
Project :  
Version :  
Date : 1/29/2011  
Author : F4CG  
Company : F4CG  
Comments:
```

```
Chip type : ATmega16  
Program type : Application  
Clock frequency : 11.059200 MHz  
Memory model : Small  
External SRAM size : 0  
Data Stack size : 256
```

```
*****/
```

```
#include <mega16.h>  
#include <delay.h>  
#include <stdio.h>
```

```
// Alphanumeric LCD Module functions
```

```
#asm  
 .equ __lcd_port=0x18 ;PORTB  
#endasm  
#include <lcd.h>
```

```
#define R_data1 PORTC.0  
#define R_data2 PORTC.1  
#define R_pwm PORTC.2
```

```
#define L_data1 PORTC.7  
#define L_data2 PORTC.6  
#define L_pwm PORTC.5
```

```
#define mtr_naik1 PORTA.1  
#define mtr_naik2 PORTA.2
```

```
#define mtr_buka1 PORTA.3  
#define mtr_huka2 PORTA.4
```

```
##define sens_1 PIND.7  
##define sens_2 PIND.6  
#define sens_3 PIND.7  
#define sens_4 PIND.6
```

```

// #define sens_5 PIND.3
// #define sens_6 PIND.2

#define rot_kiri PIND.0
#define rot_kanan PIND.1

#define tombol_turun PIND.2
#define tombol_naik PIND.3

#define tombol_tutup PIND.4
#define tombol_buka PIND.5

#define tombol_sim PINA.7
#define tombol_real PINA.6

#define servo PORTA.0

unsigned char lpwm_value = 0;
unsigned char rpwm_value = 0;
unsigned char servo_value = 0;
unsigned char loop = 0;

unsigned int tenth_of_sec = 0;
unsigned int count_rotki = 0;
unsigned int count_rotka = 0;
unsigned char line_count = 0;

bit L1_value = 0;
bit L2_value = 0;
bit R1_value = 0;
bit R2_value = 0;

bit recent_kiri;
bit recent_kanan;

bit do_count = 0;
bit enable_pwm = 0;
bit flag = 0;

// Timer 0 overflow interrupt service routine
interrupt [TIM0_OVF] void timer0_ovf_isr(void)
{
    // Reinitialize Timer 0 value
    TCNT0=0xD4;

    loop++;

    if (loop < lpwm_value){
        L_data1 = L1_value;
        L_data2 = L2_value;
    }
}

```

```

}
else {
    L_data1 = L_data2 = 0;
}

if (loop < rpwm_value) {
    R_data1 = R1_value;
    R_data2 = R2_value;
}
else {
    R_data1 = R_data2 = 0;
}

if ((loop % 20 != 0) && (loop % 20 <= servo_value)) {
    servo = 1;
}
else {
    servo = 0;
}

if (loop >= 100) {
    loop = 0;
    tenth_of_sec++;
}
}

// Timer 2 overflow interrupt service routine
interrupt [TIM2_OVF] void timer2_ovf_isr(void)
{
    // Reinitialize Timer 2 value
    TCNT2=0xF5;
    if (do_count == 1) {
        if (rot_kiri == 0) {
            if (recent_kiri == 1) {
                count_rotki++;
            }
            recent_kiri = 0;
        }
        else {
            recent_kiri = 1;
        }

        if (rot_kanan == 0) {
            if (recent_kanan == 1) {
                count_rotka++;
            }
            recent_kanan = 0;
        }
        else {
            recent_kanan = 1;
        }
    }
}

```

```

    }
}

if (sens_3 == 0 || sens_4 == 0) {
    if (flag == 0) {
        line_count++;
        flag = 1;
    }
}
else {
    flag = 0;
}

}

void reset_counter() {
    count_rotka = 0;
    count_rotki = 0;
    line_count = 0;
}

// Declare your global variables here

void main(void)
{
    // Declare your local variables here
    int i;
    bit simulation = 0;

    char s_count[16];

    // Input/Output Ports initialization
    // Port A initialization
    // Func7=In Func6=In Func5=Out Func4=Out Func3=Out Func2=Out Func1=Out
    // Func0=Out
    // State7=T State6=T State5=0 State4=0 State3=0 State2=0 State1=0 State0=0
    PORTA=0x00;
    DDRA=0x3F;

    // Port B initialization
    // Func7=Out Func6=Out Func5=Out Func4=Out Func3=Out Func2=Out Func1=Out
    // Func0=Out
    // State7=0 State6=0 State5=0 State4=0 State3=0 State2=0 State1=0 State0=0
    PORTB=0x00;
    DDRB=0xFF;

    // Port C initialization
    // Func7=Out Func6=Out Func5=Out Func4=Out Func3=Out Func2=Out Func1=Out
    // Func0=Out
    // State7=0 State6=0 State5=0 State4=0 State3=0 State2=0 State1=0 State0=0

```

```
PORTC=0x00;
DDRC=0xFF;

// Port D initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T State0=T
PORTD=0x00;
DDRD=0x00;

// Timer/Counter 0 initialization
// Clock source: System Clock
// Clock value: 43.200 kHz
// Mode: Normal top=FFh
// OC0 output: Disconnected
TCCR0=0x04;
TCNT0=0xD4;
OCR0=0x00;

// Timer/Counter 1 initialization
// Clock source: System Clock
// Clock value: Timer 1 Stopped
// Mode: Normal top=FFFFh
// OC1A output: Discon.
// OC1B output: Discon.
// Noise Canceler: Off
// Input Capture on Falling Edge
// Timer 1 Overflow Interrupt: Off
// Input Capture Interrupt: Off
// Compare A Match Interrupt: Off
// Compare B Match Interrupt: Off
TCCR1A=0x00;
TCCR1B=0x00;
TCNT1H=0x00;
TCNT1L=0x00;
ICR1H=0x00;
ICR1L=0x00;
OCR1AH=0x00;
OCR1AL=0x00;
OCR1BH=0x00;
OCR1BL=0x00;

// Timer/Counter 2 initialization
// Clock source: System Clock
// Clock value: 11059.200 kHz
// Mode: Normal top=FFh
// OC2 output: Disconnected
ASSR=0x00;
TCCR2=0x02;
TCNT2=0xFE;
OCR2=0x00;
```

```

// External Interrupt(s) initialization
// INT0: Off
// INT1: Off
// INT2: Off
MCUCR=0x00;
MCUCSR=0x00;

// Timer(s)/Counter(s) Interrupt(s) initialization
TIMSK=0x41;

// Analog Comparator initialization
// Analog Comparator: Off
// Analog Comparator Input Capture by Timer/Counter 1: Off
ACSR=0x80;
SFIOR=0x00;

// LCD module initialization
lcd_init(16);

// Global enable interrupts
#asm("sei")

// while (1) {
//   R1_value = 1;
//   R2_value = 0;
//   L1_value = 1;
//   L2_value = 0;
//
//   rpwm_value = lpwm_value = 75;
//   delay_ms(5000);
//   rpwm_value = lpwm_value = 50;
//   delay_ms(5000);
//   rpwm_value = lpwm_value = 25;
//   delay_ms(5000);
// }

while(1) {
  lcd_gotoxy(0,0);
  lcd_putsf("Ready to use");

  if (tombol_sim == 0) {
    simulation = 1;
  }

  if (tombol_naik == 0) {
    mtr_naik1 = 1;
    mtr_naik2 = 0;
    delay_ms(100);
  }
  else if (tombol_turun == 0) {

```

```

    mtr_naik1 = 0;
    mtr_naik2 = 1;
    delay_ms(100);
}
else {
    mtr_naik1 = 0;
    mtr_naik2 = 0;
}

if (tombol_buka == 0) {
    mtr_buka1 = 0;
    mtr_buka2 = 1;
    delay_ms(500);
}
else if (tombol_tutup == 0) {
    mtr_buka1 = 1;
    mtr_buka2 = 0;
    delay_ms(500);
}
else {
    mtr_buka1 = 0;
    mtr_buka2 = 0;
}

if (tombol_real == 0) {
    lcd_clear();
    lcd_gotoxy(0,0);
    lcd_putsf("Run in 1 second");
    delay_ms(1000);
    break;
}
if (simulation == 1) {
    lcd_clear();
    lcd_gotoxy(0,0);
    lcd_putsf("Bisma Pranadipta");
    lcd_gotoxy(0,1);
    lcd_putsf("06.12.213");

    delay_ms(2000);
    lcd_clear();

    for (;;) {
        lcd_gotoxy(0,0);
        lcd_putsf("Test Sensor");
        lcd_gotoxy(0,1);
//        if (sens_1 == 0) {
//            lcd_putchar('*');
//        } else {
//            lcd_putchar('-');
//        }
    }
}

```

```

//     if (sens_2 == 0) {
//         lcd_putchar('*');
//     } else {
//         lcd_putchar('-');
//     }
    if (sens_3 == 0) {
        lcd_putchar('*');
    } else {
        lcd_putchar('-');
    }
    if (sens_4 == 0) {
        lcd_putchar('*');
    } else {
        lcd_putchar('-');
    }
//     if (sens_5 == 0) {
//         lcd_putchar('*');
//     } else {
//         lcd_putchar('-');
//     }
//     if (sens_6 == 0) {
//         lcd_putchar('*');
//     } else {
//         lcd_putchar('-');
//     }

    if (tombol_sim == 0) {
        lcd_clear();
        lcd_gotoxy(0,0);
        lcd_putsf("Counter Rotary");
        delay_ms(300);
//         delay_ms(1000);
//         lcd_clear();
        do_count = 1;
        while (tombol_sim == 1) {
            sprintf(s_count, "ki: %3i ka: %3i", count_rotki, count_rotka);
            lcd_gotoxy(0,1);
            lcd_puts(s_count);
        }
        lcd_clear();
        lcd_gotoxy(0,0);
        lcd_putsf("Simulasi Selesai");
        delay_ms(1000);
        lcd_clear();

        while (tombol_real != 0) {
            //tunggu tombol diteken
            lcd_gotoxy(0,1);
            lcd_putsf("Tekan Start");
        }
    }

```

```

        break;
    }
}
lcd_clear();
lcd_gotoxy(0,0);
lcd_putsf("Run in 2 seconds");
delay_ms(2000);
lcd_clear();
lcd_putsf("Running");
break;
}
}

```

```

reset_counter();
//i = tenth_of_sec;
while (line_count < 1) {
    R1_value = 1;
    R2_value = 0;
    R_pwm = 1;
    rpwm_value = 72;

    L1_value = 1;
    L2_value = 0;
    L_pwm = 1;
    lpwm_value = 50;
}

```

```

reset_counter();
do_count = 1;
// enable_pwm = 1;
while (count_rotki <= 200) {
    sprintf(s_count, "ki: %3i ka: %3i", count_rotki, count_rotka);
    lcd_gotoxy(0,1);
    lcd_puts(s_count);

```

```

    R1_value = 1;
    R2_value = 0;
    R_pwm = 1;
    rpwm_value = 72;

```

```

    L1_value = 1;
    L2_value = 0;
    L_pwm = 1;
    lpwm_value = 50;

```

```

}

```

```

//BERHENTI
i = tenth_of_sec;

```

```

while (tenth_of_sec - i <= 10) {
  R1_value = 0;
  R2_value = 0;
  R_pwm = 0;
  rpwm_value = 0;

  L1_value = 0;
  L2_value = 0;
  L_pwm = 0;
  lpwm_value = 0;
}

//AMBIL BARANG
//SERVO KIRI
i = tenth_of_sec;
while (tenth_of_sec - i <= 12) {
  servo_value = 1;
}

servo_value = 0;
delay_ms(150);
//TURUN
i = tenth_of_sec;
while (tenth_of_sec - i <= 5) {
  mtr_naik1 = 0;
  mtr_naik2 = 1;
}
  mtr_naik1 = 0;
  mtr_naik2 = 0;
//TUTUP
i = tenth_of_sec;
while (tenth_of_sec - i <= 50) {
  mtr_buka1 = 1;
  mtr_buka2 = 0;
}
  mtr_buka1 = 0;
  mtr_buka2 = 0;
//NAIK
i = tenth_of_sec;
while (tenth_of_sec - i <= 10) {
  mtr_naik1 = 1;
  mtr_naik2 = 0;
}
  mtr_naik1 = 0;
  mtr_naik2 = 0;
//SERVO KANAN
i = tenth_of_sec;
while (tenth_of_sec - i <= 12) {
  servo_value = 1;
}

```

```

servo_value = 0;
delay_ms(150);

//MAJU 2
reset_counter();
do_count = 0;
//i = tenth_of_sec;
while (line_count < 1) {
    R1_value = 1;
    R2_value = 0;
    R_pwm = 1;
    rpwm_value = 72;

    L1_value = 1;
    L2_value = 0;
    L_pwm = 1;
    lpwm_value = 50;
}

reset_counter();
do_count = 1;
// enable_pwm = 1;
while (count_rotki <= 310) {
    sprintf(s_count, "ki: %3i ka: %3i", count_rotki, count_rotka);
    lcd_gotoxy(0,1);
    lcd_puts(s_count);

    R1_value = 1;
    R2_value = 0;
    R_pwm = 1;
    rpwm_value = 72;

    L1_value = 1;
    L2_value = 0;
    L_pwm = 1;
    lpwm_value = 50;
}

//BERHENTI 2
reset_counter();
i = tenth_of_sec;
while (tenth_of_sec - i <= 10) {
    R1_value = 0;
    R2_value = 0;
    R_pwm = 0;
    rpwm_value = 0;

    L1_value = 0;
    L2_value = 0;
}

```

```

    L_pwm = 0;
    lpwm_value = 0;
}

//TARUH BARANG
//SERVO KANAN
i = tenth_of_sec;
while (tenth_of_sec - i <= 12) {
    servo_value = 1;
}

servo_value = 0;
delay_ms(250);
//TURUN
i = tenth_of_sec;
while (tenth_of_sec - i <= 1) {
    mtr_naik1 = 0;
    mtr_naik2 = 1;
}
mtr_naik1 = 0;
mtr_naik2 = 0;
//BUKA
i = tenth_of_sec;
while (tenth_of_sec - i <= 50) {
    mtr_buka1 = 0;
    mtr_buka2 = 1;
}
mtr_buka1 = 0;
mtr_buka2 = 0;
//NAIK
i = tenth_of_sec;
while (tenth_of_sec - i <= 20) {
    mtr_naik1 = 1;
    mtr_naik2 = 0;
}
mtr_naik1 = 0;
mtr_naik2 = 0;
//SERVO KIRI
i = tenth_of_sec;
while (tenth_of_sec - i <= 12) {
    servo_value = 2;
}

servo_value = 0;
delay_ms(250);

//MUNDUR
reset_counter();
// while (line_count < 2) {
while (count_rotki <= 432) {

```

```

sprintf(s_count, "ki: %3i ka: %3i", count_rotki, count_rotka);
lcd_gotoxy(0,1);
lcd_puts(s_count);

R1_value = 0;
R2_value = 1;
R_pwm = 1;
rpwm_value = 60;

L1_value = 0;
L2_value = 1;
L_pwm = 1;
lpwm_value = 50;
}

//BERHENTI
i = tenth_of_sec;
while (tenth_of_sec - i <= 10) {
    R1_value = 0;
    R2_value = 0;
    R_pwm = 0;
    rpwm_value = 0;

    L1_value = 0;
    L2_value = 0;
    L_pwm = 0;
    lpwm_value = 0;
}

lcd_clear();
lcd_gotoxy(0,1);
lcd_putsf("Selesai..");
while (1)
{
    R1_value = 0;
    R2_value = 0;
    R_pwm = 0;
    rpwm_value = 0;

    L1_value = 0;
    L2_value = 0;
    L_pwm = 0;
    lpwm_value = 0;

    servo_value = 0;
};
}

```



