TA / SEKJUR TE/ 2007/ 055

KENDALI ARAH SUDUT ROBOT TANK DENGAN KOMPAS DIGITAL

TUGAS AKHIR

Diajukan untuk melengkapi salah satu syarat untuk memperoleh gelar sarjana Teknik pada Jurusan Teknik Elektro Universitas Islam Indonesia Yogyakarta



oleh :

Nama : Yudi Heryanto

No. Mahasiswa : 02 524 077

JURUSAN TEKNIK ELEKTRO

FAKULTAS TEKNOLOGI INDUSTRI

UNIVERSITAS ISLAM INDONESIA

YOGYAKARTA

2007

LEMBAR PENGESAHAN PEMBIMBING

KENDALI ARAH SUDUT ROBOT TANK DENGAN KOMPAS DIGITAL

TUGAS AKHIR

Disusun oleh :

Nama : Yudi Heryanto

No. Mahasiswa : 02 524 077

Yogyakarta, 10 Mei 2007

Pembimbing I

(Tito Yuwono,ST,M.Sc.)

Pembimbing II

(Yusuf Aziz Amrullah,ST.)

LEMBAR PENGESAHAN DOSEN PENGUJI

KENDALI ARAH SUDUT ROBOT TANK DENGAN KOMPAS DIGITAL

TUGAS AKHIR

Oleh :

Nama : Yudi Heryanto

No.Mahasiswa : 02 524 077

Telah Dipertahankan Di Depan Sidang Penguji Sebagai Salah Satu Syarat Untuk Memperoleh Gelar Sarjana Teknik Elektro, Fakultas Teknologi Industri Universitas Islam Indonesia

Tim Penguji,

Tito Yuwono, ST, M.Sc.
Ketua

Hendra Setiawan, ST, MT.
Anggota I

Yusuf Aziz Amrullah, ST.
Anggota II

Mengetahui,
Ketua Jurusan Teknik Elektro

UNIVERSITAS ISLAM INDONESIA ★ YOGYAKARTA ★ FAKULTAS TEKNOLOGI INDUSTRI

(Tito Yuwono, ST, M.Sc)

HALAMAN PERSEMBAHAN

Dengan penuh cinta

Karya ini kupersembahkan kepada :

H.wartomo dan Acih selaku Ayah dan Ibuku tercinta, Atas

Curahan Kasih Sayangnya.

Syair Hidayat dan Juju Juariah selaku orang tua angkatku

yang tanpa pamrih memberikan kasih sayangnya.

Serta

Kakak- kakak serta adik- adikku yang selalu kusayangi.

Juga

Gerhani Yulianti Sahabat setia hidupku

dan

Seluruh keluargaku

MOTTO

Sahabat sejatimu adalah orang yang selalu berterus terang kepadamu jika engkau berada dalam kebingungan, Dia pun rela berkorban dan mencurahkan tenaga demi dapat berkumpul denganmu.

(Ali Bin Abi Thalib. RA)

Ngelmu iku kalakonne kanthi laku, lekase lawan khas, tegese khas nyantosani, setya budya pangekese dur angkara.

(Soemarsaid Martono, 984:163)

Aja dumeh kuwasa, tumindake daksura lan daksia-marang sapada-pada.
Aja dumeh pinter,tumindak keblinger.
Aja dumeh kuat lan gagah, tumindak sarwa gegabah.
Aja dumeh sugih, tumindak lali karo sing ringkih.
Aja dumeh menang tumindake sewenang-wenang.

(Anonim)

Aja nggugu benere dhewe

(Aryanto, 1992:158)

KATA PENGANTAR

بسم الله الرحمن الرحيم

Assalamu'alaikum Wr. Wb.

Alhamdulillah, segala puji dan syukur ditunjukan hanya bagi Allah SWT. Atas rahmat dan taufik–Nya penulis akhirnya dapat menyelesaikan tugas akhir ini yang berjudul **"Kendali Arah Sudut Robot Tank Dengan Kompas Digital "** dengan lancar.

Adapun maksud dan tujuan penyusunan tugas akhir ini adalah sebagai salah satu syarat dalam menempuh gelar sarjana pada Jurusan Teknik Elektro Universitas Islam Indonesia, Yogyakarta.

Pada kesempatan ini dengan segala rasa syukur dan kerendahan hati penulis sampaikan ucapan terima kasih yang sebesar-besarnya kepada :

1. Bpk Fathul Wahid, ST. M.Sc. Selaku dekan Fakultas Teknologi Industri (FTI) Universitas Islam Indonesia (UII).
2. Bpk Tito Yuwono, ST. selaku Kajur Teknik Elektro dan dosen pembimbing 1.
3. Bpk Yusuf Aziz Amrullah, ST. Selaku dosen pembimbing II atas masukannya juga dengan sabar telah memberi bimbingan.
4. Ibu dan Bapakku tercinta yang tidak henti–hentinya memberikan dorongan serta doa restu yang tulus sehingga memberiku kekuatan untuk menyelesaikan tugas akhir ini.

5. Kedua kakak dan kedua adiku yang tercinta serta seluruh keluargaku yang senantiasa tidak henti – hentinya memberiku semangat dan doanya.
6. Ibu Juju dan Bapak Syair Hidayat sebagi Orang tua angkatku senantiasa memberikan ketulusan cinta dan kasih sayangnya dengan ikhlas sehingga aku mengerti akan arti kasih sayang dan cara menghargai dalam hidup ini.
7. Untuk tunanganku sekaligus sahabat hidupku "Gerhani Yulianti", terima kasih atas kesetiaannya dan tanpa henti – hentinya memberikan dorongan serta motivasinya.
8. Adik adiku tercinta (Lani , Deni, Rani, Desi) terima kasih.
9. Mr.Arif, Mr.Peyet dan seluruh kru MP terima kasih atas bantuannya serta doanya.
10. Sahabat–sahabatku "GEMPAR" yang telah memberikan arti persahabatan selama ini, terima kasih atas dukunganya.
11. Dosen dan karyawan Fakultas Teknologi Industri UII, yang bersedia dengan sabar membantu dan berbagi ilmunya.
12. Seluruh Staf TU dan Perpustakaan Jurusan yang telah banyak membantu segala kelancaran penulis.
13. Seluruh Staf Laboran Mas Heri, Mas Agung, yang telah amat sangat banyak membantu penulis.
14. Doel, Dd, Eko, N, Marcel,Soni,Dadang, Enggar dan M. situr serta seluruh teman–teman mahasiswa jurusan Teknik Elektro UII yang tidak bisa saya sebutkan satu persatu.

15. Teman–teman kontrakan (Agoeng, Epry) yang senantiasa saling mengingatkan dan memberikan semangat.
16. Bpk. Kost (Mas Han, Mas Tono) sekeluarga, terima kasih atas semua kebaikannya. Dan juga terima kasih atas informasi serta pemberian material yang dibutuhkan dalam pembuatan robot ini.
17. Seluruh pihak yang tidak dapat disebutkan satu-persatu, yang telah memberikan *support* dan doa.

Akhir kata penulis hanya dapat memanjatkan doa kehadirat Allah SWT, kiranya dapat melimpahkan rahmat dan kesejahteraan kepada semua pihak atas segala budi baik yang telah diberikan kepada penulis. Amien...

Wassalamu'alaikum Wr. Wb

Yogyakarta, mei 2007

(Yudi Heryanto)

DAFTAR ISI

HALAMAN JUDUL i

LEMBAR PENGESAHAN PEMBIMBING ii

LEMBAR PENGESAHAN DOSEN PENGUJI iii

HALAMAN PERSEMBAHAN iv

MOTTO v

KATA PENGANTAR vi

ABSTRAK ix

DAFTAR ISI x

DAFTAR TABEL xii

DAFTAR GAMBAR xiii

BAB I. PENDAHULUAN

1.1. Latar Belakang Masalah 1

1.2. Identifikasi Masalah 2

1.3. Batasan Masalah 2

1.4. Tujuan dan Manfaat 3

1.5. Sistematika Penulisan 3

BAB II. DASAR TEORI

2.1. CMPS03-*Robot Compass Module* 5

2.1.1. Metode I^2C 6

2.1.2. Metode PWM 7

2.2. Mikrokontroler 8

2.2.1. Arsitektur Mikrokontroler AT89S52 8

2.2.2. Organisasi Memori 9

2.2.3. Memori Program 11

2.2.4. Memori Data 11

2.2.5. Fungsi – fungsi pin Mikrokontroler AT89S52 12

2.3. LCD (*Liquid Crystal Display*) 14

2.3.1. DDRAM (*Display Data Random Access Memory*) 16

2.3.2. CGRAM 16

2.3.3. CGROM 17

2.3.4. Konfigurasi PIN 17

2.3.5. Register 18

2.3.5.1. register Perintah 18

2.3.5.2. Register Data 18

2.3.5.3. Penulisan Data ke register Perintah Dan Register Data 18

2.4. DC Motor 19

2.5. Keypad 20

2.6. PWM 22

BAB III. PERANCANGAN

3.1 Gambaran umum sistem 24

3.2 Perancangan Perangkat Keras 25

3.2.1.Rangkaian *Keypad* 25

3.2.2.Rangkaian sensor 27

3.2.3. Perancangan Rangkaian LCD 28

3.2.4. Perancangan catu daya 29

3.2.5 Mikrokontroler AT89S52 30

3.2.5.1. Osilator 30

3.2.5.2. Reset 30

3.2.6. Perancangan Rangkaian PWM 31

3.3 Perancangan Perangkat lunak 33

3.3.1. Program Inisialisasi Mikrokontroler AT89S52 34

3.3.2. Program utama 35

3.3.3. Program LCD 36

3.3.4. Sub rutin Ukur Sudut 37

3.3.5. Sub rutin *Keypad* 39

3.3.6 Sub rutin *Entry data* 41

3.3.7. Pengendalian Motor 42

BAB IV. HASIL PENGAMATAN DAN PEMBAHASAN

4.1.Metode Pengujian 44

4.2. Pengujian Fungsional 45

4.2.1. Pengujian rangkaian sensor 45

4.2.2. Pengujian Rangkaian Keypad 51

4.2.3. Pengujian Rangkaian mikrokontroler.....52

4.2.3.1. Rangkaian Clock.....53

4.2.3.2. Rangkaian Reset.....53

4.2.4. Pengujian Rangkaian LCD.....53

4.2.5. Pengujian Rangkaian Driver PWM dan Pengendali Motor...54

4.3 Pengujian Kinerja Sistem dan Pembahasan Sistem keseluruhan.....55

4.3.1. Pengkalibrasian dan pengujian.....59

4.3.2. Pengujian set Point Dengan Aktuasi Motor.....63

4.3.3. Pengujian kompas digital,PWM dengan perhitungan rumus.67

BAB V. PENUTUP

5.1. Kesimpulan 68

5.2. Saran-saran.....69

DAFTAR PUSTAKA

LAMPIRAN

DAFTAR TABEL

Tabel 3.1.Pengkodean *keypad*.....27

Tabel 3.2.Kebenaran pada rangkaian driver motor.....33

Tabel 4.1.Output PWM dan frekuensi dari sensor CMPS03 per sudut 46

Tabel 4.2 Hasil pengamatan *keypad* 52

Tabel 4.3. Ketentuan nilai sudut dengan arah.....56

Tabel 4.4. Perbandingan nilai sudut kompas dengan analog 59

Tabel 4.5. Pengaruh medan magnet terhadap sensor.....63

Tabel 4.6 Set point VS aktuasi motor 64

DAFTAR GAMBAR

Gambar 2.1. Modul kompas digital /sensor kompas 5

Gambar 2.2. Arsitektur mokrokontroler AT89S52 9

Gambar 2.3. Memori data internal 11

Gambar 2.4. konfigurasi pin Mikrokontroler AT89S52 13

Gambar 2.5. Bentuk fisik LCD M1632 15

Gambar 2.6. Dimensi LCD M1632 ..16

Gambar 2.7. Alamat DDRAM M1632 ..16

Gambar 2.8. Blok diagram pengendalian motor DC 20

Gambar 2.9. Keypad matriks 3 x 4 21

Gambar 2.10. Contoh penekanan tombol 21

Gambar 2.11. Efek *bouncing* pada penekanan saklar 22

Gambar 2.12. Pulsa PWM 23

Gambar 3.1. Diagram blok 24

Gambar 3.2. Rangkaian *keypad* 26

Gambar 3.3. Bentuk fisik *keypad* ..26

Gambar 3.4. Rangkaian sensor 28

Gambar 3.5. Rangkaian LCD 29

Gambar 3.6. Rangkaian catu daya 5V DC 30

Gambar 3.7. Rangkaian catu daya 12V DC 30

Gambar 3.8. Rangkaian minimum AT89S52 31

Gambar 3.9 Rangkaian mikro ke driver motor.....32

Gambar 3.10. Rangkaian H-Bridge.....33

Gambar 3.11. *Flowchart* Program utama.....36

Gambar 3.12 *Flowchart* program LCD.....37

Gambar 3.13. Ukur sudut.....38

Gambar 3.14. *Flowchart* scan keypad.....40

Gambar 3.15. Sub rutin entry data 41

Gambar 3.16. Pengendalian motor.....42

Gambar 4.1. Grafik pengaruh perubahan sudut terhadap PWM 50

Gambar 4.2. Output PWM dari mikro 54

Gambar 4.3. Output PWM dari driver 55

Gambar 4.4. Tampilan LCD sebelum memakai sensor 55

Gambar 4.5. Tampilan LCD kompas digital 56

Gambar 4.6. Tampilan setelah menekan tanda bintang 57

Gambar 4.7. Tampilan setelah memasuki nilai 57

Gambar 4.8. Tampilan ketika setelah menekan enter 57

Gambar 4.9. Tampilan ketika sesaat motor berhenti 58

Gambar 4.10.Tampilan akhir 58

Gambar 4.11. Grafik kompas digital vs kompas analog 62

Gambar 4.12. Grafik aktuasi motor VS set point 1.....65

Gambar 4.13. Grafik aktuasi motor VS set point II.....66

Gambar 4.14. Grafik aktuasi motor VS set point III.....66

ABSTRAK

Penelitian ini bertujuan untuk memperoleh suatu Aplikasi Mikrokontroler AT89S52 untuk kendali arah sudut robot tank dengan kompas digital. serta untuk mengetahui cara kerjanya. Rangkaian ini secara keseluruhan terdiri dari tiga bagian, yaitu Masukan berupa sensor compas CMPSO3, keypad, pengolah berupa AT89S52 dan keluaran berupa LCD dan sebuah motor DC sebagai peralatan yang dikendalikan. Hasil penelitian ini dapat direalisasikan dengan menggunakan metode PWM yang merupakan output dari sensor kompas sebagai kompas digital. Untuk menggerakan robot agar dapat diatur kecepatan serta ketepatan nilai sudut yang diinginkan bisa dilakukan dengan cara mengatur lebar pulsanya (PWM) dengan menggunakan mikrokontroler dan juga rangkaian driver H-Bridge sebagai penguatan untuk menjalankan motor. Hasil pembacaan yang diperoleh dari sistem kompas digital memiliki tingkat error sekitar 0–8%. Untuk pembacaan sudut yang didapat pada metode ini hanya dari 0^0 sampai 355^0, sehingga untuk perintah sudut 356^0-360^0 terhadap robot ini secara otomatis akan menghasilkan pembacaan sudut 0^0 pada kompas dan ini berarti robot akan berhenti ketika diarah sudut 0^0 sehingga sudut-sudut ini bisa disebut sudut mati. Kadang terdapat perbedaan atau selisih pembacaan antara kompas digital dengan sudut robot yang diinginkan, namun tingkat error tersebut hanyalah 1-2% saja. Diharapkan penelitian ini bisa dikembangkan lagi sehingga bisa melengkapi kekurangan dari sistem ini dan juga diharapkan penelitian ini berguna bagi mereka yang menggeluti dunia robotika.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Navigasi sangatlah penting bagi kebutuhan manusia maupun robot, agar pada saat bergerak,dapat menuju ke arah yang tepat sesuai yang dituju. Dari dulu sampai sekarang, manusia suka membaca arah jika akan bepergian jauh, seperti ketika di tengah lautan, di gunung bahkan di dalam suatu kota yang baru dan asing bagi mereka. Namun banyak cara yang dilakukan manusia untuk mengetahui arah yang mereka cari, salah satu cara untuk membaca arah yaitu dengan menggunakan kompas. Kompas adalah alat navigasi yang menunjukan arah utara, selatan, barat dan timur dimana alat ini memiliki berbagai bentuk dan versi. Yang sering beredar atau sering di lihat mungkin hanya kompas manual yang umumnya berbentuk bulat atau sebagainya yang didalamnya ada suatu cairan dengan sebatang magnet berbentuk jarum, dan jarang sekali melihat kompas digital dengan tampilan layar.

Pada penelitian ini akan dibuat suatu alat yang berupa kompas digital *portable* untuk keperluan navigasi, serta aplikasi pengembangannya dalam dunia robotika. Untuk aplikasinya dalam dunia robotika, peneliti akan mencoba menggunakan kompas digital ini sebagai kendali arah atau navigasi sudut sebuah robot tank, sehingga diharapkan alat ini bisa mengendalikan putaran robot tank tersebut dalam bentuk sudut (derajat) sesuai dengan yang diinginkan. Sistem ini akan dilengkapi dengan tampilan LCD yang akan menampilkan jumlah sudut

dalam bentuk derajat dan juga dilengkapi dengan nama arah–arahnya seperti utara, selatan, barat, timur sehingga pengguna akan semakin mudah mengetahui keakuratan arah yang dihasilkan dari putaran robot tersebut.

Dengan output digital, diharapkan penelitian ini akan sangat membantu bagi mereka yang menggeluti robotika, sehingga ini dapat digunakan sebagai acuan aplikasi robot lainnya.

1.2 Rumusan Masalah

Untuk mewujudkan tercapainya pembuatan alat yang dapat memecahkan masalah pembuatan kompas digital, dapat diidentifikasikan sebagai berikut:

1. Bagaimana membangun dan merealisasikan suatu instrumen yang dapat mengetahui arah sebagai kompas digital
2. Bagaimana unjuk kerja aplikasi mikrokontroler AT89S52 untuk penggunaan pembuatan kompas digital
3. Bagaimana efektifitas penggunaan sensor yang digunakan
4. Bagaimana pengaplikasiannya terhadap robot tank
5. Bagaimana unjuk kerja aplikasi mikrokontroler AT89S52 untuk kompas digital sebagai kendali arah sudut robot tank.

1.3 Batasan masalah

Agar dalam penulisan penelitian ini lebih terarah, maka pembahasan penulisan ini dibatasi pada ruang lingkup pembahasan sebagai berikut :

1. Sistem menggunakan mikrokontroler AT89S52 sebagai pemroses utama.

2. Sistem berfungsi mengendalikan sebuah sensor, keypad dan penampil dan juga sebuah motor DC.
3. Sistem menggunakan sensor CMPSO3 yang digunakan untuk mendeteksi posisi arah medan magnet bumi.
4. Sistem menggunakan sebuah keypad 3x4 sebagai salah satu input dan sebuah LCD 16 x2 sebagai penampil dan juga sebuah motor DC.

1.4 Tujuan dan manfaat

Tujuan dan manfaat yang akan dicapai dalam penulisan penelitian ini adalah sebagai berikut :

1. Membangun dan merealisasikan suatu peralatan yang dapat berfungsi sebagai navigasi dengan berbasis mikrokontroler AT89S52 dan perangkat-perangkat pendukung lainnya.
2. Mengetahui unjuk kerja aplikasi mokrokontroler AT89S52 untuk kompas digital dan juga aplikasinya sebagai kendali arah sudut robot tank
3. Mengetahui efektifitas sensor yang digunakan.
4. Mengetahui manfaat dari kompas digital dan dalam hal ini aplikasinya sebagai navigasi robot tank

1.5 Sistematika Penulisan

Sistematika penulisan dari laporan akhir penelitian ini yang berupa karya tulis (skripsi) dibagi dalam lima bab, dengan isi masing-masing bab diuraikan sebagai berikut :

BAB I Pendahuluan

Berisi latar belakang masalah, perumusan masalah, batasan masalah, dan sistematika penulisan.

BAB II Landasan Teori

Menguraikan tentang teori-teori yang menjadi acuan dalam pembuatan tugas akhir, diantaranya teori tentang mikrokontroler keluarga MCS-51 dalam hal ini mikrokontroler AT89S52, teori tentang LCD, keypad, teori tentang sensor arah, teori tentang motor DC.

BAB III Perancangan Sistem

Berisi tentang proses perancangan dan pembuatan sistem aplikasi mikrokontroller AT89S52 untuk perancangan kompas digital dan aplikasinya sebgagai kendali navigasi sudut robot tank dari mulai diagram blok, prinsip kerja, serta pembahasan mengenai *software* sebagai otak sistem pengendalinya.

BAB IV Hasil Pengamatan dan Pembahasan

Berisi pembahasan dari hasil pengamatan.

BAB V Kesimpulan dan Saran

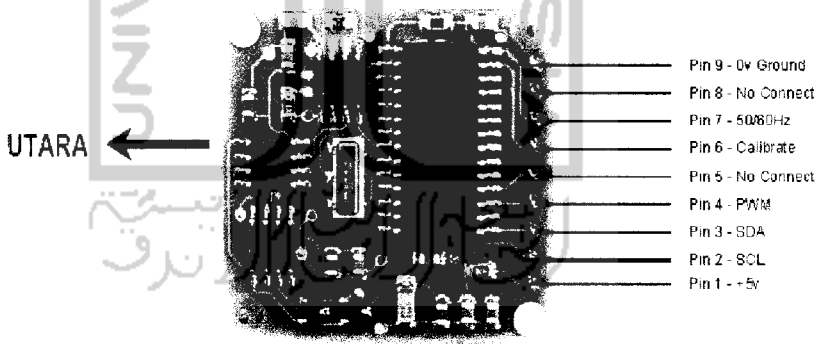
Berisi kesimpulan dari pembahasan dan saran berdasarkan hasil yang diperoleh.

BAB II

DASAR TEORI

2.1 CMPS03 – Robot Compass Module

Komponen utama dari kompas digital ini adalah CMPS03 – *robot compass module* yang berfungsi sebagai sensor. Alat ini telah terprogram untuk dapat mengetahui arah dan memiliki output sudah dalam bentuk digital. CMPS03 ini memiliki resolusi hingga 0,1 derajat dan juga memiliki 2 pilihan antarmuka/*interface* yaitu dalam bentuk I^2C (*inter integrated circuit*) dan PWM(*pulse width modulation*). Modul CMPS03 telah terkalibrasi di pabriknya namun karena lokasi pabriknya berbeda dengan Indonesia dalam hal sudut inklinasinya, maka modul ini perlu dikalibrasi ulang.



Gambar 2.1. Modul kompas digital/sensor kompas

Dibutuhkan tegangan +5VDC untuk menjalankan modul kompas ini dan dengan arus kira – kira 15mA. Ada dua cara untuk mengolah data dari modul kompas ini, yang pertama dengan cara PWM yaitu dengan membaca lebar pulsa

yang merupakan output dari sensor kompas digital ini dan cara yang kedua adalah dengan metode I^2C yaitu dengan cara membaca dua output yang terdiri dari SDA dan SCL.

2.1.1 Metode I^2C

Pada saat ini desain elektronik dituntut untuk semakin ringkas dan fleksibel, dimana ukuran fisik IC semakin diperkecil dan jumlah pin diminimalkan dengan tetap menjaga *fleksibilitas* dan kompabilitas IC sehingga mudah untuk digunakan dalam berbagai keperluan desain yang berbeda, oleh karenanya banyak perusahaan semikonduktor yang berusaha mengembangkan cara baru komunikasi antar IC yang lebih akomodatif terhadap tuntutan diatas sebagai alternatif dari hubungan antar IC secara paralel (parallel bus) yang sudah kita kenal luas. Salah satu metode yang telah matang dan dipakai secara luas adalah IIC (sering ditulis juga I^2C) singkatan dari *inter integrated circuit* bus yang dikembangkan oleh Philips Semiconductor sejak tahun 1992, dengan konsep dasar komunikasi 2 arah antar IC dan/atau antar sistem secara serial menggunakan 2 kabel. Adapun keuntungan yang didapat dengan menggunakan I^2C antara lain :

1. Meminimalkan jalur hubungan antara IC (bandingkan dengan paralel bus)
2. Menghemat luasan PCB yang dibutuhkan .
3. Membuat sistem yang didesain berorientasi *software* (mudah diekspan dan diupgrade).
4. Membuat sistem yang didesain menjadi standar, sehingga dapat dihubungkan dengan sistem lain yang juga menggunakan I^2C bus.

Jika digunakan *device* yang telah dilengkapi dengan kemampuan I^2C secara *built in*, akan sangat mudah untuk melaksanakan komunikasi baik single master maupun multi master.

Adapun I^2C memiliki fitur-fitur utama sebagai berikut :

1. Hanya melibatkan dua kabel yaitu serial data line (selanjutnya disebut SDA) dan serial clock line (selanjutnya disebut SCL).
2. Setiap IC terhubung dalam I^2C memiliki alamat yang unik yang dapat diakses secara *software* dengan master/slave protocol yang sederhana, dan mampu mengakomodasikan multimaster.
3. I^2C merupakan serial bus dengan orientasi data 8 bit (byte),komunikasi 2 arah, dengan kecepatan transfer data sampai 100Kbit/s pada mode standar dan 3,4 Mbit/s pada mode kecepatan tinggi.
4. Jumlah slave terbatas dan jumlah IC yang dapat dihubungkan pada I^2C bus hanya dibatasi oleh beban kapasitansi pada bus yaitu maksimum 400pF.

2.1.2 Metode PWM

PWM (*pulse width modulation*) adalah metode yang akan dipakai pada perancangan ini caranya yaitu dengan memakai lebar pulsa yang merupakan output dari sensor CMPS03. Setiap kali sensor bergerak atau berubah posisi maka lebar pulsa yang dikeluarkannya akan berubah sehingga perubahan inilah yang akan dibaca oleh sistem (mikrokontroler) yang kemudian ditampilkan kedalam bentuk sudut.

2.2 Mikrokontroler

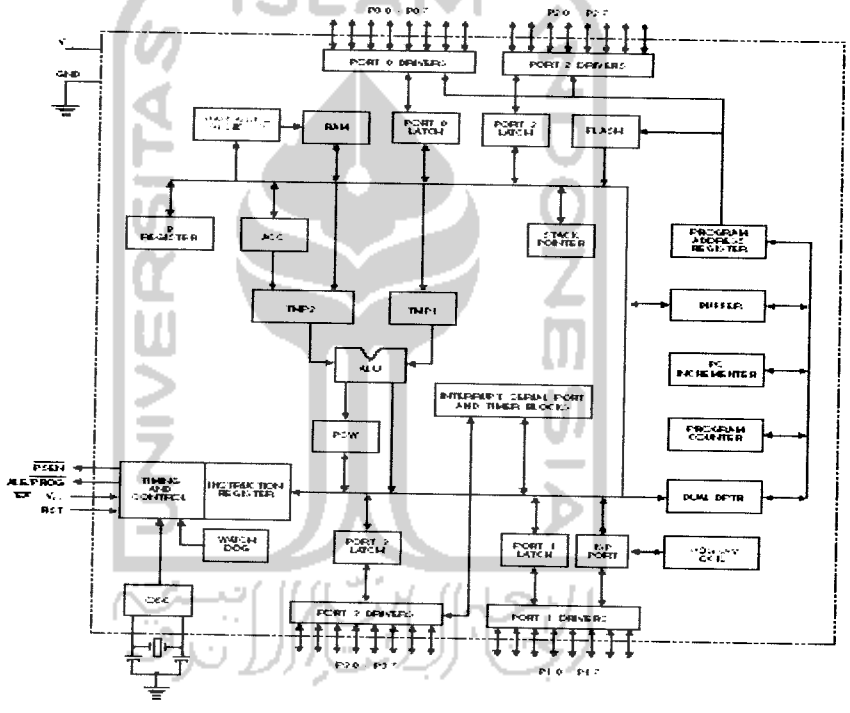
Mikrokontroler adalah suatu terobosan teknologi mikroprosesor dan mikrokomputer, yang mana teknologi ini adalah teknologi semikonduktor dengan kandungan transistor yang lebih banyak, namun hanya membutuhkan ruang yang kecil serta dapat diproduksi secara masal (dalam jumlah banyak) sehingga harganya menjadi lebih murah.

Mikrokontroler ini kemampuan digitalnya menirukan fungsi otak manusia, sehingga meliputi fungsi atau instruksi aritmatika (berhitung), logika (mempertimbangkan suatu kondisi), dan memori. Mikrokontroler ini berbeda halnya dengan mikroprosesor yang hanya pemrosesannya terdiri dari *central processing unit* (CPU) dan register-register, tanpa memori, tanpa I/O, dan *periferal* yang dibutuhkan oleh suatu sistem supaya dapat bekerja. Namun apabila mikroprosesor ini dikombinasikan dengan I/O dan memori (RAM dan ROM) akan menghasilkan sebuah mikrokontroler, yang mana kombinasi dari komponen-komponen ini sudah terdapat dalam satu chip *integrated circuit* (IC).

2.2.1 Arsitektur Mikrokontroler AT89S52

Mikrokontroler AT89S52 adalah sebuah mikrokomputer yang terdiri dari 8 bit dengan *low power suply* dan kinerja tinggi yang terdiri dari CMOS dengan *flash programmable* sebesar 8 Kbyte, yang mana alat ini dibuat dengan menggunakan teknologi *nonvolatile* berdensitas tinggi dari ATMEL yang kompatibel dengan keluarga 80C51, dengan demikian perintah-perintah dan fungsi pin-pinnya sama dengan mikrokontroler keluarga 8051 yang merupakan suatu standar industri.

Dengan mengkombinasikan sebuah *versatile* CPU 8 bit dengan sebuah *chip monolithic* pada sistem *flash programmable*, AT89S52 merupakan mikrokontroler yang *powerful* yang menyediakan fleksibilitas yang tinggi dan solusi biaya yang efektif untuk beberapa aplikasi kendali. AT89S52 menyediakan fitur-fitur yang standar, yaitu mempunyai kecepatan 8 Kbyte, RAM 256 byte, 32 jalur I/O, 2 data pointer, tiga 16 bit timer/counter, 7-vektor 2 level arsitektur interup, sebuah port serial *full duplex*, *oscillator* on-chip, dan sirkuit *clock*.



Gambar 2.2. Arsitektur mikrokontroler AT89S52

[Sumber : *Atmel Literature Requests*, www.atmel.com/literature]

2.2.2 Organisasi Memori

Organisasi memori pada mikrokontroler *flash* AT89S52 sama dengan AT89C51/52 dari ATMEL yang memiliki ruang alamat memori data dan memori

program yang terpisah, pemisahan memori data dan program tersebut membolehkan memori data diakses dengan alamat 8-bit, sehingga dapat dengan cepat dan mudah disimpan dan dimanipulasi oleh CPU 8 bit, namun demikian alamat memori data 16-bit bisa dihasilkan dengan melalui Register DPTR. Memori data eksternal dan memori program eksternal dapat dikombinasikan dengan cara menggabungkan sinyal RD dan PSEN melalui gerbang AND dan keluarannya sebagai tanda baca ke memori data/program eksternal.

Mikrokontroler AT 89S52 mempunyai alamat yang terpisah untuk memori program dan memori data. Memori untuk progran dapat mencapai 64 Kbyte. Ada 4 jenis memori pada mikroprosesor ini, yaitu :

1. On-Chip memori. Memori ini sejenis memori yang secara fisik terdapat pada mikroposesor itu sendiri. Memori ini terdiri dari internal RAM (IRAM) dan SFR (*Spesial Function Register*). IRAM mempunyai besar 128 Kbyte dengan 8 alamat awal diisi oleh register banks. IRAM mempunyai alamat 00-7Fh sedangkan SFR mempunyai alamat 80h-FFh.
2. RAM eksternal. Memori ini adalah memori tambahan yang digunakan untuk menyimpan kelebihan data yang tidak tersimpan pada internal RAM
3. Memori kode (*Code Memori*). Memori ini adalah memori yang menyimpan program yang akan dilaksanakan
4. Memori bit. Memori bit adalah memori yang dapat digunakan untuk merubah nilai bit secara langsung

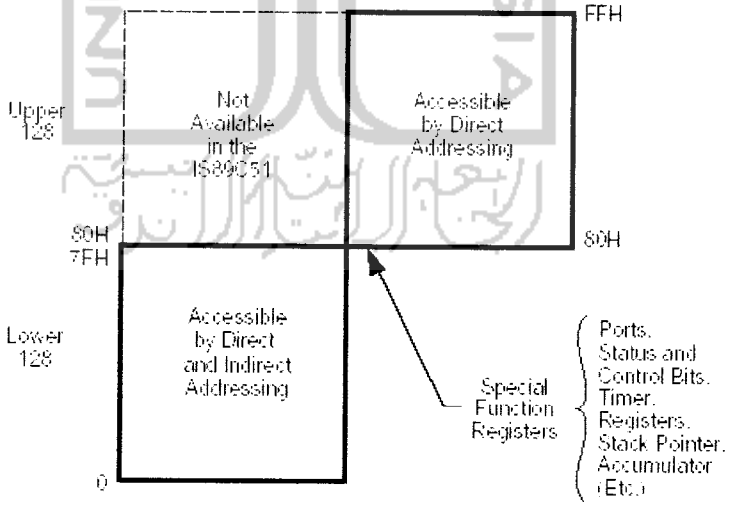
2.2.3 Memori Program

Memori program hanya bisa dibaca saja, terdapat memori yang bisa diakses langsung hingga 64 Kbyte. Sedangkan *strobe* (tanda) untuk akses program memori eksternal melalui sinyal PSEN *(Program Strobe Enable)*.

Jika pin $\overline{EA}$ dihubungkan ke ground, maka semua pengambilan instruksi langsung dilakukan pada memori eksternal. Pada AT89S52, jika $\overline{EA}$ disambung ke VCC, pengambilan instruksi pada lokasi 0000H hingga 1FFFH untuk memori internal dan pengambilan instruksi untuk memori eksternal pada lokasi 2000H sampai FFFFH.

2.2.4 Memori Data

AT89S52 mempunyai 256 byte memori RAM. 128 byte atas ditempati secara pararel oleh *special function register* (SFR). Ini berarti bahwa 128 byte atas mempunyai alamat yang sama dengan SFR, walaupun secara fisik terpisah.



Gambar 2.3. Memori data internal

[Sumber : *Atmel Literature Requests*, www.atmel.com/literature]

2.2.5 Fungsi-fungsi Pin Mikrokontroler AT89S52

Mikrokontroler AT89S52 terdapat 4 port, yang mana setiap portnya memiliki masing-masing 8 bit yang mempunyai fungsi-fungsi tertentu, seperti diutarakan dibawah ini.

- Port 0 (P0.0-P0.7)

  Port ini adalah port I/O bi-directional. Port ini jika diberi logika 1 dalam kondisi mengapung akan menjadi input *high-impedance*. Port ini juga menjadi bus jalur alamat dan jalur data untuk byte rendah jika ada memori luar yang digunakan.

- Port 1 (P1.0-P1.7)

  Port ini adalah port I/O bi-directional dengan internal *pull-up*. Dua kaki port 1 memiliki ragam fungsi, yang mempunyai fungsi khusus :

  T2 (P1.0) : Masukan eksternal pencacah 2

  T2EX (P1.1) : Pemicu reload pewaktu 2

- Port 2 (P2.0-P2.7)

  Port ini adalah port I/O bi-directional dengan internal *pull-up* . Port ini juga menjadi bus jalur alamat dan jalur data untuk byte tinggi jika ada memori luar yang digunakan (pengalamatan 16 bit).

- Port 3 (P3.0-P3.7)

  Port ini adalah port I/O bi-directional dengan *internal pull-up*. Port 3 dapat digunakan sebagai fungsi khusus yaitu:

  RxD (P3.0) : Serial Input port

  TxD (P3.1) : Serial Output Port

INT0 (P3.2) : External Interrupt

INT1 (P3.3) : External Interrupt

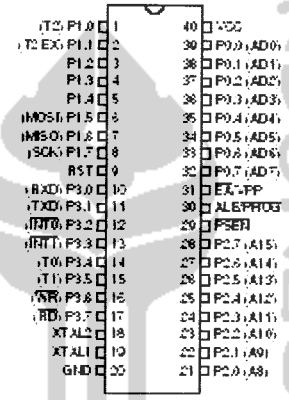
T0 (P3.4) : Timer 0 external input

T1 (P3.5) : Timer 1 external input

WR (P3.6) : External data memory *write strobe*

RD (P3.7) : External data memory *read strobe*

Adapunn susunan dari pin-pin mikrokontroler AT89S52 diperlihatkan pada Gambar 2.4 dibawah ini.



Gambar 2.4. Konfigurasi pin mikrokontroler AT89S52

[Sumber : *Atmel Literature Requests*, www.atmel.com/literature]

Adapun penjelasan dari pin-pinnya adalah sebagai berikut :

- ALE/PROG (Pin 30)

  Pin ini berfungsi meng-*enable* alamat byte rendah A0-A7 bila digunakan EPROM tambahan diluar, tetapi bila digunakan EPROM di dalam CHIP maka kaki ini berfungsi untuk menerima pulsa program selama proses pemrograman.

- $\overline{\text{PSEN}}$

  Sinyal keluaran yang mengaktifkan EPROM luar (*Program Store Enable*)

- $\overline{\text{EA}}$/Vpp

  Jika HIGH, mikro menjalankan perintah dari EPROM. Jika LOW, mikro menjalankan perintah dari EPROM luar. Pada saat pemrograman EPROM dalam, kaki ini menerima 21V untuk pemrograman EPROM.

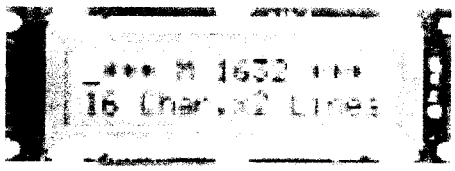
- RST (reset) : Merupakan master reset 89S52
- XTAL 1 : Masukan ke penguat osilator
- XTAL 2 : Output penguat osilator
- GND : Sebagai *Ground*
- VCC :Sebagai *Power suply*

2.3 LCD (*Liquid Crystal Display*)

LCD (*Liquid Crystal Display*) adalah suatu display dari bahan cairan kristal yang pengoperasiannya menggunakan sistem dot matriks. LCD yang digunakan dalam tugas akhir ini adalah LCD M1632 dengan tampilan 16 x 2 baris dengan konsumsi daya rendah yang dilengkapi dengan mikrokontroler HD44780U buatan Hitachi. Perangkat ini sudah dilengkapi dengan fasilitas *high-contrast* dan tampilan karakter yang besar (*Wide-View*) TN LCD dan CMOS LCD.

LCD M1632 memiliki tiga memori, yaitu DDRAM (*Display Data Random Access Memory*), CGRAM (*Character Generator Random Access Memory*), dan

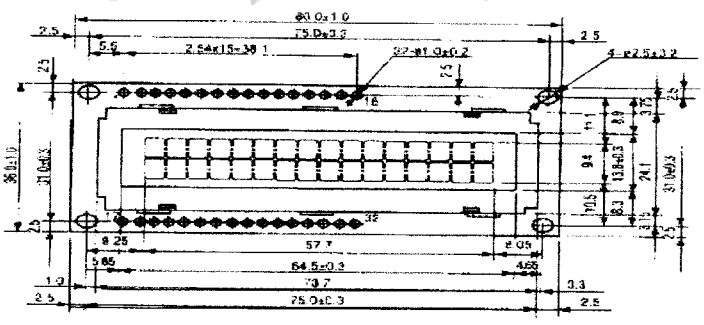
CGROM (*Character Generator Read Only Memory*). Bentuk fisik LCD M1632 ditunjukkan pada Gambar 2.5.

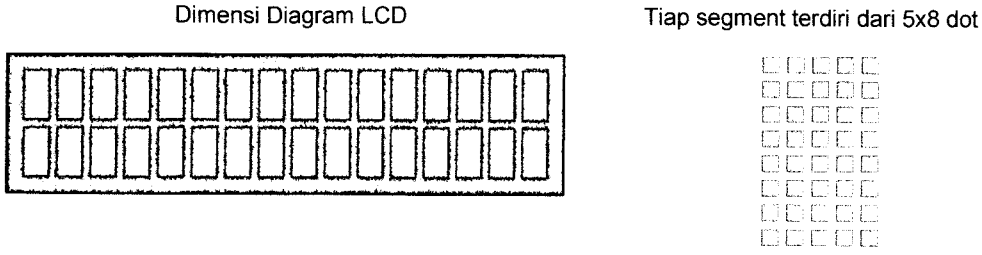


Gambar 2.5. Bentuk fisik LCD M1632

LCD (*Liquid Crystal Display*) M1632 memiliki fitur sebagai berikut:

a. Terdiri dari 16 segmen karakter, tiap segmen terdiri dari 5 x 7 dot matrix ditambah dengan kursor.
b. Menggunakan tegangan operasi yang rendah yaitu antara 2,7 V sampai 5,5 V.
c. Sesuai untuk aplikasi yang membutuhkan kerja dengan kecepatan tinggi.
d. LCD M1632 mempunyai 80 x 8-bit display sehingga mampu menulis 80 karakter maksimal (ditunjukkan pada Gambar 2.5).
e. Banyak fungsi instruksi yang dapat digunakan seperti: *display clear*, *cursor home*, display ON/OFF, *cursor* ON/OFF, *display character blink, cursor shift, dan display shift*.
f. Instruksi yang digunakan adalah standart CMOS.





Gambar 2.6. Dimensi LCD M1632

2.3.1 DDRAM (*Display Data Random Access Memory*)

DDRAM merupakan memori tempat karakter yang ditampilkan berbeda. Contoh, untuk karater 'A' ditulis pada alamat 00, maka karakter tersebut akan tampil pada baris pertama dan kolom pertama dari LCD. Apabila karater tersebut ditulis pada alamat 40, maka karakter tersebut akan tampil pada baris kedua kolom pertama dari LCD. Untuk lebih jelasnya bisa dilihat pada Gambar 2.7. Berikut ini.

Display position	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
DDRAM address	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
	40	41	42	43	44	45	46	47	48	49	4A	4B	4C	4D	4E	4F

Gambar 2.7. Alamat DDRAM M1632

2.3.2 CGRAM (*Character Generator Random Access Memory*)

CGRAM merupakan memori untuk menggambarkan pola sebuah karakter dimana bentuk dari karakter dapat diubah-ubah sesuai keinginan, tetapi memori ini akan hilang saat catu daya dimatikan/*power supply* tidak aktif, sehingga pola karakter akan hilang.

2.3.3 CGROM (*Character Generator Read Only memory*)

CGROM merupakan memori untuk menggambarkan pola sebuah karakter dimana pola tersebut sudah ditentukan secara permanen dari HD44780U sehingga pengguna tidak dapat mengubah lagi. Tetapi karena ROM bersifat permanen, maka pola karakter tersebut tidak akan hilang walaupun *power supply* tidak aktif.

Sebagai contoh pada saat HD44780U akan menampilkan data 41H yang tersimpan pada DDRAM, maka HD44780U akan mengambil data dialamat 41H (01000001b) yang ada pada CGROM yaitu pola karakter 'A'.

2.3.4 Konfigurasi PIN

LCD M1632 memiliki 14 PIN dengan fungsi yang berbeda-beda. Terdapat 3 PIN sebagai bit kontrol yaitu E sebagai input *clock*, R/W sebagai input untuk memilih *read* atau *write* dari RS sebagai *register select*, 8 PIN sebagai bit data yaitu DB0 sampai DB7 dan dua pin lainnya adalah VCC dan *Ground*. Konfigurasi pin LCD M1632 ditunjukkan pada Tabel 2.1.

Tabel 2.1. Konfigurasi pin LCD M1632

No	Nama pin	Keterangan
1	GND	0
2	VCC	+ 5 V
3	VEE	Tegangan kontras LCD
4	RS	Register Select, 0 = Register Perintah, 1 = Register Data
5	R/W	1 = Read, 0 = Write
6	E	Enable Clock LCD, Logika 1 setiap kali pengiriman atau pembacaan data
7	D0	Data Bus 0
8	D1	Data Bus 1
9	D2	Data Bus 2
10	D3	Data Bus 3

11	D4	Data Bus 4
12	D5	Data Bus 5
13	D6	Data Bus 6
14	D7	Data Bus 7
15	Led +	+ 5 V
16	Led -	0

2.3.5 Register

HD44780U mempunyai dua buah register yang aksesnya diatur dengan menggunakan kaki RS. Pada saat RS berlogika '0', maka register yang akan diakses adalah register perintah dan pada saat RS berlogika 1, maka register yang diakses adalah Register data.

2.3.5.1 Register Perintah

Register ini adalah register dimana perintah dari mikrokontroler ke HD44780U pada saat proses penulisan data atau tempat status dari HD44780U dapat dibaca pada saat pembacaan data.

2.3.5.2 Register Data

Register ini adalah register dimana perintah mikrokontroler dapat menuliskan atau membaca data ke atau dari DDRAM. Penulisan data pada register ini akan menempatkan data tersebut ke DDRAM sesuai dengan alamat yang telah diatur sebelumnya.

2.3.5.3 Penulisan Data ke Register Perintah dan Register Data

Penulisan data ke register perintah dilakukan dengan tujuan mengatur tampilan LCD, inisialisasi dan mengatur *address counter* ataupun *address data*. Kondisi RS berlogika 0 menunjukkan akses data ke register perintah. RW berlogika 0 yang menunjukkan proses penulisan data. Sedangkan penulisan data

pada register data dilakukan untuk mengirimkan data yang akan ditampilkan pada LCD. Proses diawali dengan adanya logika 1 pada RS yang menunjukkan akses ke register data, kondisi R/W diatur pada logika 0 yang menunjukkan proses penulisan data. Pengiriman data dari data Bus DB0 – DB7 diawali dengan pemberian pulsa logika 1 pada E *clock*. Dan akhiri dengan pulsa logika 0 pada E *clock*.

2.4 DC Motor

Motor DC adalah mesin yang mengubah energi listrik menjadi energi mekanis yang berupa kecepatan atau putaran, dalam hal ini energi listrik yang diubah adalah listrik arus searah (DC). Dengan memperhatikan fungsi dari motor tersebut maka konstruksi motor DC sangat mirip dengan generator DC.

DC motor atau istilah lain dikenal sebagai dinamo ialah motor yang paling banyak digunakan untuk *mobile robot*. DC motor tidak berisik, dan dapat memberikan daya yang memadai untuk tugas – tugas berat. Motor DC standar berputar secara bebas berbeda halnya dengan motor steper.

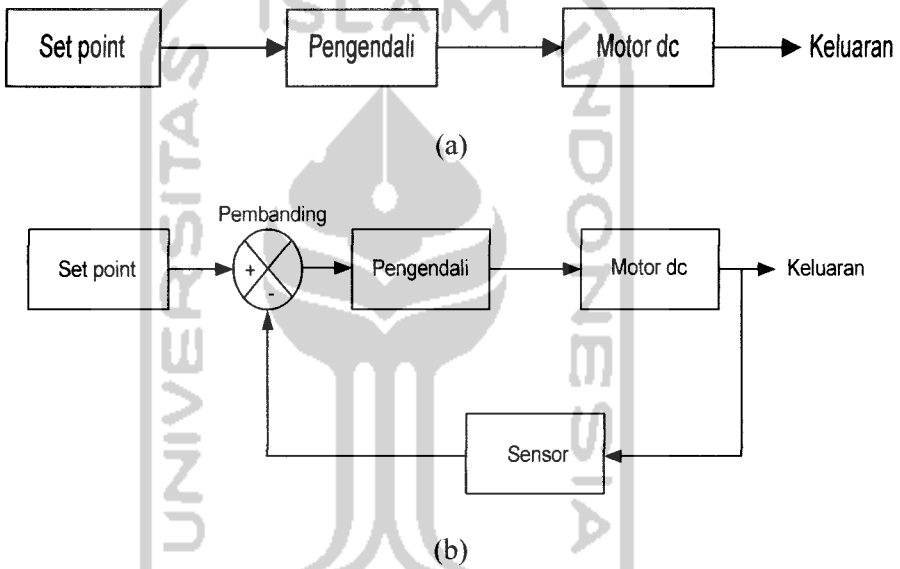
Untuk mengetahui berapa banyak putaran biasanya digunakan mekanisme feedback. Pada beberapa aplikasi *mobile* robot, biasanya dibuat agar motor dapat:

- Bergerak dengan arah maju mundur (dua arah)
- Memodifikasi kecepatannya sesuai dengan yang diinginkan

Oleh karena itu dibutuhkan dengan apa yang disebut H-Bridge, yang dapat dipenuhi dengan transistor daya. Namun saat ini sudah banyak IC yang berfungsi

sebagai H-Bridge dengan arus yang cukup besar, salah satunya adalah L293D dari ST-SGS-Thomson.

Pada motor dc apabila ingin memodifikasi kecepatannya pada umumnya menggunakan metoda pengendalian kalang tertutup. Pada motor dc terdapat beberapa metode pengendalian kecepatan motor, dimana pengendali dan algoritma kendali yang diterapkan harus menyesuaikan dengan metode pengendali kecepatan motor.



Gambar 2.8. Blok diagram pengendalian motor DC

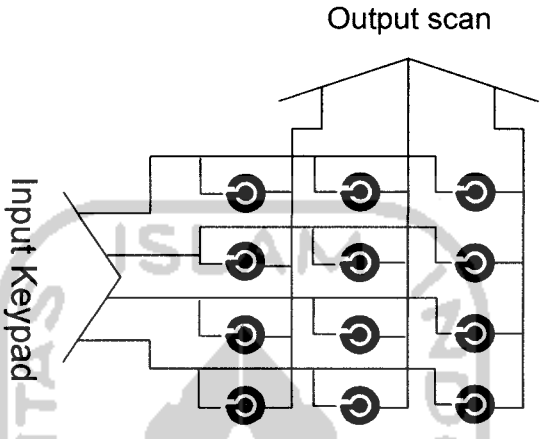
Keterangan:

a. Blok diagram pengendalian motor DC kalang terbuka
b. Blok diagram pengendalian motor DC kalang tertutup

2.5 Keypad

Keypad merupakan susunan dari beberapa saklar yang disusun sedemikian rupa sehingga bisa digunakan sebagai piranti masukan pada mikrokontroler.

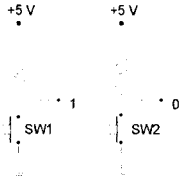
Salah satu jenis keypad adalah keypad matriks. Keypad matriks yang digunakan ini, merupakan piranti yang prinsip kerjanya berfungsi sebagai pengganti kerja Keyboard.



Gambar 2.9. Keypad matriks 3 x 4

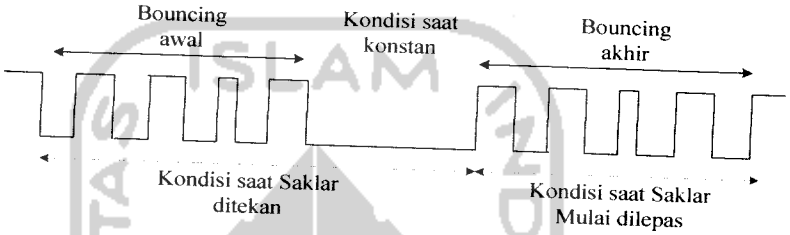
Keypad matriks terdiri dari beberapa saklar yang tersusun dalam bentuk kolom dan baris yang masing-masing dihubungkan dengan port I/O mikrokontroler. Saklar-saklar tersebut dihubungkan sedemikiran rupa, sehingga apabila salah satu saklar ditekan maka akan menghubungkan salah satu baris dengan salah satu kolomnya. Susunan keypad matrik 3 x 4 dapat dilihat pada Gambar 2.9.

Pada saklar mekanis hal yang perlu dipertimbangkan adalah saat saklar ditekan dan dilepas terjadi *bouncing* (logika saklar yang tidak dapat diperkirakan).



Gambar 2.10. Contoh penekanan tombol

Misalkan seperti Gambar 2.11, saat penekanan tombol kondisi yang diharapkan adalah '0' tetapi karena adanya efek *bouncing* kondisi yang diharapkan tersebut tidak dapat diperkirakan. Untuk mengatasi hal tersebut maka pembacaan data saklar dilakukan setelah selang waktu tertentu untuk saklar mekanis biasanya sekitar 10-20 ms. Gambar 2.11 menunjukkan efek *bouncing* dari saklar mekanis.



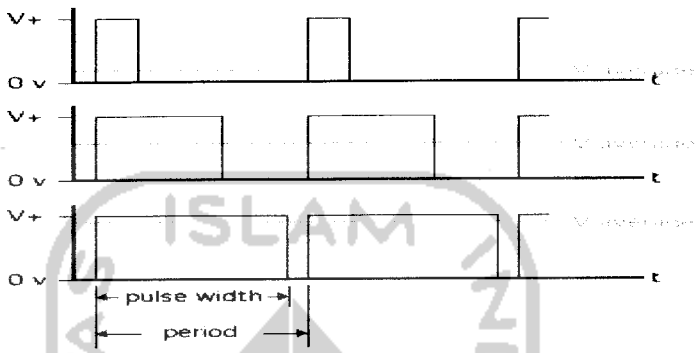
Gambar 2.11. Efek *bouncing* pada penekanan saklar

2.6 PWM

Pulse width modulation atau PWM adalah metode untuk mengatur kecepatan motor dan menghindarkan motor mengkonsumsi daya yang berlebih. PWM dapat mengatur kecepatan motor, karena tegangan yang diberikan dalam selang waktu tertentu saja. PWM dapat dibangkitkan dengan modifikasi pada *software*.

Control PWM bekerja dengan cara penyaklaran on off pada *power supply* sebagai sumber tegangan ke motor dengan sangat cepat sekali. Tegangan DC dirubah ke sebuah sinyal gelombang kotak, secara bolak balik diantara sinyal full on (12Vdc) dan sinyal 0, hal ini akan memberikan motor kekuatan sebagai sebuah gerak awalan pada motor. Jika penyaklaran frekuensi cukup tinggi, motor berputar

di kecepatan yang mantap dan momentum yang baik. Dengan penambahan *duty cycle* pada sinyal PWM, lamanya waktu ketika ON, daya rata-rata dan kecepatan motor dapat divariasi.



Gambar 2.12. Pulsa PWM

Pengaturan control sinyal PWM ini bisa dilakukan dengan sebuah mikro yaitu dengan cara memberikan program untuk pembuatan sinyal PWM sehingga bisa membuat lebar pulsa yang diinginkan.

BAB III

PERANCANGAN

3.1 Gambaran Umum sistem

Perangkat keras yang akan dibangun adalah suatu aplikasi mikrokontroler untuk kompas digital yang nantinya akan digunakan untuk arah gerak robot tank (motor DC). Input untuk aplikasi ini berupa sensor arah CMPSO3 dan sebuah keypad. Outputnya berupa LCD dan sebuah motor DC. Diagram blok sistem dapat dilihat pada Gambar 3.1. berikut ini.



Gambar 3.1. Diagram blok

Sistem kompas yang dibangun dapat menampilkan arah utara, selatan, timur, barat bumi dan juga output dalam bentuk sudut (derajat) yang nantinya dapat digunakan sebagai feedback motor dc sehingga dengan input dari sebuah keypad (nilai sudut), motor tersebut akan bergerak sesuai dengan derajat atau sudut yang sesuai dengan nilai yang diinginkan. Sistem ini dilengkapi dengan penampil berupa LCD dengan tampilan output jumlah nilai sudut dalam bentuk

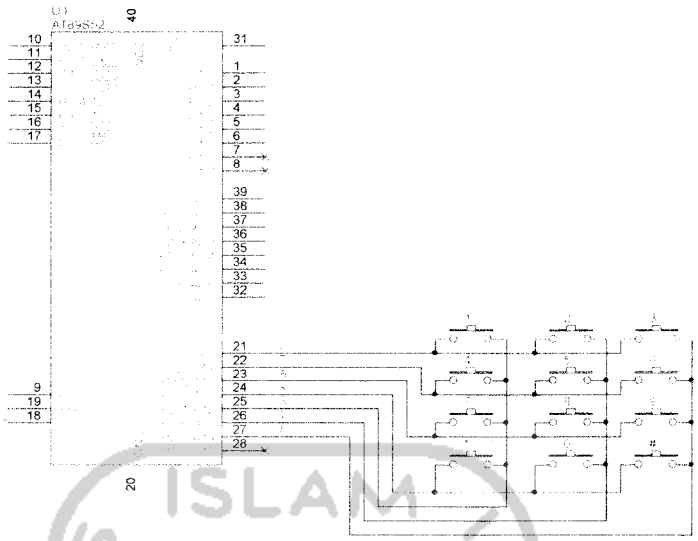
derajat dan juga sekaligus dengan arah–arahnya seperti utara, selatan, barat, dan timur. Sehingga kita bisa dengan mudah melihat arah atau posisi motor ketika waktu bergerak ataupun ketika motor telah menunjukan arah yang diinginkan (diam).

Metode yang digunakan untuk sistem kompas digital ini dengan metode PWM. Yaitu dengan mengatur lebar pulsa.

3.2 Perancangan Perangkat Keras

3.2.1 Rangkaian *Keypad*

Keypad merupakan rangkaian saklar yang berfungsi sebagai masukan, dalam alat ini yang digunakan adalah *keypad* matrik 3 x 4, yang mana 4 bit barisnya dihubungkan dengan P2.0–P2.3, sedangkan 3 bit kolomnya dihubungkan dengan P2.4–P2.6. P2.0–P2.3 berfungsi sebagai keluaran *keypad* mikrokontroler dan P2.4–P2.6 berfungsi sebagai masukan *keypad* mikrokontroler. Cara kerja dari *keypad* matriks ini adalah dengan sistem *scanning* tombol satu per satu. Mikrokontroler akan memberikan kondisi '0' pada setiap port keluaran secara bergantian sehingga apabila pada saat yang sama ada penekanan tombol maka pada port masukan akan memiliki kondisi yang sama yaitu '0' seperti ditunjukkan pada Gambar 3.2.



Gambar 3.2. Rangkaian *keypad*

Apabila port keluaran dikondisikan dengan bit 1110, lalu pada tombol kolom kedua tersebut ditekan, maka masukan yang dihubungkan oleh tombol tersebut menjadi '0' dan kemudian oleh mikrokontroler diterjemahkan menjadi angka tertentu, seperti ditunjukkan pada Tabel 3.1.

1	2	3
4	5	6
7	8	9
*	0	#

Gambar 3.3. Bentuk fisik *keypad*

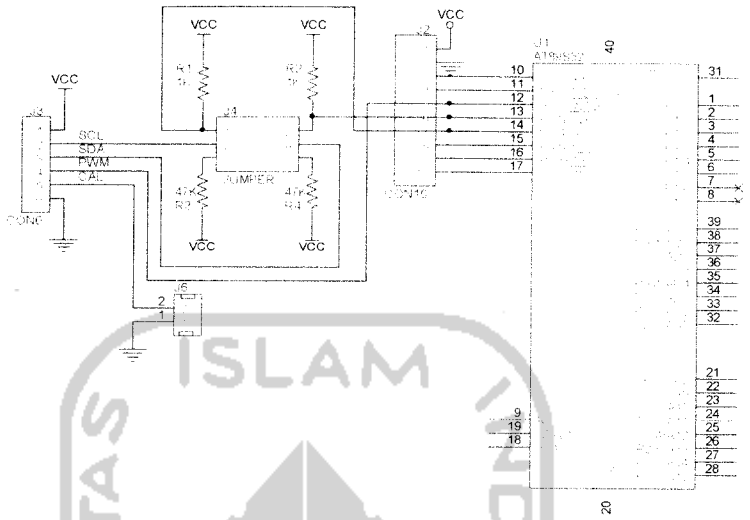
Tabel 3.1. Pengkodean *keypad*

N0	Data tombol	Kombinasi kolom P2.6 – P2.4	Kombinasi baris P2.3 – P2.0
1.	1	110	1110
2.	2	101	
3.	3	011	
4.	4	110	1101
5.	5	101	
6.	6	011	
7.	7	110	1011
8.	8	101	
9.	9	011	
10.	SET SUDUT	110	0111
11.	0	101	
12.	ENTER	011	

3.2.2 Rangkaian Sensor

Pada sensor kompas CMPSO3 terdiri dari beberapa pin yaitu VCC, GND merupakan pin untuk catu daya 5V DC, dan 3 pin yang merupakan output yaitu PWM, SDA, SCL,dan juga 1 pin yang berfungsi untuk kalibrasi yaitu CAL. Output PWM dihubungkan ke P3.2 yang merupakan port interupsi 0 (INT0) dan untuk I2C yaitu SDA dan SCL dihubungkan ke P3.3 (INT1) dan P3.4 (T0). Rangkaian ini juga diberikan saklar yang berfungsi untuk memilih metode yang akan digunakan yaitu antara metode PWM atau I2C. Namun pada sistem ini hanya memakai metode PWM yaitu pada P3.2. Pada rangkaian ini juga disediakan saklar yang berfungsi sebagai kalibrasi arah kompas. Kalibrasi ini bisa dipakai berulang

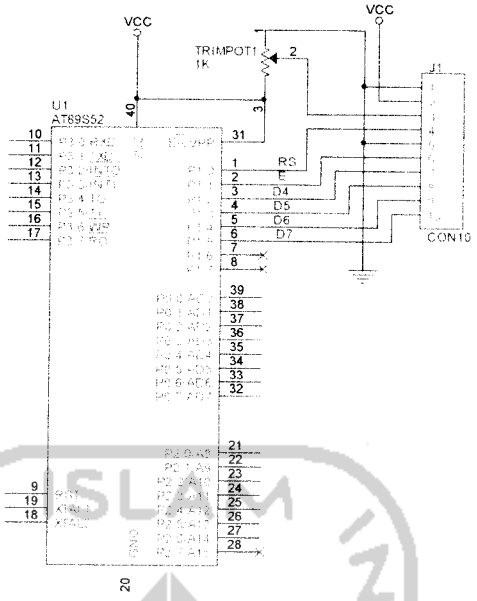
ulang sampai arah kompas sesuai dengan yang diinginkan yaitu sesuai standar posisi bumi. Untuk lebih jelasnya bisa dilihat Gambar 3.4 berikut ini.



Gambar 3.4. Rangkaian sensor.

3.2.3 Perancangan Rangkaian LCD

LCD adalah alat yang berfungsi sebagai penampil. Pada perancangan ini digunakan LCD H1602B 2 x 16 baris buatan Hitachi yang mempunyai 16 pin yang berfungsi sebagai jalur data, dengan konsumsi daya rendah menggunakan catu daya sebesar 5V. Untuk jalur LED atau *backlight* yang terdapat pada pin 15 diberi sebuah dioda agar tegangan tidak melebihi kebutuhan sehingga tidak merusak LCD sedangkan untuk mengatur kontras agar layar di LCD terlihat jelas pada rangkaian diberi V-R sebesar 1KΩ. Rangkaian selengkapnya ditunjukkan pada Gambar 3.5.

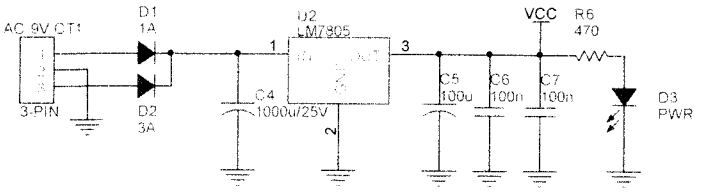


Gambar 3.5. Rangkaian LCD

Untuk koneksi LCD ke mikrokontroler dibutuhkan data bus dan kontrol RS, dan E. Untuk data bus digunakan P1, sedangkan untuk RS = P1.0 dan E=P1.1.

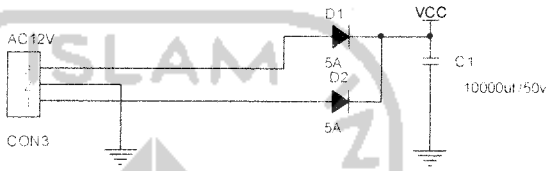
3.2.4 Perancangan Catu Daya

Untuk rangkaian catu daya digunakan transformator CT (*Center Tap*). Tegangan yang diambil dari trafo sebesar 12V CT AC, untuk mengubahnya menjadi DC digunakan 2 dioda 1N4002 yang berfungsi menyearahkan arus, output dari diode ini belum benar – benar bersih sehingga digunakan kapasitor untuk menghilangkan *noise*. Output dari catu daya ini harus dibuat se-stabil mungkin agar rangkaian berjalan dengan normal dan tidak terjadi *noise* atau gangguan, digunakan regulator yang juga digunakan untuk menurunkan tegangan sesuai kebutuhan semua rangkaian. Rangkaian catu daya selengkapnya ditunjukan pada Gambar 3.6.



Gambar 3.6. Rangkaian catu daya 5v dc

Rangkaian catu daya pada Gambar 3.6 digunakan untuk mengoperasikan sensor CMPSO3, dan juga untuk mengoperasikan mikrokontroler, dan LCD.



Gambar 3.7. Rangkaian catu daya 12 v dc

Rangkaian pada Gambar 3.7 adalah rangkaian untuk mengoperasikan motor dc dengan output 12V DC. Rangkaian tersebut menggunakan 2 buah dioda 5 amper dan sebuah kapasitor 10000uf dengan trafo CT 5A.

3.2.5 Mikrokontroler AT89S52

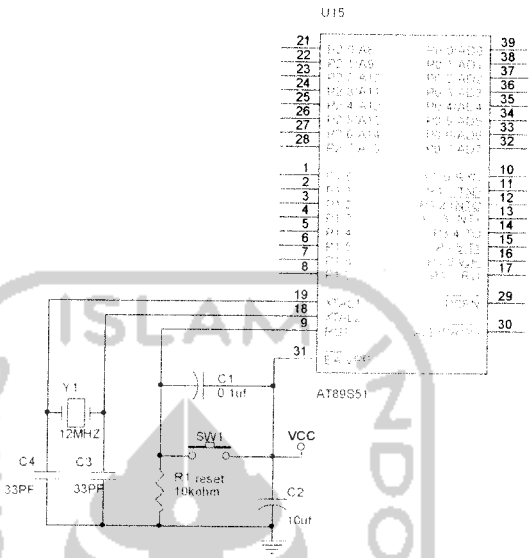
3.2.5.1 Osilator

Pada rangkaian osilator ini digunakan kristal 12 MHz seperti ditunjukkan Gambar 3.8. Menurut *datasheet* crystal yang bisa digunakan untuk mikrokontroler AT89S52 adalah 0 – 33 MHz dan dua kapasitor 30 ± 10 pF.

3.2.5.2 Reset

Rangkaian *reset* digunakan untuk menghentikan kerja mikrokontroler dengan kembali ke alamat 0000/*reset*. Rangkaian *reset* dapat dilihat pada Gambar 3.8. Untuk mereset mikrokontroler AT89S52 yaitu dengan memberikan logika

tinggi pada pin reset (pin 9) mikrokontroler AT89S52, logika tinggi ini dibuat minimal dua *machine cycles*. Jika menggunakan kristal 12 MHz maka logika tinggi minimal selama 2 us.

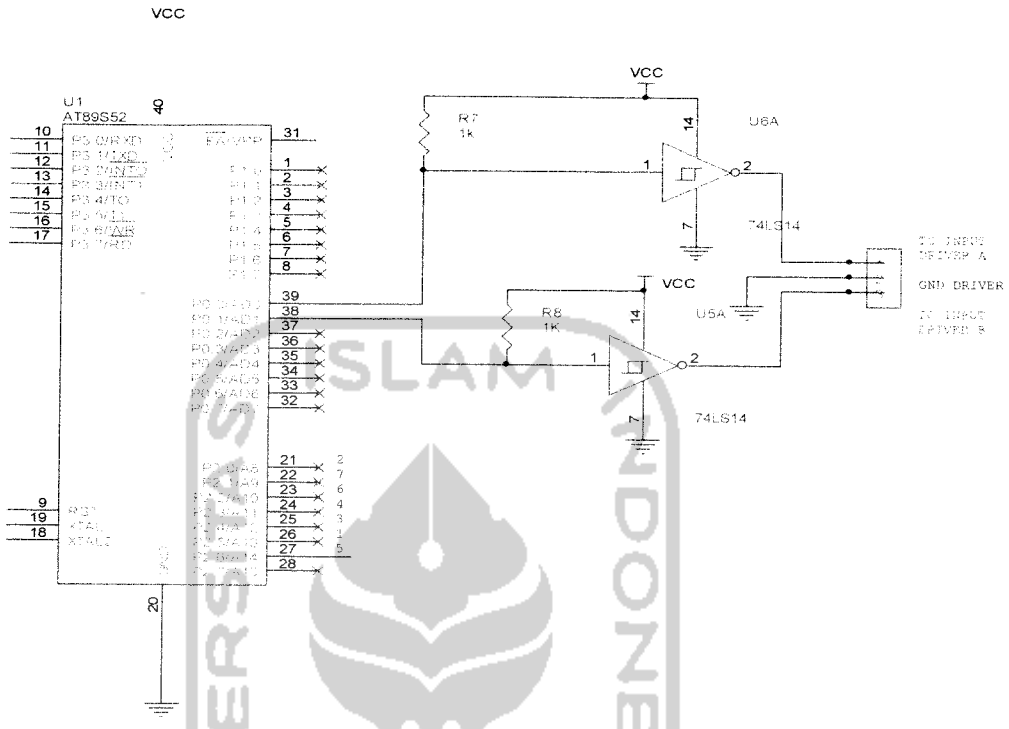


Gambar 3.8. Rangkaian minimum AT89S52

3.2.6 Perancangan rangkaian PWM

Rangkaian yang digunakan merupakan rangkaian driver motor dengan daya kira-kira sampai 200 watt atau 5 amper 40 volt. Dan untuk motor dc yang mengkonsumsi tegangan 6 volt sampai 40 volt. Pada rangkaian ini digunakan rangkain H-Bridge dengan menggunakan IC darlington yaitu TIP 125 dan TIP 120. Rangkaian ini memiliki dua input logika yang kompatibel dan dua output yaitu masing-masing terdiri dari A dan B. Jika input A diberi logika high dan B diberi logika low maka output akan menggerakan motor ke satu arah dan jika input diberi logika sebaliknya maka motor akan bergerak berlawanan arah dengan arah pertama. Driver ini jika input keduanya diberi logika sama maka driver tidak

bekerja (diam) dan rangkaian tidak mengkonsumsi daya sama sekali. Pada sistem yang digunakan rangkaian ini dihubungkan dengan 12V DC.

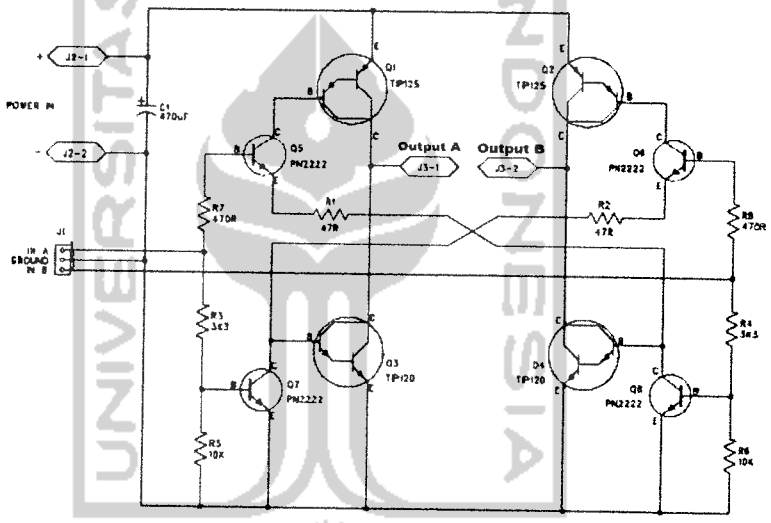


Gambar 3.9. Rangkaian mikro ke driver motor

Out put untuk driver motor dari mikro menggunakan P0.0 dan P0.1 yang kemudian masing masing diberi resistor 1k ohm yang terhubung ke VCC yang berfungsi sebagai pull up. Lalu kemudian masing - masing output dari IC ini akan di masukan ke driver motor untuk lebih jelasnya bisa dilihat Gambar 3.9 dan juga Gambar 3.10 rangkaian driver H-Bridge.

Tabel 3.2. Kebenaran pada rangkaian driver motor

Input		Out put	
A	B	A	B
0	0	Diam	
0	1	0	1
1	0	1	0
1	1	Diam	



Gambar3.10. Rangkaian H-Bridge

3.3 Perancangan Perangkat Lunak

Agar sistem bekerja dengan sebagaimana mestinya, maka diperlukan perangkat lunak yang mengatur kerja dari keseluruhan rangkaian. Pertama yang dibuat adalah diagram alir (*Flowchart*) dan kemudian dilakukan pembuatan program. Pembuatan program ditulis dalam bahasa C dengan menggunakan

SDCC (*Small Device C Compiller*) Mikrokontroler AT89S52, dan program tersebut disimpan dalam memori flash. Pada mikrokontroler AT89S52 terdapat memori program sebesar 4 kbyte flash dan memori data 128 byte RAM.

3.3.1 Program Inisialisasi Mikrokontroler AT89S52

Program inisialisasi mikrokontroler meliputi inisialisasi bit, inisialisasi alamat interupsi, inisialisasi interupsi *timer*, seting peralatan pada kondisi awal, inisialisasi penyimpanan program di *flash* memori, dan pengosongan alamat memori RAM. Mikrokontroler AT89S52 mempunyai dua *timer/counter* yaitu *timer* 0 dan *timer* 1. Dalam pembuatan alat ini hanya digunakan *timer* 0 untuk pengaturan peralatan

Interupsi *timer* berperan penting, dengan fasilitas ini akan lebih mudah membuat program yang memerlukan waktu pengontrol untuk menjalankan aplikasi yang terkait pada program tersebut. Untuk menjalankan fasilitas timer memerlukan prosedur sebagai berikut;

a. Pengisian register TH dan TL sesuai dengan lamanya interupsi yang diinginkan.
b. Pengesetan bit EA yang mengizinkan adanya interupsi.
c. Pengesetan bit ET yang mengizinkan adanya interupsi timer.
d. Menetukan mode timer
e. Pengesetan bit TR untuk menjalankan timer.

Mode timer mikrokontroler yang digunakan adalah mode 0, dengan maksimal cacahan 65.536. Timer dibuat dengan seting setiap 50.000 melakukan

interupsi. Nilai 50.000 dipengaruhi oleh kristal yang digunakan. Jika kristal yang digunakan adalah 12 MHz maka 1 *cycle* mikrokontroler adalah 1 *microsecond.* Untuk membentuk 1 detik dibutuhkan 20 kali interupsi. Sehingga data yang dimasukkan ke register tl dan th adalah 50.000

$$1\ Cycle = \frac{1}{f.crystal} x12$$

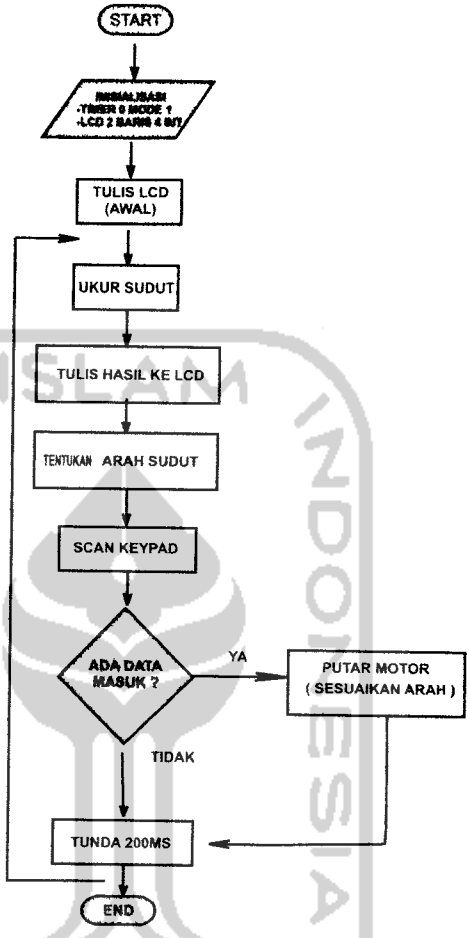
$$1\ Cycle = \frac{1}{12MHz} X12$$

$$1\ Cycle = 1\ microsecond$$

3.3.2 Program utama

Program utama meliputi sub rutin–sub rutin dari tiap-tiap rangkaian. Dan mekanismenya adalah yang pertama dimulai program akan menginisialisasi *timer* 0 dan *mode* 1 dan berikut LCD 16x2 sebagai penampil yang digunakan, kemudian mikro akan menampilkan tulisan awal ke LCD. Selanjutnya mikrokontroler akan mengukur sudut dengan cara membaca output dari sensor dan kemudian hasil dari pengukuran tersebut ditampilkan ke LCD yang sebelumnya telah ditentukan terlebuh dahulu arah dari pembacaan sudut yang terukur. Lalu kemudian program akan mensekan *keypad,* jika ada data masuk (tombol pada *keypad* ada yang ditekan) maka program akan mengerakan motor kearah yang sesuai dengan data yang dimasukan dan jika tidak ada data yang masuk (tombol *keypad* tidak ada yang ditekan), maka program akan berhenti sekitar 200 milidetik dan kemudian akan kembali lagi tahap mengukur sudut dan

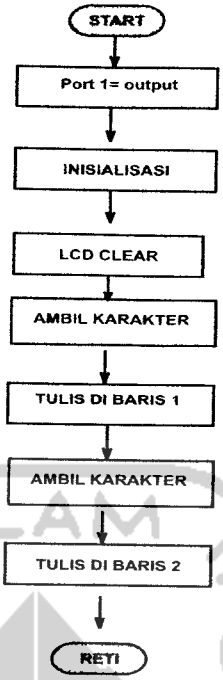
program akan terus-menerus mengalami pengulangan seperti yang telah dijelaskan. Berikut adalah diagram alir dari program utama.



Gambar 3.11. Flowchart program utama.

3.3.3 Program LCD

LCD 2x16 terdiri dari 16 karakter 2 baris yang dapat menampilkan karakter (angka, huruf, simbol) dalam format ASCII. *Flowchart* Pemrograman LCD dapat dilihat pada Gambar 3.12 berikut ini.

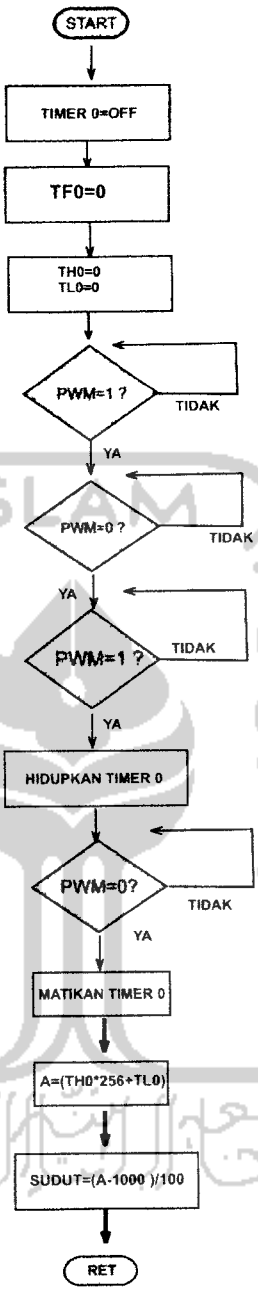


Gambar 3.12. *Flowchart* program LCD

Pada sub rutin LCD, program pertama kali akan menginisialisai port 1 pada mikrokontroler sebagai output dan program juga akan menginisialisasi program lainnya. Kemudian LCD akan mengambil karakter yang telah ditulis didalam program dan ditampilkan di baris pertama pada LCD yang sebelumnya LCD di bersihkan (reset) terlebih dahulu oleh program kemudian LCD akan kembali mengambil karakter dan selanjutnya ditampilkan pada baris kedua.

3.3.4 Sub rutin Ukur Sudut

Pada sub ini menjelaskan tentang bagaimana pengukuran sudut pada sistem sehingga akan ditampilkan di LCD. Pada sub ini memakai timer 0 mode 1.



Gambar 3.13. Ukur sudut

Pertama kali program akan mengosongkan register timer 0 yaitu TF0,TH0,TL0 dengan cara memberikan data 0. kemudian mikrokontroler akan membaca masukan berupa PWM yang merupakan output dari sensor pada *port*

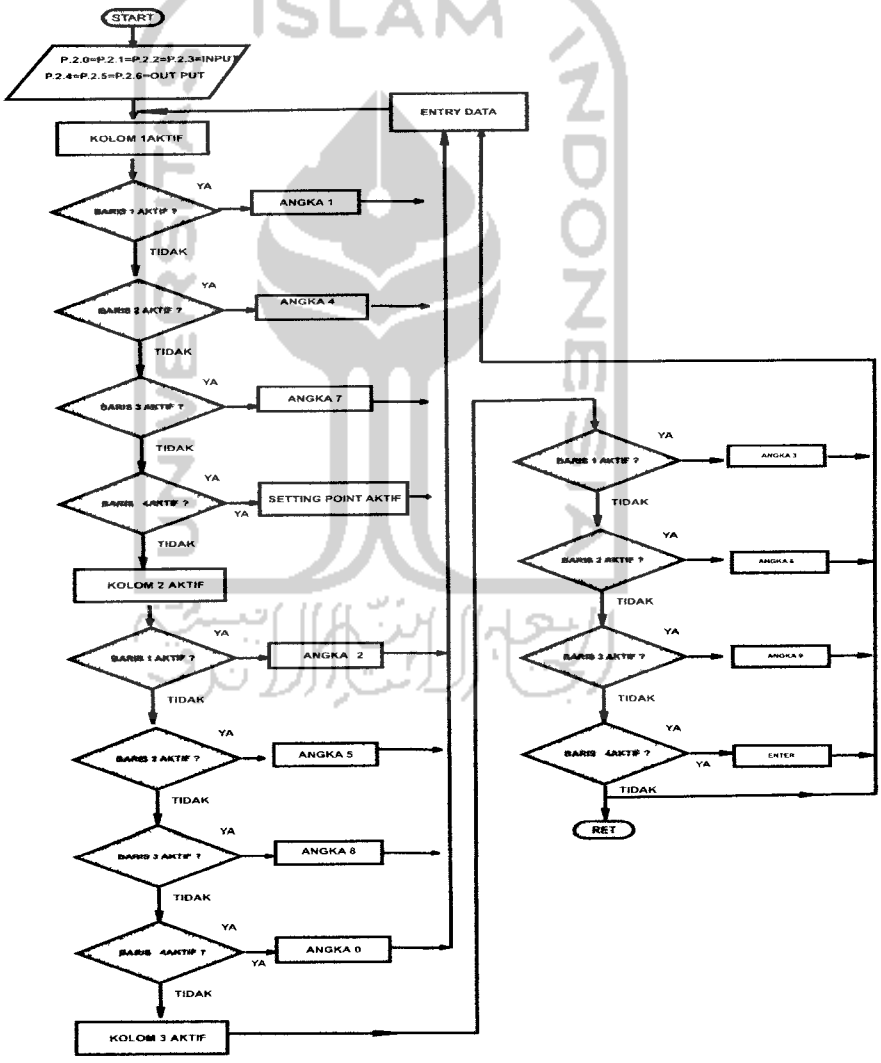
3.4. Setiap satu periode PWM dalam keadaan *high* dan *low* (ON) maka maka program akan menghidupkan dan mematikan *timer 0* (*high = timer on, low = timer off*). Kemudian setelah itu register TH0 dan TL0 akan mencacah PWM dalam keadaan (ON) tersebut. lalu kemudian hasil cacahan tersebut akan dimasukan kedalam rumus sudut hingga dihasilkan nilai sudut.

3.3.5 Sub rutin Keypad

Berisikan tentang *scan keypad* yaitu scaning tombol-tombol pada *keypad.* Pada *keypad*, terdiri dari tiga kolom dan akan diberi nilai biner yang akan berubah ubah dan empat baris yang diberi nilai biner 1111 secara konstan dan hanya akan berubah jika terdapat tombol *keypad* yang ditekan sehingga dihasilkan kombinasi-kombinasi bilangan biner yang mewakili masung-masing angka pada *keypad* dan kemudian dibaca oleh mikrokontroler dan ditampilkan dalam bentuk angka sesuai dengan tombol yang ditekan. untuk lebih jelasnya bisa dilihat pada Tabel 3.1. Pengkodean *keypad* halaman 32.

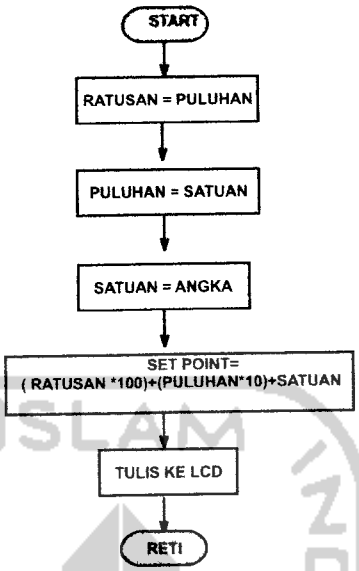
Yang pertama dimulai dengan inisialisasi port input dan output yang dipakai lalu kemudian program akan mengaktifkan kolom satu dengan memberikan logika 110 kemudian program akan *menscan* baris pertama apakah aktif, jika aktif maka ada tombol yang ditekan dan ini akan menghasilkan kombinasi pada baris berubah dari 1111 menjadi 1110 dan ini berarti pada pada baris dan kolom akan menghasilkan kombinasi 1101110 dan ini mewakili angka 1 kemudian tulis angka 1 pada *entry data.* Kemudian program akan kembali pada kolom satu aktif dan kemudian akan *menscan* baris pertama, kedua, ketiga dan

keempat apakah aktif atau tidak, Jika aktif, maka tulis ke *entry data* jika tidak maka akan lanjut dengan mengaktifkan kolom kedua dan baris dengan memberikan kombinasi-kombinasi bilangan biner. Proses *scaning* sama dan pemberian kombinasi bilangan pada kolom adalah sama dengan sebelumnya yang berbeda hanyalah pemberian kombinasi pada baris. Begitu juga dengan kolom ketiga. Proses ini mengalami pengulangan dengan sistem *scaning* secara terus menerus.



Gambar 3.14. Flowchart scan keypad

3.3.6 Sub rutin Entry Data

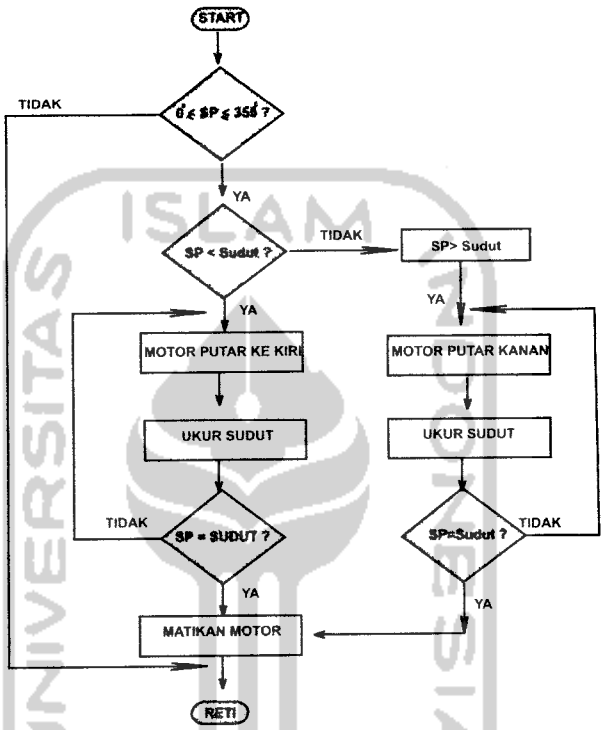


Gambar 3.15. Sub rutin *entry data*

Pada sub rutin ini, program akan mengklasifikasikan angka-angka yang dimasukan lewat tombol kemudian merangkainya menjadi puluhan, ratusan, satuan dengan cara pergeseran pada LCD. Dimulai dengan program menginisialisasi ratusan sama dengan puluhan, puluhan sama dengan satuan, satuan akan sama dengan angka. Pada *entry data* masukan nilai sudut maksimal terdiri dari tiga digit sehingga maksimal nilai yang dimasukan adalah dalam bentuk ratusan. Hal ini berarti jika ada tiga angka yang dimasukan maka angka pertama akan digeser menjadi puluhan setelah ada angka yang kedua masuk dan angka yang kedua ini akan dianggap sebagai satuan, kemudian angka yang pertama akan digeser kembali dari puluhan menjadi ratusan dan juga angka yang kedua dari satuan akan menjadi puluhan dan angka yang baru dimasukan sebagai

satuan. Sehingga tampilan di LCD akan terdiri dari tiga digit yaitu ratusan, puluhan, dan satuan.

3.3.7 Pengendalian Motor



Gambar.3.16. Pengendalian motor

Sub rutin ini akan aktif jika pada program utama terdapat *entry data* artinya jika ada masukan pada *keypad* yang disebut *set point (*SP*).* Pertama kali program mengeksekusi apakah ada SP, jika tidak ada maka program akan langsung ke sub rutin berikutnya dan jika ada maka program akan membandingkan SP tersebut apakah nilainya kurang dari sudut kompas yang terbaca saat itu atau tidak, jika nilai SP lebih kecil dari sudut yang terbaca kompas saat itu maka motor akan berputar ke kiri. Kemudian selama motor berputar, sudut

yang dibaca motor akan dibandingkan apakah sudah sama dengan SP. Artinya apakah gerak motor sudah sama dengan SP yang diinginkan. Jika iya, maka sistem akan mematikan motor dan jika tidak, maka motor akan berputar lagi hingga akhirnya arah putaran motor sama dengan SP. Kemudian jika nilai SP lebih besar dari sudut yang terbaca, maka motor akan berputar ke kanan lalu kemudian sudut motor tersebut akan diukur dan dibandingkan dengan SP. Jika sudut yang terukur motor sama dengan SP maka motor akan berhenti dan jika nilainya masih belum sama maka motor akan berputar terus hingga akhirnya sudut yang terukur oleh motor sama dengan SP.

BAB IV

HASIL PENGAMATAN DAN PEMBAHASAN

4.1 Metode Pengujian

Berdasarkan spesifikasi sistem yang telah dijelaskan sebelumnya, selanjutnya dilakukan pengujian terhadap sistem menggunakan beberapa metode pengujian. Tujuan pengujian ini untuk membuktikan apakah sistem yang diimplementasikan telah memenuhi spesifikasi yang telah direncanakan sebelumnya. Hasil pengujian akan dimanfaatkan untuk menyempurnakan kinerja sistem dan sekaligus digunakan dalam pengembangan lebih lanjut.

Metode pengujian dipilih berdasarkan fungsi operasional dan beberapa parameter yang mendudukung dan yang ingin diketahui dari sistem tersebut. Data yang diperoleh dari metode pengujian yang dipilih tersebut dapat memberikan informasi yang cukup untuk keperluan penyempurnaan sistem.

Dalam penelitian ini dipilih dua macam metode pengujian, yaitu pengujian fungsional dan pengujian kinerja sistem. Pengujian fungsional digunakan untuk membuktikan apakah sistem yang diimplementasikan dapat memenuhi persyaratan fungsi operasional seperti yang direncanakan.

Pengujian kinerja sistem dimaksudkan untuk memperoleh beberapa parameter yang dapat menunjukkan kemampuan dan kehandalan sistem dalam menjalankan fungsi operasionalnya.

4.2 Pengujian Fungsional

Sebagaimana dijelaskan sebelumnya, pengujian fungsional bertujuan untuk memeriksa fungsi operasional sistem yang diimplementasikan apakah telah sesuai dengan spesifikasi yang direncanakan dan sistem menjalankan fungsinya sesuai dengan tujuan pengembangannya.

Ada dua macam metode pengujian fungsional yang dilakukan. Metode pertama adalah pengujian fungsional bagian demi bagian, sedangkan yang kedua adalah pengujian sistem secara keseluruhan.

4.2.1 Pengujian Rangkaian Sensor

Sensor yang digunakan adalah CMPSO3 yang berfungsi untuk mendeteksi arah utara dengan cara mendeteksi magnet bumi. Sensor ini memiliki output digital yang berupa PWM. Pengujian rangkaian ini dengan cara mengamati perubahan bentuk output PWM yang dikeluarkan oleh sensor dan juga perubahan frekuensi per tiap sudutnya, sehingga bisa diketahui perubahan yang terjadi. Pengamatan dilakukan dengan menggunakan osiloskop Tetronik dengan time/div 5ms dan resolusi yang didapat adalah pembacaan 2 derajat dengan lebar pulsa 0.20ms. Dan ini berarti terdapat perubahan lebar pulsa pada osiloskop setiap kelipatan 2 derajat dengan perubahan lebar pulsa 0.20ms per dua derajatnya. Tabel. 4.1 di bawah ini merupakan Tabel hasil pengamatan yang telah dilakukan.

Tabel.4.1. Output PWM dan frekuensi dari sensor CMPS03 per sudut.

Sudut Kompas (^{0})	Lebar PWM (ms)	Frekuensi (Hz)	Sudut Kompas (^{0})	Lebar PWM (ms)	Frekuensi (Hz)
0	1.00	1000	49		
1			50	6.00	166.7
2	1.20	833.3	51		
3			52	6.20	161.3
4	1.40	714.3	53		
5			54	6.40	156.3
6	1.60	625	55		
7			56	6.60	151.5
8	1.80	555.6	57		
9			58	6.80	147.1
10	2.00	500	59		
11			60	7.00	142.9
12	2.20	454.5	61		
13			62	7.20	138.9
14	2.40	416.7	63		
15			64	7.40	135.1
16	2.60	384.6	65		
17			66	7.60	131.6
18	2.80	357.1	67		
19			68	7.80	128.2
20	3.00	333.3	69		
21			70	8.00	125
22	3.20	312.5	71		
23			72	8.20	122
24	3.40	294.1	73		
25			74	8.40	119.0
26	3.60	277.8	75		
27			76	8.60	116.3
28	3.80	263.2	77		
29			78	8.80	113.6
30	4.00	250	79		
31			80	9.00	111.1
32	4.20	238.1	81		
33			82	9.20	108.7
34	4.40	227.3	83		
35			84	9.40	106.4
36	4.60	217	85		
37			86	9.60	104.4
38	4.80	208.3	87		
39			88	9.80	102
40	5.00	200	89		
41			90	100	100
42	5.20	192.3	91		
43			92	10.20	98.04
44	5.40	185.2	93		
45			94	10.40	96.15
46	5.60	187.6	95		
47			96	10.60	94.34
48	5.80	172.4	97		

Sudut Kompas (^{0})	Lebar PWM (ms)	Frekuensi (Hz)	Sudut Kompas (^{0})	Lebar PWM (ms)	Frekuensi (Hz)
98	10.80	92.59	147		
99			148	15.80	63.29
100	11.00	90.91	149		
101			150	16.00	62.50
102	11.20	89.29	151		
103			152	16.20	61.73
104	11.40	87.72	153		
105			154	16.40	60.98
106	11.60	86.21	155		
107			156	16.60	60.24
108	11.80	84.75	157		
109			158	16.80	59.52
110	12.00	83.33	159		
111			160	17.00	58.82
112	12.20	81.97	161		
113			162	17.20	58.14
114	12.40	80.65	163		
115			164	17.40	57.47
116	12.60	79.37	165		
117			166	17.60	56.82
118	12.80	78.13	167		
119			168	17.80	56.18
120	13.00	76.92	169		
121			170	18.00	55.58
122	13.20	75.76	171		
123			172	18.20	54.95
124	13.40	74.63	173		
125			174	18.40	54.35
126	13.60	73.53	175		
127			176	18.60	53.76
128	13.80	72.46	177		
129			178	18.80	53.19
130	14.00	71.43	179		
131			180	19.00	52.63
132	14.20	70.42	181		
133			182	19.20	52.08
134	14.40	69.44	183		
135			184	19.40	51.55
136	14.60	68.49	185		
137			186	19.60	51.02
138	14.80	66.67	187		
139			188	19.80	50.51
140	15.00	65.67	189		
141			190	20.00	50.00
142	15.20	64.79	191		
143			192	20.20	49.50
144	15.40	64.94	193		
145			194	20.40	49.02
146	15.60	64.10	195		

Sudut Kompas (^{0})	Lebar PWM (ms)	Frekuensi (Hz)	Sudut Kompas (^{0})	Lebar PWM (ms)	Frekuensi (Hz)
196	20.60	48.54	245		
197			246	25.60	39.06
198	20.80	48.08	247		
199			248	25.80	38.76
200	21.00	47.62	249		
201			250	26.00	38.46
202	21.20	47.17	251		
203			252	26.20	38.17
204	21.40	46.73	253		
205			254	26.40	37.88
206	21.60	46.30	255		
207			256	26.60	37.59
208	21.80	45.87	257		
209			258	26.80	37.31
210	22.00	45.45	259		
211			260	27.00	37.04
212	22.20	45.05	261		
213			262	27.20	36.76
214	22.40	44.64	263		
215			264	27.40	36.50
216	22.60	44.25	265		
217			266	27.60	36.23
218	22.80	43.86	267		
219			268	27.80	35.97
220	23.00	43.48	269		
221			270	28.00	35.71
222	23.20	43.10	271		
223			272	28.20	35.46
224	23.40	42.74	273		
225			274	28.40	35.21
226	23.60	42.37	275		
227			276	28.60	34.97
228	23.80	42.02	277		
229			278	28.80	34.72
230	24.00	41.67	279		
231			280	29.00	34.48
232	24.20	41.32	281		
233			282	29.20	34.25
234	24.40	40.98	283		
235			284	29.40	34.01
236	24.60	40.65	285		
237			286	29.60	34.78
238	24.80	40.32	287		
239			288	29.80	33.56
240	25.00	40	289		
241			290	30.00	33.33
242	25.20	39.68	291		
243			292	30.20	33.11
244	25.40	39.37	293		

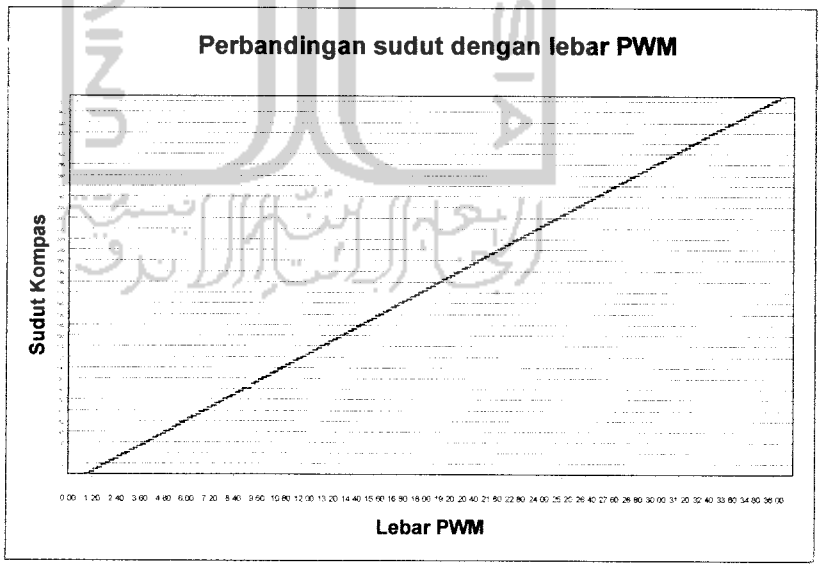
Sudut Kompas (^{0})	Lebar PWM (ms)	Frekuensi (Hz)	Sudut Kompas (^{0})	Lebar PWM (ms)	Frekuensi (Hz)
294	30.40	32.89	328	33.80	29.59
295			329		
296	30.60	32.68	330	34.00	29.41
297			331		
298	30.80	32.47	332	34.20	29.24
299			333		
300	31.00	32.26	334	34.40	29.07
301			335		
302	31.20	32.05	336	34.60	28.90
303			337		
304	31.40	31.85	338	34.80	28.74
305			339		
306	31.60	31.65	340	35.00	28.57
307			341		
308	31.80	31.45	342	35.20	28.41
309			343		
310	32.00	31.25	344	35.40	28.25
311			345		
312	32.20	31.05	346	35.60	28.09
313			347		
314	32.40	30.86	348	35.80	27.93
315			349		
316	32.60	30.67	350	36.00	27.78
317			351		
318	32.80	30.49	352	36.20	27.62
319			353		
320	33.00	30.30	354	36.40	27.47
321			355		
322	33.20	30.12	356	1.00	1000
323			357	1.00	1000
324	33.40	29.94	358	1.00	1000
325			359	1.00	1000
326	33.60	29.76	360	1.00	1000
327					

Dilihat dari hasil pengamatan, perubahan pada tiap-tiap sudut memiliki bentuk lebar pulsa yang berbeda–beda, begitu juga dengan frekuensinya. Jika diperhatikan Pada pengujian osiloskop didapatkan perubahan lebar pulsa tiap satu sudutnya berkisar kira-kira antara 0.10ms, namun hal ini tidak bisa dilihat perubahannya karena resolusi yang didapat oleh osiloskop berkisar 2 derajat dengan lebar pulsa 0.20ms. Bisa dikatakan perubahan lebar pulsa pada tiap sudutnya sangat kecil sekali per 1 derajatnya, sehingga perubahan lebar pulsa

ketika sudut 1 derajat dengan 2 derajat kalau dilihat secara sekilas tidak nampak pada osciloscope yang digunakan.

Frekuensi kerja maksimum pada sensor ini adalah 1KHz dimana jika diaplikasikan kedalam sudut, nilai maksimum ini dimiliki ketika sudut 0^0 dan sudut-sudut mati dengan lebar pulsa ketika ON adalah 1ms. Semakin besar sudut, maka semakin kecil frekuensinya dan PWM semakin bertambah lebar ketika keadaan ON. Untuk lebih jelasnya bisa dilihat Tabel 4.1 dan juga untuk contoh gambar – gambar yang didapat dalam bentuk PWMnya bisa dilihat di lampiran.

Dari hasil pengamatan, pertambahan lebar pulsa per sudutnya tetap, dan bisa dikatakan linear. Pada tabel di atas, terdapat lebar pulsa dan frekuensi yang dikosongkan, hal ini dikarenakan sudutnya terbaca di kompas tetapi perubahan PWM pada osiloskop tidak terbaca (masih tetap), hal ini dikarenakan resolusi yang dimiliki osiloskop yang digunakan.



Gambar 4.1. Grafik pengaruh perubahan sudut terhadap PWM

Grafik di atas menununjukan pengaruh perubahan sudut terhadap lebar pulsa yang di keluarkan sensor. Seperti yang telah dijelaskan diatas tiap 2 derajat diwakili oleh 0.20 ms PWM dalam keadaan ON dan ini berarti jika setiap sudut bertambah kelipatan 2 derajat maka lebar PWM akan bertambah dengan kelipatan 0.20 ms. Dan perubahan ini akan tetap linear sampai sudut maksimum. Untuk lebih jelasnya bisa dilihat Gambar 4.1.

Hasil pembacaan sensor CMPS03 dengan metode PWM ini hanya bisa membaca dari 0^0 sampai 355^0 dan untuk sudut 356^0 sampai 359^0 bisa disebut sebagai sudut mati. Hal ini sudah merupakan karasteristik dari sensor itu sendiri yang apabila pembacaannya dilakukan dengan metode PWM hanya akan didapat pembacaan dari 0^0sampai 355^0. Dari hasil pengujian yang didapat rangkaian sensor sudah berhasil berjalan sesuai rancangan.

4.2.2 Pengujian Rangkaian *Keypad*

Rangkaian *keypad* digunakan untuk memberi masukan data nilai sudut dalam bentuk angka dan data kendali peralatan. Pengujian *keypad* dengan melakukan penekanan tombol mulai dari tombol 0-9, tombol *, dan tombol #. Hasil pengujian ditampilkan pada Tabel 4.2.

Dari Tabel 4.2 dapat dilihat *keypad* dapat berfungsi sesuai dengan yang diharapkan. Hal utama dalam keberhasilan penggunaan *keypad* adalah dalam mengatasi adanya bouncing ketika penekanan tombol *keypad.* Efek yang ditimbulkan oleh *bouncing* yaitu terdeteksinya penekanan tombol dua kali atau lebih. Dengan pemakaian delay setelah penekanan *keypad* dapat menghilangkan adanya efek *bouncing.*

Tabel 4.2. Hasil pengamatan *keypad*

No.	Angka *Keypad*	Hasil Penekanan
1.	1	1
2.	2	2
3.	3	3
4.	4	4
5.	5	5
6.	6	6
7.	7	7
8.	8	8
9.	9	9
10.	*	SETTING ARAH
11.	0	0
12.	#	ENTER

4.2.3 Pengujian Rangkaian Mikrokontroler

Mikrokontroler merupakan pemroses utama dalam perancangan alat ini. Pengujian rangkaian mikrokontroler dilakukan dengan cara memberikan logika *high* dan *low* pada keempat port masukan dan keluarannya, kemudian membaca kondisi logika pada masing masing port.

Dari hasil pengujian, pada semua port mempunyai logika sesuai dengan yang diprogram, sehingga rangkaian mikrokontroler ini dapat digunakan sebagai pemroses utama.

4.2.3.1 Rangkaian *Clock*

Rangkaian *clock* berfungsi memberikan pewaktuan kepada CPU. Saat catu daya dihidupkan maka rangkaian *clock* otomatis bekerja. Rangkaian ini menggunakan kristal 12 MHz, sehingga setiap *cycle*:

$$1\ cycle = \frac{1}{cystal} x12$$

$$1\ cycle = \frac{1}{12MHz} X12$$

$$1\ cycle = 1\ microsecond$$

4.2.3.2 Rangkaian Reset

Rangkaian reset berfungsi menghentikan kerja CPU dan kemudian mengulang dari awal (program *counter* ke alamat 0000). Saat catu daya dihidupkan rangkaian reset menunda kerja dari CPU hingga tegangan stabil (*power on reset*). Reset pada mikrokontroler AT89S52 adalah aktif tinggi.

Pengujian dilakukan dengan menghidupkan mikrokontroler mengamati kerja mikrokontroler. Dari hasil pengamatan dapat disimpulkan reset mikrokontroler dapat berfungsi dengan baik, *power on reset* dapat berfungsi dengan baik, dengan tanda mikrokontroler dapat langsung bekerja ketika power dihidupkan. Begitu juga ketika dilakukan reset pada mikrokontroler melalui tombol maka mikrokontroler dapat melakukan reset.

4.2.4 Pengujian Rangkaian LCD

Rangkaian LCD digunakan untuk menampilkan data kompas digital ketika motor berputar maupun diam dan juga untuk menampilkan nilai setingan arah yang diinginkan. Pengujian dilakukan dengan menampilkan data pada LCD, dan

hasilnya dilihat langsung. Dari hasil pengamatan, LCD bisa menampilkan karakter dengan baik. Jumlah karakter yang bisa ditampilkan sebanyak 16 x 2. Kontras LCD dapat diatur melalui potensiometer, pengaturan dengan memberikan tegangan pada kaki VEE. Rangkaian yang digunakan menggunakan rangkaian pembagi tegangan.

4.2.5 Pengujian Rangkaian Driver PWM dan Pengendali Motor

Pengujian pada rangkain driver dilakukan dengan memberikan sinyal input PWM yang berasal dari mikro kemudian dilihat perubahan sinyalnya. Output dari rangkaian driver ini digunakan untuk menjalankan motor. Dan dalam perancangan ini, digunakan sebuah motor DC power window dengan tegangan kerja 12 Vdc.



Gambar 4.2. Output PWM dari mikro

Terjadi perubahan PWM pada keluaran rangkaian driver yaitu terjadi pada tegangan PWM sedangkan untuk lebar pulsa dan frekuensi tetap sama. Hal ini dikarenakan pada rangkaian driver bekerja pada tegangan 12Vdc. Sehingga sebelumnya tegangan PWM (dari mikro) 5Vdc dan belum kuat untuk menggerakan motor DC, maka PWM dari driver ini dengan tegangan 12Vdc sudah bisa untuk menggerakan motor DC.



Gambar 4.3. Output PWM dari driver

Pada pengujian ini motor bergerak sesuai dengan pulsa yang diinginkan dan rangkain berhasil sesuai dengan rancangan.

4.3. Pengujian Kinerja Sistem dan Pembahasan Sistem Keseluruhan

Untuk memulai menjalankan alat ini yaitu dengan menekan tombol power. Tampilan pada baris pertama LCD sebelum menggunakan sensor kompas akan tampil kata "*KOMPAS DIGITAL*" berikut baris kedua dengan tampilan kata "Sudut = " bisa dilihat pada Gambar 4.4. Sedangkan tampilan LCD pada baris pertama setelah diberi masukan dari sensor kompas akan tampil arah seperti " TIMUR ". Dan pada baris ke dua LCD akan tampil "Sudut = 90^0 " Nilai 90^0 yang tertera merupakan nilai sudut yang terukur pada sensor pada posisi saat itu untuk lebih jelasnya bisa dilihat pada Gambar 4.5.



Gambar.4.4. Tampilan LCD sebelum memakai sensor.

Dan kata "TIMUR" adalah arah untuk posisi sudut 90^0. Nama arah Dan nilai inilah yang akan berubah - ubah sesuai dengan pembacaan sensor dengan kata lain sensor berpindah posisi (motor bergerak) sesuai yang diinginkan nantinya. Adapun ketetapan nama arah dengan sudut yang telah diberikan pada sistem ini dapat dilihat pada Tabel 4.3 berikut ini:

Tabel 4.3 Ketentuan nilai sudut dengan arah

ARAH	SUDUT
UTARA	316^0 s/d 45^0
TIMUR	46^0 s/d 135^0
SELATAN	136^0 s/d 225^0
BARAT	226^0 s/d 315^0



Gambar.4.5. Tampilan LCD kompas digital

Pada tampilan LCD di atas, sistem berfungsi sebagai kompas digital dan belum bisa menggerakan motor. Untuk menggerakan motor, perlu adanya perintah penyetingan arah terlebih dahulu dengan cara memberikan nilai sudut yang di inginkan pada *keypad* terlebih dahulu dengan menekan *keypad.* Berikut ini cara atau tahapan untuk menyeting arah gerak motor pada yang diinginkan pada sistem :

1. Tekan tanda * (bintang) sampai ada tampilan SETTING ARAH pada baris pertama tampilan LCD.



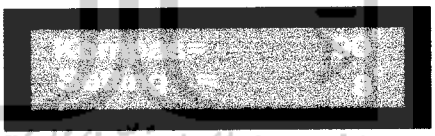
Gambar 4.6. Tampilan setelah menekan tanda bintang

2. kemudian masukan nilai sudut yang diinginkan (maksimal 355^0). Contoh : arah yang dituju 8^0 maka tekan angka 8 maka akan tampil pada Gambar 4.7 sbb:



Gambar 4.7. Tampilan ketika memasukan nilai

3. tekan tanda # (pagar) pagar sebagai enter.lihat Gambar!

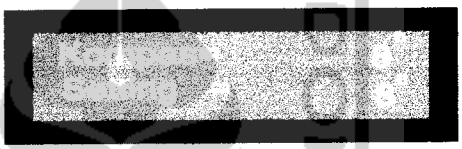


Gambar 4.8. Tampilan ketika setelah menekan enter

Setelah kita menyeting sudut dan menekan enter (#) pada *keypad* maka sistem akan membaca setingan lalu sistem akan mengirimkan data PWM dengan waktu tertentu sesuai setingan ke driver, lalu driver akan menggerakan motor sesuai dengan setingan tersebut sampai menemukan sudut yang sama dengan setingan. Disamping motor bergerak kearah yang dituju, sensor kompas akan tetap

menampilan perubahan sudut yang dia baca ketika sensor itu bergerak sejalan dengan motor sehingga kita dapat mengetahui sudut–sudut yang dilewati per satu derajatnya. Lebih jelasnya bisa dilihat Gambar 4.8. Nilai sudut 90^0 pada baris pertama LCD merupakan pembacaan kompas digital ketika bergerak dan nilai ini berubah ubah sesuai pembacaan arah sensor yang dilewati.

Pergerakan motor akan berhenti sampai sensor kompas membaca sudut sesuai dengan nilai setingan yang diinginkan.lalu sistem akan menampilkan hasil pergerakan akhir motor dengan jumlah sudut sesuai dengan pembacaan kompas digital lihat Gambar 4.9.



Gambar 4.9. Tampilan ketika sesaat motor berhenti



Gambar 4.10. Tampilan akhir

Sensor kompas sendiri di sistem ini berfungsi sebagai *feedback* sedangkan setingan sudut yang di inputkan berfungsi sebagai set point. Sehingga motor akan terus bergerak sampai mendapatkan hasil yang sama dengan set point.

4.3.1 Pengkalibrasian dan pengujian

Selanjutnya dilakukan pengkalibrasian yaitu antara kompas digital yang dibuat dengan kompas analog yang telah memenuhi standar. Setelah kompas digital terkalibrasi. Pengujian kompas digital dengan kompas analog harus dilakukan untuk mengetahui seberapa akurat dan presisi kompas digital yang dibuat. Berikut ini adalah tabel hasil dari pengujian kompas digital dengan cara membandingkan antara pembacaan dengan menggunakan kompas digital dan dengan menggunakan kompas analog. Pengukuran dibawah ini dilakukan dalam satu kalibrasi

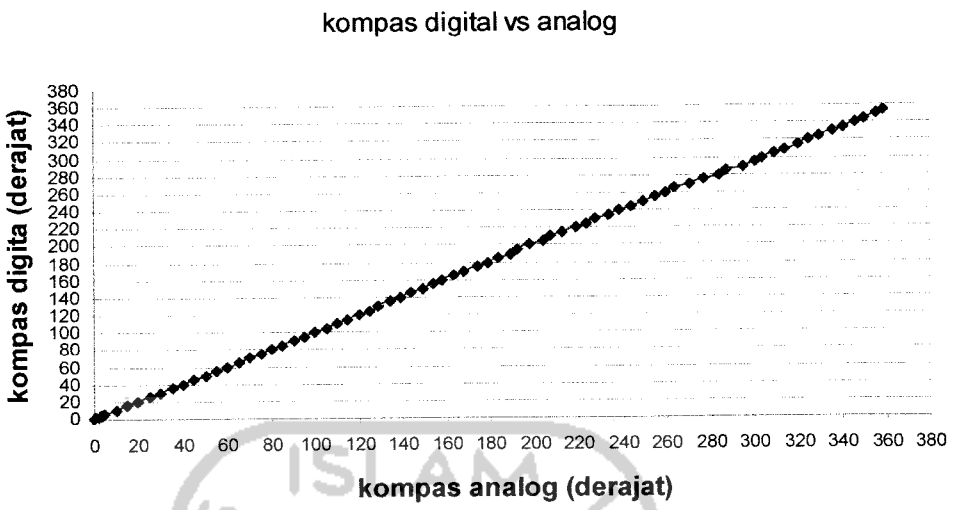
Tabel 4.4. Perbandingan nilai sudut kompas digital dengan analog

SUDUT TERBACA		
Kompas Digital (derajat)	**Kompas Analog (derajat)**	**Tingkat error (%)**
0	0	0.0
1	1	0.0
2	2	0.0
3	3	0.0
4	4	0.0
5	5	0.0
10	10	0.0
15	15	0.0
20	20	0.0
25	25	0.0
30	30	0.0
35	35	0.0
40	40	0.0
45	45	0.0
50	50	0.0
55	55	0.0
60	60	0.0
65	65	0.0
70	70	0.0
75	75	0.0
80	80	0.0

SUDUT TERBACA		
Kompas Digital (derajat)	Kompas Digital (derajat)	Kompas Digital (derajat)
85	85	0.0
90	90	0.0
95	95	0.0
100	100	0.0
105	105	0.0
110	110	0.0
115	115	0.0
120	120	0.0
125	125	0.0
130	129	0.3
135	134	0.3
140	139	0.3
145	144	0.3
150	149	0.3
155	154	0.3
160	158	0.6
165	163	0.6
170	168	0.6
175	174	0.3
180	180	0.0
185	184	0.3
190	189	0.3
195	192	0.8
200	198	0.6
205	204	0.3
210	207	0.8
215	213	0.6
220	219	0.3
225	224	0.3
230	228	0.6
235	234	0.3
240	239	0.3
245	244	0.3
250	250	0.0
255	255	0.0
260	260	0.0
265	264	0.3
270	270	0.0

SUDUT TERBACA		
Kompas Digital (derajat)	Kompas Digital (derajat)	Kompas Digital (derajat)
275	277	0.6
280	284	1.1
285	287	0.6
290	295	1.4
295	301	1.7
300	304	1.1
305	309	1.1
310	314	1.1
315	320	1.4
320	325	1.4
325	330	1.4
330	336	1.7
335	341	1.7
340	346	1.7
345	350	1.4
350	356	1.7
355	359	1.1

Dari Tabel di atas bisa diamati bahwa untuk sudut 356^0-359^0 pada kompas digital tidak didapat, akan tetapi pada kompas analog ada. Terjadi ketidaksamaan hasil pembacaan antara kompas analog dengan kompas digital yaitu dimulai ketika kompas digital membaca nilai sudut 130^0 sedangkan kompas analog membaca 129^0. terjadi ketidaksamaan pembacaan antara kompas analog dengan kompas digital dengan tingkat error yang variatif. Tingkat error maksimal pada sistem kompas digital ini adalah 0.8% yang berarti terdapat selisih pembacaan antara kompas analog dengan kompas digital sekitar 6^0 dan minimal 0.3% yang berarti 1^0. Berikut adalah Gambar grafik perbandingan antara kompas analog dengan digital.



Gambar 4.11. Grafik kompas digital vs kompas analog.

Dari hasil pengamatan, Bisa dilihat bahwa terjadi perubahan kelinearitasan nilai sudut dari mulai nilai 130 sampai 355. Untuk nilai sudut 360^0 sama nilainya dengan sudut 0^0. Lebih jelasnya bisa dilihat Gambar 4.11 di atas. Selain itu juga pengaruh medan magnet disekeliling sensor sangat mempengaruhi pembacaan sensor. Hal ini akan mengakibatkan pembacaan sensor kacau dan nilai hasil pembacaan akan berubah. Nilai perubahannya tergantung dari kuat tidaknya medan magnet yang didekatkan. Disamping itu juga jarak medan magnet sangat mempengaruhi perubahan nilai yang terjadi. Berikut ini adalah tabel hasil pengujian sensor sebelum dan sesudah didekatkan terhadap medan magnet. Medan magnet yang didekatkan didapat dari manet yang berasal dari kompas analog yang dipakai untuk kalibrasi. Untuk lebih jelasnya bisa dilihat pada Tabel 4.5 di bawah ini.

Tabel 4.5. Pengaruh medan magnet terhadap sensor kompas

PENGARUH MEDAN MAGNET TERHADAP SENSOR		
Pembacaan awal Kompas Digital (derajat)	**Pembacaan setelah didekatkan medan magnet (derajat)**	**Tingkat error (%)**
0	35	9.7
1	12	3.1
2	39	10.3
3	25	6.1
4	15	3.1
5	26	5.8
10	30	5.6
15	24	2.5
20	33	3.6
25	45	5.6
30	42	3.3
35	54	5.3
40	55	4.2
45	54	2.5
50	55	1.4
55	68	3.6
60	85	6.9
65	80	4.2
70	90	5.6
75	94	5.3
80	90	2.8
85	90	1.4
90	125	9.7
95	100	1.4
100	110	2.8
110	98	3.3
120	287	46.4
130	148	5.0
140	127	3.6
200	87	31.4
250	270	5.6
270	258	3.3
300	280	5.6
350	65	79.2
355	25	91.7

4.3.2 Pengujian Set point dengan Aktuasi Motor

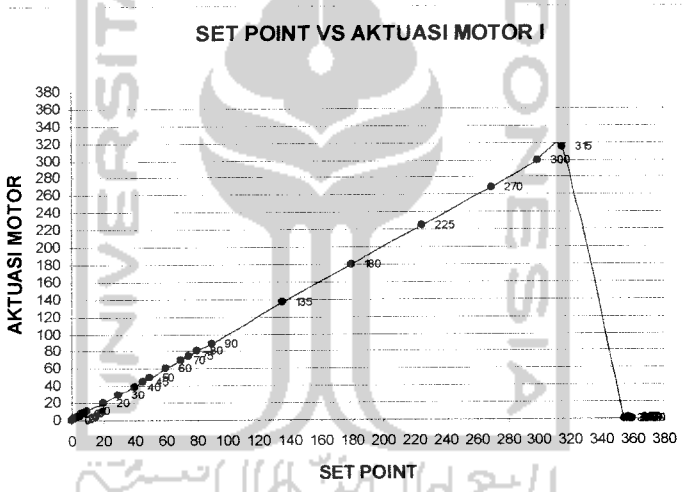
Pengujian dilakukan tiga kali untuk set point yang sama, lalu hasilnya dibandingkan dengan aktuasi motor. Pengujian ini dilakukan untuk mengetahui

ketepatan antara motor yang dikendalikan dengan set point, sehingga kita bisa mengetahui bahwa motor bergerak tepat sesuai dengan sudut yang diinputkan (set point) atau tidak dan juga bisa mengetahui tingkat errornya. Berikut ini adalah Tabel hasil dari pengujian.

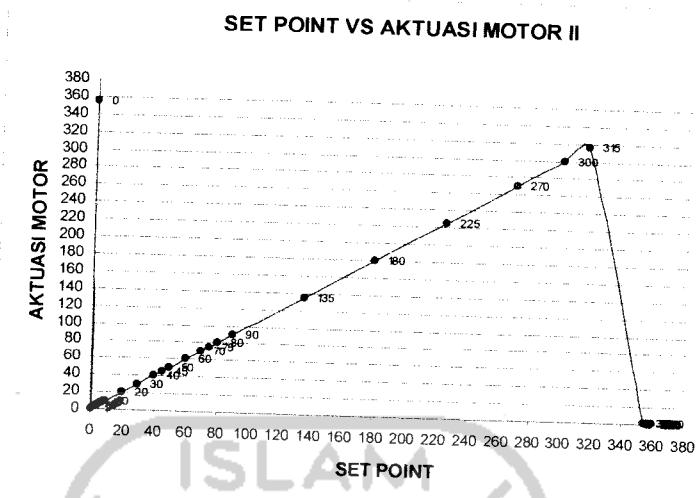
Tabel 4.6 Set point vs aktuasi motor

Set Point			Aktuasi Motor		
I	II	III	I	II	III
0	0	0	0	355	0
1	1	1	1	0	1
2	2	2	2	2	3
3	3	3	3	3	3
4	4	4	4	4	4
5	5	5	5	5	5
6	6	6	7	6	6
7	7	7	7	7	7
8	8	8	8	8	8
9	9	9	9	9	9
10	10	10	11	10	10
20	20	20	20	20	20
30	30	30	30	30	30
40	40	40	39	40	40
45	45	45	45	45	45
50	50	50	50	50	50
60	60	60	60	60	60
70	70	70	70	70	70
75	75	75	75	75	75
80	80	80	80	80	80
90	90	90	89	90	90
135	135	135	136	135	135
180	180	180	180	180	180
225	225	225	225	225	225
270	270	270	269	270	270
300	300	300	300	300	300
315	315	315	315	316	315
355	355	355	0	1	0
356	356	356	0	1	0
357	357	357	1	0	0
358	358	358	0	1	1
359	359	359	0	0	0
360	360	360	0	1	0

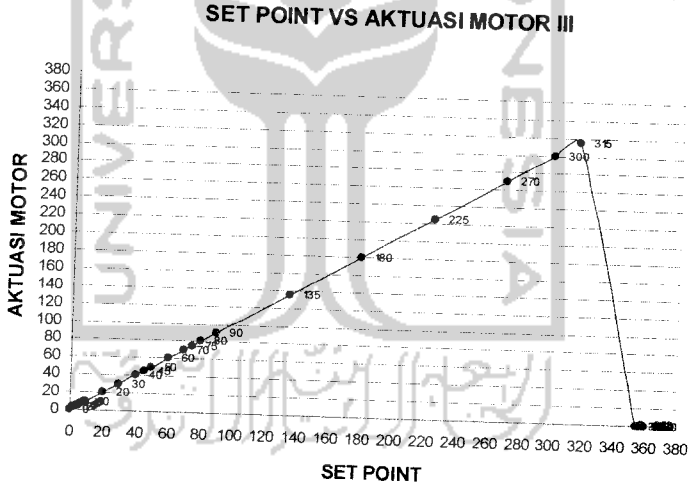
Dari hasil pengujian diatas, bisa kita lihat pada ketiga hasil pengujian yang didapat, hasilnya ada yang akurat dan ada yang tidak dan ketidak akuratanya atau tingkat error yang didapat hanya 1^0, misalnya kita lihat pada pengujian pertama ketika set point di set 0^0, maka yang terbaca adalah kadang 0^0dan kadang 1^0 atau bahkan kadang 355^0 tetapi kelebihan atau kekurangan tingkat errornya tidak melebihi atau kurang dari 1^0. Hal ini bisa saja terjadi karena diakibatkan adanya slip pada gear–gear motor atau bisa juga karena adanya getaran–getaran yang mempengaruhi sensornya sendiri mengingat sensor CMPSO3 sangat sensitiv terhadap perubahan kedudukan.



Gambar 4.12 Grafik aktuasi motor vs set point 1



Gambar 4.13 Grafik aktuasi motor vs set point II



Gambar 4.14 Grafik aktuasi motor vs set point III

Boleh dikatakan aktuasi motor pada set point 99% akurat atau sama. Untuk lebih jelasnya bisa dilihat Gambar-Gambar grafik di atas. Pemberian

perintah pada set point dengan nilai sudut mati (356^0 – 359^0) , sistem akan otomatis menghasilkan sudut 0 derajat hal ini berarti aktuasi motor akan berhenti pada sudut 0 derajat. Begitu juga dengan perintah dengan nilai sudut 360^0, sistem akan menghasilkan sudut 0^0 hal ini karena posisi nilai sudut 360^0 sama dengan posisi ketika 0^0 sehingga nilai 360^0 sama dengan nilai 0^0. Dari gambar – gambar grafik diatas bisa dilihat bahwa kelinearitasan antar keduanya masih bisa terjaga.

4.3.3 Pengujian Kompas Digital , PWM Dengan Perhitungan Rumus

Pada pengujian ini dilakukan untuk membandingkan antara nilai sudut dengan nilai PWM dan juga perhitungan nilai PWM yang digunakan didalam program untuk mencari nilai sudut kompas yang akan ditampilkan. Kemudian hasil perhitungan dibandingkan dengan sudut kompas yang terbaca. Adapun rumus perhitungan sudut yang digunakan adalah sebagai berikut :

$$Sudut = \frac{(PWM - 1000)}{100}$$

Nilai PWM yang terdapat didalam rumus dalam *mikro second* (*us*) sehingga nilai PWM yang didapat harus dirubah kedalam *us*. Untuk Tabel hasil pengamatan PWM dan juga hasil perhitungannya kedalam sudut bisa dilihat di lampiran.

Dari Tabel di lampiran, bisa diamati bahwa nilai sudut hasil perhitungan PWM yang didapat dengan nilai sudut yang terbaca pada kompas adalah sama dan ini membuktikan hasil yang didapat sesuai dengan rancangan.

BAB V

PENUTUP

5.1 Kesimpulan

Berdasarkan perancangan, pembuatan, pengujian alat, dan pembahasan, maka dapat diambil beberapa kesimpulan sebagai berikut :

1. Rangkaian aplikasi mikrokontroler AT89S52 untuk Kompas Digital Sebagai Arah Sudut Robot Tank terdiri dari bagian *input* yaitu sensor kompas CMPSO3 dan keypad, bagian pengolah berupa mikrokontroler AT89S52 dan bagian *output* berupa LCD dan motor DC.
2. Pembacaan yang dihasilkan oleh sensor CMPSO3 ini dengan menggunakan metode PWM hanya bisa membaca dari 0^0 sampai 355^0 sehingga untuk sudut 356^0 dan 359^0 bisa dikatakan sudut mati dalam kompas ini. Hal ini sudah merupakan karakteristik dari sensor tersebut yang apabila dilakukan pembacaan melalui metode PWM.
3. Dampak dari pemberian perintah sudut-sudut mati (356^0-359^0), secara otomatis akan menghasilkan sudut 0^0 pada sistem dan ini berarti robot tank akan berhenti pada titik sudut 0^0.
4. Sensitivitas dari sensor ini sangat tinggi terhadap medan magnet sehingga jika didekatkan dengan benda yang memiliki medan magnet akan mempengaruhi pembacaan sensor (error).

5.2 Saran-saran

Penelitian ini dapat dikembangkan lagi untuk mencapai hasil yang lebih baik. Beberapa pengembangan yang dimungkinkan adalah:

1. *Software / hardware* dapat dikembangkan lagi untuk kendali jarak jauh
2. Pembacaan sensor CMPSO3 dengan menggunakan metode I^2C yang lebih memungkinkan mendapatkan pembacaan sudut dari 0 derajat hingga 359 derajat.
3. Pemasangan sensor sebaiknya didesain dengan bahan yang kokoh sehingga getaran tidak mempengaruhinya dan juga hindarkan sensor dari motor DC sejauh mungkin hal ini dikarenakan magnet yang dihasilkan oleh motor akan mempengaruhi sensor.
4. Untuk hasil yang lebih, dibuat sistem jarak jauh dan interfacing ke komputer dengan menggunakan delfi ataupun lainnya sebagai tampilan, sehingga kita dapat mengendalikan pergerakan robot tank dari jarak jauh dengan PC.
5. Dibuat pengembangannya untuk gerak secara elevasi.

DAFTAR PUSTAKA

Achyanto, Djoko, 1997. ***Mesin-mesin Listrik Edisi ke-Empat***, Penerbit Erlangga, Jakarta.

Anonim, 1997, ***"Atmel Literature Requests"***, web site www.atmel.com/literature

Budioko, Totok, 2005. ***Belajar Dengan Mudah dan Cepat Pemrograman Bahasa C Dengan SDCC (Small Device C Compiler) Pada Mikrokontroler AT89X051/AT89C51/52 Teori, Simulasi dan Aplikasi***, Penerbit Gava Media ,Yogyakarta.

Instalasi dan Mesin Listrik, Laboratorium, 2004. ***Modul praktikum Teknik Tenaga Listrik***, Penerbit Jurusan Teknik Elektro Fakultas Teknologi Industri Universitas Islam Indonesia, Yogyakarta.

Steeman, JPM, 1993, *"**Data Sheet Book** 2"*, Elex Media Computindo, Jakarta

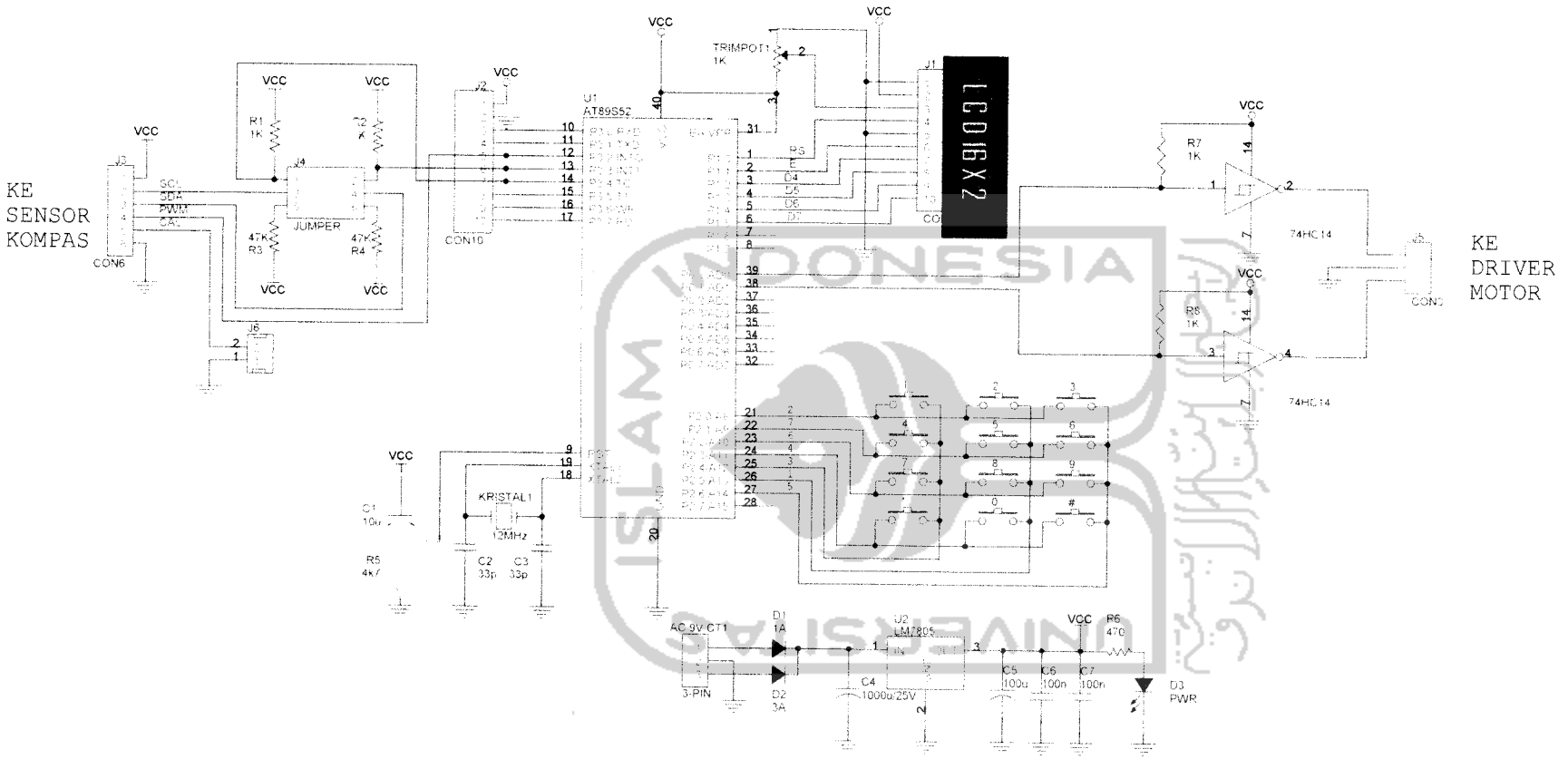
Widodo Budiharto, S.Si, M.Kom. ***Interfacing Komputer Dan Mikrokontroler***, Penerbit Media Elek Komputindo, Jakarta.

Widodo Budiharto. ***Membuat Robot Cerdas,*** Penerbit Media Elek Komputindo, Jakarta.

L.A.M.P.I.R.A.N.

- Layout sistem
- Data set kompas
- Gambar output PWM
- Tabel pengujian pwm dengan rumus
- Listing program
- Gambar robot

KOMPAS DIGITAL

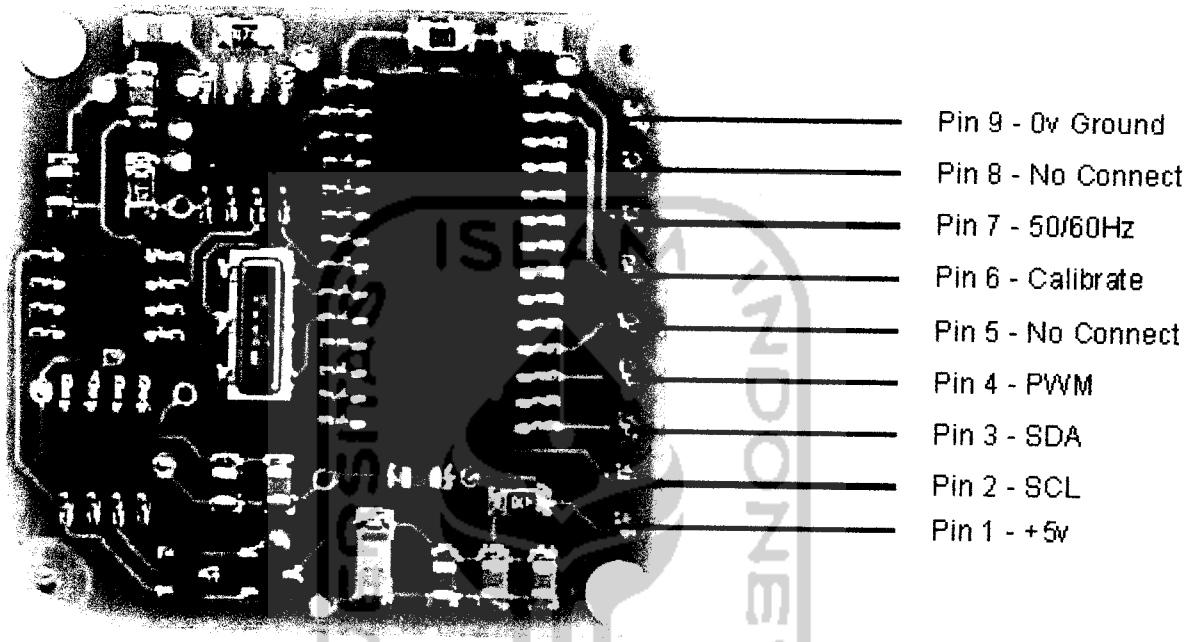


Title	Yudi Heryanto	
Size B	Document Number KOMPAS DIGITAL	Rev
Date: Monday, May 14, 2007	Sheet 1 of 1	

CMPS03 - Robot Compass Module

This compass module has been specifically designed for use in robots as an aid to navigation. The aim was to produce a unique number to represent the direction the robot is facing. The compass uses the Philips KMZ51 magnetic field sensor, which is sensitive enough to detect the Earths magnetic field. The output from two of them mounted at right angles to each other is used to compute the direction of the horizontal component of the Earths magnetic field. We have examples of using the Compass module with a wide range of popular controllers.

Connections to the compass module



The compass module requires a 5v power supply at a nominal 15mA.

There are two ways of getting the bearing from the module. A PWM signal is available on pin 4, or an I2C interface is provided on pins 2,3.

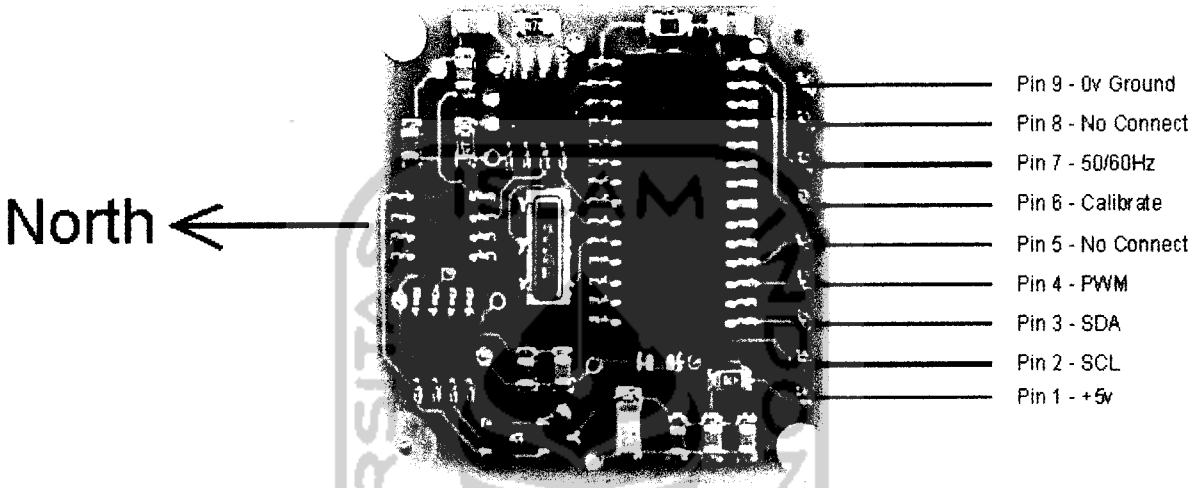
The PWM signal is a pulse width modulated signal with the positive width of the pulse representing the angle. The pulse width varies from 1mS (0°) to 36.99mS (359.9°) - in other words 100uS/° with a +1mS offset. The signal goes low for 65mS between pulses, so the cycle time is 65mS + the pulse width - ie. 66ms-102ms. The pulse is generated by a 16 bit timer in the processor giving a 1uS resolution, however I would not recommend measuring this to anything better than 0.1° (10uS). Make sure you connect the I2C pins, SCL and SDA, to the 5v supply if you are using the PWM, as there are no pull-up resistors on these pins.

Pin 2,3 are an I2C interface and can be used to get a direct readout of the bearing. If the I2C interface is not used then these pins should be pulled high (to +5v) via a couple of resistors. Around 47k is ok, the values are not at all critical.

Calibrating the CMPS01, CMPS03 Robot Compass Modules

CMPS03 - Calibration procedure is the same as CMPS01 Rev 7.
CMPS01 - As from Software revision 7, the calibration procedure has changed.
Both methods are detailed below

Calibration only needs to be done once - the calibration data is stored in EEPROM on the PIC16F872 chip. You do not need to re-calibrate every time the module is powered up. The module has already been calibrated in our workshop for our inclination, which is 67 degrees. If your location is close to this, you may like to try the compass without re-calibrating at all.



Compass module orientation to produce 0 degrees reading.

Register	Function
0	Software Revision Number
1	Compass Bearing as a byte, i.e. 0-255 for a full circle
2,3	Compass Bearing as a word, i.e. 0-3599 for a full circle, representing 0-359.9 degrees.
4,5	Internal Test - Sensor1 difference signal - 16 bit signed word
6,7	Internal Test - Sensor2 difference signal - 16 bit signed word
8,9	Internal Test - Calibration value 1 - 16 bit signed word
10,11	Internal Test - Calibration value 2 - 16 bit signed word
12	Unused - Read as Zero
13	Unused - Read as Zero
14	Calibration Done Flag - Zero in calibrate mode when un-calibrated, 255 otherwise - unused in Rev 7 software & CMPS03
15	Calibrate Command - Write 255 to enter calibrate mode, write zero to exit. See text.

Register 0 is the Software revision number (originally 3, now 7 with new calibration routines and 8 for the CMPS03).

Registers 14 & 15 are used to calibrate the compass. The procedure changed with Rev 7 software. Full calibration information is here

IMPORTANT - The compass module must be kept flat (horizontal and parallel to the earths surface) with the components on top and for the CMPS01 the sensors underneath. Keep the module away from metallic - especially magnetic - objects.

Calibrating Rev 3 Software - Recognized by lack of revision number on CPU chip, or read revision number from register 0

I2C Method

To calibrate the compass using the I2C bus, you only have to write 255 to

register 15 and rotate the module very slowly through 360° . Writing zero to register 15 will store the calibration values in the processors internal EEPROM. Readings are taken by the processor at four compass points and these values are used to generate the calibration values. Register 14 reads 255 during normal operation. It reads zero when Calibrate mode is entered and 255 again when the four compass points have been measured. Register 14 will therefore indicate that the four points have been acquired and that zero can be written to register 15 to store the calibration and return to normal operation. It is necessary to rotate the compass very slowly during calibration to avoid missing the required compass points and to keep it horizontal to ensure the calibration figures are accurate.

Pin Method

Pins 5,6 are used to calibrate the compass. The calibrate input (pin 6) has an on-board pull-up resistor and can be left unconnected after calibration. To calibrate the compass you only have to take the calibrate pin low and rotate the module very slowly through 360° . Taking the calibrate pin high will store the calibration values in the processors internal EEPROM. Readings are taken by the processor at four compass points and these values are used to generate the calibration values. The CalDone output pin (pin 5) is high during normal operation. It goes low when the Calibrate pin is pulled low and high again when the four compass points have been measured. The CalDone pin will therefore indicate that the four points have been acquired and that the Calibrate pin can be raised high again. It is necessary to rotate the compass slowly during calibration to avoid missing the required compass points and to keep it horizontal to ensure the calibration figures are accurate.

Calibrating Rev 7 Software - Recognized by revision number label on CMPS01 CPU chip, or read revision number from register 0.
Also applies to **Calibrating the CMPS03 Module**.

Note that pin 5 (CalDone) and register 14 (Calibration Done Flag) are not used with Rev 7 software or the CMPS03. Pin 5 should be left unconnected and register 14 ignored. When calibrating the compass, you must know **exactly** which direction is North, East, South and West. Don't guess at it. Get a magnetic needle compass and check it.

I2C Method

To calibrate using the I2C bus, you only have to write 255 (0xff) to register 15 for each of the four major compass points North, East, South and West. The 255 is cleared internally automatically after each point is calibrated. The compass points can be set in any order, but all four points must be calibrated. For example

1. Set the compass module flat, pointing North. Write 255 to register 15
2. Set the compass module flat, pointing East. Write 255 to register 15
3. Set the compass module flat, pointing South. Write 255 to register 15
4. Set the compass module flat, pointing West. Write 255 to register 15

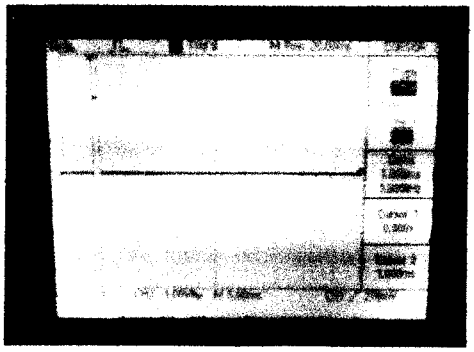
That's it.

Pin Method

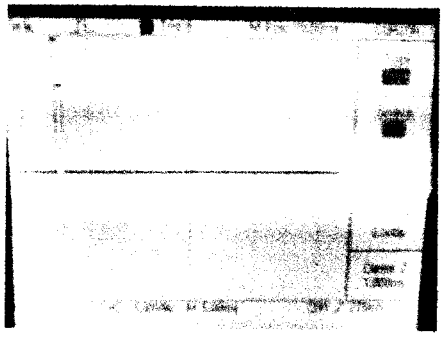
Pin 6 is used to calibrate the compass. The calibrate input (pin 6) has an on-board pull-up resistor and can be left unconnected after calibration. To calibrate the compass you only have to take the calibrate pin low and then high again for each of the four major compass points North, East, South and West. A simple push switch wired from pin6 to 0v (Ground) is OK for this. The compass points can be set in any order, but all four points must be calibrated. For example

1. Set the compass module flat, pointing North. Press and release the switch
2. Set the compass module flat, pointing East. Press and release the switch
3. Set the compass module flat, pointing South. Press and release the switch
4. Set the compass module flat, pointing West. Press and release the switch

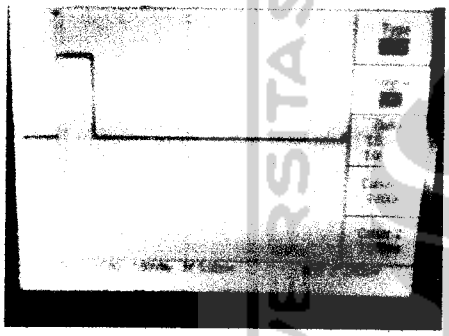
That's it.



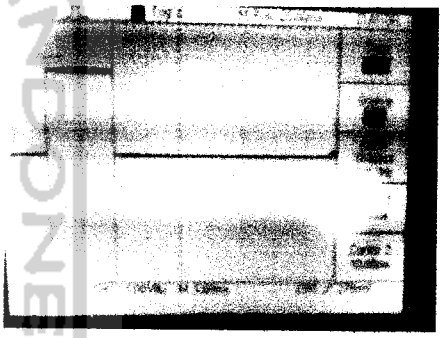
Gambar PWM ketika sudut 0^0



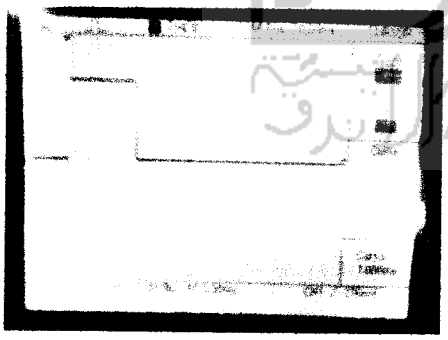
Gambar PWM ketika sudut 1^0



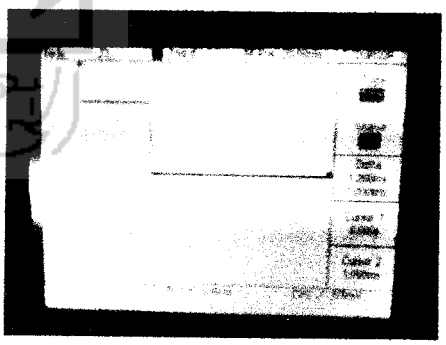
Gambar PWM ketika sudut 46^0



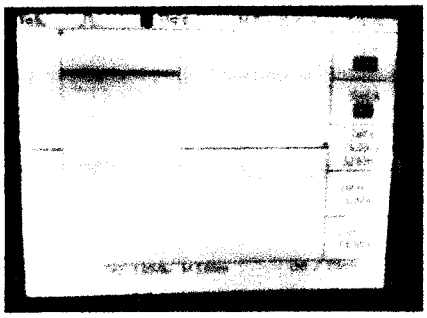
Gambar PWM ketika sudut 90^0



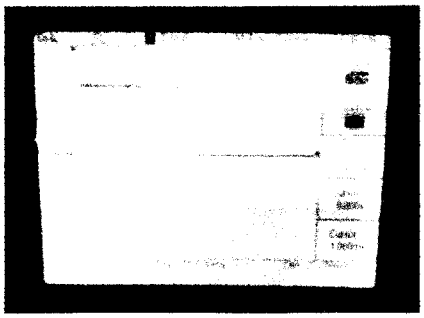
Gambar PWM ketika sudut 100^0



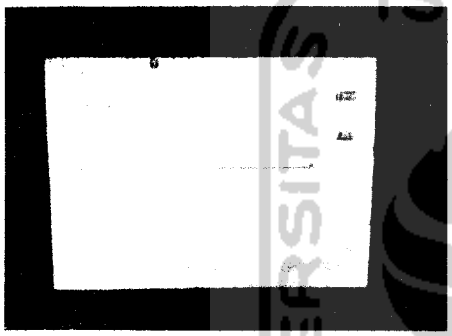
Gambar PWM ketika sudut 116^0



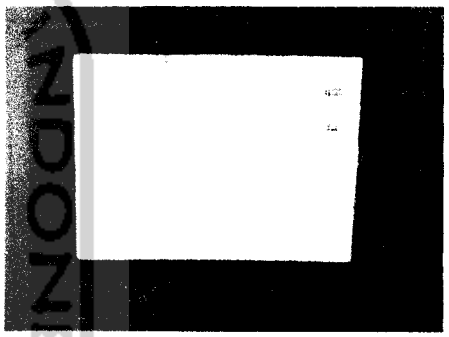
Gambar PWM ketika sudut 180^0



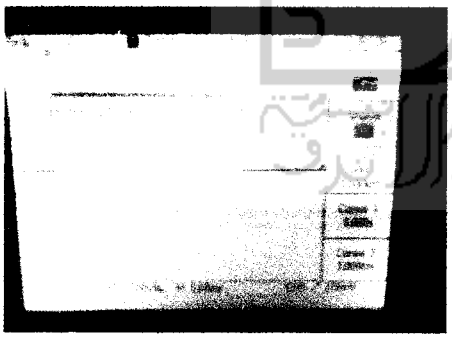
Gambar PWM ketika sudut 200^0



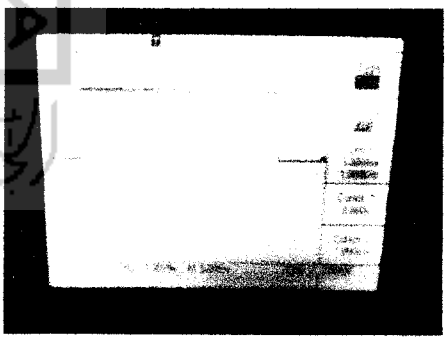
Gambar PWM ketika sudut 250^0



Gambar PWM ketika sudut 270^0



Gambar PWM ketika sudut 300^0



Gambar PWM ketika sudut 350^0

Tabel. Perbandingan antara sudut kompas,lebar pulsa dengan perhitungan.

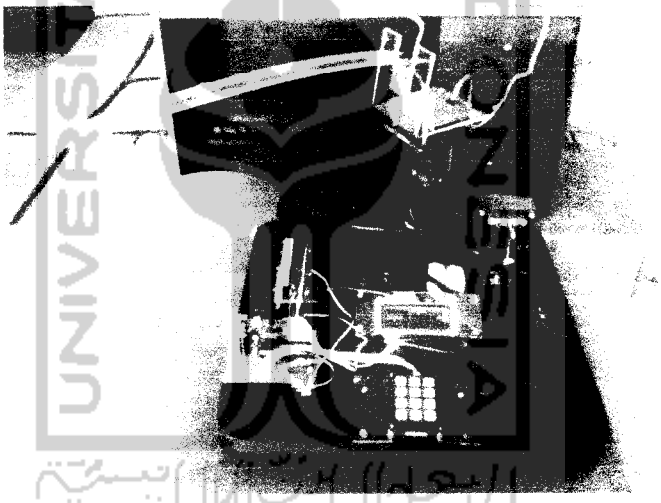
Sudut Kompas (^{0})	Lebar PWM (ms)	Rumus (PWM(us)-1000)/100	Sudut Kompas (^{0})	Lebar PWM (ms)	Rumus (PWM us)-1000)/100
0	1.00	0	49		
1			50	6.00	50
2	1.20	2	51		
3			52	6.20	52
4	1.40	4	53		
5			54	6.40	54
6	1.60	6	55		
7			56	6.60	56
8	1.80	8	57		
9			58	6.80	58
10	2.00	10	59		
11			60	7.00	60
12	2.20	12	61		
13			62	7.20	62
14	2.40	14	63		
15			64	7.40	64
16	2.60	16	65		
17			66	7.60	66
18	2.80	18	67		
19			68	7.80	68
20	3.00	20	69		
21			70	8.00	70
22	3.20	22	71		
23			72	8.20	72
24	3.40	24	73		
25			74	8.40	74
26	3.60	26	75		
27			76	8.60	76
28	3.80	28	77		
29			78	8.80	78
30	4.00	30	79		
31			80	9.00	80
32	4.20	32	81		
33			82	9.20	82
34	4.40	34	83		
35			84	9.40	84
36	4.60	36	85		
37			86	9.60	86
38	4.80	38	87		
39			88	9.80	88
40	5.00	40	89		
41			90	100	90
42	5.20	42	91		
43			92	10.20	92
44	5.40	44	93		
45			94	10.40	94
46	5.60	46	95		
47			96	10.60	96
48	5.80	48	97		

Sudut Kompas (^{0})	Lebar PWM (ms)	Rumus (PWM us)-1000)/100	Sudut Kompas (^{0})	Lebar PWM (ms)	Rumus (PWM us)-1000)/100
98	10.80	98	147		
99			148	15.80	148
100	11.00	100	149		
101			150	16.00	150
102	11.20	102	151		
103			152	16.20	152
104	11.40	104	153		
105			154	16.40	154
106	11.60	106	155		
107			156	16.60	156
108	11.80	108	157		
109			158	16.80	158
110	12.00	110	159		
111			160	17.00	160
112	12.20	112	161		
113			162	17.20	162
114	12.40	114	163		
115			164	17.40	164
116	12.60	116	165		
117			166	17.60	166
118	12.80	118	167		
119			168	17.80	168
120	13.00	120	169		
121			170	18.00	170
122	13.20	122	171		
123			172	18.20	172
124	13.40	124	173		
125			174	18.40	174
126	13.60	126	175		
127			176	18.60	176
128	13.80	128	177		
129			178	18.80	178
130	14.00	130	179		
131			180	19.00	180
132	14.20	132	181		
133			182	19.20	182
134	14.40	134	183		
135			184	19.40	184
136	14.60	136	185		
137			186	19.60	186
138	14.80	138	187		
139			188	19.80	188
140	15.00	140	189		
141			190	20.00	190
142	15.20	142	191		
143			192	20.20	192
144	15.40	144	193		
145			194	20.40	194
146	15.60	146	195		

Sudut Kompas (0)	Lebar PWM (ms)	Rumus (PWM us)-1000)/100	Sudut Kompas (0)	Lebar PWM (ms)	Rumus (PWM us)-1000)/100
196	20.60	196	245		
197			246	25.60	246
198	20.80	198	247		
199			248	25.80	248
200	21.00	200	249		
201			250	26.00	250
202	21.20	202	251		
203			252	26.20	252
204	21.40	204	253		
205			254	26.40	254
206	21.60	206	255		
207			256	26.60	256
208	21.80	208	257		
209			258	26.80	258
210	22.00	210	259		
211			260	27.00	260
212	22.20	212	261		
213			262	27.20	262
214	22.40	214	263		
215			264	27.40	264
216	22.60	216	265		
217			266	27.60	266
218	22.80	218	267		
219			268	27.80	268
220	23.00	220	269		
221			270	28.00	270
222	23.20	222	271		
223			272	28.20	272
224	23.40	224	273		
225			274	28.40	274
226	23.60	226	275		
227			276	28.60	276
228	23.80	228	277		
229			278	28.80	278
230	24.00	230	279		
231			280	29.00	280
232	24.20	232	281		
233			282	29.20	282
234	24.40	234	283		
235			284	29.40	284
236	24.60	236	285		
237			286	29.60	286
238	24.80	238	287		
239			288	29.80	288
240	25.00	240	289		
241			290	30.00	290
242	25.20	242	291		
243			292	30.20	292
244	25.40	244	293		

Sudut Kompas (^{0})	Lebar PWM (ms)	Rumus (PWM us)-1000)/100	Sudut Kompas (^{0})	Lebar PWM (ms)	Rumus (PWM us)-1000)/100
294	30.40	294	328	33.80	328
295			329		
296	30.60	296	330	34.00	330
297			331		
298	30.80	298	332	34.20	332
299			333		
300	31.00	300	334	34.40	334
301			335		
302	31.20	302	336	34.60	336
303			337		
304	31.40	304	338	34.80	338
305			339		
306	31.60	306	340	35.00	340
307			341		
308	31.80	308	342	35.20	342
309			343		
310	32.00	310	344	35.40	344
311			345		
312	32.20	312	346	35.60	346
313			347		
314	32.40	314	348	35.80	348
315			349		
316	32.60	316	350	36.00	350
317			351		
318	32.80	318	352	36.20	352
319			353		
320	33.00	320	354	36.40	354
321			355		
322	33.20	322	356	1.00	0
323			357	1.00	0
324	33.40	324	358	1.00	0
325			359	1.00	0
326	33.60	326	360	1.00	0
327					

GAMBAR ROBOT TANK



UNIVERSITAS ISLAM INDONESIA
FAKULTAS TEKNOLOGI INDUSTRI

KARTU KONSULTASI BIMBINGAN TUGAS AKHIR

Nama : Yudi Haryanto

Nomor_Mhs : 02524077

Pembimbing I : Tito Yuwono, ST, MSc.

Pembimbing II : Yusup Azis Ambiyah, ST.

Proposal Disetujui Tanggal : 19 Juli 2005

Tanda Tangan Mhs

Yudi Haryanto
Nama Mhs

Judul Tugas Akhir : KOMPAS DIGITAL PORTABLE

No	Tanggal	Keterangan	T T. Pembimbing
1	2/4/07	demo alat	
2	3/5/07	demo alat	
3	4/5/07	Demo OK	
4	12/6/07	Skripsi siap sidang	
5	12/6/07	acc laporan	

Pembimbing I

(.....)

Pembimbing II

(.....)

Nb: *Blangko ini sebagai syarat pendadaran*