

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



**PERANCANGAN DAN PEMBUATAN SISTEM PEMANTAU POSISI
KAPAL NELAYAN DENGAN GPS UNTUK PENENTU POSISI BUBU
BERBASIS MIKROKONTROLLER AT89S52**

SKRIPSI

Disusun Oleh :

ESTEFANO NUNO XIMENES

07.12.910



AGUSTUS 2010

FOURTH PART

CHAPTER

ARTICLE

SECTION

ARTICLE



ARTICLE

ARTICLE

ARTICLE

**INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO
PROGRAM STUDI ELEKTRONIKA**



**PERANCANGAN DAN PEMBUATAN SISTEM PEMANTAU POSISI KAPAL
NELAYAN DENGAN GPS UNTUK PENENTU POSISI BUBU BERBASIS
MIKROKONTROLLER AT89S52**

SKRIPSI

**Disusun Oleh:
Estefano Nuno Ximenes
07.12.910**

AGUSTUS 2010



THE NATIONAL ARCHIVES
COLLEGE PARK, MARYLAND

RECORDS OF THE NATIONAL ARCHIVES
COLLEGE PARK, MARYLAND

DATE: _____
BY: _____

GENERAL INFORMATION

1. Name of the organization: _____
2. Address: _____
3. City: _____
4. State: _____
5. Zip: _____

6. Name of the individual: _____

7. Title: _____
8. Organization: _____
9. Address: _____

PERSONAL INFORMATION

10. Name: _____
11. Address: _____
12. City: _____
13. State: _____
14. Zip: _____

ADDITIONAL INFORMATION

15. _____

16. _____

17. _____

18. _____

19. _____

LEMBAR PERSETUJUAN

**PERANCANGAN DAN PEMBUATAN SISTEM PEMANTAU POSISI KAPAL
NELAYAN DENGAN GPS UNTUK PENENTU POSISI BUBU BERBASIS
MIKROKONTROLLER AT89S52**

SKRIPSI

*Disusun dan Diajukan Untuk Melengkapi dan Memenuhi
Syarat Guna Mencapai Gelar Sarjana Teknik*

**Disusun Oleh :
Estefano Nuno Ximenes
07.12.910**

Mengetahui

Jurusan Teknik Elektro S-1

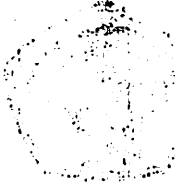
Diperiksa dan Disetujui

Dosen Pembimbing


Ir. Yusuf Ismail Nakhoda, MT
NIP. 1.1018800189


I Komang Somawirata, ST. MT.
NIP.P. 1030100361

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



STATE OF ARIZONA DEPARTMENT OF REVENUE
TAXPAYER IDENTIFICATION NUMBER (TIN) STATEMENT

This statement is prepared by the Department of Revenue for the taxpayer named above. It is based on information received from the taxpayer and other sources. It is subject to change without notice.

ARIZONA DEPARTMENT OF REVENUE, 1700 W. WASHINGTON AVENUE, PHOENIX, ARIZONA 85007

STATE OF ARIZONA DEPARTMENT OF REVENUE
TAXPAYER IDENTIFICATION NUMBER (TIN) STATEMENT

STATE OF ARIZONA DEPARTMENT OF REVENUE
TAXPAYER IDENTIFICATION NUMBER (TIN) STATEMENT
STATE OF ARIZONA DEPARTMENT OF REVENUE
TAXPAYER IDENTIFICATION NUMBER (TIN) STATEMENT
STATE OF ARIZONA DEPARTMENT OF REVENUE
TAXPAYER IDENTIFICATION NUMBER (TIN) STATEMENT
STATE OF ARIZONA DEPARTMENT OF REVENUE
TAXPAYER IDENTIFICATION NUMBER (TIN) STATEMENT
STATE OF ARIZONA DEPARTMENT OF REVENUE
TAXPAYER IDENTIFICATION NUMBER (TIN) STATEMENT

STATE OF ARIZONA DEPARTMENT OF REVENUE
TAXPAYER IDENTIFICATION NUMBER (TIN) STATEMENT

STATE OF ARIZONA DEPARTMENT OF REVENUE
TAXPAYER IDENTIFICATION NUMBER (TIN) STATEMENT

STATE OF ARIZONA DEPARTMENT OF REVENUE
TAXPAYER IDENTIFICATION NUMBER (TIN) STATEMENT

STATE OF ARIZONA DEPARTMENT OF REVENUE
TAXPAYER IDENTIFICATION NUMBER (TIN) STATEMENT

STATE OF ARIZONA DEPARTMENT OF REVENUE
TAXPAYER IDENTIFICATION NUMBER (TIN) STATEMENT

STATE OF ARIZONA DEPARTMENT OF REVENUE
TAXPAYER IDENTIFICATION NUMBER (TIN) STATEMENT

STATE OF ARIZONA DEPARTMENT OF REVENUE
TAXPAYER IDENTIFICATION NUMBER (TIN) STATEMENT

STATE OF ARIZONA DEPARTMENT OF REVENUE
TAXPAYER IDENTIFICATION NUMBER (TIN) STATEMENT

STATE OF ARIZONA DEPARTMENT OF REVENUE
TAXPAYER IDENTIFICATION NUMBER (TIN) STATEMENT

STATE OF ARIZONA DEPARTMENT OF REVENUE
TAXPAYER IDENTIFICATION NUMBER (TIN) STATEMENT



PERKUMPULAN PENGELOLAH PENDIDIKAN DAN TEKNOLOGI NASIONAL MALANG

INSTITUT TEKNOLOGI NASIONAL MALANG

FAKULTAS TEKNOLOGI INDUSTRI

FAKULTAS TEKNIK SIPIL DAN PERENCANAAN

PROGRAM PASCASARJANA MAGISTER TEKNIK

(PERSERO) MALANG
KAMPUS NIAGA MALANG

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

**BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI**

Nama Mahasiswa : Estefano Nuno Ximenes
N.I.M : 07.12.910
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Elektronika
Judul Skripsi : *“Perancangan dan Pembuatan Sistem Pemantau Posisi Kapal Nelayan Dengan GPS Untuk Penentu Posisi Bubu Berbasis Mikrokontroller AT89S52”*

Dipertahankan dihadapan majelis penguji skripsi Jenjang Strata Satu (S-1) pada :

Hari : Sabtu
Tanggal : 21 Agustus 2010
Dengan Nilai : 83,4 (A) *BY*

PANITIA UJIAN SKRIPSI

KETUA


Ir. Yusuf Ismail Nakhoda, M.P.

NIP.Y. 1018800189

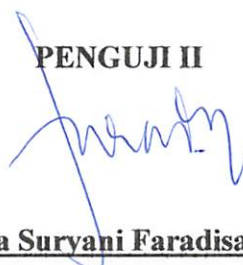
ANGGOTA PENGUJI

PENGUJI I



Sunny Prasetio, ST. MT

PENGUJI II



Irmalia Suryani Faradisa, ST. MT

NIP. P 1030100365

1951 11 10

1951 11 10

1951 11 10

1951 11 10

1951 11 10

1951 11 10

1951 11 10

1951 11 10

1951 11 10

1951 11 10

1951 11 10

1951 11 10

1951 11 10

1951 11 10

1951 11 10

1951 11 10

1951 11 10

1951 11 10

1951 11 10

1951 11 10

1951 11 10

1951 11 10

1951 11 10

1951 11 10

1951 11 10

1951 11 10





LEMBAR PERSEMBAHAN

Penulis memanjatkan sembah dan puji syukur yang sedalam-dalamnya kehadirat **Tuhan Yang Maha Esa** atas rahmat dan karunia yang Dia berikan. Dengan ijin-Nya jualah Penulis dapat menyelesaikan skripsi dan akhirnya menuntaskan perkuliahan guna mendapatkan gelar Sarjana Teknik. Banyak suka dan duka yang Penulis hadapi dalam penulisan dan penyelesaian skripsi ini, namun berhasil Penulis hadapi dengan rahmat-Nya melalui hamba-hamba-Nya yang telah dan akan dengan setia mendukung Penulis. Dengan ini rasa terimakasih Penulis sembahkan kepada :

- **Bapak, ibunda** tercinta yang dengan segala daya upaya memberi semangat dan dukungannya kepada Penulis. Tanpa nya, pasti tambah susaaa...k dech. Makasih yah Mom n Dad.....atas semua pengorbanannya **J LOVE YOU Mam n Dad**
- Semua saudara dan keluarga ku, **Amancio, Nobia, Adina n Avia** yang dengan senang Hati nemanin Ortu selama tanpa kehadiranku
- **MY wife n my life LOKUDO** yang tercinta n **my sweety Heart GRACA** yang telah banyak memberi arti dalam hidupku. Makasih selalu mendampingiku selama di Malang.dan setia menanti kehadiran ku di **TJMOR-LESTE. J LOVE YOU** aku selalu merindukan kalian berdua dalam hidupku selama-lamanya.
- **My friends at campus JTN Malang** di saat-saat mencemaskan. **You're all the best supporting guys.**
- **Last but not last my brother AJE** serta **mana Aucay** yang telah banyak membantu, selama aku merantau di kota Malang

Malang, sept 2010

Penulis

KATA PENGANTAR

Puji syukur kupersembahkan kehadirat Tuhan Yang Maha Esa yang telah memberikan berkat dan rahmat-Nya, sehingga dapat menyelesaikan skripsi yang berjudul “Perancangan dan Pembuatan Sistem Pemantauan Posisi Mobil Travel Dengan GPS Berbasis Mikrokontroler AT89S51 dengan Tampilan PC” ini dengan lancar. Skripsi ini merupakan persyaratan kelulusan Studi di Jurusan Teknik Elektro S-1 Konsentrasi Teknik Elektronika ITN Malang dan untuk mencapai gelar Sarjana Teknik.

Keberhasilan penyelesaian laporan skripsi ini tidak lepas dari dukungan dan bantuan berbagai pihak. Untuk itu penyusun menyampaikan terima kasih kepada :

1. Bapak Prof.Dr.Eng.Ir. Abraham Lomi, MSEE selaku Rektor Institut Teknologi Nasional Malang.
2. Bapak Ir.Sidik Noetjahjono, MT selaku Dekan Fakultas Teknologi Industri ITN Malang.
3. Bapak Ir. F. Yudi Limpraptono, MT selaku Ketua Jurusan Teknik Elektro S-1.
4. Bapak Sotjihadi, ST, selaku Dosen Pembimbing.
5. Ayah dan Ibu serta saudara-saudara kami yang telah memberikan do'a restu, moral,dorongan, semangat, dan biaya.
6. *Istri tercinta serta my sweetie heart Graca Love you.*
7. Semua yang telah membantu dalam penyelesaian penyusunan skripsi ini.

Penyusun telah berusaha semaksimal mungkin dan menyadari sepenuhnya akan keterbatasan pengetahuan dalam menyelesaikan laporan ini. Untuk itu penyusun mengharapkan saran dan kritik yang membangun dari pembaca demi kesempurnaan laporan ini.

Harapan penyusun semoga laporan skripsi ini memberikan manfaat bagi perkembangan ilmu pengetahuan dan pembaca.

Malang, Agustus 2010

Penyusun

DAFTAR ISI

	Halaman
ABSTRAK	i
KATA PENGANTAR	ii
DAFTAR ISI	iv
DAFTAR GAMBAR	viii
DAFTAR TABEL	x
BAB I. PENDAHULUAN	
1.1. Latar Belakang	1
1.2. Rumusan Masalah	2
1.3. Tujuan	3
1.4. Ruang lingkup pembahasan	4
1.4.1 Hardware (<i>perangkat keras</i>)	4
1.5. Metodologi Penulisan	5
1.6. Metodologi Penelitian	6
1.7. Sistematika Penulisan	8
BAB II. LANDASAN TEORI	
2.1. Umum	10
2.2. Global Positioning System	10
2.2.1 Pendahuluan	10
2.2.2 Element-Element GPS	11
2.2.3 Cara Kerja GPS	12
2.2.3.1 Menerima Titik Potong (Titik Triangulasi)	13
2.2.3.2 Sistem Pewaktuan	15

2.2.3.3	Penentuan Posisi Satelit di Angkasa	15
2.2.3.4	Koreksi Kesalahan	16
2.2.3.4.1	Perjalanan Sinyal Melalui Atmosfir	16
2.2.3.4.2	Multipath Error	17
2.2.3.4.3	GDOP	17
2.2.3.4.4	Selective Availability (SA)	19
2.3.	Protokol Komunikasi NMEA dan Bahasa RMC	19
2.4.	Liquid Crystal Display (LCD)	21
2.4.1	Konfigurasi LCD	21
2.4.2	Instruksi Operasi Dasar	23
2.4.2.1	Register	23
2.4.2.2	Busy Flag	24
2.4.2.3	Address Counter	24
2.4.2.4	Display Data RAM (DD RAM)	24
2.4.2.5	Character Generator ROM (CG ROM)	25
2.4.2.6	Character Generator RAM (CG RAM)	25
2.5.	Mikrokontroler AT89S51	26
2.5.1	Pendahuluan	26
2.5.2	Perangkat Keras Mikrokontroler AT89S52	26
2.5.3	Konfigurasi Kaki-Kaki Mikrokontroler AT89S52	28
2.5.4	Organisasi Memori Mikrokontroler AT89S52	31
2.5.4.1	Memori Program	31
2.5.4.2	Memori Data	31
2.5.4.3	Sistem Interupsi	34

2.5.4.4	Timer / Counter	35
2.5.4.5	Metode Pengalamatan	38
2.5.5	Bahasa Assembler MCS-51	39
2.6.	Komunikasi Serial	40
2.7.	Metode Transmisi Serial	41
2.8.	Serial Electrically Erasable an Programmable Read Only Memory (Serial EEPROM) AT24C16	43
2.8.1	Fitur-fitur yang dimiliki serial EEPROM	43
2.8.2	Konfigurasi Pin	44
2.8.3	Block diagram AT24C16	44
2.9.	Keypad	45

BAB III. PERENCANAAN DAN PEMBUATAN ALAT

3.1.	Perancangan Sistem	46
3.2.	Perancangan Hardware	49
3.2.1	Perancangan Komunikasi data GPS dengan Mikrokontroler AT89S52	49
3.2.2	Pengaturan <i>Boud Rate</i>	51
3.2.3	Perancangan Rangkaian Reset	53
3.2.4	Perancangan Rangkaian Serial EEPROM	55
3.2.5	LCD (Liquid Cristal Display)	56
3.2.6	Keypad	57
3.3.	Perangkat Lunak	59
3.3.1	Mikrokontroler	59
3.3.2	Flow Chart	60

BAB IV. ANALISA DAN PENGUJIAN ALAT

4.1. Pendahuluan	61	
4.2. Pengujian Modul GPS	61	
4.3. Pengujian Keseluruhan Sistem	63	
4.3.1 Bahasa pemograman yang digunakan oleh mikrokontroller AT89S52.....	64	
4.3.2 Program untuk baca GPS dan Tulis GPS	64	4.3.2
Program Akses memory (I2C protokol) at24C16.....	66	
4.3.3 Program untuk mengakses LCD.....	67	
4.3.4 Program untuk mengakses keypad.....	68	
4.4 Hasil pengujian system.....	70	
4.5. Analisa Pengujian	74	
4.5.1 Perhitungan hasil Error	75	
4.5.2 Perhitungan Hasil Error Rata-Rata	76	

BAB V. PENUTUP

5.1. Kesimpulan	80
5.2. Saran	81

DAFTAR GAMBAR

	Halaman
Gambar 2.1 Bagian Utama GPS	12
Gambar 2.2 Hasil Pengukuran Terhadap Satelit Pertama	13
Gambar 2.3 Hasil Pengukuran Dengan 2 Satelit	14
Gambar 2.4 Hasil Pengukuran Dengan 3 Satelit	14
Gambar 2.5 Perjalanan Sinyal Melalui Atmosfer.....	16
Gambar 2.6 Multipath Error	17
Gambar 2.7 GDOP Yang Buruk.....	18
Gambar 2.8 GDOP Yang Baik	18
Gambar 2-9. Diskripsi pin pada LCD 20 Char x 4 Line	22
Gambar 2.10 Blok Diagram AT89S52	27
Gambar 2.11. Bentuk Fisik AT89S52	28
Gambar 2.12 AT89S52 SFR dan <i>Reset Values</i>	32
Gambar 2.13 Asynchronous Serial Data Frame (8E1).....	42
Gambar 2.14 Bentuk fisik AT24C16	43
Gambar 2.15 Block diagram AT24C16	44
Gambar 3.1 Bubu.....	46
Gambar 3.2 Block diagram system.....	47
Gambar 3.3 Block diagram sistem perjalanan pertama dan kedua.....	48
Gambar 3.4 Block diagram sistem perjalanan.....	49
Gambar 3.5 Rangkaian interface antara GPS dengan MCU	50
Gambar 3.6 Rangkaian Mikrokontroler AT89S52	51
Gambar 3.7. Rangkaian Osilator	51

Gambar 3.8 Rangkaian Reset untuk MCU AT89S52	53
Gambar 3.9. Rangkaian Liquid Crystal Display (LCD) 20 Char x 4 Line.....	57
Gambar 3.10 Rangkaian interface keypad 4x4.....	57
Gambar 3.11. Rangkaian dasar keypad 4x4	58
Gambar 3.12 Flow chart	60
Gambar 4.1. Blok Diagram Pengujian Modul GPS.....	61
Gambar 4.2. Hasil Pengujian Modul GPS.....	62
Gambar 4.3 Block diagram pengujian sesuai dengan sistem perjalanan ...	63
Gambar 4.4 Foto keseluruhan alat.....	71
Gambar 4.5 Foto koordinat saat menentukan posisi data ke1 untuk Record	71
Gambar 4.6 foto koordinat saat menentukan posisi data ke25 untuk Record	71
Gambar 4.7 Foto koordinat saat menentukan posisi data ke60 untuk Record	72
Gambar 4.8 Foto koordinat saat menentukan posisi data ke1 untuk erase	72
Gambar 4.9 Foto koordinat saat menentukan posisi data ke23 untuk erase	73
Gambar 4.10 Foto perbandingan koordinat kampus 1 ITN	73
Gambar 4.11 Foto perbandingan koordinat Terminal Arjosari.....	73

DAFTAR TABEL

	Halaman
Tabel 2-1. Konfigurasi Pin-pin LCD	23
Tabel 2-2. Register Seleksi	24
Tabel 2-3. Fungsi Terminal pada LCD	25
Tabel 2.4 Konfigurasi port 1.5 sampai 1.7	29
Tabel 2.5 Alamat Sumber Interupsi	34
Tabel 2.6 Kegunaan <i>Interrupt Enable Register</i>	35
Tabel 2.7 <i>Register TMOD</i>	36
Tabel 2.8 <i>Timer/Counter Mode Control Register</i>	37
Tabel 2.9 Mode Operasi <i>Timer/Counter</i>	37
Table 2.10 konfigurasi pin	44
Tabel 3.1 Rangkaian Serial EEPROM	55
Table 3.2 Perincian dan Urutan <i>Pe-record</i>	55
Table 3.3 Memory mapping	56
Table 3.4 Metode Penghapusan dan <i>pe-record</i>	56
Tabel 4.1. Hasil Pengujian Modul GPS EG T-10	62
Tabel 4.2 Hasil data perjalanan pertama saat record bubu 1-50	64
Tabel 4.3 Perjalanan kedua saat record data 51-100	65
Tabel 4.4 Perjalanan kedua saat erase atau pengambilan bubu 50-1	66
Tabel 4.4 Hasil koordinat yang ditentukan oleh GPS standard	68
Tabel 4.5 Hasil koordinat yang ditentukan oleh GPS yang dirakit	69
Tabel 4.6 Perhitungan error untuk lintang S	69
Tabel 4.7 Perhitungan error untuk Bujur E	70

**PERANCANGAN DAN PEMBUATAN SISTEM PEMANTAU POSISI KAPAL NELAYAN DENGAN GPS
UNTUK PENENTU POSISI BUBU BERBASIS MIKROKONTROLLER AT89S52**

ESTEFANO NUNO XIMENES
Jurusan Teknik Elektro S-1
Konsentrasi Teknik Elektronika
Fakultas Teknologi Industri
Insitut Teknologi Nasional Malang
Email : nochén_encha@yahoo.com

ABSTRAK

GPS merupakan sistem navigasi yang berbasis satelit yang terdiri dari jaringan 24 orbit satelit berada 11.000 mil laut (11.000 × 1.85 km) diatas bumi dan dalam 6 lintasan yang berbeda dimana satelit tersebut bergerak dengan teratur dan mengitari bumi dua kali dalam waktu hampir 24 jam. Dalam hal ini penggunaan GPS pada kapal nelayan yang sering menangkap ikan-ikan besar biasanya berada ditengah laut yang menggunakan BUBU sebagaiperangkap. Dalam hal ini penggunaan GPS adalah salah satunya pemanfaatan pada peralatan perangkap bubu yang biasa dipakai oleh para nelayan untuk menangkap ikan. Kapal merupakan transportasi laut yang biasanya digunakan oleh para nelayan untuk mencari ikan, Kapal yang digunakan untuk menangkap ikan dibagi menjadi 3 jenis yaitu motor temple ,perahu tanpa motor dan kapal motor. Kapal penangkap ikan memiliki metode penangkapan ikan dengan menggunakan pukat tarik, jaring, pancing dan perangkap seperti bubu. Kapal penangkapan ikan yang menggunakan pancing biasanya hanya menagkap ikan di tepi laut, sedangkan kapal, yang menggunakan bubu dan jaring dapat ditebarkan pada semua daerah baik di tepi maupun tengah laut. Tingkat efisiensi GPS inilah yang menjadi pendorong ide dalam pembuatan tugas akhir Metodologi penulisan laporan ini melalui beberapa tahap terdiri atas : Library Research, Tahap perencanaan, Tahap pembuatan, Tahapan pengujian. Berdasarkan hasil pengujian yang dilakukan di beberapa tempat dikota Malang total error yang diperoleh adalah 0,001% setelah dibandingkan dengan hasil yang diperoleh GPS standard

ABSTRACT

GPS is a satellite-based navigation system consisting of a network of 24 satellites in orbit 11 000 nautical miles (1.85 × 11 000 km) above the earth, and in six different paths where the satellite is moving in an orderly and around the earth twice in a time nearly 24 hours. In this case the use of GPS on a fishin boat that often catch the big fish are usually located in the middle of sea which uses Bubu. In this case the use of GPS is one of them use the equipment Bubu trap that is usually used by the nelayan to catch fish. The ship is a marine transportation which is usually used by fishermen to catch fish, vessels used to catch fish is divided into three types: outboard motors, boat motors and boats without motors. Fishing vessels have a method of fishing using trawl, nets, fishing and trapping as Bubu. Fishing vessel that uses fishing rod fish to catch usually only on the seashore, while the ship, which uses a trap and nets can be cast on all the good areas on the edge or middle of the sea. Efficiency level GPS this is the major driving force in making the final idea of writing this report methodology through several stages consisting of: Library Research, planning stage, stage production, stage of testing. Based on the results of tests conducted in several towns of Malang, where the total errors obtained was 0.001% when compared with results obtained by the GPS standard

BAB I

PENDAHULUAN

1.1 Latar Belakang

Kemajuan teknologi dibidang elektronika dewasa ini berkembang dengan cepat dan berpengaruh dalam pembuatan alat-alat canggih, yaitu alat yang dapat bekerja secara otomatis dan memiliki ketelitian tinggi salah satunya dengan bantuan *Global Positioning System (GPS)*.

Dikutip dari penulis buku Abidin Z Hasanuddin, Penentuan Posisi dengan GPS dan Aplikasinya, PT Pradnya Paramita, Jakarta, 1995. Dimana GPS merupakan sistem navigasi yang berbasisan satelit yang terdiri dari jaringan 24 orbit satelit berada 11.000 mil laut (11.000×1.85 km) diatas bumi dan dalam 6 lintasan yang berbeda dimana satelit tersebut bergerak dengan teratur dan mengintari bumi dua kali dalam waktu hampir 24 jam.

Dalam hal ini penggunaan GPS adalah salah satunya pemanfaatan pada peralatan perangkap bubu yang biasa dipakai oleh para nelayan untuk menangkap ikan. Kapal merupakan transportasi laut yang biasanya digunakan oleh para nelayan untuk mencari ikan, Kapal yang digunakan untuk menangkap ikan dibagi menjadi 3 jenis yaitu motor temple ,perahu tanpa motor dan kapal motor. Kapal penangkap ikan memiliki metode penangkapan ikan dengan menggunakan pukot tarik, jaring, pancing dan perangkap seperti bubu. Kapal penangkapan ikan yang menggunakan pancing biasanya hanya menangkap ikan di tepi laut, sedangkan kapal, yang menggunakan bubu dan jaring dapat ditebarkan pada semua daerah baik di tepi maupun tengah laut.

Pada awalnya kapal penangkap ikan dengan menggunakan bubu, membawa 50 bubu kosong sebab pada umumnya kapal penangkap ikan ini memiliki kapasitas

untuk menampung 50 bubu, kemudian akan ditebarkan 50 bubu baru yang masih kosong pada daerah tempat berkumpulnya ikan-ikan dan melakukan pengambilan bubu lama yang sudah terisi ikan sebanyak 50 buah. Pada proses pengambilan bubu, para nelayan sering kali bingung untuk mencari diman posisi bubu yang telah ditebarkan dan terkadang membutuhkan waktu yang lama untuk mencari bubu yang telah ditebar. Sebab bubu yang telah mereka tebarkan, berjumlah relatif banyak dan terkadang ada yang terlupa posisi ditebarkannya dimana. Untuk mengetahui posisi bubu yang ditebarkan, dibutuhkan suatu alat yang dapat *me-record* posisi bubu yang telah ditebarkan salah satunya dengan menggunakan GPS (*global positioning system*).

Tingkat efisiensi GPS inilah yang menjadi pendorong ide dalam pembuatan tugas akhir.

1.2 Rumusan Masalah

Mengacu pada permasalahan yang telah diuraikan pada latar belakang, maka rumusan masalah dapat disusun sebagai berikut:

1. Bagaimana mendesain suatu alat yang dapat *me-record* suatu posisi dan ditampilkan pada LCD Char x 4 Line. sehingga dapat diketahui posisi-posisi yang *di-record* pada LCD Char x 4 Line.
2. Bagaimana membuat program mikrokontroler AT89S52 untuk dapat membaca GPS dimana *output* GPS berupa data serial (ASCII).
3. Bagaimana membuat program mikrokontroler AT89S52 untuk dapat membaca dan menulis pada serial EEPROM.
4. Bagaimana membuat program mikrokontroler AT89S52 untuk dapat menampilkan suatu posisi pada LCD 20 Char x 4 Line.

1.3 Tujuan

Adapun tujuan yang ingin dicapai dalam pembuatan tugas akhir ini yang terdiri dari:

1. Mendesain suatu alat yang memiliki fitur sebagai berikut:
 - Dapat *me-record* 100 titik dengan asumsi 50 titik dianggap sebagai *database* (posisi bubu yang lama dan 50 titik kemudian adalah posisi bubu yang baru).
 - Alat ini dilengkapi dengan 6 tombol untuk *me-record*, menghapus posisi bubu, *back record*, *forward record*, *back erase* dan *forward erase*.
2. Mendesain suatu system yang menggunakan GPS, dimana GPS berfungsi sebagai pengirim informasi. Kemudian informasi dari GPS tersebut dapat dibaca dan diolah oleh mikrokontroler AT89S52.
3. Mendesain suatu system yang dapat dipakai untuk menyimpan data-data dari hasil pembacaan GPS oleh mikrokontroler, untuk menyimpan data pilih serial EEPROM (24C16) sebagai memorinya.
4. Mendesain system dimana mikrokontroler dapat mengolah data yang di dapat dari GPS untuk di tampilkan di LCD 20 Char x 4 Line.
5. Membantu atau memudahkan para nelayan untuk mencari posis dimana bubu ditebarkan.

1.4 Ruang lingkup pembahasan

Agar perancangan dan pembuatan alat ini dapat dilakukan secara sistematis dan terarah, maka ruang lingkup pembahasan dan perancangan alat dibatasi sebagai berikut:

1. Penggunaan alat digunakan diluar ruang serta dibatasi hanya untuk suatu wilayah tertentu yang telah ditentukan sebagai simulasi, yaitu disekitar wilayah kampus ITN II Malang.
2. Tingkat akurasi dari lokasi yang ditunjukkan tergantung pada akurasi GPS *receiver* yang digunakan..
3. Tidak membahas secara detail bagian GPS *receiver*.
4. Tidak membahas bahasa pemrograman yang digunakan pada mikrokontroler AT89S52
5. Tidak membahas catu daya yang digunakan.

1.4.1 Hardware (*perangkat keras*)

- GPS (*Global Positioning system*)

GPS module yang dipakai adalah EGT-10 dengan kepresisian sebesar 25 meter.

- Serial EEPROM

Serial EEPROM yang dipakai adalah 24C16 dengan kapasitas memory 16KB sehingga serial EEPROM ini sangat mendukung dalam pembuatan aplikasi GPS pada kapal penangkap ikan.

- Mikrokontroler

Mikrokontroler yang digunakan merupakan produk ATMEL yaitu AT89S52 .

1.5 Metodologi Penulisan

Metodologi penulisan tugas akhir ini adalah sebagai berikut:

A. Library Research.

Yaitu penelitian yang dilakukan dengan mengumpulkan data dan membaca buku-buku acuan yang ada hubungannya dengan masalah yang dibahas dalam skripsi ini.

B. Tahap pencanaan:

- ✓ Perencanaan tiap-tiap blok diagram dengan komponen yang sesuai.
- ✓ Perencanaan sistem secara keseluruhan.

C. Tahap pembuatan:

- ✓ Penggabungan tiap-tiap blok menjadi suatu kesatuan utuh yang diharapkan nantinya dapat mendukung keseluruhan sistem
- ✓ Pembuatan perangkat lunak pada microcontroller AT89S52 menggunakan *instruction set* yang telah disediakan oleh intel.

D. Tahapan pengujian, meliputi:

- ✓ Pengujian perangkat keras:
Pengujian ini dilakukan untuk mengetahui apakah sistem perangkat keras yang dibuat berjalan dengan baik ataukah tidak. Pengujian dengan bantuan perangkat lunak.
- ✓ Pengujian perangkat lunak:
Pengujian perangkat lunak pada system mikrokontroler AT89S52 dilakukan dengan melakukan simulasi pada program simulasi AT89S52.

1.6 Metodologi penelitian

Metodologi penelitian yang dilakukan dalam penulisan tugas akhir ini meliputi Sembilan bagian, yaitu tinjauan pustaka, pengumpulan data, perencanaan perangkat keras, perencanaan perangkat lunak, pembuatan perangkat keras, pembuatan perangkat lunak, pengujian system, pengambilan kesimpulan, dan penulisan laporan tugas akhir.

Tinjauan pustaka :

Tinjauan pustaka ini meliputi studi mengenai teori, instruksi-instruksi dan fitur-fitur dari mikrokontroler AT89S52, *GPS module*, *graphic LCD 20 Char x 4 Line*. desain *hardware* menggunakan LCD 20 Char x 4 Line dan keypad juga pemrograman assembly.

A. Pengumpulan Data

Pengumpulan data yang meliputi pengumpulan data mikrokontroler AT89S52, *GPS module*, LCD 20 Char x 4 Line, serial EEPROM AT24C16, desain *hardware* menggunakan LCD, keypad, dan pemrograman assembly

B. Perencanaan Perangkat Keras

Perangkat perangkat keras pada aplikasi GPS pada kapal penangkap ikan ini meliputi:

- Perencanaan perangkat keras pada PCB converter GPS module (EG-T10) ke mikro.
- Perencanaan perangkat keras mikrokontroler *single chip* serta memory EEPROM 24C16.
- Perencanaan perangkat keras display yang berupa *graphic LCD 20 Char x 4 Line*.
- Perencanaan perangkat keras keypad untuk *me-record* dan menghapus koordinat.

C. Perencanaan Perangkat Lunak

Perencanaan perangkat lunak pada aplikasi GPS pada kapal penangkap ikan meliputi:

- Perencanaan perangkat lunak pada mikrokontroler dengan pemrograman basic compiler.

D. Pembuatan Perangkat Keras

Pembuatan perangkat keras pada aplikasi GPS pada kapal penangkap ikan meliputi :

- Perencanaan perangkat keras pada *converter GPS* module ke mikro.
- Perencanaan perangkat keras mikrokontroler *single chip* serta memori EEPROM 24C16.
- Perencanaan perangkat keras display pada LCD 20 Char x 4 Line.
- Perencanaan perangkat keras keypad untuk *me-record* dan menghapus koordinat.

E. Pembuatan Perangkat Lunak

Pembuatan perangkat lunak pada aplikasi GPS pada kapal penangkap ikan meliputi :

- Perencanaan perangkat lunak pada mikrokontroler dengan Basic Compiler.

F. Pengujian Sistem

Pengujian system pada tugas akhir ini dilakukan dengan :

- Pengujian untuk *me-record* posisi koordinat pada serial EEPROM dan ditampilkan ke LCD. Pengujian dilakukan didarat secara simulasi dengan mengganti posisi bucu yang ditebarkan pada nomor rumah pada jalan tertentu. Sehingga dapat diketahui kepresisian alat pada saat posisi *pe-record-an* dan penghapusan.
- Pengujian dilakukan dengan simulasi menggunakan mikrokontroler yang deprogram untuk mengeluarkan data sesuai dengan data yang dikeluarkan

oleh GPS. Hal ini dilakukan untuk pengujian system *pe-recorder-an* dan penghapusan data.

G. Pengambilan Kesimpulan

Pengambilan kesimpulan berdasarkan hasil pengujian yang telah dilakukan terhadap aplikasi GPS pada kapal penangkap ikan yang telah dibuat, meliputi antara lain : kesimpulan tentang kepresisian dari GPS ketika *pe-record-an* dan penghapusan data (pemasangan bubu dan pengambilan bubu), pengaruh GPS terhadap cuaca dan pengujian system saat *pe-record-an* data maupun penghapusan data.

H. Penulisan Lapoan Tugas Akhir

Penulisan laporan tugas akhir secara keseluruhan.

1.7 Sistematika Penulisan

- Bab 1 : PENDAHULUAN

dalam bab ini dipaparkan hal-hal yang yang berkaitan dengan latar belakang, tujuan, perumusan masalah, ruang lingkup pembahasan serta metodologi penelitian yang digunakan dalam pembuatan tugas akhir ini.

- Bab2 : TEORI PENUNJANG

Dalam bab ini akan dijelaskan mengenai mikrokontroler AT89S52, GPS *module*, *Graphic* LCD 20 Char x 4 Line. serial EEPROM AT24C16, desain *hardware* menggunakan *Graphic* LCD dan keypad, dan pemrograman assembly.

- Bab 3 : PERENCANAAN SISTEM

Dalam bab ini akan dibahas mengenai perencanaan dan perealisasiian aplikasi GPS pada kapal penangkap ikan secara keseluruhan, baik perangkat keras maupun perangkat lunak.

- **Bab 4 : PENGUJIAN SISTEM**

Bab ini membahas tentang pengujian dan hasil yang diperoleh dari aplikasi GPS pada kapal penangkap ikan yang telah dibuat.

- **Bab 5 : KESIMPULAN DAN SARAN**

Bab ini berisi kesimpulan - kesimpulan yang diperoleh dari perencanaan dan perealisasi tugas akhir ini serta saran-saran yang diberikan untuk penyempurnaan dan pengembangan system lebih lanjut.

BAB II

LANDASAN TEORI

2.1 Umum

Dalam perencanaan dan pembuatan sistem Pemantauan Posisi Kapal Nelayan Dengan GPS (*Global Positioning System*) Berbasis Mikrokontroler AT89S52 dibutuhkan beberapa teori yang menjadi landasan, yaitu teori yang berhubungan dengan komponen-komponen yang digunakan dalam perancangan dan pembuatan sistem ini.

2.2 Global Positioning System

2.2.1 Pendahuluan

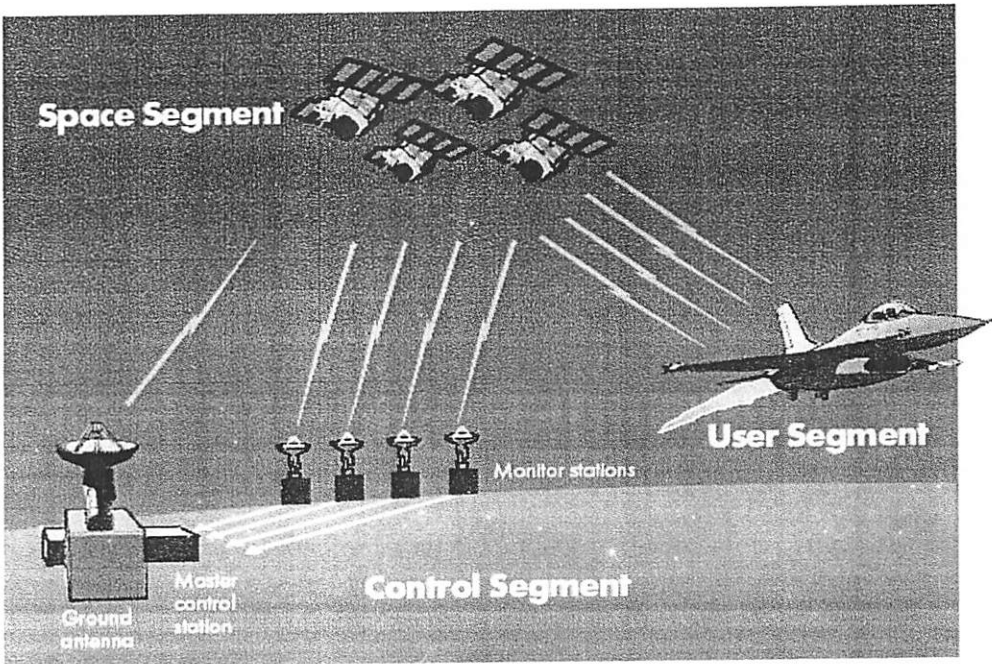
Global Positioning System (GPS) adalah sistem navigasi yang berbasiskan satelit yang terdiri dari jaringan 24 orbit satelit berada 11.000 mil laut (11.000×1.85 km) diatas bumi dan dalam 6 lintasan orbit yang berbeda. Jika dilihat dari ketinggiannya, satelit-satelit GPS ini termasuk dalam satelit MOE, dimana satelit tersebut bergerak secara teratur dan mengitari bumi dua kali dalam waktu hampir 24 jam (kurang 4 menit). Dengan cara ini, maka minimal akan terdapat 6 satelit pada setiap saat yang dapat diterima sinyalnya dari tempat manapun dimuka bumi. Satelit GPS adalah satelit Navstar dan pembiyaannya ditanggung pemerintah Amerika Serikat.

2.2.2 Element-element GPS

Global Positioning System (GPS) dibagi atas 3 bagian utama, yaitu : *space segment*, *user segment*, dan *control segment*. *Space segment* berupa satelit-satelit GPS, pada awalnya berjumlah 24 satelit, masing-masing berada pada orbit dengan ketinggian 11.000 mil laut diatas permukaan bumi. Saat ini terdapat 27 satelit GPS yang beroperasi. Satelit- satelit ini mampu melakukan transmisi pada frekuensi 1.575,42 MHz (sinyal L1) dan 1.227,6MHz (sinyal L2).sinyal L1 digunakan secara umum oleh kalangan sipil sedangkan sinyal L2 hanya digunakan untuk kalangan terbatas oleh pihak militer Amerika Serikat.

User segment terdiri atas berbagai penerima GPS (GPS Receiver), penerima GPS ini melakukan pendeteksian, penkodean dan pemrosesan sinyal dari satelit GPS. Saat ini terdapat ratusan model penerima GPS berbeda yang telah dipergunakan.

Control Segment terdiri atas stasiun bumi (*ground station*) yang berfungsi untuk memastikan bahwa satelit-satelit GPS tersebut tetap bekerja dengan benar. Stasiun ini berlokasi di beberapa tempat diseluruh dunia (Hawai dan Kwajalein disamudra pasifik, diego garcia disamudra hindia, kepulauan Ascension disamudra Atlantik, dan Colorado Springs di Colorado). Stasiun utama (*master ground Station*) berada dimarkas.



Gambar 2.1 Bagian Utama GPS^[3]

Falcon Air Force Clorado Springs. Stasiun bumi ini bertugas mendata posisi dan kondisi setiap satelit serta memberikan perbaikan (koreksi) informasi.

2.2.3 Cara Kerja GPS

GPS mampu memberikan informasi suatu posisi dipermukaan bumi secara akurat melalui serangkaian proses pengukuran dan kalkulasi yang rumit, namun pada dasarnya cara kerja GPS dapat dijelaskan dalam beberapa langkah dasar sebagai berikut:

1. Dasar dari proses penentuan posisi pada GPS adalah triangulasi (pertemuan titik potong) dari satelit-satelit GPS. Triangulasi juga sering disebut sebagai trilaterasi.
2. Agar dapat melakukan proses triangulasi dari satelit-satelit GPS, pesawat penerima GPS (GPS Receiver) harus dapat mengukur jarak masing-masing satelit dipesawat penerima dengan menghitung jarak waktu tempuh sinyal radio dari satelit ke penerima.

3. Untuk menghitung waktu tempuh seperti diatas maka GPS memerlukan suatu sistem pewaktuan (timing system) yang sangat akurat.
4. Harus diketahui secara pasti posisi masing-masing satelit diangkasa, sehingga dapat dilakukan posisi titik potong (titik triangulasi) yang merupakan posisi dari penerima.
5. Yang terakhir adalah koreksi terhadap hasil yang diterima karena adanya faktor-faktor yang menyebabkan terjadinya kesalahan pengukuran jarak antara satelit dan penerima.

2.2.3.1 Menerima Titik Potong (Titik Triangulasi)

Sebenarnya GPS berlandaskan pada prinsip triangulasi, yaitu dengan menjadikan satelit-satelit di angkasa sebagai titik-titik referensi untuk menentukan suatu lokasi pada bumi.

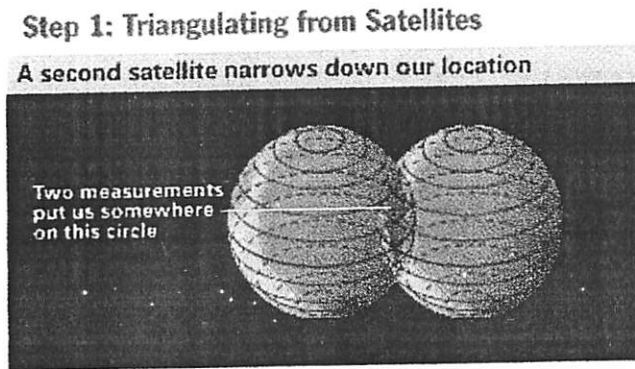
Sebagai contoh sebuah penerima GPS menerima sinyal dari 3 satelit GPS. Dari satelit pertama diperoleh jarak 11.000 mil. Dengan menganggap satelit pertama sebagai pusat sebuah bola dengan jari-jari 11.000 mil maka posisi penerima merupakan suatu titik pada permukaan bola tersebut.



Gambar 2.2. Hasil Pengukuran Terhadap Satelit Pertama^[5]

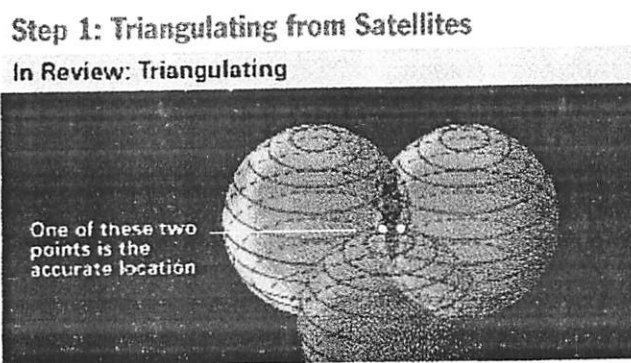
Kemudian dari satelit kedua diperoleh jarak antara penerima dengan satelit kedua adalah 12.000 mil. Seperti pada satelit pertama, maka diperoleh sebuah luasan bola dengan jari-jari 12.000 mil. Kedua luasan bola pertama dan kedua berpotongan yang menghasilkan

sebuah luas lingkaran. Maka posisi penerima adalah sebuah titik pada lingkaran perpotongan tersebut.



Gambar 2.3. Hasil Pengukuran Dengan 2 Satelit^[6]

Dengan perhitungan dari satelit ketiga diperoleh jarak antara penerima dan satelit ketiga sejauh 13.000 mil. Sehingga diperoleh sebuah luasan bola berjari-jari 13.000 mil. Ketiga luasan bola itu akan saling berpotongan sehingga menghasilkan dua titik pada irisannya. Posisi penerima adalah pada salah satu titik potong tersebut, sedangkan titik yang lain pada umumnya merupakan titik yang tidak mungkin. Akan tetapi kadang diperlukan tambahan perhitungan keempat dengan memanfaatkan sebuah hasil perhitungan jarak terhadap satelit lain. Akhirnya dapat ditentukan posisi penerima GPS pada permukaan bumi.



Gambar 2.4. Hasil Pengukuran Dengan 3 Satelit^[8]

2.2.3.2 Sistem Pewaktuan

Waktu tempuh sinyal radio merupakan kunci dari GPS, maka diperlukan suatu sistem pewaktuan yang benar-benar baik. Satu kesalahan dalam seperseribu detik saja dapat mengakibatkan kesalahan perhitungan sampai 200 mil. Kesalahan akibat pewaktuan ini biasanya disebut pseudo range. Pada satelit, pewaktuan biasa dikatakan hampir sempurna karena menggunakan jam atom dengan tingkat kepresisian tinggi. Tetapi pada sisi penerima sangatlah tidak mungkin untuk menerapkan hal yang sama agar dapat terjadi sinkronisasi. Hal ini dikarenakan harga jam atom yang sangat mahal, antara USD 50.000 sampai USD 100.000. Akhirnya diterapkan sebuah solusi dimana dibuat sebuah pengukuran kesalahan (error) setiap saat yang menjadi offset bagi waktu penerima GPS, kemudian offset ini dikurangkan pada perhitungan waktu sebelumnya, sehingga diperoleh pewaktuan yang setaraf dengan jam atom.

2.2.3.3 Penentuan Posisi Satelit di Angkasa

Sebagai titik referensi (*reference point*) untuk perhitungan posisi suatu obyek, satelit-satelit GPS harus diketahui secara pasti posisinya di angkasa. Satelit-satelit GPS bergerak pada orbitnya sedemikian rupa sehingga di bagian bumi manapun akan terlihat minimal 5 satelit. Posisi dari setiap satelit (data almanak satelit) telah diprogramkan ke setiap penerima GPS di bumi yang memberikan informasi dimana posisi satelit pada setiap saatnya.

Orbit dari masing-masing satelit GPS adalah tetap, akan tetapi untuk menyempurnakannya masing-masing tersebut terus-menerus dimonitor. Sehingga setiap satelit tetap berada pada posisi, ketinggian dan kecepatan yang benar. Penyimpangan posisi satelit terjadi dikarenakan adanya gravitasi bulan dan matahari. Kesalahan akibat penyimpangan posisi satelit ini disebut kesalahan ephemeris (*ephemeric errors*). Kesalahan

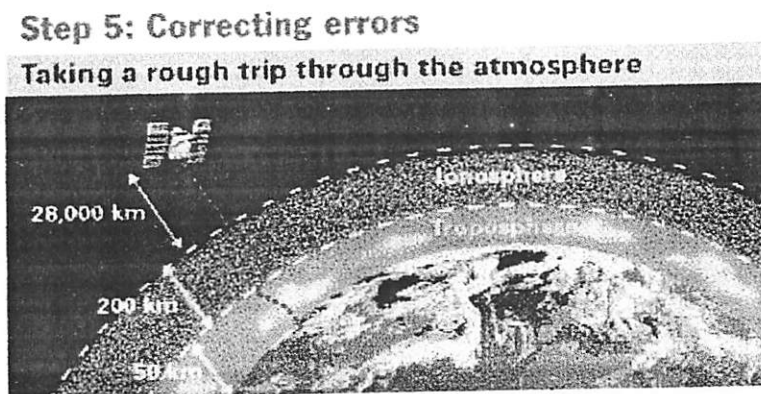
ini biasanya sangat kecil dan dapat diabaikan, namun untuk memperoleh hasil yang sangat tepat, maka kesalahan tersebut harus diperhitungkan.

2.2.3.4 Koreksi Kesalahan

Secara teoritis, cara kerja GPS sangat sederhana dan mudah bila langkah-langkah diatas berjalan lancar dan tanpa hambatan dengan asumsi sinyal yang dipancarkan tidak terganggu. Tetapi pada kenyataannya terdapat banyak hal yang dapat terjadi pada sinyal GPS yang mengakibatkan terjadinya kesalahan. Berikut beberapa uraian tentang faktor-faktor penyebab kesalahan pada sistem GPS :

2.2.3.4.1 Perjalanan Sinyal Melalui Atmosfer

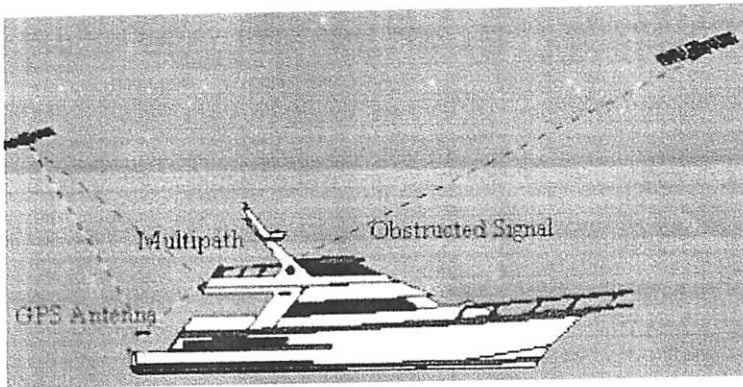
Dalam perjalanannya, sinyal GPS yang dipancarkan satelit GPS ke penerima GPS di bumi harus melalui lapisan-lapisan atmosfer bumi yang terdiri dari berbagai macam zat yang dapat mempengaruhi sinyal GPS. Lapisan ionosfer yang terdiri dari banyak partikel bermuatan (ion) yang dapat mempengaruhi sinyal elektromagnetik menjadi hambatan terbesar pada transmisi sinyal GPS. Selain itu, lapisan troposfer memberikan pengaruh yang cukup besar dikarenakan pada lapisan ini banyak terjadi peristiwa cuaca yang dapat mengganggu perjalanan sinyal elektromagnetik.



Gambar 2.5. Perjalanan Sinyal Melalui Atmosfer^[8]

2.2.3.4.2 Multipath Error

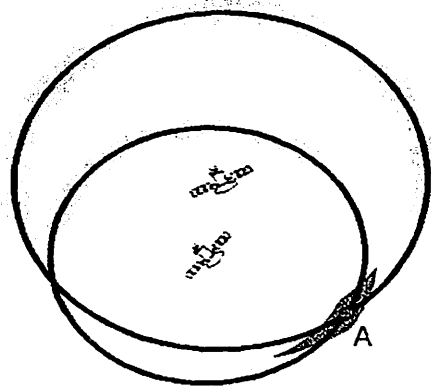
Multipath Error merupakan kesalahan yang terjadi akibat pemantulan sinyal GPS oleh benda-benda dipermukaan bumi, sehingga penerima GPS mengalami kesulitan membedakan sinyal yang datang secara langsung dari satelit dengan sinyal yang telah terpantul. Jika sinyal pantulan cukup kuat dapat mengacaukan penerima dan menyebabkan kesalahan perhitungan pos.



Gambar 2.6. Multipath Error^[9]

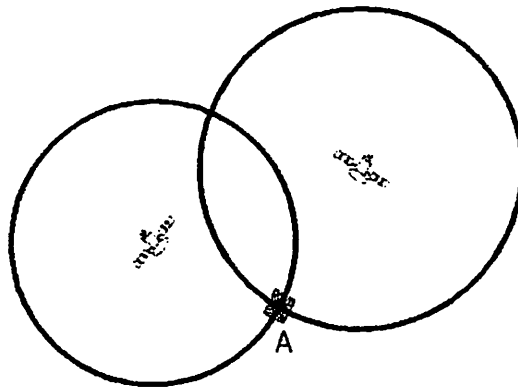
2.2.3.4.3 GDOP (Geometric Dillution of Precision)

Geometric Dillution of Precision (GDOP) merupakan suatu faktor yang dapat menyebabkan kesalahan pengukuran yang dikarenakan struktur geometri yang terjadi antara satelit-satelit GPS terhadap penerima GPS dibumi.



Gambar 2.7. GDOP Yang Buruk^[10]

Dalam gambar 2.7 diatas tampak bahwa penerima GPS menerima sinyal dari dua satelit GPS yang berdekatan. Ini berakibat pada perpotongan yang terjadi antara dua lingkaran yang dibentuk dua satelit itu akan menghasilkan ketidakpastian yang besar pula. Sedangkan dalam gambar 2.8 berikut tampak bahwa perpotongan kedua lingkarannya akan makin sempit sehingga faktor ketidakpastiannya lebih kecil.



Gambar 2.8. GDOP Yang Baik^[11]

2.2.3.4.4 Selective Availability (SA)

Selective Availability atau yang lebih dikenal sebagai SA merupakan degradasi (penurunan) ketepatan dari GPS yang digunakan secara umum yang ditentukan Departemen Pertahanan Amerika Serikat. Dengan SA, ketepatan GPS diturunkan sampai maksimal 100 meter, tetapi biasanya kesalahan diatas 30 meter tidak wajar terjadi. Penurunan ketepatan GPS ini dilakukan untuk mencegah penggunaan GPS oleh pihak yang tidak bertanggung jawab. Sehingga GPS dengan akurasi yang tinggi dapat digunakan untuk kepentingan militer. Sedangkan yang dapat dimanfaatkan untuk kalangan umum adalah GPS yang sudah terdegradasi.

2.3 Protokol Komunikasi NMEA dan Bahasa RMC

Dengan beraneka ragamnya peralatan-peralatan yang dapat dihubungkan dengan sebuah GPS receiver maka haruslah dibentuk suatu protocol komunikasi yang standard sehingga suatu GPS receiver dapat berhubungan dengan peralatan-peralatan yang lain, dan suatu peralatan yang dapat berhubungan dengan berbagai macam GPS receiver yang ada dipasaran. Untuk keperluan itulah maka pada tahun 1980, sebuah grup yang terdiri dari profesional-profesional membentuk suatu standard komunikasi untuk interface peralatan-peralatan kelautan. Hasilnya adalah NMEA 0180 (*National Marine Elektronik Association*) yang kemudian dikembangkan menjadi NMEA 0182 dan dengan melihat kekurangan yang ada pada NMEA 0180 dan NMEA 0182 disempurnahkan menjadi NMEA 0183.

Pada NMEA 0180 selain kompatibel dengan RS 232 dengan komponen IC TTL juga kompatibel dengan dengan standart RS 422 yang merupakan standart komunikasi dengan sistem diferensial. Sistem ini bekerja pada 4800 bps dengan sistem pengiriman secara data paket, selain itu semua data yang dikirimkan sudah bukan berupa data byte langsung

melainkan berupa kode-kode ASCII yang dapat dicetak. Berikut adalah format dari protocol komunikasi ini :

\$	Talker ID	Sentence ID	Data Field	..	Chek Sum	< CR >	< LF >
----	-----------	-------------	------------	----	----------	--------	--------

Keterangan :

1. "\$" Merupakan karakter awal yang dikirimkan menandakan awal dari pengiriman suatu paket data.
2. Talker ID Terdiri dari dua huruf yang menunjukkan jenis GPS receiver yang sedang dipakai.

Contoh:

- a. GPS Global Positioning System receiver
 - b. LC Loran-C receiver
 - c. OM Omega Navigation Receiver
 - d. I I Integrated Instrument Receiver
3. Sentence ID terdiri dari 3 huruf yang menunjukkan jenis paket yang sedang dikirim.

Contoh:

- a. GLL Geograpic Positioning, Longitude, Latitude
 - b. BWC Bearing and Distance to Waypoint
 - c. GSA Gips Dops And Active Satelite
 - d. GSV Satelites In View
 - e. RMB Recommended Minimum Navigation Information
4. Data Fields merupakan data dari jenis paket yang telah disebutkan pada Sentence ID. Apabila pada segment ini terdiri dari beberapa jenis data lagi maka perlu dipisah-pisahkan dengan koma. Apabila data yang bersangkutan tidak dapat

diberikan oleh GPS receiver, karena misalnya GPS receiver tidak menerima sinyal informasi apapun, maka yang dikirim hanya komanya saja tanpa diisi apapun

5. Check Sum bagian hanya optional (boleh ada / boleh tidak). Check Sum ini selalu di dalului dengan karakter '*' dan terdiri dari dua digit hexsadesimal. Check Sum ini digunakan sebagai pendeteksi error apakah paket data yang diterima itu sudah benar dan sesuai dengan saat dikirimkan. Check Sum ini merupakan XOR dari semua karakter, tetapi tidak termasuk '\$' dan '*'.
6. <CR> <LF> ASCII Carrierd Return dan Line Feed ini dikirimkan sebagai tanda bahwa transmisi untuk paket bersangkutan sudah berakhir.

RMC (*Recommended Minimum Specific*) GPS/ Transmit Data, yang merupakan salah satu bahasa dalam protocol komunikasi NMEA 0183 yang berguna sebagai sarana komunikasi GPS receiver dengan peralatan-peralatan luar. RMC ini memberikan informasi tentang lokasi geografis meliputi Longitude dan Latitude. Format RMC adalah sebagai berikut:

```
$ GPRMC, 06014, A, 0756.5394, S, 11237.0438, E, 000.0, 060604, 001.4, E*67
```

Pada contoh diatas berisi data posisi 07⁰ 56, 5394'LS dan 112⁰,0438' BT.

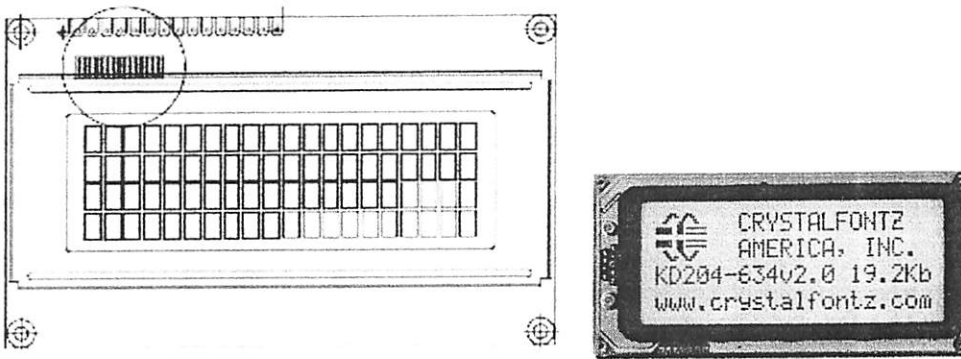
2.4. Liquid Crystal Display (LCD)

2.4.1. Konfigurasi LCD

Liquid Crystal Display adalah modul tampilan berkonsumsi daya yang relatif rendah dan terdapat sebuah controller CMOS didalamnya. Kontroler tersebut sebagai pembangkit karakter dari ROM/RAM dan display data RAM. Semua fungsi tampilan dikontrol oleh suatu intruksi dan modul LCD dapat dengan mudah diinterfacekan dengan mikroprosesor/mikrokontroler. Input yang diperlukan untuk mengendalikan modul ini berupa bus data yang termultipleks dengan bus alamat dan 3 bit sinyal

control, pengendali dot matriks LCD dilakukan secara internal pada modul LCD sendiri.

LCD merupakan suatu bentuk kristal cair yang akan beremulsi apabila dikenakan tegangan padanya. Tampilan ini berupa dot matriks 5×7 LCD sehingga jenis huruf yang akan dapat ditampilkan akan lebih banyak dan lebih baik resolusinya jika dibandingkan dengan 7 segment.



Gambar 2-9. Diskripsi pin pada LCD 20 Char x 4 Line.

LCD tipe 20 Char x 4 Line. memiliki cirri-ciri sebagai berikut :

- LCD ini terdiri dari 20 kolom 4 baris untuk karakter dengan tampilan yang terdiri dari 5×7 dot matrix dengan cursor
- Karakter generator ROM dengan 192 tipe karakter
- Karakter generator RAM dengan 8 tipe karakter
- 80×8 display data RAM
- Dapat diinterfacekan ke MPU 8 atau 4
- Dilengkapi fungsi tambahan : *display clear*, *cursor home*, *display ON/OFF*, *cursor ON/OFF*, *display character blink*, *cursor shift*, dan *display shift*.
- Internal Data

- Internal Otomatis, reset pada saat power ON
- + 5 volt PSU Tunggu

Tabel 2-1. Konfigurasi Pin-pin LCD^[12]

PIN NUMBER	SYMBOL	FUNCTION
1	V _{ss}	GND
2	V _{dd}	+ 3V or + 5V
3	V _o	Contrast Adjustment
4	RS	H/L Register Select Signal
5	R/ \bar{W}	H/L Read/Write Signal
6	E	H → L Enable Signal
7	DB0	H/L Data Bus Line
8	DB1	H/L Data Bus Line
9	DB2	H/L Data Bus Line
10	DB3	H/L Data Bus Line
11	DB4	H/L Data Bus Line
12	DB5	H/L Data Bus Line
13	DB6	H/L Data Bus Line
14	DB7	H/L Data Bus Line
15	A/V _{ee}	+ 4.2V for LED (RA = 0); Negative Voltage Output
16	K	Power Supply for B/L (0V)

2.4.2. Interuksi Operasi Dasar

2.4.2.1. Register

Controller LCD mempunyai dua buah register 8 bit yaitu register interuksi (IR) dan register data (RD). IR menyimpan interuksi seperti *display clear*, *cursor shift*, dan *diplay data* (DD RAM) serta *karakter generator* (CG RAM). DR menyimpan data untuk ditulis di DD RAM atau CG RAM atau membaca data dari DD RAM atau CG RAM. Ketika data ditulis di DD RAM atau CG RAM maka DR akan secara otomatis menulis data ke DD RAM atau CG RAM dan data pada DD RAM dan CG RAM hendak dibaca maka alamat data ditulis pada IR sedangkan data alamat dimasukkan melalui DR dan mikroprosesor membaca data dari DR.

Tabel 2-2. Register Seleksi^[12]

RS	R/W	OPERASI
0	0	Seleksi IR, IR Write Display Clear
0	0	Busy Flag (DB7) @counter (DB0 – DB7) Read
1	0	Seleksi DR, DR Write
1	1	Seleksi DR, DR Read

2.4.2.2. Busy Flag

Busy Flag menunjukkan bahwa modul siap untuk menerima instruksi selanjutnya. Sebagaimana yang terlihat pada table register seleksi sinyal akan melalui DB7. Jika RS=0 dan RW=1. Jika bernilai 1 maka modul sedang melakukan kerja internal dan intruksi tidak dapat diterima. Sehingga status dari Flag ini harus diperiksa sebelum melakukan intruksi selanjutnya.

2.4.2.3. Address Counter

AC menunjukkan lokasi memori dalam modul LCD. Pemilihan lokasi alamat itu diberikan lewat regis interuksi (IR). Ketika data ada pada A, maka AC secara otomatis menaikkan atau menurunkan alamat tergantung dari *Entry Mode Set*.

2.4.2.4. Display Data RAM (DD RAM)

Pada LCD masing-masing line mempunyai range alamat tersendiri. Alamat ini diekspresikan dengan bilangan *Hexadecimal*. Untuk itu 1 range alamat berkisar 00H-0FH sedangkan untuk line 2 range alamat berkisar antara 40H-4FH.

2.4.2.5. Character Generator ROM (CG ROM)

CG ROM mempunyai tipe dot matrik 5×7 dan data pada LCD telah tersedia ROM sebagai pembangkit Character dalam kode ASCII.

2.4.2.6. Cracter Generator RAM (CG RAM)

CG RAM digunakan untuk pembuatan karakter tersendiri melalui program.

Tabel 2-3. Fungsi Terminal pada LCD^[12]

Nama Signal	Jumlah Terminal	I/O	Tujuan	Fungsi
DB0 – DB3	4	I/O	MPU	Sebagai lalu lintas data dan intruksi ke atau dari MPU Low Byte
DB4 – DB7	4	I/O	MPU	Sebagai lalu lintas data atau intruksi 2 arah upper byte. DB7 sebagai busy flag
E	1	1	MPU	Sinyal Start (read / write)
R/W	1	1	MPU	Seleksi Sinyal 0 = Write 1 = read
RS	1	1	MPU	Seleksi Register
VLS	1	-	PS	0 = interuksi reg (wr) Busy flag addr counter (rd) 1 = data reg (wr dan rd)
7	1	-	PS	Mengatur Tampilan LCD
Vss	1	-	PS	+ 5 Volt

2.5 Mikrokontroler AT89S52

2.5.1. Pendahuluan

Perbedaan mendasar antara mikrokontroler dan mikroprosesor adalah mikrokontroler selain memiliki CPU juga dilengkapi memori dan *input output* yang merupakan kelengkapan sebagai sistem minimum mikrokomputer sehingga sebuah mikrokontroler dapat dikatakan sebagai mikrokomputer dalam keping tunggal (*Single Chip Mikrokomputer*) yang dapat berdiri sendiri.

Mikrokontroler AT89S52 merupakan mikrokontroler ATMEL yang kompatibel penuh dengan mikrokontroler keluarga MCS-51, membutuhkan daya rendah, memiliki *performance* yang tinggi dan merupakan mikrokomputer 8 bit yang dilengkapi EEPROM (*Electrical Eraseable and Programable Read Only Memory*) dan 4 Kbyte RAM internal dengan kapasitas 256 Kbyte. Program memori yang dapat diprogram ulang dalam system atau menggunakan programmer *Nonvolatile* memori *konvensional*. Dalam sistem mikrokontroler terdapat dua hal yang mendasar, yaitu : perangkat lunak dan perangkat keras yang keduanya saling terkait dan mendukung.

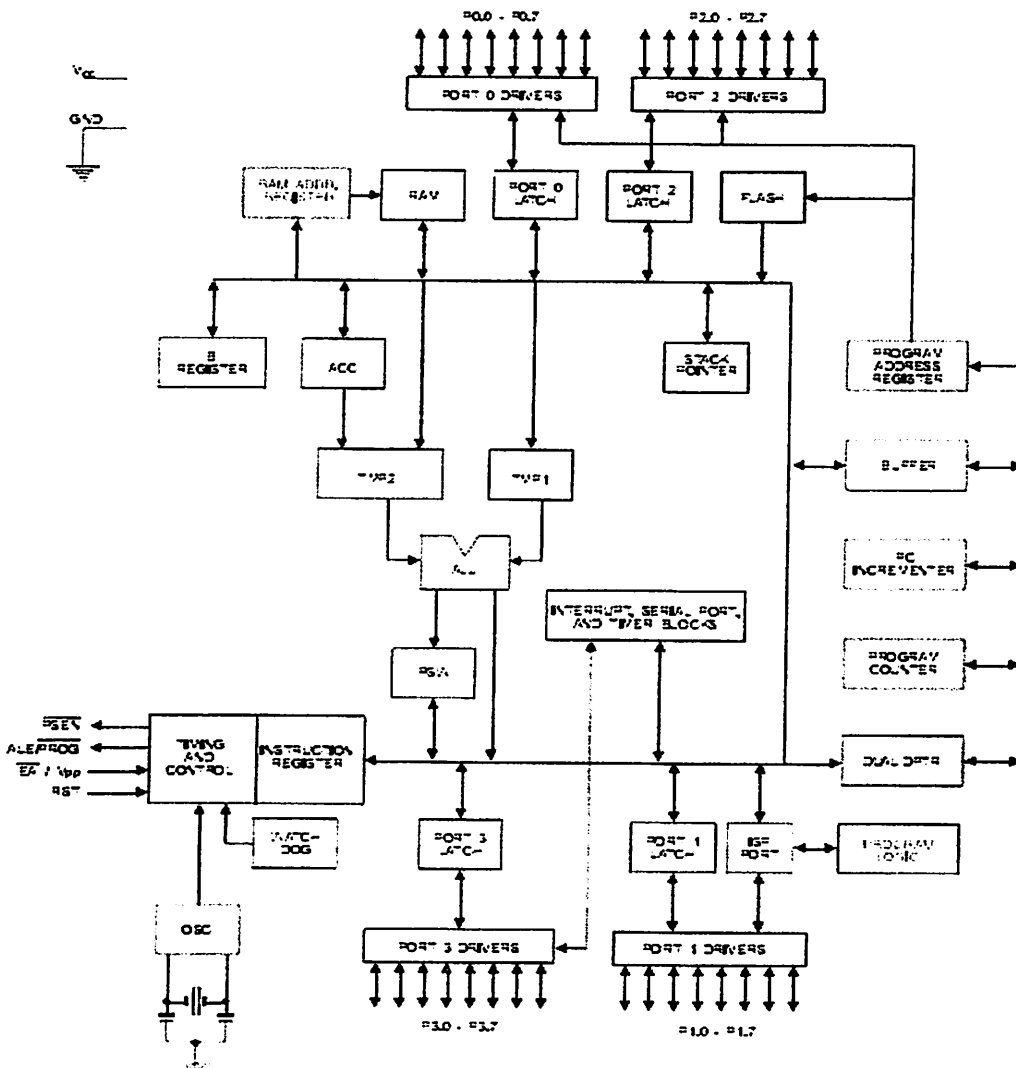
2.5.2. Perangkat Keras Mikrokontroler AT89S52

Secara umum mikrokontroler AT89S52 memiliki:

- 8 bit CPU (*Central Processing Unit*) dengan register A dan B
- 2 *Data pointer* (DPTR)
- 8 bit *Program Status Word* (PSW)
- 4 Kbyte *EPROM* (*Eraseable and Programable Read Only Memory*)
- 256 x 8 bit internal *RAM*
- 32 pin *input / output* tersusun sebagai 4 port
- 2 buah *timer / counter* 16 bit

- 2 Serial Iport Full Duplex yaitu SBUF
- Kecepatan pelaksanaan instruksi per siklus 1 us pada frekuensi clock 12 Mhz
- Watchdog Timer
- 2 buah eksternal interrupt dan 3 buah internal interrupt
- Fleksibel ISP Programming

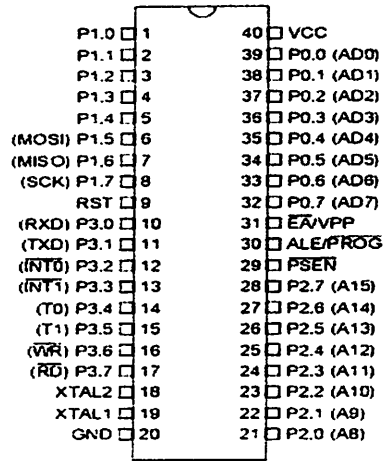
Dengan keistimewaan diatas pembuatan alat menggunakan AT89S52 menjadi lebih sederhana dan tidak memerlukan IC pendukung yang banyak. Adapun blok diagram dari mikrokontroller AT89S52 adalah sebagai berikut:



Gambar 2.10 Blok Diagram AT89S52^[1]

2.5.3 Konfigurasi Kaki-kaki Mikrokontroler AT89S52

Berikut ini adalah bentuk fisik dari AT89S52



Gambar 2.11. Bentuk Fisik AT89S52^[1]

Fungsi-fungsi dari tiap-tiap pin sebagai berikut:

➤ **Pin 40: Vcc**

Dihubungkan dengan sumber tegangan + 5 volt.

➤ **Pin 20 GND**

Dihubungkan dengan *Ground* Rangkaian atau media pertanahan.

➤ **Pin 32-39 : Port 0 (P0.0 – P0.7)**

Merupakan port *input-output* 2 arah, 8 bit dua arah, tanpa *internal pull-up* dan konfigurasi sebagai multipleks bus alamat rendah (A0-A7) dan data selama pengaksesan program *memory eksternal*.

➤ **Pin 1-8 : Port 1 (P1.0 – P1.7)**

Merupakan port *input-output* dua arah dengan *internal pull-up* selain itu port 1 merupakan port yang digunakan sebagai *ISP header*.

Tabel 2.4 Konfigurasi port 1.5 sampai 1.7^[2]

Port Pin	Fungsi
P1.5	MOSI (master data <i>output</i> , slave data <i>input</i> untuk chanel SPI)
P1.6	MISO (master data <i>input</i> , slave data <i>output</i> untuk chanel SPI)
P1.7	SCK (master <i>clock output</i> , slave <i>clock input</i> untuk chanel SPI)

➤ **Pin 21-28 : port 2 (P2.0 – P2.7)**

Merupakan port *input/output* dengan *internal pull-up*. Port ini dapat digunakan sebagai alamat bus *byte* tinggi selama adanya akses ke memori program atau memori data luar. Mengeluarkan *address* tinggi selama pengambilan (*fetching*) program memori *external*.

➤ **Port 3 (P3.0 – P3.7)**

Merupakan port *input-output* dengan *internal pull-up* dimana port 3 juga memiliki fungsi khusus sebagai berikut:

- RXD (P3.0), masukan data serial atau sebagai keluaran data.
- TXD (P3.1), keluaran pengiriman data untuk serial *port* (*asynchronous*) atau sebagai keluaran *clock* (*sybchronous*).
- INT 0 (P3.2), masukan *interrupt* 0.
- INT 1 (P3.3), masukan *interrupt* 1.
- T 0 (P3.4), masukan dari pewaktu/pencacah 0.
- T 1 (P3.5), masukan dari pewaktu/pencacah 1.
- WR (P3.6), sinyal penulisan memori data luar.
- RD (P3.7), sinyal pembacaan memori data luar.

➤ **Pin 9: RST/ VPD**

Merupakan pin *input* yang aktif jika pin aktif tinggi selama dua siklus mesin maka ketika osilator bekerja akan mereset peralatan.

➤ **Pin 30 : ALE (*Address Latch Enable*)**

Pin ALE (aktif tinggi) mengeluarkan pulsa *output* untuk menyangga (*latch*) satu byte alamat rendah selama mengakses ke memori *external*. ALE dapat mengendalikan 8 beban TTL. Pin ini juga merupakan *input* pulsa program yang aktif rendah selama pemrograman EPROM. Pada operasi normal, ALE dikeluarkan pada suatu kecepatan yang konstan yaitu 1/6 dari frekuensi osilator dan dapat digunakan untuk *timing external* tujuan membuat *clock*.

➤ **Pin 29 : PSEN (*Program Strobe Enable*)**

Pin ini aktif rendah yang merupakan *strobe* pembacaan ke *memory external*.

➤ **Pin 18-19 : XTAL 1 & XTAL 2**

Pin XTAL1 merupakan pin input ke penguat osilator internal dan XTAL2 merupakan pin *output* dari penguat osilator internal.

➤ **Pin 31 : EA, VPP (*External Access/Programming Supply Voltage*)**

Dapat diberikan logika rendah (*Ground*) atau logika tinggi (+5V). Jika diberikan logika tinggi maka mikrokontroller akan mengakses program dari ROM internal (EEPROM/ *Flash Memory*), dan jika diberikan logika rendah maka mikrokontroller akan mengakses program dari *memory eksternal*.

2.5.4 Organisasi Memori Mikrokontroler AT89S52

Organisasi memori pada mikrokontoller AT89S52 dapat dibagi menjadi dua bagian besar yaitu memori program dan memori data. Pembagian tersebut didasarkan atas fungsi dari penyimpanan data maupun program. Memori program dapat digunakan untuk menyimpan instruksi-instruksi yang akan dijalankan oleh mikrokontroller, sedangkan memori data digunakan sebagai tempat data yang sedang diolah mikrokontroller.

2.5.4.1 Memori Program

Memori program adalah memori yang berfungsi untuk menyimpan instruksi program. Mikrokontroller AT89S52 memiliki program memori sebesar 4 Kbyte dengan ruang alamat 0000H-0FA0H. Jika alamat-alamat program lebih tinggi dari pada 0FA0H dimana melebihi kapasitas ROM *internal*, menyebabkan AT89S52 secara otomatis mengambil kode byte dari program memori *external*. Kode *byte* juga dapat diambil hanya dari memori external dengan alamat 0000H-FFFFH dengan cara pin EA ke *ground*.

2.5.4.2. Memori Data

Memori data adalah tempat penyimpanan data yang bersifat sementara. Data yang dimaksudkan disini adalah data yang diperoleh dari GPS seperti koordinat. Sehingga pada memori data bersifat *volatile* yaitu data akan hilang bila tidak dicatu. Memori data lebih dikenal dengan nama RAM (*Random Acces Memory*) yaitu dapat dilakukan pembacaan dan penulisan data pada alamat yang tersedia. Memori data pada AT89S52 dapat memiliki pengalamatan sampai 64 Kbyte (\$00000H sampai \$0FFFFH) yang didapat dari memori data eksternal, sebagaimana yang ditunjukkan

pada gambar berikut. Pada AT89S52 mempunyai 256 byte RAM *internal* ditambah *Special Function Register (SFR)*.

0FBH								0FFH
0FCH	B 00000000							0F7H
0E8H								0EFH
0E9H	ACC 00000000							0E7H
0D8H								0DFH
0D9H	PSW 00000000							0D7H
0C8H								0CFH
0C9H								0C7H
0B8H	IP xx000000							0BFH
0B9H	P3 11111111							0B7H
0ABH	IE 0x000000							0AFH
0ACH	P2 11111111		AUXR1 xxxxxxxx0				WDTRST xxxxxxxx	0A7H
98H	SCON 00000000	SBUF xxxxxxxx						9FH
99H	P1 11111111							97H
88H	TCON 00000000	TMOD 00000000	TLO 00000000	TL1 00000000	TH0 00000000	TH1 00000000	AUXR xxx00xx0	8FH
89H	P0 11111111	SP 00000111	DP0L 00000000	DP0H 00000000	DP1L 00000000	DP1H 00000000	PCON 0xxx0000	87H

Gambar 2.12 AT89S52 SFR dan Reset Values^[1]

Pada Keluarga MCS-51, ruang memori data eksternal terbagi menjadi 3 blok yang disebut *lower 128*, *upper 128* dan ruang SFR sebagaimana yang ditunjukkan pada gambar diatas.

Pada *lower 128* lokasi memori terbagi menjadi tiga bagian yaitu :

1. *Register Banks 0-3*

Lokasi *register bank* dimulai dari alamat 00H-1FH yang terdiri dari 32 byte. *Register bank* ini terdiri dari 4 buah register 8 bit yang dapat dipilih melalui pengaturan *Program Status Word Register*.

2. *Bit Addressable*

Terdiri dari 16 byte yang dimulai dari 20H-2FH. Masing-masing dari 128 bit lokasi ini dapat dialamati secara langsung yaitu dari \$00H hingga \$7FH.

3. *Scratch pad Area*

Lokasinya dari alamat \$30H sampai \$7FH atau sebanyak 80 byte yang dapat digunakan sebagai data bagi RAM.

Pada 128 byte diatas ditempati oleh suatu register yang memiliki fungsi khusus yang disebut *Special Function Register (SFR)*. Ruang dari register fungsi khusus ini adalah dari 80H sampai FFH. Berikut ini adalah contoh isi dari vector alamat pada SFR.

1. *Accumulator (ACC)* dan register B

Kedua register tersebut digunakan untuk operasi perkalian dan pembagian.

2. *Program Status Word (PSW)*

Register ini meliputi bit-bit : CY (*Carry*), AC (*Auxiliary Carry*), FO (sebagai *Flag*), RS0 dan RS1 (untuk pemilihan *register bank*), OV (*overflow*) dan parity (*Parity Flag*).

3. *Stack Pointer (SP)*

SP merupakan suatu register yang digunakan untuk alamat. Register ini berguna apabila digunakan suatu *routine* pada program.

4. *Data pointer High (DPH)* dan *Data pointer Low (DPL)*

DPTR adalah suatu register yang digunakan untuk pengalamatan tidak langsung. Register ini digunakan untuk memori program baik internal maupun eksternal juga digunakan untuk alamat eksternal data. DPTR ini dikontrol oleh 2 buah register 8 bit yaitu DPH dan DPL.

2.5.4.3. Sistem Interupsi

Mikrokontroler AT89S52 mempunyai 9 buah sumber interupsi yang dapat mengakibatkan permintaan interupsi, yaitu INT0, INT1, T0, T1 port serial dan beberapa port lainnya. Saat terjadi interupsi mikrokontroler secara otomatis akan menuju ke *subrutin* pada alamat tersebut. Setelah interupsi *service* selesai dikerjakan, mikrokontroler akan mengerjakan program semula. Sumber interupsi *external* adalah INT0, INT1, dimana kedua interupsi *external* ini akan aktif pada transmisi rendah selain itu juga ada *Timer/Counter 0*, *Timer/Counter 1* dan interupsi dari port serial (*receiver*). Interupsi serial dibangkitkan dengan melakukan operasi OR pada R1 dan T1. Tiap-tiap sumber interupsi dapat *di-enable* atau *di-disable* secara *software*. Tingkat prioritas semua sumber interupsi dapat diprogram sendiri-sendiri dengan *set* atau *clear* bit pada SFRS IP (*Interrupt Priority*).

Tabel 2.5 Alamat Sumber Interupsi^[2]

Sumber Interupsi	Alamat Awal
<i>Power On Reset</i>	0000H
<i>Interrupt</i> luar 0 (INT 0)	0003H
Pewaktu/Pencacah 0 (T0)	000BH
<i>Interrupt</i> luar 1 (INT 1)	0013H
Pewaktu/Pencacah 1 (T1)	001BH
Port 110 Serial	0023H

Register yang berperan dalam mengatur aktif tidaknya interupsi adalah *interrupt enable register*, susunan dari bit-bit beserta kegunaannya adalah:

Tabel 2.6 Kegunaan *Interrupt Enable Register*^[2]

Data	Simbol	Posisi	Fungsi/Arti
D0	EX0	IE.0	Diatur secara <i>software</i> untuk interupsi dari INT1
D1	ET0	IE.1	Diatur secara <i>software</i> untuk interupsi dari <i>timer/counter 1</i>
D2	EX1	IE.2	Diatur secara <i>software</i> untuk interupsi dari INT1
D3	ET1	IE.3	Diatur secara <i>software</i> untuk interupsi dari <i>timer/counter 1</i>
D4	ES	IE.4	Untuk mengatur <i>enable</i> atau <i>disables</i> atau interupsi R1/T1
D5	-	IE.5	-
D6	-	IE.6	-
D7	EA	IE.7	Jika diatur 0 maka semua interupsi <i>di-disable</i> , jika diatur 1 maka interupsi diatur <i>di-disable</i> atau <i>di-enable</i> menurut masing-masing bit.

2.5.4.4. Timer/Counter

Pengendalian kerja dari *timer/counter* dilakukan dengan pengaturan register yang berhubungan dengan kerja dari *timer/counter* yaitu melalui sebuah *timer/counter mode control*. Untuk mengaktifkan *timer/counter* yang meliputi penentuan fungsi sebagai *timer* atau sebagai *counter* serta pemilihan *mode* operasi dapat diatur melalui TMOD. Konfigurasi dari register TMOD seperti yang ditunjukkan dalam table berikut ini:

Tabel 2.7 Register TMOD^[2]

Data	Simbol	Posisi	Fungsi/Arti
D0	IT0	TCON.0	<i>Interrupt 0 type control bit</i>
D1	IE0	TCON.1	<i>External interrupt 0 edge flag</i>
D2	IT1	TCON.2	<i>Interrupt type 1 control bit. Diatur oleh software untuk menentukan aktif low atau high trigger dari external</i>
D3	IE1	TCON.3	<i>External interrupt 1 edge flag. Diatur oleh hardware ketika external interrupt terdeteksi dan nol-kan melalui software ketika interrupt diproses.</i>
D4	TR0	TCON.4	<i>Timer 0 dan control bit. Diatur oleh software ketika timer counter 0.</i>
D5	TF1	TCON.5	<i>Timer 0 overflow flag control bit. Diatur oleh software ketika timer/counter 0 overflow.</i>
D6	TR1	TCON.6	<i>Timer 1 control bit. Diatur oleh software ketika timer counter 0.</i>
D7	TF1	TCON.7	<i>Timer 1 overflow flag control bit. Diatur oleh software ketika timer/counter 0 overflow.</i>

Tabel 2.8 Timer/Counter Mode Control Register^[2]

Data	Simbol	Fungsi/Arti
D0	<i>Timer 0; M0 (0)</i>	Untuk memilih mode <i>timer</i> .
D1	<i>Timer 0; M1 (0)</i>	Untuk memilih mode <i>timer</i> .
D2	<i>Timer 0; C/T (0)</i>	1 = <i>Counter</i> & 0 = <i>Timer</i> .
D3	<i>Timer 0; GATE (0)</i>	<i>Timer</i> akan berjalan jika bit di <i>set</i> dan INT0 (untuk <i>timer 0</i>) atau INT1 (untuk <i>Timer 1</i>).
D4	<i>Timer 1; M0 (1)</i>	Untuk memilih mode <i>timer</i>
D5	<i>Timer 1; M1 (1)</i>	Untuk memilih mode <i>timer</i>
D6	<i>Timer 1; C/T (0)</i>	1 = <i>Counter</i> & 0 = <i>Timer</i>
D7	<i>Timer 1; GATE (1)</i>	<i>Timer</i> akan berjalan jika bit di <i>set</i> dan INT0 (untuk <i>Timer 0</i>) atau INT1 (untuk <i>Timer 1</i>)

Tabel 2.9 Mode Operasi Timer/Counter^[2]

M1	M0	Operating Mode
0	0	Timer 13 bit
0	1	Timer/Counter 16 bit
1	0	8 bit Auto reload Timer/Counter
1	1	TL0 dari <i>Timer</i> adalah 8 bit <i>Timer/Counter</i> dikendalikan oleh control bit <i>Timer 0</i> . TH0 adalah 8 bit yang dikendalikan oleh <i>Timer 1 control bit</i>

2.5.4.5. Metode Pengalamatan

Metode pengalamatan yang digunakan pada MCS-51 terbagi menjadi dua jenis, pengalamatan langsung dan pengalamatan tidak langsung, (Moh.Ibnu Malik,1997:36)

➤ *Pengalamata Tidak Langsung*

Operasi pengalamatan tidak langsung menunjukkan sebuah register yang berisi alamat memori yang akan digunakan dalam suatu operasi.

Lokasi yang nyata tergantung dari isi register saat interuksi dijalankan.

Untuk pengalamatan tidak langsung digunakan symbol @. Misalnya:

- ADD A, @R0 : tambahkan isi R0 dengan Acc dan hasil di Acc
- DEC @R1 : kurangi isi dari alamat R1

➤ *Pengalamatan Langsung*

Pengalamatan langsung dilakukan dengan memberikan nilai ke suatu register secara langsung. Untuk melakukan hal tersebut digunakan tanda #. Misalnya :

- MOV A,#01 H : isi Acc dengan data 01H
- MOV DPTR,#19H : isi DPTR data 19H

Pengalamatan data langsung dari 0 sampai 127 akan mengakses RAM internal, sedangkan pengalamatan dari 128 sampai 255 akan mengakses register perangkat keras. Misalnya :

- MOV P3,A : pindahkan isi Acc ke alamat port 3 (BOH)
- INC 50 : naikan lokasi 50 (desimal) dalam memori

2.5.5 Bahasa Assembler MCS-51

Bahasa *Assembler* digunakan dalam setiap operasi CPU dalam bentuk bahasa simbol yang disusun berurutan dalam pernyataannya. Masing-masing pernyataan akan diterjemahkan ke dalam interuksi bahasa mesin atau sering disebut *operation code/opcode*. Dalam penulisan bahasa mesin ini, terdapat berbagai macam kelompok interuksi, diantaranya:

- Perpindahan data

Interuksi ini digunakan untuk memindahkan data antara register, memori, register-memori, antara muka register dan muka memori.

Contohnya :

MOV A,R0 : memindahkan isi register R0 ke Acc.

MOV A,@R0 : memindahkan isi alamat R0 ke Acc.

- Operasi Aritmatika

Interuksi ini melaksanakan operasi aritmatika yang meliputi penjumlahan, pengurangan, perkalian, maupun pembagian.

Contohnya :

ADD A,#data : menambahkan Acc dengan data.

ADC A,#data : menambahkan Acc dengan data dan carry.

INC R6 : menambah isi R6 dengan 1.

DEC R7 : mengurangi isi R7 dengan 1.

MUL AB : mengalikan isi Acc dengan isi register B.

DIV AB : membagi isi Acc dengan isi register B.

- Operasi percabangan

Interuksi ini mengubah urutan normal pelaksanaan suatu program untuk melaksanakan pada lain tempat yang kita perlukan pada saat itu. Contohnya :

1. CJNE (*Compare Jump Not Equal*)

Interuksi ini membandingkan isi lokasi memori tertentu dengan isi Acc, jika sama interuksi ini selanjutnya akan dieksekusi. Jika tidak sama eksekusi akan kembali ke alamat kode yang telah di tunjuk.

2. JB (*Jump If Bit Set*)

Interuksi ini akan menguji suatu alamat bit isi satu, eksekusi akan menuju ke alamat kode dan jika tidak interuksi akan dilanjutkan.

3. JNB (*Jump Bit Not Set*)

Interuksi ini menguji suatu alamat bit. Jika berisi 0 maka eksekusi akan menuju ke alamat kode. Jika berisi 1 maka interuksi selanjutnya yang akan dieksekusi.

2.6 Komunikasi Serial

Dalam dunia komunikasi ada dua cara pemindahan data yaitu secara paralel dan serial. Perbedaan dua cara tersebut terdapat pada jumlah bit yang dipindahkan. Setiap data merupakan kelompok dari bit-bit tersebut, dikenal dengan istilah *byte* yang merupakan kelompok yang terdiri dari 8 bit, serta *word* yaitu kelompok yang terdiri dari 16 bit.

Pada komunikasi serial terjadi pemindahan data satu bit pada satuan waktu, sedangkan pada komunikasi paralel terjadi pemindahan data secara berurutan dari kelompok bit pada satuan waktu.

Ditinjau dari arah komunikasi data, dikenal ada 3 cara, yaitu : komunikasi *simplex*, *half duplex*, dan *full duplex*.

1. Komunikasi *simplex* adalah sistem komunikasi data yang arah perpindahan datanya satu arah saja.
2. Komunikasi *half duplex* adalah sistem komunikasi data yang arah perpindahan datanya dua arah, namun proses pemindahannya tidak bersamaan.
3. Komunikasi *full duplex* adalah sistem komunikasi yang arah perpindahan datanya dua arah secara bersamaan.

Yang dimaksud dengan komunikasi secara serial adalah komunikasi dengan memanfaatkan hanya satu saluran sinyal untuk pengiriman dan penerimaan data. Untuk data digital yang merupakan serangkain bit maka data dikirim bit per bit. Pengiriman akan dimulai dari LSB (*Least Significant Bit*), dan diakhiri dengan MSB (*Most Signifikan Bit*). Setiap karakter yang dikirimkan disusun sesuai dengan suatu urutan bit tertentu.

2.7 Metode Transmisi Serial

Ada dua metode trasmisi serial yang digunakan yaitu :

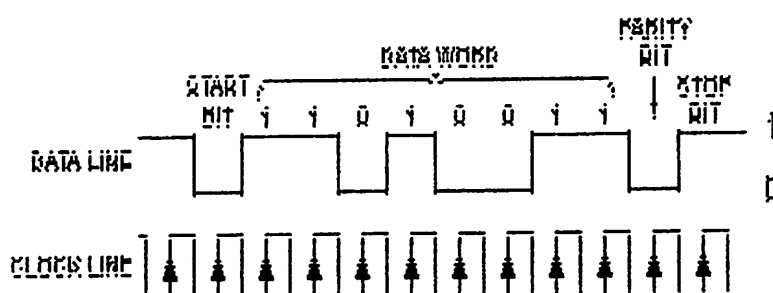
1. Komunikasi Serial Sinkron (*Synchronous Serial Communication*)

Pada komunikasi metode ini, pengiriman dan penerimaan akhir komunikasi adalah secara sinkron atau serempak menggunakan *clock* yang tepat, waktu dalam periode terpisah tiap bit dengan pengecekan *clock* pada penerimaan akhir. Salah

satu aspek penting pada metode ini jika akhir dari komunikasi kehilangan sinyal *clock*, maka komunikasi akan *terminasi*.

2. Komunikasi Serial *Asinkron* (*Asynchronous Serial Communication*)

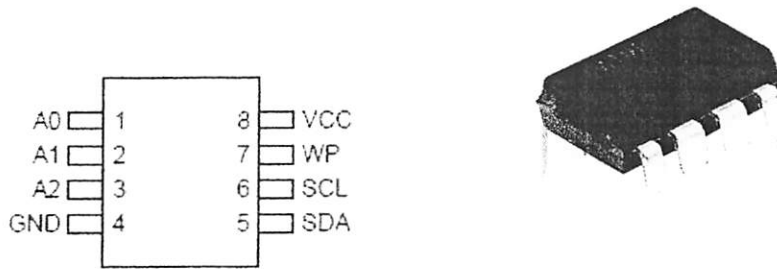
Komunikasi metode ini sinyal *clock* pengirim dan penerima data tidak harus *sikron* untuk tiap bit data yang ditransmisikan. Karena itu sinyal *clock* tidak perlu dikirimkan. Hal ini bukan berarti tidak ada usaha *sinkronisasi* sama sekali antara pengirim dan penerima data. *Sinkronisasi* dilakukan dengan pengiriman sebuah bit awal (*bit start*) sebelum serangkaian bit data dikirimkan. Agar selama penerimaan data, *clock* penerima tidak meleset jauh, maka jumlah bit dalam satu kali pengiriman dibatasi. Umumnya batasan maksimal 8 bit data (1 *byte*) dalam satu kali pengiriman. Oleh sebab itu komunikasi serial *asinkron* biasanya digunakan pada perangkat yang menghasilkan data dengan laju rendah. Kemudian untuk menjamin adanya transisi sinyal minimal satu kali diantara dua pengiriman rangkaian bit data, maka pengiriman data perlu diakhiri dengan bit penutup (*bit stop*) yang memiliki polaritas berbeda dengan bit pembuka.



Gambar 2.13 Asynchronous Serial Data Frame (8E1)^[1]

Bagian penting yang lain pada setiap sinyal serial *asinkronous* adalah *bit rate* pada data yang dikirimkan. Kecepatan data yang diterima didasarkan pada kecepatan minimum 300bps (*bit per second*).

2.8 Serial Electrically Erasable an Programmable Read Only Memory (Serial EEPROM) AT24C16



Gambar 2.14 bentuk fisik AT24C16

Serial EEPROM yang digunakan pada tugas akhir ini adalah AT24C16 (serial EEPROM 64kbit) berikut pembahasan jenis serial EEPROM tersebut.

2.8.1. Fitur-fitur yang dimiliki serial EEPROM

Fitur-fitur yang dimiliki serial EEPROM ini sebagai berikut :

- Waktu akses baca sangat cepat yaitu 50ms
- Proteksi data secara *hardware*
- Menggunakan dua jalur dalam komunikasi secara serial (SDA dan SCL)
- Transfer data *protocol* bersifat *bidirectional*
- Waktu akses tulis yang sangat cepat
- *Page write cycle time* maksimum 10ms
- 1-8 byte *page write operation*
- Beroperasi pada *low voltage* dan *standard voltage*
 - Versi 2.7 ($V_{cc} = 2.7V - 5.5V$)
 - Versi 1.8 ($V_{cc} = 1.8V - 5.5V$)
- Keandalan teknologi yang tinggi

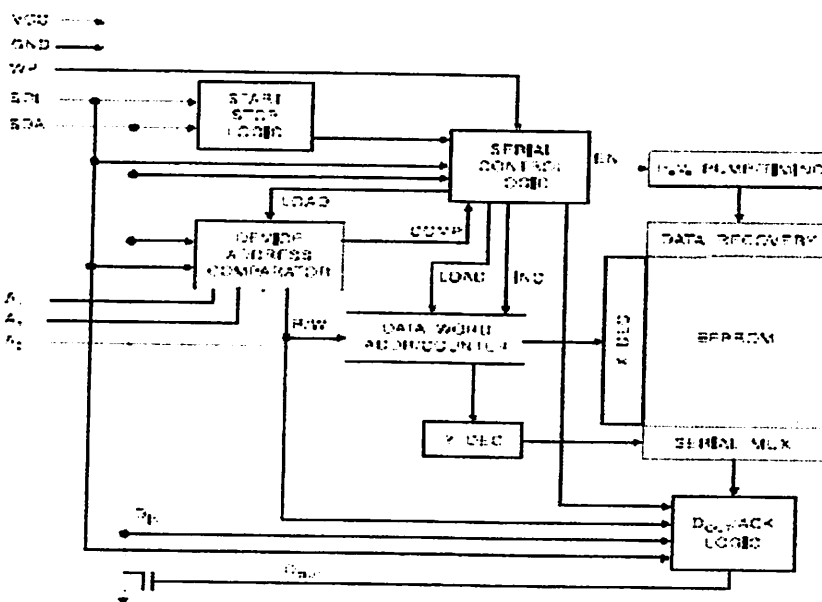
- Daya tahan pada AT24C16 adalah 1 juta Write cycles.
 - Lama penyimpan data selama 100 tahun
 - ESD protection lebih dari 3.000V
- Bekerja pada temperature industrial dan komersial.
 - Maksimal 8 Device address inputs

2.8.2 konfigurasi Pin

Table 2.10 konfigurasi pin

Pin Name	Function
A0 - A2	Address Inputs
SDA	Serial Data
SCL	Serial Clock Input
WP	Write Protect
NC	No Connect
GND	Ground
VCC	Power Supply

2.8.3 Block diagram AT24C16



Gambar 2.15 Block diagram AT24C16

Konfigurasi pengoperasian pin sebagai berikut :

- ❖ *Start condition* perubahan transisi turun pada Pin SDA dan pada saat SCL dalam keadaan *high* akan menandakan sebuah *start condition* yang mengawali proses *read/write* pada AT24C16
- ❖ *Stop condition* perubahan transisi naik pada pin SDA dan pada saat pin SCL dalam keadaan *high* akan menandakan sebuah *stop condition* (yang mengakhiri proses *read/write* pada AT24C16)
- ❖ *Clock dan transisi data*. Data pada pin SDA hanya boleh berubah pada waktu kondisi SCL *low*. Jika perubahan SCL pada waktu SCL *high*, akan menandakan *start condition* atau *stop condition*.

2.9 Keypad

Keypad yang dipakai dalam tugas akhir ini adalah keypad push button 2 pin. Keypad di desain aktif low. Prinsip kerjanya adalah keypad akan aktif low, jika ditekan. keypad yang dipakai sebanyak 6 buah, yang masing-masing keypad mempunyai fungsi sebagai berikut:

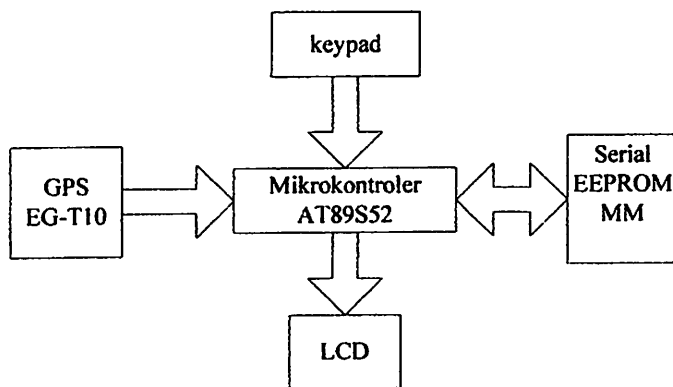
- Keypad tombol COR: *adres* untuk melakukan *record (write to EEPROM)*
- Keypad tombol MEN: *adres* untuk melakukan *erase (erase data from EEPROM)*
- Keypad tombol anak panah atas : : untuk menambah *counter record/erase* ke *counter record/erase* berikutnya
- Keypad tombol anak panah bawah: untuk kembali ke *counter record/erase* ke *counter record/erase* berikutnya
- Keypad tombol ENT: sebagai tombol enter
- Keypad tombol CAN: sebagai tombol CANSE

Sebab bubu yang mereka tebarkan terkadang ada yang terlupa posisi ditebarkan bubu tersebut.

Untuk mengatasi permasalahan tersebut, maka dibuat suatu alat yang bernama Aplikasi GPS Pada Kapal Penangkap Ikan. Alat ini dapat merecord dan menghapus posisi bubu yang telah ditebarkan. Alat ini menggunakan modul GPS untuk mengetahui koordinat lintang dan bujur tempat ditebarkan bubu oleh nelayan. Alat ini sangat membantu nelayan untuk mengetahui posisi bubu yang telah ditebarkan (terekam didalam memory).

Berikut ini adalah blok diagram sistem Aplikasi GPS Pada Kapal Penangkap Ikan

3.1 Blok Diagram Sistem



Gambar 3.2 block diagram system

Data yang dihasilkan oleh GPS berupa data serial yang dikeluarkan pada pin TX. Kemudian data tersebut dibaca oleh mikrokontroler AT89S52 dan diolah untuk dijadikan koordinat yang akan ditampilkan pada LCD. Pada mikrokontroler juga dipasang 6 keypad yaitu keypad untuk *record*, keypad untuk *erase*, keypad untuk *back record* (kembali ke counter record sebelumnya), keypad untuk *back erase* (kembali ke counter erase sebelumnya), keypad untuk *forward record* (menambah nilai counter record ke counter record sesudahnya), keypad untuk *forward erase* (menambah nilai counter erase ke counter erase sesudahnya). Ketika keypad *record* ditekan, maka mikrokontroler akan *me-record*

BAB III

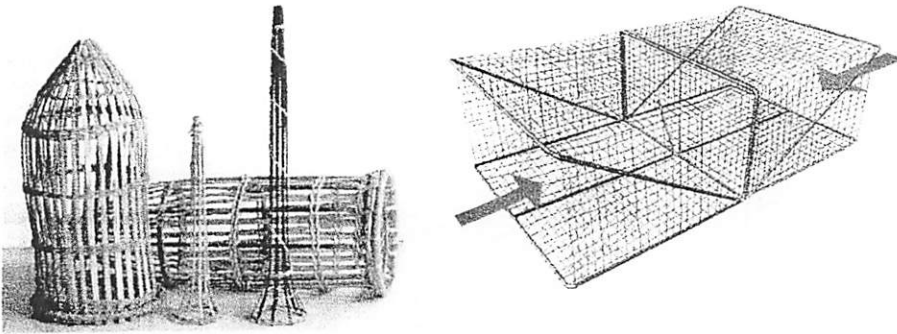
PERANCANGAN SISTEM

3.1 Perancangan Sistem

Perancangan sistem terbagi menjadi beberapa bagian yaitu :

- Perancangan rangkaian pada GPS
- Perancangan rangkain mikrokontroller
- Perancangan rangkaian serial EEPROM
- Perancangan rangkaian keypad
- Perancangan rangkaian LCD 20 Char x 4 Line.

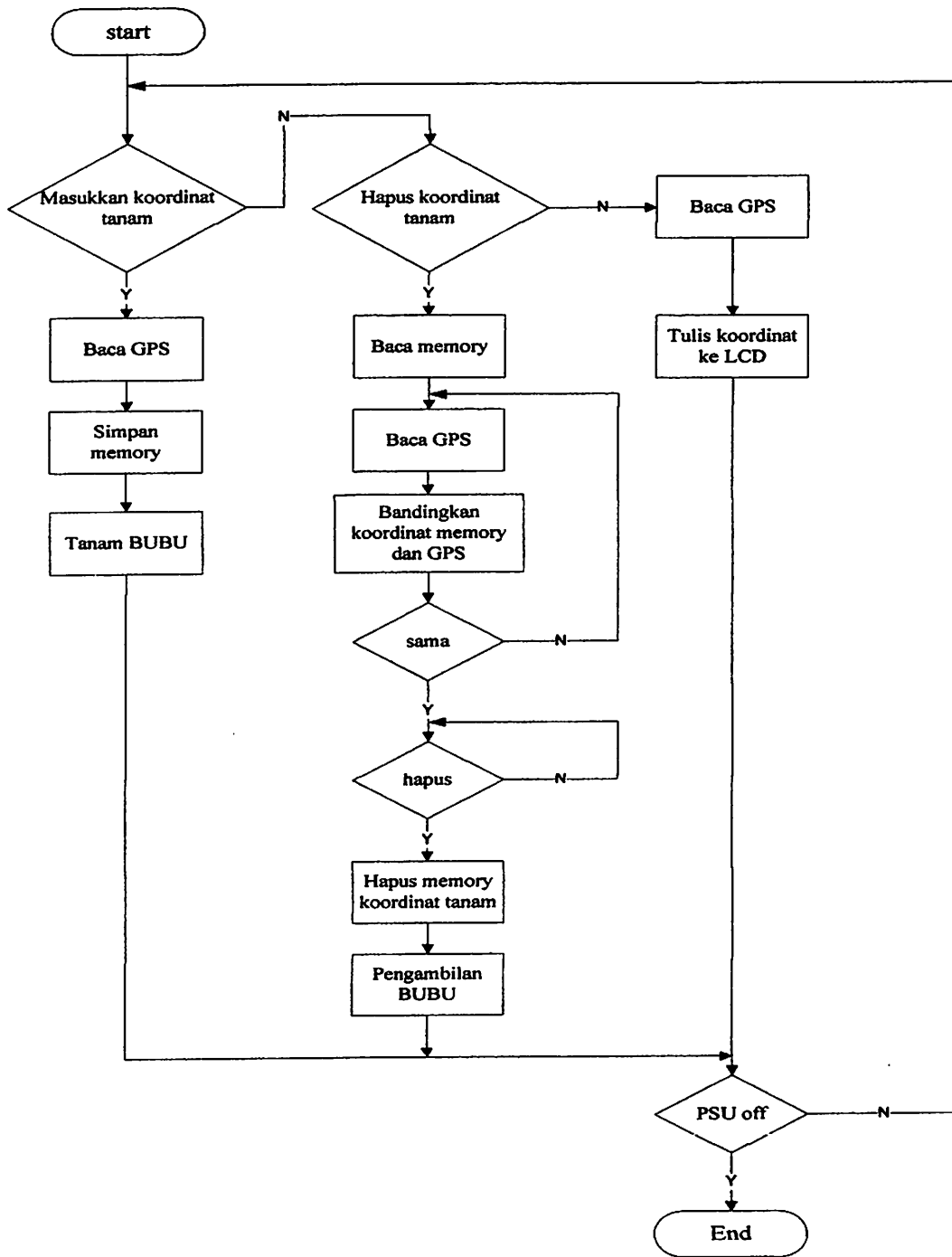
Berdasarkan web page pusat informasi pelabuhan, pada umumnya kapal penangkap ikan memiliki kapasitas untuk menampung 50 bubu kosong . bubu tersebut memiliki panjang 1 meter dan diameter 80cm seperti pada gambar 3.1



Gambar 3.1 bubu

Pada awal penangkapan ikan, bubu kosong yang berada pada kapal penangkap ditebarkan pada daerah tempat berkumpulnya ikan-ikan. Kemudian dilakukan pengambilan bubu lama yang sudah ditebarkan pada hari sebelumnya sebanyak 50 buah. Pada proses pengambilan bubu, para nelayan seringkali mengalami kesulitan untuk mencari dimana posisi bubu yang telah ditebarkan dan membutuhkan waktu yang lama untuk mencari bubu yang telah ditebar.

3.3.2 Flow Chart



Gambar 3.12 flow chart

3.3. Perangkat Lunak

Perangkat lunak yaitu pada program mikrokontroler menggunakan bahasa *assembly*.

3.3.1. Mikrokontroler

Untuk pemakaian mikrokontroler dalam suatu sistem, perlu direncanakan perangkat lunaknya yang dapat mengatur sistem tersebut. Perangkat lunak disini adalah susunan perintah-perintah (program) didalam memori yang harus dilaksanakan oleh mikrokontroler. Didalam suatu mikro memori merupakan suatu fasilitas utama karena disini terdapat perintah yang harus dilakukan oleh mikro. Memori disini dapat dibedakan menurut fungsinya menjadi memori program dan memori data.

Perencanaan perangkat lunak (software) didasarkan pada perencanaan perangkat keras yang telah dibuat sebelumnya, untuk mendapatkan sistem kerja yang diharapkan. Software dari alat terdapat pada diagram alirnya sebagai berikut :

```

mov     Dly0,#100
lcall  delay0
ret

```

4.3.4 Program untuk mengakses keypad

```

scnkpd: mov     R0,#10
        lcall  delay0
col1:   mov     P1,#11111110b
        mov     A,P1
c1b1:   cjne   A,#11101110b,c1b2
        mov     R0,#1
c1b2:   cjne   A,#11011110b,c1b3
        mov     R0,#2
c1b3:   cjne   A,#10111110b,c1b4
        mov     R0,#3
c1b4:   cjne   A,#01111110b,col2
        mov     R0,#13
;
col2:   mov     P1,#11111101b
        mov     A,P1
c2b1:   cjne   A,#11101101b,c2b2
        mov     R0,#4
c2b2:   cjne   A,#11011101b,c2b3
        mov     R0,#5
c2b3:   cjne   A,#10111101b,c2b4
        mov     R0,#6
c2b4:   cjne   A,#01111101b,col3
        mov     R0,#14
;
col3:   mov     P1,#11111011b
        mov     A,P1
c3b1:   cjne   A,#11101011b,c3b2
        mov     R0,#7
c3b2:   cjne   A,#11011011b,c3b3
        mov     R0,#8
c3b3:   cjne   A,#10111011b,c3b4
        mov     R0,#9
c3b4:   cjne   A,#01111011b,col4
        mov     R0,#15
;
col4:   mov     P1,#11110111b
        mov     A,P1
c4b1:   cjne   A,#11100111b,c4b2
        mov     R0,#11
c4b2:   cjne   A,#11010111b,c4b3
        mov     R0,#0
c4b3:   cjne   A,#10110111b,c4b4
        mov     R0,#12
c4b4:   cjne   A,#01110111b,back
        mov     R0,#16
back:   ret
;
tg_tkn: lcall  scnkpd
        lcall  delay0
tg_tk0: cjne   R0,#16,tg_tk1
        ljmp  tg_tkn
tg_tk1: cjne   R0,#15,tg_tk2
        ljmp  tg_tkn

```

```

        setb    ISDA                ;\
        clr     ISDA                ; | C=1, D=1, D=0, C=0
        clr     ISCL                ;/
        ret

;
i2cstp: clr     ISDA                ; i2c stop
        setb    ISCL                ;\ D=0, C=1, D=1, C=0
        setb    ISDA                ;/
        clr     ISCL
        ret

```

4.3.3 Program untuk mengakses LCD

```

line1:  mov     P0,#080h
        lcall  w_ins
        ret

;
line2:  mov     P0,#0C0h
        lcall  w_ins
        ret

;
line3:  mov     P0,#094h
        lcall  w_ins
        ret

;
line4:  mov     P0,#0D4h
        lcall  w_ins
        ret

;
line5:  mov     P0,#0F0h
        lcall  w_ins
        ret

;
tulis:  clr     A
        lcall  wr_chr
        inc   DPTR
        djnz  Char,tulis
        ret

;
wr_chr: movc   A,@A+DPTR
        mov   P0,A
        lcall w_chr
        ret

;
w_ins:  clr     Enbl
        clr     Rest
        setb    Enbl
        clr     Enbl
        mov     Dly0,#100
        lcall  delay0
        ret

;
w_chr:  clr     Enbl
        setb    Rest
        setb    Enbl
        clr     Enbl

```

4.3.2 Program Akses memory (I2C protokol) at24C16

```

wr_mem: mov     A,Adbm                ; AT24C16 write address
        lcall  adrtx
        mov   A,Admm                ; address hardware memory
        lcall dtatx
        mov   A,Dtmm                ; data memory
        lcall dtatx
        lcall i2cstp                ; i2c stop
        lcall wt_wr
        ret

;
wt_wr:  mov     Dly0,#000
        mov     Dly1,#025
wtwr:   lcall  delay0
        djnz   Dly1,wtwr
        ret

;
adrtx:  lcall  i2cstr                ; send address
        lcall  putbit
        ret

;
dtatx:  lcall  putbit                ; send data
        ret

;
dtarx:  lcall  getbit                ; get data
        ret

;
putbit: mov     R6,#8
putbt:  RLC     A
        mov   ISDA,C
        setb ISCL
        clr  ISCL
        djnz R6,putbt
        setb ISDA
        lcall getack
        ret

;
getbit: mov     R6,#8
getbt:  setb   ISCL
        mov   C,ISDA
        RLC   A
        clr  ISCL
        djnz R6,getbt
        setb ISDA
        ret

;
getack: setb   ISDA                ; tunggu ack
        setb  ISCL                ; \
ackbit: mov   C,ISDA                ; | D=1, C=1
        jc   ackbit                ; | tunggu D=0, C=0
        clr  ISCL                ; /
        ret

;
givack: clr    ISDA                ; kirim ack
        setb  ISCL                ; \
        clr  ISCL                ; | D=0, C=1, C=0, D=1
        setb ISDA                ; /
        ret

;
i2cstr: setb   ISCL                ; i2c start

```

```

    mov     Ch0D,R7
    lcall  bc_srl
    lcall  bc_srl
    lcall  bc_srl
    lcall  bc_srl
    mov     Ch0E,R7          ; asli data bujur (E)
    ret

;
tlgps0: mov     P0,Ch00
    lcall  w_chr
    mov     P0,Ch01
    lcall  w_chr
    mov     P0,Ch02
    lcall  w_chr
    mov     P0,Ch03
    lcall  w_chr
    mov     P0,#'.'
    lcall  w_chr
    mov     P0,Ch04
    lcall  w_chr
    mov     P0,Ch05
    lcall  w_chr
    mov     P0,#' '
    lcall  w_chr
    mov     P0,Ch06
    lcall  w_chr
    mov     P0,#' '
    lcall  w_chr
    mov     P0,Ch07
    lcall  w_chr
    mov     P0,Ch08
    lcall  w_chr
    mov     P0,Ch09
    lcall  w_chr
    mov     P0,Ch0A
    lcall  w_chr
    mov     P0,Ch0B
    lcall  w_chr
    mov     P0,#'.'
    lcall  w_chr
    mov     P0,Ch0C
    lcall  w_chr
    mov     P0,Ch0D
    lcall  w_chr
    mov     P0,#' '
    lcall  w_chr
    mov     P0,Ch0E
    lcall  w_chr
    ret

```


4.3.1 Bahasa pemrograman yang digunakan oleh mikrokontroller AT89S52

Pada bagian ini akan dibahas beberapa program yang dipakai dalam sistem ini

antara lain:

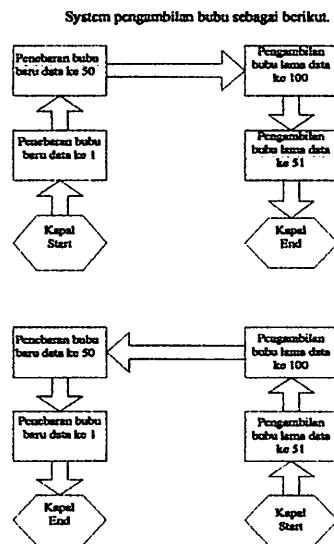
- Program untuk baca GPS dan Tulis GPS
- Program Akses memory (I2C protokol) at24C16
- Program untuk mengakses LCD
- Program untuk mengakses keypad

4.3.2 Program untuk baca GPS dan Tulis GPS

```
bc_rmc: lcall    bc_fbk                ;\  
        cjne    R7, #'R', bc_rmc      ; | baca gps  
        mov    Cntr, #16              ; | yang diawali character R  
bcrmc:  lcall    bc_srl                ; | 17 character dihilangkan  
        djnz   Cntr, bcrmc            ;/  
        lcall   bc_srl  
        mov    Ch00, R7  
        lcall   bc_srl  
        mov    Ch01, R7  
        lcall   bc_srl  
        mov    Ch02, R7  
        lcall   bc_srl  
        mov    Ch03, R7  
        lcall   bc_srl  
        lcall   bc_srl  
        mov    Ch04, R7  
        lcall   bc_srl  
        mov    Ch05, R7  
        lcall   bc_srl  
        lcall   bc_srl  
        lcall   bc_srl  
        lcall   bc_srl  
        mov    Ch06, R7                ; asli data lintang (S)  
        lcall   bc_srl  
        lcall   bc_srl  
;  
        mov    Ch07, R7  
        lcall   bc_srl  
        mov    Ch08, R7  
        lcall   bc_srl  
        mov    Ch09, R7  
        lcall   bc_srl  
        mov    Ch0A, R7  
        lcall   bc_srl  
        mov    Ch0B, R7  
        lcall   bc_srl  
        lcall   bc_srl  
        mov    Ch0C, R7  
        lcall   bc_srl
```

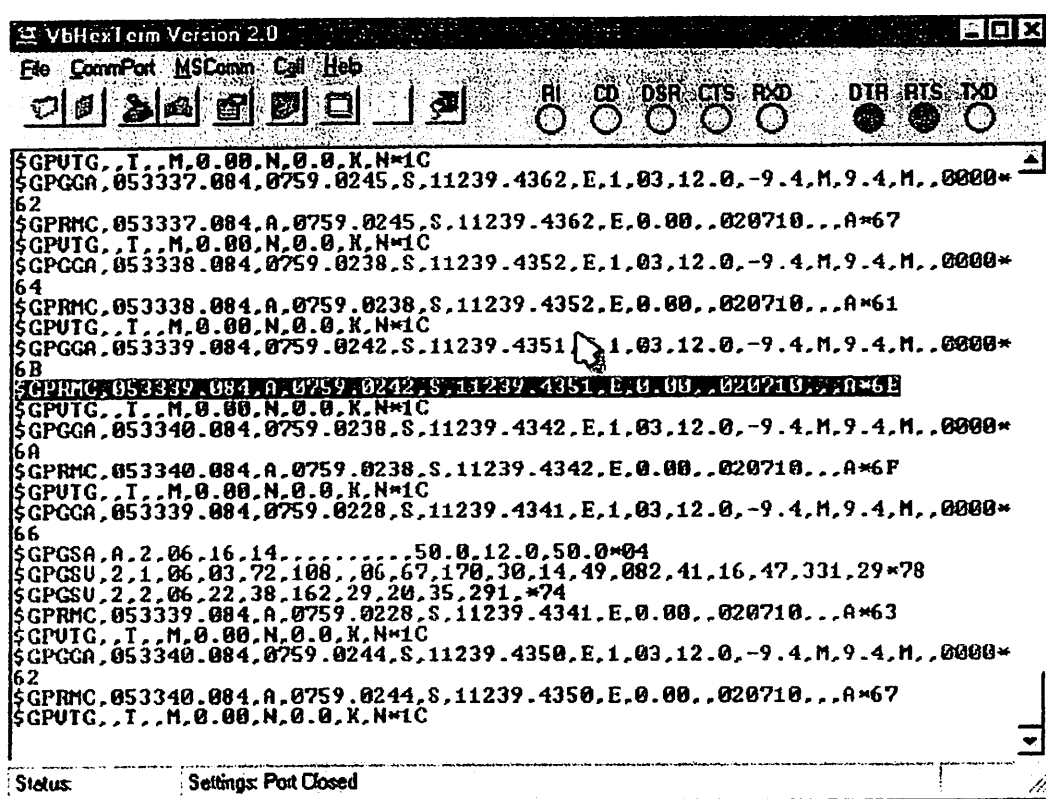
4.3. Pengujian Keseluruhan Sistem

Pengujian ini dapat dilakukan setelah semua tahap pengujian perbagian telah dilakukan dengan baik, pengujian dilakukan dalam dua tahap, yaitu pengujian untuk menentukan posisi bubu yang akan ditebarkan dengan jumlah 50 buah pada perjalanan pertama, yaitu penebaran bubu 1-50. Untuk perjalanan kedua melakukan penebaran bubu yang ke 51-100, bersamaan itu melakukan pengambilan bubu yang telah ditebarkan pada perjalanan pertama yaitu mulai pengambilan bubu 50-1. Kapal yang digunakan yaitu kapal nelayan dengan GPS Receiver yang diletakan dalam kapal dan antena GPS yang diletakan diluar kapal. System perjalanan akan dilakukan sesuai dengan block diagram dibawah ini.



Gambar 4.3 block diagram pengujian sesuai dengan sistem perjalanan

Dari hasil pengujian didapat data yang dikirimkan oleh modul GPS sebagai berikut :



Gambar 4.2. Hasil Pengujian Modul GPS

\$GPRMC,053339.084,A,0759.0242,S,11239.4351,E,0.00,020710,A*6E

Tabel 4.1. Hasil Pengujian Modul GPS EG T-10

\$GPRMC	RMC Protocol Header
053339.419	Diambil : 05:33:39:419 UTC
A	A, Data Valid ;V, Data Not Valid
0759.0242,S	Latitude : 07o59'0242 S
11239.4351,E	Longitude : 112o39'4351 E
0.00	Kecepatan diatas tanah 0,00 Knot
020710	Tanggal dalam format ddmmyy (02 Juli 2010)
A*6E	Checksum

BAB IV

ANALISA DAN PENGUJIAN SISTEM

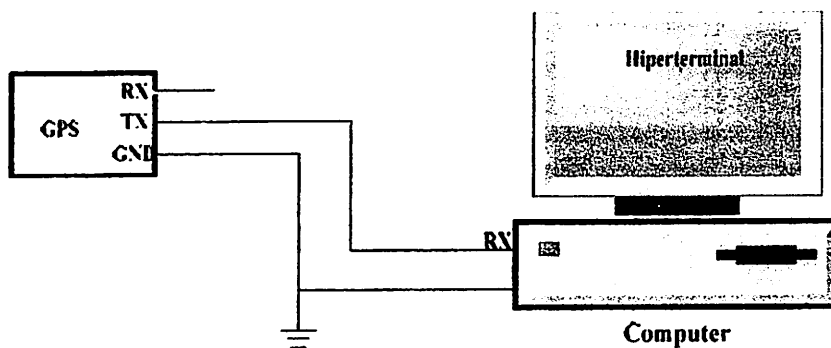
4.1 Pendahuluan

Dalam bab ini membahas tentang pengujian dan pengukuran dari peralatan yang dibuat. Secara umum pengujian ini bertujuan untuk mengetahui apakah piranti yang telah direalisasikan dapat bekerja sesuai dengan perencanaan yang telah direncanakan. Berhubung pada batasan masalah pegujian hanya dilakukan pada area kampus ITN2 dan tidak diuji secara langsung dilaut.

Pada bab ini akan diperlihatkan hasil pengujian alat aplikasiGPS pada kapal penangkap ikan dengan tampilan pada LCD. Sebagian besar pengujian dilakukan dengan menggunakan mikrokontroller uji dan GPS karena sistem yang dirancang berhubungan dengan data serial.

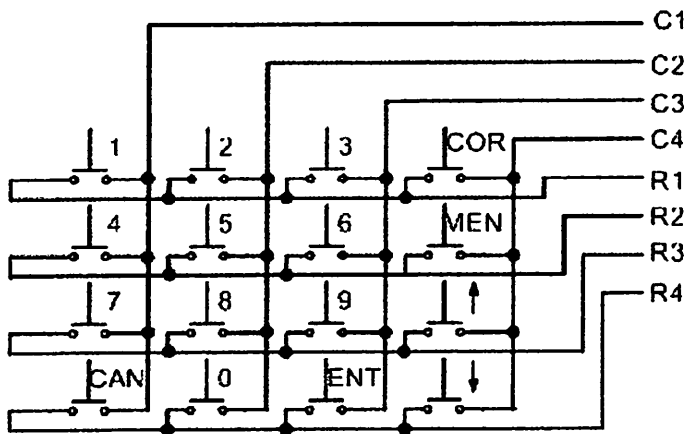
4.2 Pengujian Modul GPS

Pengujian Modul GPS dilakukan dengan menggunakan PC dengan Komunikasi RS 232. Data yang diterima oleh PC ditampilkan dengan menggunakan program Hiperterminal.



Gambar 4.1. Blok Diagram Pengujian Modul GPS

Keypad serig digunakan sebagai suatu input pada beberapa peralatan yang berbasis mikroprocessor atau mikrokontroller. Keypad sesungguhnya terdiri dari sejumlah saklar, yang terhubung sebagai baris dan kolom dengan susunan seperti yang ditunjukkan pada gambar 3.8. Agar mikrokontroller dapat melakukan scan keypad, maka port mengeluarkan salah satu bit dari 4 bit yang terhubung pada kolom dengan logika low "0" dan selanjutnya membaca 4 bit pada baris untuk menguji jika ada tombol yang ditekan pada kolom tersebut. Sebagai konsekuensi, selama tidak ada tombol yang ditekan, maka mikrokontroller akan melihat sebagai logika high "1" pada setiap pin yang terhubung ke baris.

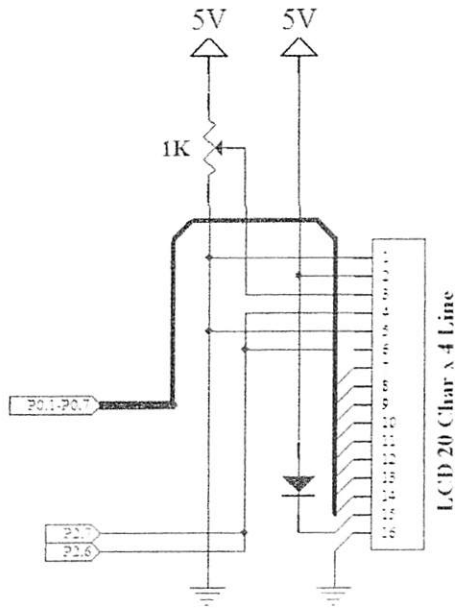


Gambar 3.11. Rangkaian dasar keypad 4x4

Keterangan :

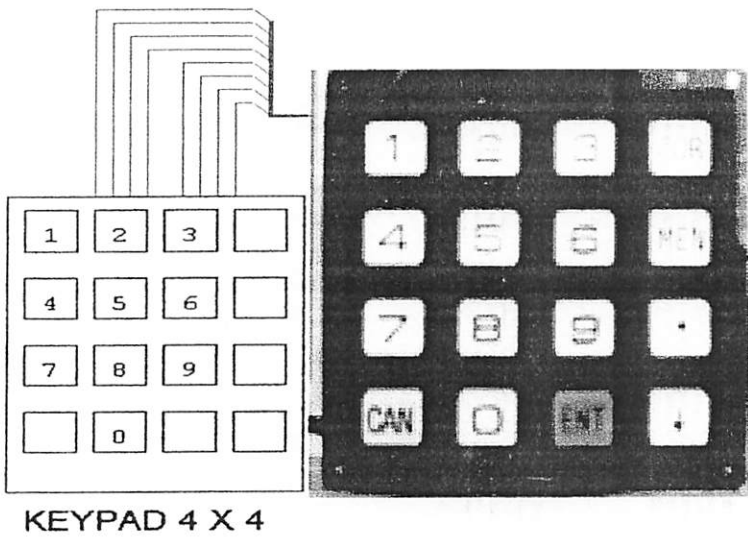
(C) = coulomb (*kolom*)

(R) = row (*baris*)



Gambar 3.9. Rangkaian Liquid Crystal Display (LCD) 20 Char x 4 Line

3.2.6. Keypad



Gambar 3.10 Rangkaian interface keypad 4x4

Table 3.3 memory mapping

koordinat		Alamat
100 titik		1-400 <i>byte</i>
Counter record	MSB	401 <i>bytes</i>
	LSB	402 <i>bytes</i>
Counter erase	MSB	403 <i>bytes</i>
	LSB	404 <i>bytes</i>

Dengan penghapusan dan pe-record data sebagai berikut

Table 3.4 Metode Penghapusan dan pe-record

Perjalanan	Record	Erase
1	Data ke 1-50 (1-200 <i>byte</i>)	Data ke100-51 (400-201 <i>byte</i>)
2	Data ke 51-100 201	Data ke 50-1 (200-1 <i>byte</i>)

3.2.5. LCD (Liquid Cristal Display)

Sebagai unit keluaran sistem mikrokontroler, port 0 digunakan sebagai jalur data bagi penampilan LCD. Dalam aplikasi ini tidak dilakukan operasi pembacaan dari register intruksi internal ataupun register data internal modul LCD. Oleh karena jalur kontrol R/W dihubungkan langsung ke ground sehingga mode operasi yang dilakukan selalu operasi penulisan ke register interuksi maupun register data internal modul LCD. Dengan mengubah nilai V_{EE} menggunakan potensiometer 1 k Ω akan diperoleh tingkat kecerahan yang berbeda pada tampilan LCD. Bila LCD yang digunakan adalah backlight, dioda perlu perlu dipasang. Dalam gambar 3-6, memperlihatkan rangkaian LCD yang direncanakan.

3.2.4 Perancangan Rangkaian Serial EEPROM

Pada perancangan aplikasi GPS pada kapal penangkap ikan digunakan serial EEPROM sebagai *memory* dengan rangkaian sebagai berikut :

Tabel 3.1 Rangkaian Serial EEPROM

No pin	Nama	Koneksi pin I/O AT24C16
5	SDA	P0.0
6	SCL	P0.1

Pada perancangan aplikasi GPS pada kapal penangkap ikan digunakan serial EEPROM sebagai *memory* untuk tempat penyimpanan data yang akan di-*record* AT24C16 memiliki kapasitas *memory* 16 Kbyte satu titik yang di-*record* membutuhkan *memory* sebesar 4 bytes dengan perincian dan urutan pe-*record* sebagai berikut :

Table 3.2 Perincian dan Urutan Pe-*record*

koordinat	Digunakan Untuk	Jumlah Byte
X	MSB	1
	LSB	1
Y	MSB	1
	LSB	1
Jumlah total		4

Pada perencanaan awal, titik yang di-*record* sebanyak 100 titik sehingga membutuhkan 400 byte yang tersimpan mulai bytes 1-400. untuk bytes 401-402 berisi MSB dan counter LSB counter untuk keypad record. Sedangkan bytes ke 403-404 berisi MSB dan LSB untuk keypad erase. Nilai awal keypad record adalah 1 sedangkan counter erase adalah 400. dengan *memory mapping* sebagai berikut.

3.2.4 Perancangan Rangkaian Serial EEPROM

Pada perancangan aplikasi GPS pada kapal penangkap ikan digunakan serial EEPROM sebagai *memory* dengan rangkaian sebagai berikut :

Tabel 3.1 Rangkaian Serial EEPROM

<i>No pin</i>	<i>Nama</i>	<i>Koneksi pin I/O AT24C16</i>
5	SDA	P2.0
6	SCL	P2.1

Pada perancangan aplikasi GPS pada kapal penangkap ikan digunakan serial EEPROM sebagai *memory* untuk tempat penyimpanan data yang akan di-*record* AT24C16 memiliki kapasitas *memory* 16 Kbyte satu titik yang di-*record* membutuhkan *memory* sebesar 4 bytes dengan perincian dan urutan pe-*record* sebagai berikut :

Table 3.2 Perincian dan Urutan Pe-*record*

koordinat	Digunakan Untuk	Jumlah Byte
X	MSB	1
	LSB	1
Y	MSB	1
	LSB	1
Jumlah total		4

Pada perencanaan awal, titik yang di-*record* sebanyak 100 titik sehingga membutuhkan 400 byte yang tersimpan mulai bytes 1-400. untuk bytes 401-402 berisi MSB dan counter LSB counter untuk keypad record. Sedangkan bytes ke 403-404 berisi MSB dan LSB untuk keypad erase. Nilai awal keypad record adalah 1 sedangkan counter erase adalah 400. dengan *memory mapping* sebagai berikut.

$$= 9,042 \times 10^{-8} \text{ S}$$

Sehingga waktu minimal logika tinggi yang dibutuhkan untuk mereset mikrokontroler adalah

:

Reset (min) = $r \times$ periode yang dibutuhkan

$$= 9,0424 \times 10^{-8} \times 24$$

$$= 2,170 \mu\text{s}$$

Jadi mikrokontroler membutuhkan waktu minimal 2,170 μs untuk mereset. Waktu minimal

ini yang dijadikan pedoman untuk menentukan nilai R dan C. dari persamaan konstanta

waktu $r = R \times C$ dan jika nilai R ditentukan sebesar 10 K Ω , maka nilai C adalah :

$$C = \frac{t}{R}$$

$$= \frac{2,170 \times 10^{-6}}{10 \times 10^{-3}}$$

$$= 2,170 \times 10^{-12} \text{ F}$$

Kapasitor minimal yang dibutuhkan adalah 2,170 pF. Dengan menggunakan kapasitor

sebesar 10 μF , maka akan menjamin waktu reset diatas nilai minimal waktu yang dibutuhkan

untuk mereset mikrokontroler

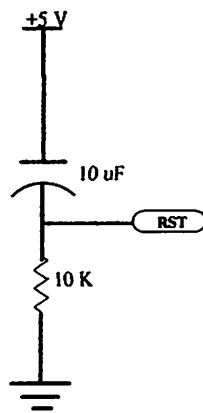
$$256 - TH_1 = \frac{11,0592 \times 10^6}{1843200}$$

$$TH_1 = 256 - 6$$

$$= 250$$

3.2.3. Perancangan Rangkaian Reset

Pin reset pada mikrokontroler merupakan aktif high (1). Pulsa transisi dari low (0) dan high (1) akan mereset mikrokontroler menuju alamat 0000H. pin reset dihubungkan dengan rangkaian power on reset diperlihatkan pada Gambar. 3.6 berikut ini :



Gambar 3.8 Rangkaian Reset untuk MCU AT89S52^[2]

Karena Kristal yang digunakan mempunyai frekuensi sebesar 11,0592 MHz, maka satu periode membutuhkan waktu sebesar :

$$T = \frac{1}{F_{XTAL}}$$

$$= \frac{1}{11,0592 \text{ MHz}} \text{ s}$$

Cara menghubungkan pada register TH1 untuk *boudrate* 19200 :

$$Boud\ Rate = \frac{Fosc}{12 \times (256 - TH1) \times 16}$$

$$19200 = \frac{11,0592\ MHz}{12 \times (256 - TH1) \times 16}$$

$$3686400 = \frac{11,0592\ MHz}{256 - TH1}$$

$$256 - TH1 = \frac{11,0592 \times 10^6}{3686400}$$

$$TH_1 = 256 - 3$$

$$= 253$$

$$= FD_H$$

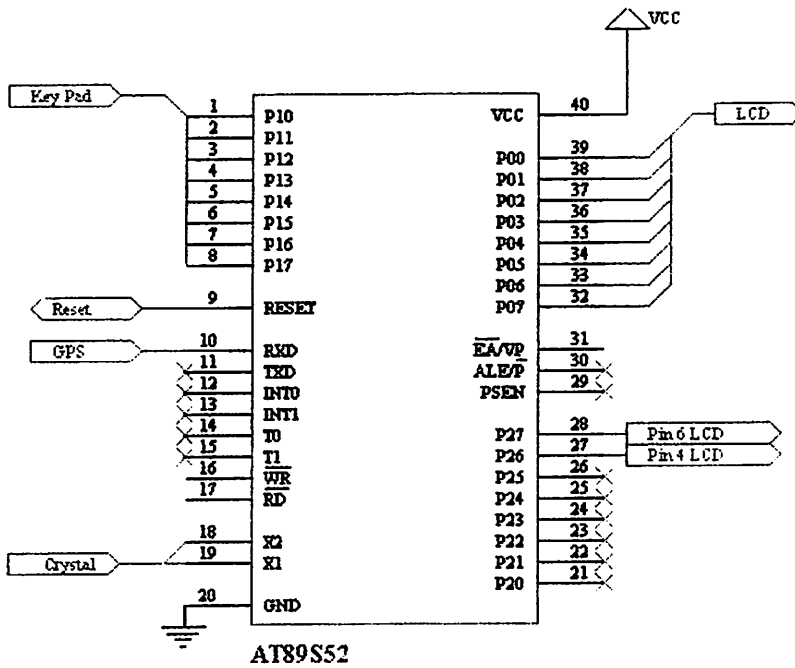
Cara menghitung pada register TH1 : untuk *boudrate* 4800 bps :

$$Boud\ Rate = \frac{Fosc}{12 \times (256 - TH1) \times 32}$$

$$4800 = \frac{11,0592\ MHz}{12 \times (256 - TH1) \times 32}$$

$$1843200 = \frac{11,0592\ MHz}{256 - TH1}$$

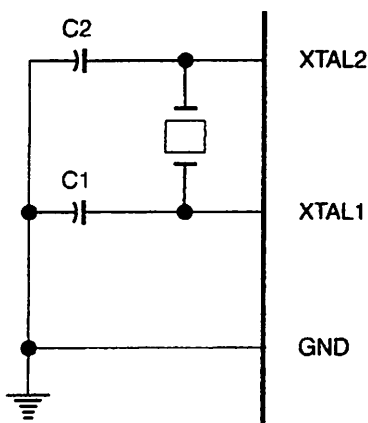
- *Reset* digunakan untuk *mereset* program kontrol MCU, dimana MCU memiliki masukan *aktif high*.



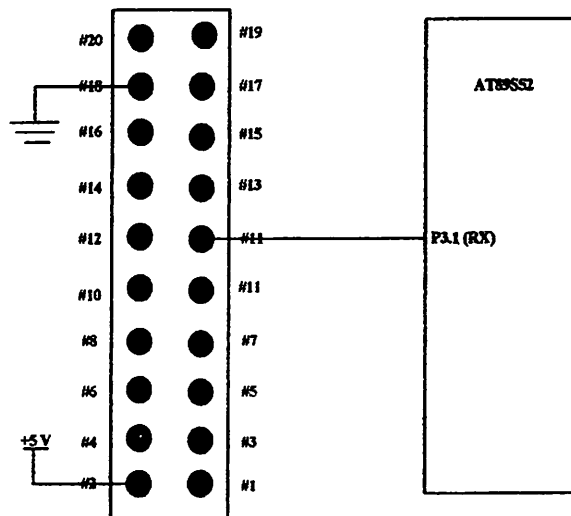
Gambar 3.6 Rangkaian Mikrokontroler AT89S52

3.2.2 Pengaturan Boud Rate

Didalam komunikasi serial antara Mikrokontroler dengan GPS terlebih dahulu ditentukan *boud rate* yang digunakan. Pada sistem ini digunakan *boud rate* 19200 bps dan 4800 bps dengan menggunakan $f_{osc} = 11,0592$ MHz.



Gambar 3.7. Rangkaian Osilator^[1]

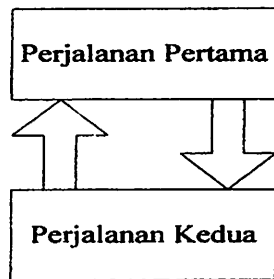


Gambar 3.5 Rangkaian interface antara GPS dengan MCU

Mikrokontroler AT89S52 digunakan untuk mendapatkan rangkaian yang praktis dan ringkas. Maka untuk mengolah data dari sistem pemantauan posisi kapal menggunakan GPS digunakan Mikrokontroler AT89S52 yang merupakan IC Mikrokontroler dengan fasilitas EEPROM sebagai tempat penyimpanan program, sehingga alat ukur menjadi lebih sederhana. Mikrokontroler AT89S52 memiliki 4 port masing-masing port digunakan sebagai berikut :

- Port 1.0-1.7 digunakan untuk driver keypad.
- Port 0.0 - 0.7, port 2.7 dan 2.6 digunakan sebagai output untuk LCD.
- Port 3.0 berfungsi sebagai Rx untuk komunikasi Serial dengan GPS.
- Kristal yang digunakan dalam mikrokontroler ini adalah 11,0592 MHz, hal ini berpengaruh pada kecepatan pengiriman sinyal data komunikasi serial. Kristal dan kapasitor C_1 dan C_2 yang masing-masing bernilai 33 pF, akan mengaktifkan pulsa clock yang menjadi penggerak bagi seluruh perasi internal MCU (*Aplication Note AT89S52 In Circuit Programming*).
- Vcc dihubungkan dengan tegangan sebesar +5V sesuai dengan tegangan operasi *chip* tunggal yang diijinkan dari *data sheet*.
- GND dihubungkan ke *ground* catu daya.

ditebarkan pada perjalanan pertama yaitu data ke (50-1). Ketika kapal start, maka dilakukan penebaran bubu untuk data 51-100 (*record*), bersamaan dilakukan pengambilan bubu lama yang dimulai dari data ke 50-1 (*erase*), kemudian kapal pulang dan selanjutnya kapal kembali mengikuti route perjalanan pertama sesuai dengan gambar 1.4



Gambar 3.4 block diagram sistem perjalanan

3.2 Perancangan Hardware

Bagian ini menguraikan perencanaan perangkat keras yang meliputi perencanaan :

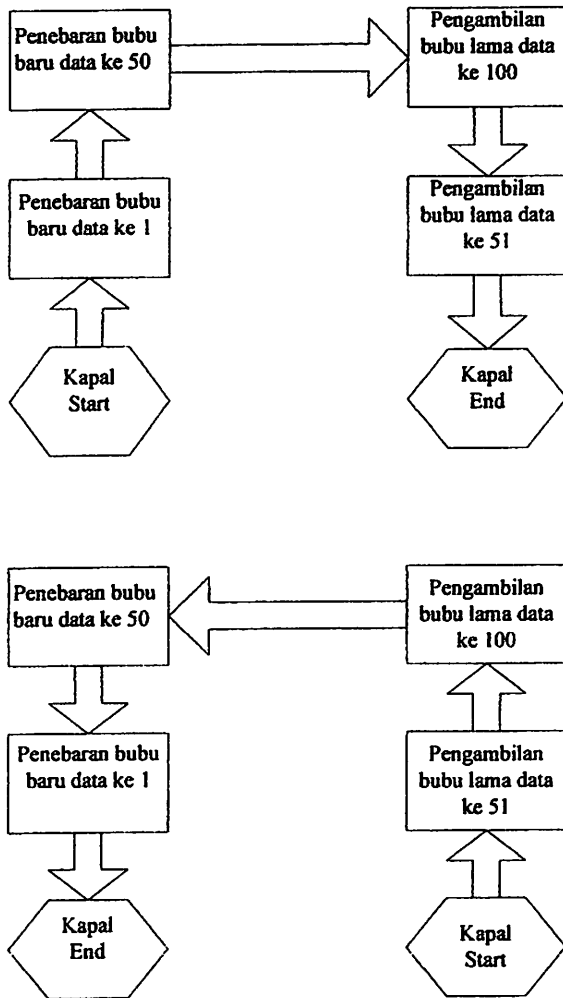
- ❖ Sistem perencanaan, terdiri dari : GPS, Mikrokontroler AT89S52, LCD M1632, key pad.

3.2.1 Perancangan Komunikasi data GPS dengan Mikrokontroler AT89S52

Pada sistem digunakan Modul GPS Receiver EG-T10 yang memiliki keluaran berupa data serial dengan level TTL dengan boudrate sebesar 4800 bps. Untuk pengiriman data ke Mikrokontroler digunakan pin ke -11 *Transmit* (Tx) yang dihubungkan dengan port 3.0 (Rx) pada Mikrokontroler dan pin ke -2 untuk Vcc sebesar 5 volt serta pin ke -18 dihubungkan dengan Ground seperti pada Gambar 3.2 berikut ini :

posisi bubu ditbarkan pada serial EEPROM (24C64), kemudian ditampilkan kedalam LCD 20 Char x 4 Line.. Ketika keypad eraser ditekan, maka mikrokontroler akan menghapus posisi bubu yang ditebarkan pada serial EEPROM (24C16), kemudian menghapus tampilan koordinat tersebut pada LCD 20 Char x 4 Line..

System pengambilan bubu sebagai berikut.



Gambar 3.3 Block diagram sistem perjalanan pertama dan kedua

Pada perjalanan pertama diasumsikan ada 50 titik sebagai data lama bubu ditebarkan (data ke 100-51).ketik kapal start, maka dilakukan penebaran bubu untuk data 1-50 (*record*). Kemudian dilakukan pengambilan bubu lama yang dimulai dari data ke 100-51 (*erase*), kemudian kapal pulang. Untuk perjalanan kedua terdapat data bubu lama yang telah


```
tg_tk2: cjne    R0,#14,tg_tk3
         ljmp    tg_tkn
tg_tk3: cjne    R0,#13,tg_tk4
         ljmp    tg_tkn
tg_tk4: cjne    R0,#12,tg_tk5
         ljmp    tg_tkn
tg_tk5: cjne    R0,#11,tg_tk6
         ljmp    tg_tkn
tg_tk6: cjne    R0,#10,tg_tk7
         ljmp    tg_tkn
tg_tk7: ret
;
tg_lps: lcall   scnkp
         lcall   delay0
         cjne    R0,#10,tg_lps
         ret
```

4.4 Hasil Pengujian system

Berikut ini adalah tabel data bubu yang disi pada serial EEPROM dan adressnya

Tabel 4.2 Hasil data perjalanan pertama saat record bubu 1-50

Data ke (n)	koordinat	
	Lintang S	Bujur E
1	0755.03 ⁰	11237.96 ⁰
2	0755.02 ⁰	11237.96 ⁰
3	0755.01 ⁰	11237.97 ⁰
4	0755.00 ⁰	11237.97 ⁰
5	0755.99 ⁰	11237.98 ⁰
6	0755.98 ⁰	11237.98 ⁰
7	0755.97 ⁰	11237.99 ⁰
8	0754.96 ⁰	11238.00 ⁰
9	0754.95 ⁰	11238.01 ⁰
10	0754.94 ⁰	11238.01 ⁰
11	0754.93 ⁰	11238.02 ⁰
12	0754.92 ⁰	11238.03 ⁰
13	0754.91 ⁰	11238.04 ⁰
14	0754.90 ⁰	11238.05 ⁰
15	0754.89 ⁰	11238.05 ⁰
16	0754.85 ⁰	11238.08 ⁰
17	0754.85 ⁰	11238.08 ⁰
18	0754.85 ⁰	11238.08 ⁰
19	0754.85 ⁰	11238.08 ⁰
20	0754.84 ⁰	11238.09 ⁰
21	0754.83 ⁰	11238.10 ⁰
22	0754.82 ⁰	11238.10 ⁰
23	0754.81 ⁰	11238.11 ⁰
24	0754.80 ⁰	11238.12 ⁰
25	0754.79 ⁰	11238.12 ⁰

Data ke (n)	koordinat	
	Lintang S	Bujur E
26	0754.78 ⁰	11238.13 ⁰
27	0754.77 ⁰	11238.13 ⁰
28	0754.76 ⁰	11238.13 ⁰
29	0754.75 ⁰	11238.14 ⁰
30	0754.74 ⁰	11238.14 ⁰
31	0754.73 ⁰	11238.14 ⁰
32	0754.72 ⁰	11238.15 ⁰
33	0754.71 ⁰	11238.15 ⁰
34	0754.70 ⁰	11238.15 ⁰
35	0754.69 ⁰	11238.16 ⁰
36	0754.68 ⁰	11238.16 ⁰
37	0754.67 ⁰	11238.17 ⁰
38	0754.66 ⁰	11238.17 ⁰
39	0754.65 ⁰	11238.17 ⁰
40	0754.64 ⁰	11238.18 ⁰
41	0754.63 ⁰	11238.18 ⁰
42	0754.62 ⁰	11238.18 ⁰
43	0754.61 ⁰	11238.19 ⁰
44	0754.60 ⁰	11238.19 ⁰
45	0754.59 ⁰	11238.20 ⁰
46	0754.58 ⁰	11238.20 ⁰
47	0754.57 ⁰	11238.20 ⁰
48	0754.56 ⁰	11238.21 ⁰
49	0754.55 ⁰	11238.21 ⁰
50	0754.54 ⁰	11238.22 ⁰

Hasil data pada tabel diatas adalah posisi yang direkam oleh GPS yang disimpan di serial EEPROM. Dimana data-data tersebut adalah nilai dari hasil record GPS untuk

tempat dimana bubu diletakan, jumlah bubu yang ditebarkan adalah 1-50. Jarak antara posisi bubu adalah 5 meter.

Tabel 4.3 perjalanan kedua saat record data 51-100

Data ke (n)	koordinat	
	Lintang S	Bujur E
51	0755.03 ⁰	11237.97 ⁰
52	0755.02 ⁰	11237.98 ⁰
53	0755.01 ⁰	11237.98 ⁰
54	0755.00 ⁰	11237.99 ⁰
55	0754.99 ⁰	11237.99 ⁰
56	0754.98 ⁰	11238. 01 ⁰
57	0754.97 ⁰	11238. 01 ⁰
58	0754.96 ⁰	11238. 02 ⁰
59	0754.95 ⁰	11238. 02 ⁰
60	0754.94 ⁰	11238. 03 ⁰
61	0754.93 ⁰	11238. 03 ⁰
62	0754.92 ⁰	11238.04 ⁰
63	0754.91 ⁰	11238.04 ⁰
64	0754.90 ⁰	11238. 05 ⁰
65	0754.89 ⁰	11238. 05 ⁰
66	0754.88 ⁰	11238.06 ⁰
67	0754.87 ⁰	11238.06 ⁰
68	0754.86 ⁰	11238.07 ⁰
69	0754.85 ⁰	11238.07 ⁰
70	0754.84 ⁰	11238.08 ⁰
71	0754.83 ⁰	11238.08 ⁰
72	0754.82 ⁰	11238.09 ⁰
73	0754.81 ⁰	11238.09 ⁰
74	0754.80 ⁰	11238.10 ⁰
75	0754.79 ⁰	11238.11 ⁰

Data ke (n)	koordinat	
	Lintang S	Bujur E
76	0754.78 ⁰	11238.12 ⁰
77	0754.77 ⁰	11238.13 ⁰
78	0754.76 ⁰	11238.14 ⁰
79	0754.75 ⁰	11238.15 ⁰
80	0754.74 ⁰	11238.16 ⁰
81	0754.73 ⁰	11238. 17 ⁰
82	0754.72 ⁰	11238.18 ⁰
83	0754.71 ⁰	11238.19 ⁰
84	0754.70 ⁰	11238.20 ⁰
85	0754.69 ⁰	11238.21 ⁰
86	0754.68 ⁰	11238.22 ⁰
87	0754.67 ⁰	11238. 23 ⁰
88	0754.66 ⁰	11238.24 ⁰
89	0754.65 ⁰	11238.25 ⁰
90	0754.64 ⁰	11238. 26 ⁰
91	0754.63 ⁰	11238.27 ⁰
92	0754.62 ⁰	11238.28 ⁰
93	0754.61 ⁰	11238. 29 ⁰
94	0754.60 ⁰	11238.30 ⁰
95	0754.59 ⁰	11238.31 ⁰
96	0754.58 ⁰	11238.32 ⁰
97	0754.57 ⁰	11238.33 ⁰
98	0754.56 ⁰	11238.34 ⁰
99	0754.55 ⁰	11238.45 ⁰
100	0754.54 ⁰	11238.36 ⁰

Setelah kapal melakukan record untuk perjalanan kedua yaitu merekam posisi bubu untuk data ke 51-100, bersamaan dengan itu, waktu kapal pulang kapal

Tabel 4.3. Kejelasan kode dan record data 21-100

Data ke	(a)	Kejelasan	
		1	2
100	0724 24 ⁰	1138 36 ⁰	1138 36 ⁰
99	0724 25 ⁰	1138 45 ⁰	1138 45 ⁰
98	0724 26 ⁰	1138 34 ⁰	1138 34 ⁰
97	0724 27 ⁰	1138 33 ⁰	1138 33 ⁰
96	0724 28 ⁰	1138 35 ⁰	1138 35 ⁰
95	0724 29 ⁰	1138 31 ⁰	1138 31 ⁰
94	0724 30 ⁰	1138 30 ⁰	1138 30 ⁰
93	0724 31 ⁰	1138 32 ⁰	1138 32 ⁰
92	0724 32 ⁰	1138 38 ⁰	1138 38 ⁰
91	0724 33 ⁰	1138 37 ⁰	1138 37 ⁰
90	0724 34 ⁰	1138 38 ⁰	1138 38 ⁰
89	0724 35 ⁰	1138 35 ⁰	1138 35 ⁰
88	0724 36 ⁰	1138 34 ⁰	1138 34 ⁰
87	0724 37 ⁰	1138 35 ⁰	1138 35 ⁰
86	0724 38 ⁰	1138 32 ⁰	1138 32 ⁰
85	0724 39 ⁰	1138 33 ⁰	1138 33 ⁰
84	0724 40 ⁰	1138 30 ⁰	1138 30 ⁰
83	0724 41 ⁰	1138 39 ⁰	1138 39 ⁰
82	0724 42 ⁰	1138 38 ⁰	1138 38 ⁰
81	0724 43 ⁰	1138 37 ⁰	1138 37 ⁰
80	0724 44 ⁰	1138 36 ⁰	1138 36 ⁰
79	0724 45 ⁰	1138 35 ⁰	1138 35 ⁰
78	0724 46 ⁰	1138 34 ⁰	1138 34 ⁰
77	0724 47 ⁰	1138 33 ⁰	1138 33 ⁰
76	0724 48 ⁰	1138 32 ⁰	1138 32 ⁰

Data ke	(b)	Kejelasan	
		1	2
75	0724 29 ⁰	1138 31 ⁰	1138 31 ⁰
74	0724 30 ⁰	1138 30 ⁰	1138 30 ⁰
73	0724 31 ⁰	1138 32 ⁰	1138 32 ⁰
72	0724 32 ⁰	1138 33 ⁰	1138 33 ⁰
71	0724 33 ⁰	1138 34 ⁰	1138 34 ⁰
70	0724 34 ⁰	1138 35 ⁰	1138 35 ⁰
69	0724 35 ⁰	1138 36 ⁰	1138 36 ⁰
68	0724 36 ⁰	1138 37 ⁰	1138 37 ⁰
67	0724 37 ⁰	1138 38 ⁰	1138 38 ⁰
66	0724 38 ⁰	1138 39 ⁰	1138 39 ⁰
65	0724 39 ⁰	1138 40 ⁰	1138 40 ⁰
64	0724 40 ⁰	1138 41 ⁰	1138 41 ⁰
63	0724 41 ⁰	1138 42 ⁰	1138 42 ⁰
62	0724 42 ⁰	1138 43 ⁰	1138 43 ⁰
61	0724 43 ⁰	1138 44 ⁰	1138 44 ⁰
60	0724 44 ⁰	1138 45 ⁰	1138 45 ⁰
59	0724 45 ⁰	1138 46 ⁰	1138 46 ⁰
58	0724 46 ⁰	1138 47 ⁰	1138 47 ⁰
57	0724 47 ⁰	1138 48 ⁰	1138 48 ⁰
56	0724 48 ⁰	1138 49 ⁰	1138 49 ⁰
55	0724 49 ⁰	1138 50 ⁰	1138 50 ⁰
54	0724 50 ⁰	1138 51 ⁰	1138 51 ⁰
53	0724 51 ⁰	1138 52 ⁰	1138 52 ⁰
52	0724 52 ⁰	1138 53 ⁰	1138 53 ⁰
51	0724 53 ⁰	1138 54 ⁰	1138 54 ⁰
50	0724 54 ⁰	1138 55 ⁰	1138 55 ⁰
49	0724 55 ⁰	1138 56 ⁰	1138 56 ⁰
48	0724 56 ⁰	1138 57 ⁰	1138 57 ⁰
47	0724 57 ⁰	1138 58 ⁰	1138 58 ⁰
46	0724 58 ⁰	1138 59 ⁰	1138 59 ⁰
45	0724 59 ⁰	1138 60 ⁰	1138 60 ⁰
44	0724 60 ⁰	1138 61 ⁰	1138 61 ⁰
43	0724 61 ⁰	1138 62 ⁰	1138 62 ⁰
42	0724 62 ⁰	1138 63 ⁰	1138 63 ⁰
41	0724 63 ⁰	1138 64 ⁰	1138 64 ⁰
40	0724 64 ⁰	1138 65 ⁰	1138 65 ⁰
39	0724 65 ⁰	1138 66 ⁰	1138 66 ⁰
38	0724 66 ⁰	1138 67 ⁰	1138 67 ⁰
37	0724 67 ⁰	1138 68 ⁰	1138 68 ⁰
36	0724 68 ⁰	1138 69 ⁰	1138 69 ⁰
35	0724 69 ⁰	1138 70 ⁰	1138 70 ⁰
34	0724 70 ⁰	1138 71 ⁰	1138 71 ⁰
33	0724 71 ⁰	1138 72 ⁰	1138 72 ⁰
32	0724 72 ⁰	1138 73 ⁰	1138 73 ⁰
31	0724 73 ⁰	1138 74 ⁰	1138 74 ⁰
30	0724 74 ⁰	1138 75 ⁰	1138 75 ⁰
29	0724 75 ⁰	1138 76 ⁰	1138 76 ⁰
28	0724 76 ⁰	1138 77 ⁰	1138 77 ⁰
27	0724 77 ⁰	1138 78 ⁰	1138 78 ⁰
26	0724 78 ⁰	1138 79 ⁰	1138 79 ⁰
25	0724 79 ⁰	1138 80 ⁰	1138 80 ⁰
24	0724 80 ⁰	1138 81 ⁰	1138 81 ⁰
23	0724 81 ⁰	1138 82 ⁰	1138 82 ⁰
22	0724 82 ⁰	1138 83 ⁰	1138 83 ⁰
21	0724 83 ⁰	1138 84 ⁰	1138 84 ⁰
20	0724 84 ⁰	1138 85 ⁰	1138 85 ⁰
19	0724 85 ⁰	1138 86 ⁰	1138 86 ⁰
18	0724 86 ⁰	1138 87 ⁰	1138 87 ⁰
17	0724 87 ⁰	1138 88 ⁰	1138 88 ⁰
16	0724 88 ⁰	1138 89 ⁰	1138 89 ⁰
15	0724 89 ⁰	1138 90 ⁰	1138 90 ⁰
14	0724 90 ⁰	1138 91 ⁰	1138 91 ⁰
13	0724 91 ⁰	1138 92 ⁰	1138 92 ⁰
12	0724 92 ⁰	1138 93 ⁰	1138 93 ⁰
11	0724 93 ⁰	1138 94 ⁰	1138 94 ⁰
10	0724 94 ⁰	1138 95 ⁰	1138 95 ⁰
9	0724 95 ⁰	1138 96 ⁰	1138 96 ⁰
8	0724 96 ⁰	1138 97 ⁰	1138 97 ⁰
7	0724 97 ⁰	1138 98 ⁰	1138 98 ⁰
6	0724 98 ⁰	1138 99 ⁰	1138 99 ⁰
5	0724 99 ⁰	1138 00 ⁰	1138 00 ⁰
4	0724 00 ⁰	1138 01 ⁰	1138 01 ⁰
3	0724 01 ⁰	1138 02 ⁰	1138 02 ⁰
2	0724 02 ⁰	1138 03 ⁰	1138 03 ⁰
1	0724 03 ⁰	1138 04 ⁰	1138 04 ⁰

Jumlah kapal melintasi record untuk pertamannya kedua akan merupakan posisi

data untuk data 21-100. Kemudian dengan itu akan kapal pulang kapal

melakukan pengambilan bubu yaitu pengambilannya mulai dari data 50-1 sesuai dengan route perjalanan pada block diagram diatas.

Tabel 4.4 perjalanan kedua saat erase atau pengambilan bubu 50-1

Data ke (n)	Perbandingan			
	Koordinat Tanam dan		Lepas	
	Koordinat tanam		Koordinat lepas	
	Lintang S	Bujur E	Lintang S	Bujur E
50	0754.54 ⁰	11238.22 ⁰	0754.54 ⁰	11238.22 ⁰
49	0754.55 ⁰	11238.21 ⁰	0754.55 ⁰	11238.21 ⁰
48	754.56	11238.21 ⁰	0754.56	11238.21 ⁰
47	0754.57 ⁰	11238.20 ⁰	0754.57 ⁰	11238.20 ⁰
46	754.58	11238.20 ⁰	0754.58	11238.20 ⁰
45	0754.59 ⁰	11238.20 ⁰	0754.59 ⁰	11238.20 ⁰
44	0754.60 ⁰	11238.19 ⁰	0754.60 ⁰	11238.19 ⁰
43	0754.61 ⁰	11238.19 ⁰	0754.61 ⁰	11238.19 ⁰
42	0754.62 ⁰	11238.18 ⁰	0754.62 ⁰	11238.18 ⁰
41	0754.63 ⁰	11238.18	0754.63 ⁰	11238.18 ⁰
40	0754.64 ⁰	11238.18 ⁰	0754.64 ⁰	11238.18 ⁰
39	0754.65 ⁰	11238.17 ⁰	0754.65 ⁰	11238.17 ⁰
38	0754.66 ⁰	11238.17 ⁰	0754.66 ⁰	11238.17 ⁰
37	0754.67 ⁰	11238.17 ⁰	0754.67 ⁰	11238.17 ⁰
36	0754.68 ⁰	11238.16 ⁰	0754.68 ⁰	11238.16 ⁰
35	0754.69 ⁰	11238.16 ⁰	0754.69 ⁰	11238.16 ⁰
34	0754.70 ⁰	11238.15 ⁰	0754.70 ⁰	11238.15 ⁰
33	0754.71 ⁰	11238.15 ⁰	0754.71 ⁰	11238.15 ⁰
32	0754.72 ⁰	11238.15 ⁰	0754.72 ⁰	11238.15 ⁰
31	0754.73 ⁰	11238.14 ⁰	0754.73 ⁰	11238.14 ⁰
30	0754.74 ⁰	11238.14 ⁰	0754.74 ⁰	11238.14 ⁰
29	0754.75	11238.14 ⁰	0754.75 ⁰	11238.14 ⁰
28	0754.76 ⁰	11238.13 ⁰	0754.76 ⁰	11238.13 ⁰
27	0754.77 ⁰	11238.13 ⁰	0754.77 ⁰	11238.13 ⁰
26	0754.78 ⁰	11238.13 ⁰	0754.78 ⁰	11238.13 ⁰

Data ke (n)	Perbandingan			
	Koordinat Tanam dan		Lepas	
	Koordinat tanam		Koordinat lepas	
	Lintang S	Bujur E	Lintang S	Bujur E
25	0754.79 ⁰	11238.12 ⁰	0754.79 ⁰	11238.12 ⁰
24	0754.80 ⁰	11238.12 ⁰	0754.80 ⁰	11238.12 ⁰
23	0754.81 ⁰	11238.11 ⁰	0754.81 ⁰	11238.11 ⁰
22	0754.82 ⁰	11238.10 ⁰	0754.82 ⁰	11238.10 ⁰
21	0754.83 ⁰	11238.10 ⁰	0754.83 ⁰	11238.10 ⁰
20	0754.84 ⁰	11238.09 ⁰	0754.84 ⁰	11238.09 ⁰
19	0754.85 ⁰	11238.08 ⁰	0754.85 ⁰	11238.08 ⁰
18	0754.85 ⁰	11238.08 ⁰	0754.85 ⁰	11238.08 ⁰
17	0754.85 ⁰	11238.08 ⁰	0754.85 ⁰	11238.08 ⁰
16	0754.85 ⁰	11238.08 ⁰	0754.85 ⁰	11238.08 ⁰
15	0754.89 ⁰	11238.05 ⁰	0754.89 ⁰	11238.05 ⁰
14	0754.90 ⁰	11238.05 ⁰	0754.90 ⁰	11238.05 ⁰
13	0754.91 ⁰	11238.04 ⁰	0754.91 ⁰	11238.04 ⁰
12	0754.92 ⁰	11238.03 ⁰	0754.92 ⁰	11238.03 ⁰
11	0754.93 ⁰	11238.02 ⁰	0754.93 ⁰	11238.02 ⁰
10	0754.94 ⁰	11238.01 ⁰	0754.94 ⁰	11238.01 ⁰
9	0754.95 ⁰	11238.01 ⁰	0754.95 ⁰	11238.01 ⁰
8	0754.96 ⁰	11238.00 ⁰	0754.96 ⁰	11238.00 ⁰
7	0755.97 ⁰	11237.99 ⁰	0755.97 ⁰	11237.99 ⁰
6	0755.98 ⁰	11237.98 ⁰	0755.98 ⁰	11237.98 ⁰
5	0755.99 ⁰	11237.98 ⁰	0755.99 ⁰	11237.98 ⁰
4	0755.00 ⁰	11237.97 ⁰	0755.00 ⁰	11237.97 ⁰
3	0755.01 ⁰	11237.97 ⁰	0755.01 ⁰	11237.97 ⁰
2	0755.02 ⁰	11237.96 ⁰	0755.02 ⁰	11237.96 ⁰
1	0755.03 ⁰	11237.96 ⁰	0755.03 ⁰	11237.96 ⁰

Pada perjalanan kedua saat kapal melakukan record data untuk bubu 51-100 selesai, kapal pulang akan melakukan pengambilan bubu lama yaitu 50-1. Data pada table diatas adalah perbandingan antara data lama saat melakukan record dengan data baru saat melakukan erase.

4.5 Analisa Pengujian

Analisa pengujian ini dilakukan untuk mengetahui hasil error dari setiap posisi yang direkam. Untuk mengetahui posisi yang sebenarnya kita menggunakan Google eart untuk membandingkan dengan peralatan yang di buat. Disini terdapat beberapa tempat yang digunakan untuk menentukan koordinat untuk melakukan perbandingan antara lain :

- ❖ Jln.Bend Dharma Belakang Kampus 1 ITN Malang
- ❖ Terninal Arjosari
- ❖ Jln.Soekarno Hatta/depan Kampus Polinema Malang
- ❖ Depan Kampus 2 ITN Malang

Tabel 4.4 hasil koordinat yang ditentukan oleh GPS standard

NO	Lokasi Pengujian	Koordinat	
		Lintang S	Bujur E
1	Kampus 1 ITN Malang	757,45 S ⁰	11236,72 ⁰ E
2	Terminal Arjosari	756,08 ⁰ S	11239,56 ⁰ E
3	Jln.Soekarno Hatta/depan Kampus Polinema Malang	756,89 ⁰ S	11236,99 ⁰ E
4	Depan Kampus 2 ITN Malang	754,94 ⁰ S	11238,02 ⁰ E

Tabel 4.5 hasil koordinat yang ditentukan oleh GPS yang dirakit

NO	Lokasi Pengujian	Koordinat	
		Lintang S	Bujur E
1	Kampus 1 ITN Malang	757,46 ⁰ S	11236,72 ⁰ E
2	Terminal Arjosari	756,08 ⁰ S	11239,55 ⁰ E
3	Jln.Soekarno Hatta/depan Kampus Polinema Malang	756,86 ⁰ S	11239,55E
4	Depan Kampus 2 ITN Malang	754, 94 ⁰ S	11238,02 ⁰ E

4.5.1 Perhitungan hasil Error

Tabel 4.6 perhitungan error untuk lintang S

No	Lokasi Pengujian	Perhitungan Error (%)	Hasil Perhitungan % Error
1	Kampus 1 ITN Malang	$E\% = \frac{757,45 - 757,46}{757,45} \times 100\%$	0.001%
2	Terminal Arjosari	$E\% = \frac{756,08 - 756,08}{756,08} \times 100\%$	0%
3	Jln.Soekarno Hatta/depan Kampus Polinema Malang	$E\% = \frac{756,89 - 756,86}{756,89} \times 100\%$	0.003%
4	Depan Kampus 2 ITN Malang	$E\% = \frac{754,94 - 754,94}{754,94} \times 100\%$	0%

Tabel 4.7 perhitungan error untuk Bujur E

No	Lokasi Pengujian	Perhitungan Error (%)	Hasil Perhitungan % Error
1	Kampus 1 ITN Malang	$E\% = \frac{11236,72 - 11236,72}{11236,72} \times 100\%$	0%
2	Terminal Arjosari	$E\% = \frac{11239,56 - 11239,55}{11239,56} \times 100\%$	0.001%
3	Jln. Soekarno Hatta/depan Kampus Polinema Malang	$E\% = \frac{11239,55 - 11239,55}{11239,55} \times 100\%$	0%
4	Depan Kampus 2 ITN Malang	$E\% = \frac{11238,02 - 11238,02}{11238,02} \times 100\%$	0%

4.5.2 Perhitungan Hasil Error Rata-Rata

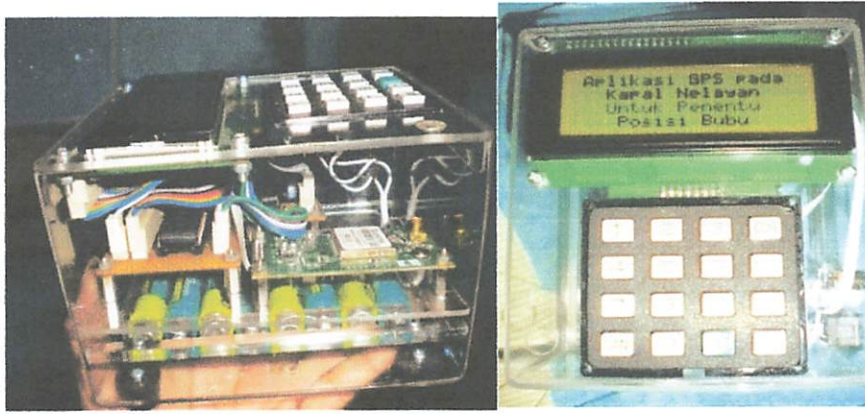
$$\%Error \text{ rata - rata} = \frac{\%Error \text{ (total)}}{\text{Jumlah Pengujian}} \times 100\%$$

$$\%Error \text{ rata - rata} = (0.001+0)+(0+0.001)+(0.003+0)+(0+0)$$

$$=0.004\%$$

$$\%Error \text{ rata - rata} = \frac{0.004}{4} \times 100\%$$

$$=-0.1\%$$



Gambar 4.4 foto keseluruhan alat



Gambar 4.5 foto koordinat saat menentukan posisi data ke1 untuk record



Gambar 4.6 foto koordinat saat menentukan posisi data ke25 untuk record



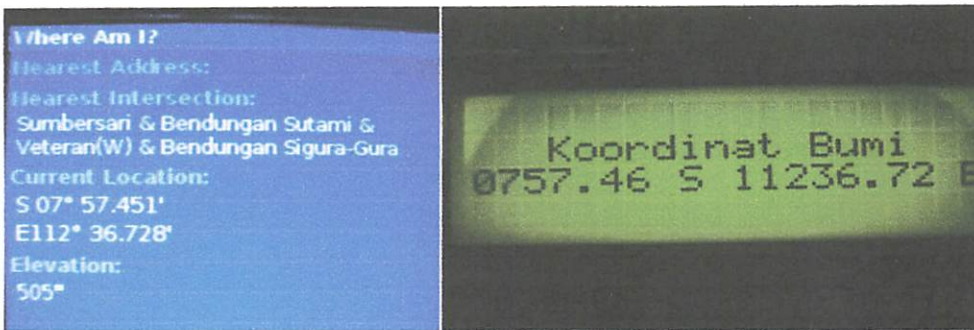
Gambar 4.7 foto koordinat saat menentukan posisi data ke60 untuk record



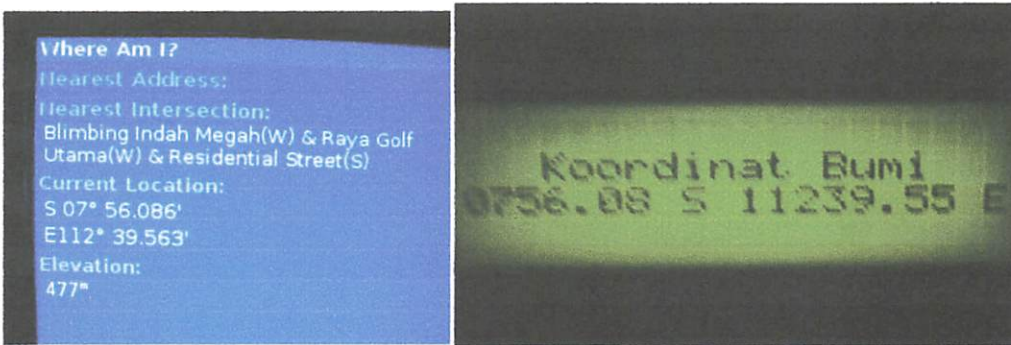
Gambar 4.8 foto koordinat saat menentukan posisi data ke1 untuk erase



Gambar 4.9 foto koordinat saat menentukan posisi data ke23 untuk erase



Gambar 4.10 foto perbandingan koordinat kampus 1 ITN



Gambar 4.11 foto perbandingan koordinat Terminal Arjosari

Figure 1. Comparison of the results of the two methods.

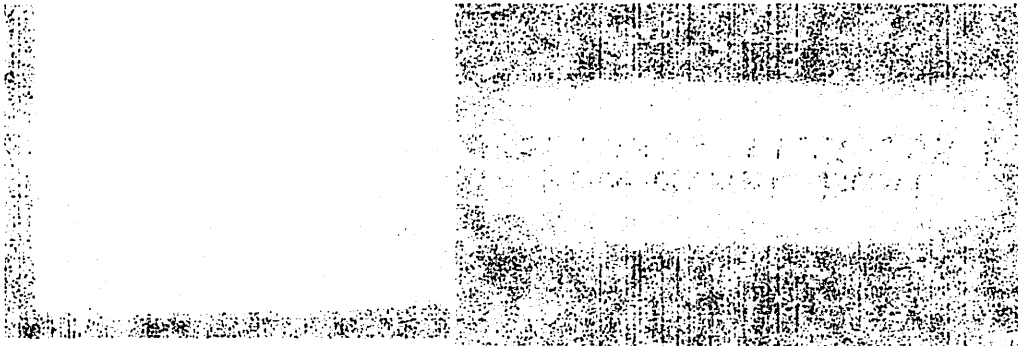


Figure 2. Comparison of the results of the two methods.

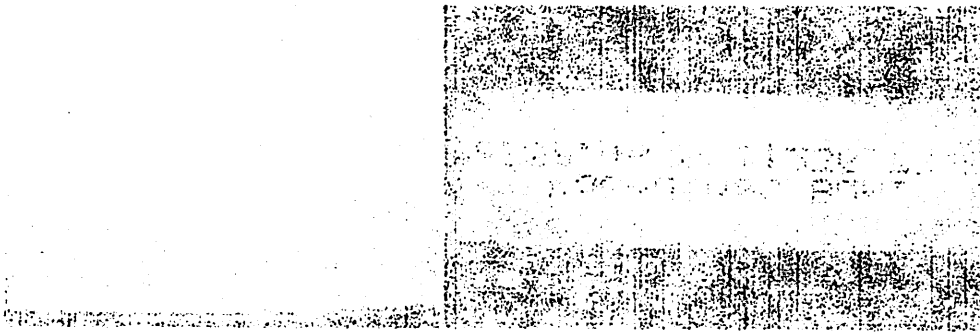


Figure 3. Comparison of the results of the two methods.



Figure 4. Comparison of the results of the two methods.

BAB V

PENUTUP

5.1. Kesimpulan

Kesimpulan yang dapat diambil dari perancangan dan pembuatan alat ini adalah :

- Pada hasil pengujian dapat diambil kesimpulan bahwa GPS dapat menentukan 1-100 titik koordinat, dimana koordinat tersebut akan dipasang alat perangkap ikan atau Bubu sebagai rangjau untuk menangkap ikan.
- Pada perancangan ini GPS memiliki fungsi untuk, memberikan informasi posisi Lintang selatan dan bujur timur, contoh 757.55^0 S dan 11236.72^0 E. dimana informasi tersebut sebagai data yang dikeluarkan dari pin Tx GPS. Kemudian data tersebut dibaca dan diolah oleh mikrokontroler AT89S52 untuk dijadikan sebagai koordinat yang akan ditampilkan pada LCD.
- Untuk pengecekan hasil error, kita melakukan perbandingan GPS standar dengan GPS yang kita buat. pengujian dilakukan pada beberapa tempat dikota Malang, dimana hasil perbandingan error adalah 0.01%

5.2 Saran

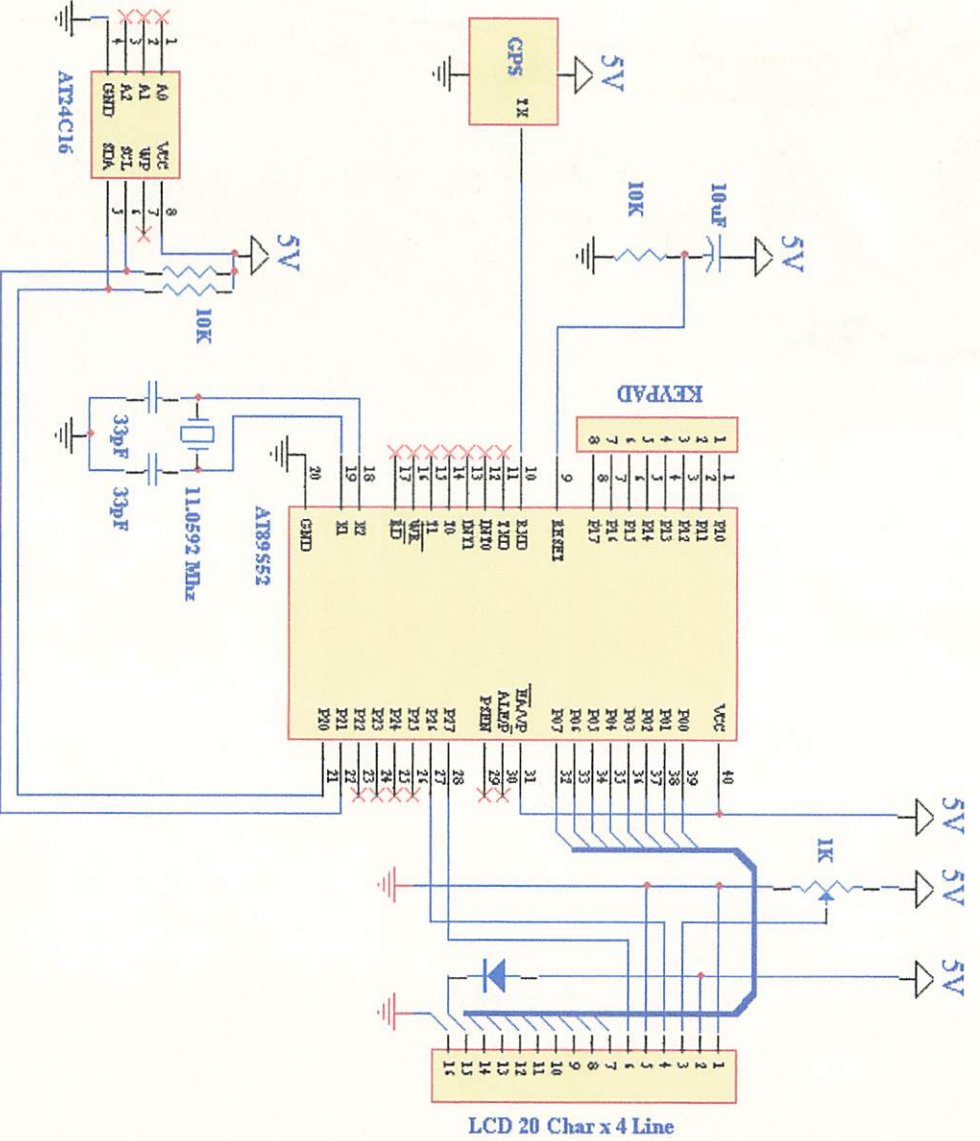
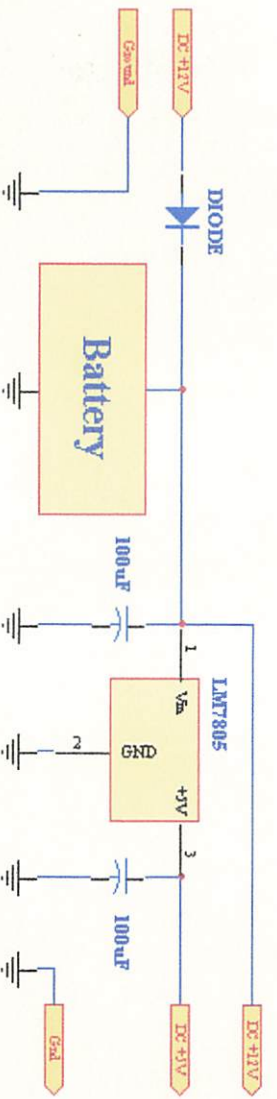
Saran yang akan diberikan untuk perancangan dan pembuatan system ini adalah :

Berhubung karena waktu yang begitu singkat untuk menyelesaikan tugas akhir ini, pengujian ini tidak dilakukan di laut melainkan di area kampus ITN2. Sehingga selanjutnya ada yang ingin menyempurnakan judul ini, sangat disarankan untuk pengujiannya dapat dilakukan langsung dilaut untuk lebih mengetahui akurasi dari peralatan yang dibuat.

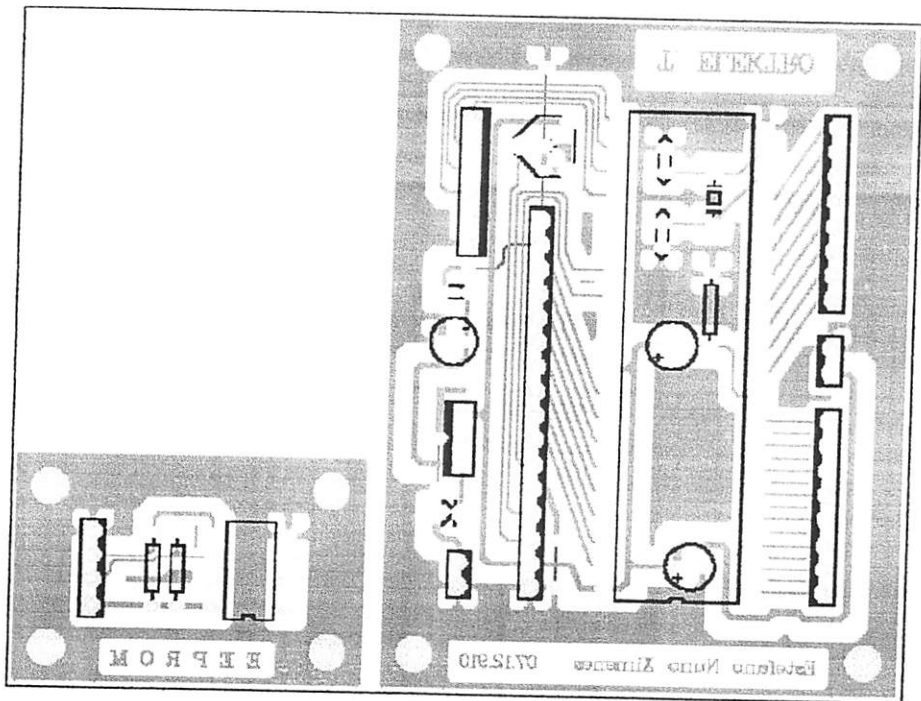
DAFTAR PUSTAKA

1. Datasheet
2. Moh.Ibnu Malik, ST, *Belajar Mikrocontroller AT8952*, Gaya Media, Edisi Pertama, 2003
3. <http://www.aero.org/publications/GPSPRIMER/Elements.html>
4. <http://www.maxim.com.html>
5. <http://www.trimble.com/gps/triangulating.html>
6. <http://www.trimble.com/gps/triangulating2.html>
7. <http://www.trimble.com/gps/triangulating3.html>
8. <http://www.trimble.com/gps/erros2.html>
9. <http://www.trimble.com/gps/erros3.html>
10. <http://www.trimble.com/gps/erros5.html>
11. <http://www.trimble.com/gps/erros6.html>
12. <http://www.google.com>
- 13 Usman, 2009, *Teknik Antar Muka + Pemrograman Mikrocontroller AT89S52*, C.V ANDI OFFSET, Yogyakarta
14. Abidin Z Hasanuddin, *Penentuan Posisi dengan GPS dan Aplikasinya*, PT Pradnya Paramita, Jakarta, 1995
15. Owen Bihop, 2004, *Dasar-Dasar Elektronika*, Erlangga

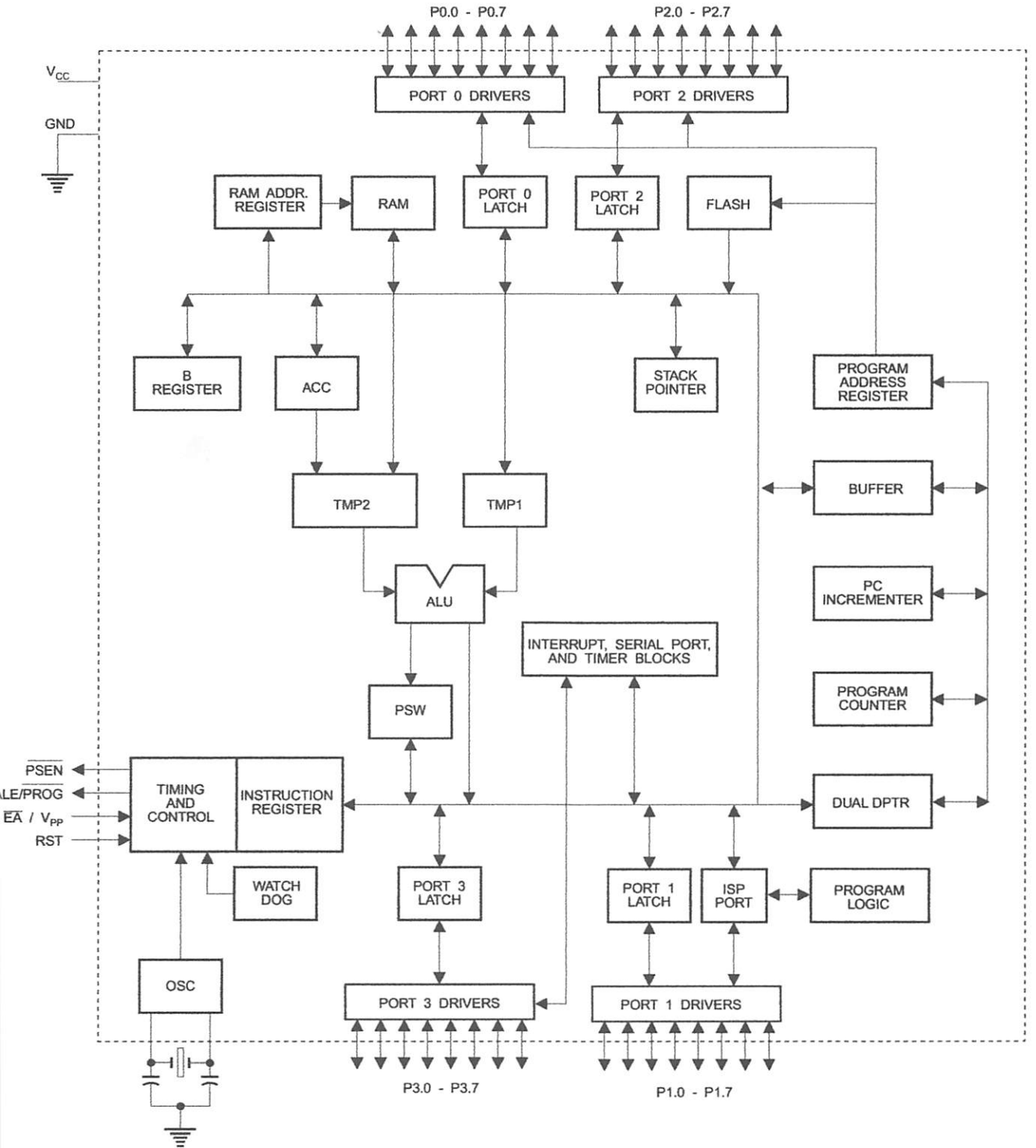
LAMPIRAN

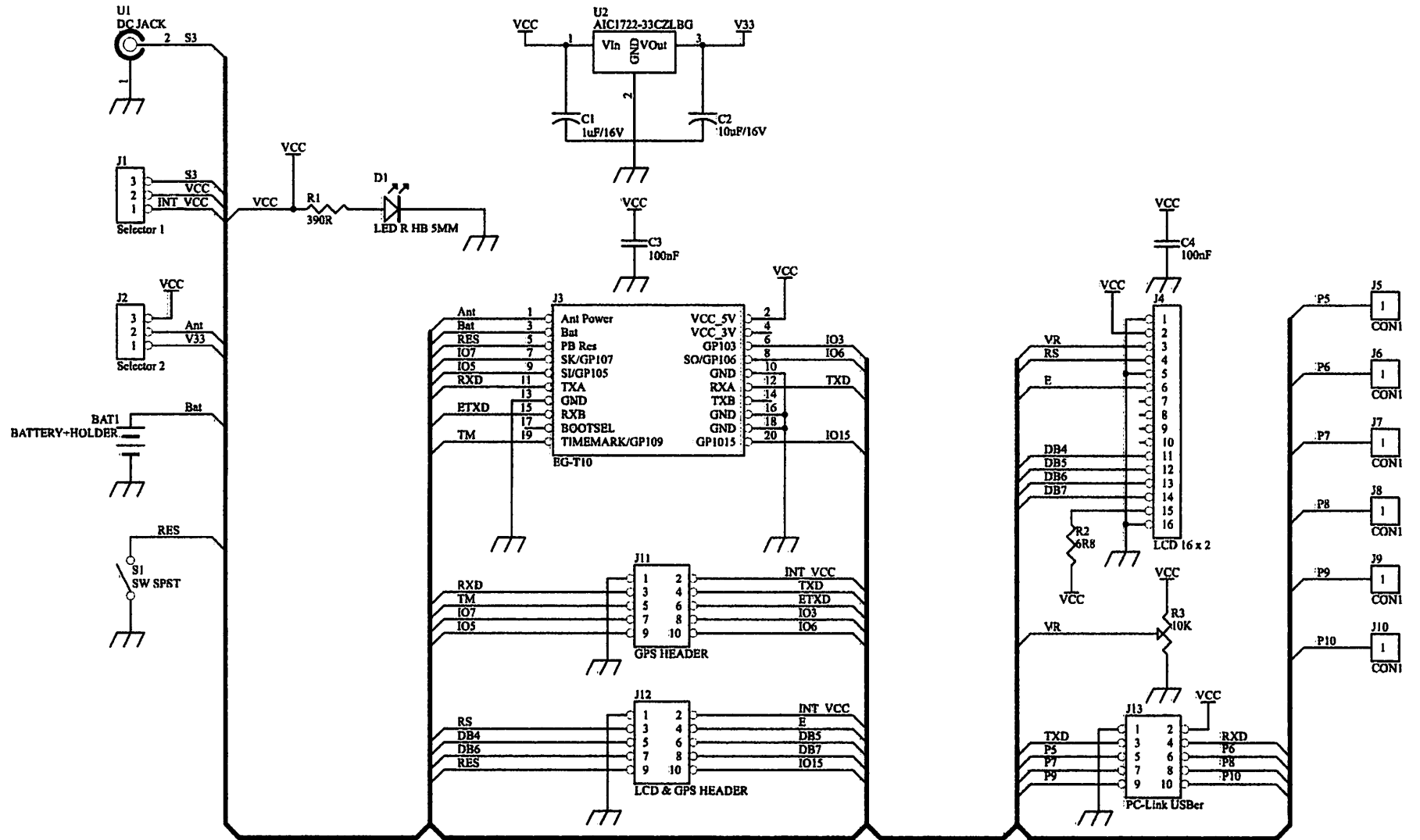


Title		Revision	
Rev	Rev No	Revision	
B			
Date:	17-Nov-2001	Sheet of	
File:	C:\MTC\OPTICAL\ETHER\NO.37H	Drawn By:	



Block Diagram





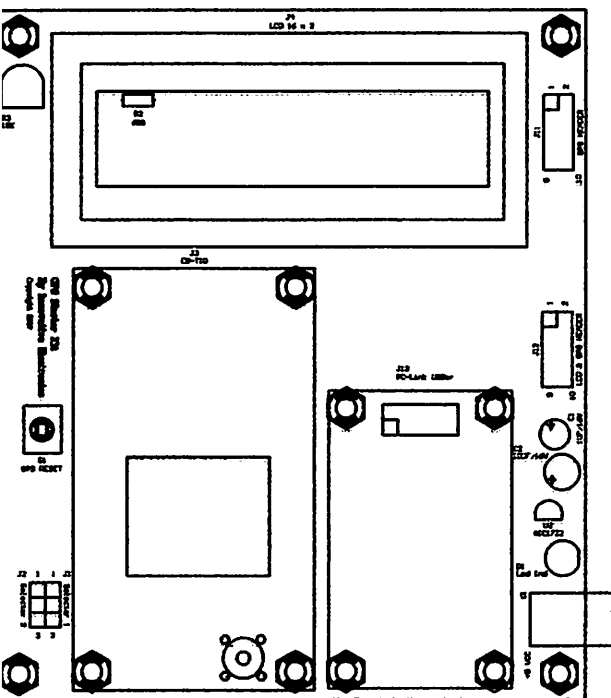
GPS Starter Kit

GPS Starter Kit merupakan suatu sarana pengembangan modul OEM GPS Receiver yang berbasis EG-T10 dan dilengkapi dengan LCD 16 karakter x 2 baris sebagai media tampilan. GPS Starter Kit dapat digunakan untuk mengevaluasi modul OEM GPS Receiver serta aplikasi-aplikasi yang berkaitan dengan GPS, antara lain: GPS navigator, GPS tracking, dsb.

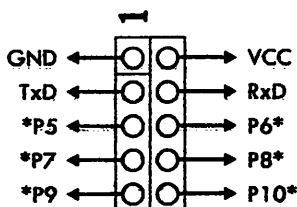
Spesifikasi

- Berbasis EG-T10, modul OEM GPS Receiver 12 channel dengan antarmuka UART-TTL dan memiliki protokol keluaran SIRF binary & NMEA-0183 versi 2.20.
- Mendukung modul OEM GPS Receiver lainnya yang bekerja pada tegangan 5 VDC dan memiliki pinout sesuai.
- Mendukung tegangan active antenna 3,3V dan 5V.
- Tersedia tombol manual reset dan socket untuk backup baterai 3V bertipe 'CR2032'.
- Dilengkapi LCD 16 karakter x 2 baris, rangkaian LCD 4-bit, dan rangkaian VR pengatur kontras LCD.
- Tegangan input catu daya 5 VDC, dilengkapi dengan regulator tegangan 3,3 VDC.
- Kompatibel dengan PC-Link USBer dan DT-51 Low Cost Series atau DT-AVR Low Cost Series, serta mendukung sistem mikrokontroler/mikroprosesor lain.

Letak dan Setting Jumper



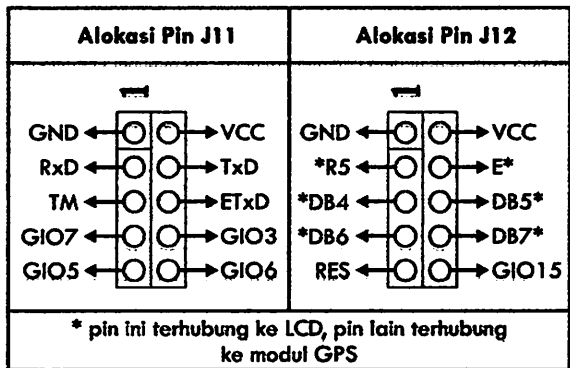
Alokasi Pin J13



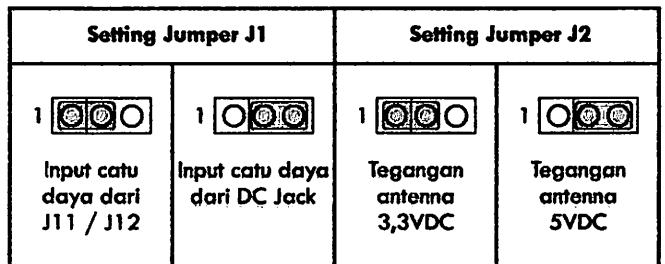
* Pin ini terhubung ke pad P5, P6, P7, P8, P9, atau P10.

J13 digunakan untuk aplikasi dengan PC-Link USBer. P5 hingga P10 memiliki fungsi yang sama dengan pin 5 hingga pin 10 PC-Link USBer.

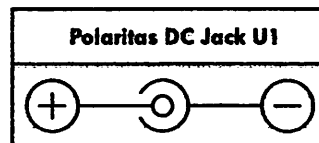
Bila menggunakan PC-Link USBer, maka J11 tidak boleh digunakan. Demikian juga sebaliknya, bila J11 digunakan, maka PC-Link USBer tidak boleh dipasang.



J11 dan J12 digunakan untuk antarmuka antara modul GPS dan LCD dengan rangkaian mikrokontroler atau mikroprosesor. VR R3 dapat diputar untuk mengatur kontras LCD.



Pemilihan sumber input catu daya dan tegangan active antenna dapat dilakukan dengan mengatur jumper J1 dan J2. Jika jumper J1 pada posisi 2-3 (input catu daya dari DC Jack) maka perhatikan polaritas berikut:



Tombol S1 dapat ditekan untuk melakukan manual reset terhadap modul GPS.

Isi CD

- Ccontoh Aplikasi dan Program Testing.
- Manual GPS Starter Kit.
- Datasheet LCD, EG-T10, dan Protokol.
- Website Innovative Electronics.

Prosedur Testing

Pengujian board ini menggunakan DT-51 Low Cost Micro System dan program gps.hex, langkah-langkah testing dapat mengikuti contoh aplikasi yang tersedia (AN GPS.PDF) pada folder Contoh Aplikasi.

* Terima Kasih atas kepercayaan Anda menggunakan produk kami, bila ada kesulitan, pertanyaan atau saran mengenai produk ini silahkan menghubungi technical support kami :

Support@innovativeelectronics.com

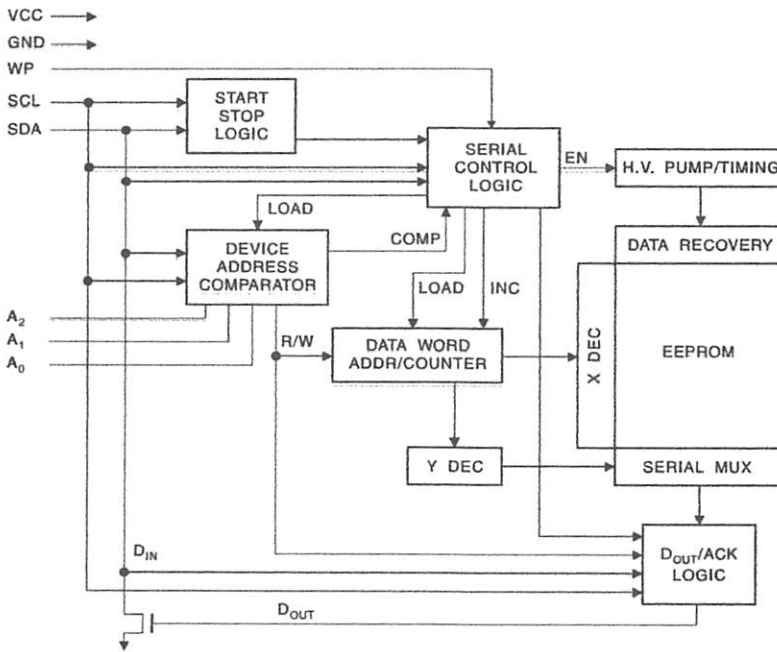


Absolute Maximum Ratings

Operating Temperature.....	-55°C to +125°C
Storage Temperature.....	-65°C to +150°C
Voltage on Any Pin with Respect to Ground.....	-1.0V to +7.0V
Maximum Operating Voltage.....	6.25V
Maximum Output Current.....	5.0 mA

***NOTICE:** Stresses beyond those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions beyond those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Block Diagram



Function Description

SERIAL CLOCK (SCL): The SCL input is used to positive edge clock data into each EEPROM device and negative edge clock data out of each device.

SERIAL DATA (SDA): The SDA pin is bi-directional for serial data transfer. This pin is open-drain driven and may be wire-ORed with any number of other open-drain or open-collector devices.

DEVICE/PAGE ADDRESSES (A2, A1, A0): The A2, A1 and A0 pins are device address inputs that are hard wired for the AT24C01A and the AT24C02. As many as eight 2K devices may be addressed on a single bus system (device addressing is discussed in detail under the Device Addressing section).

The AT24C04 uses the A2 and A1 inputs for hard wire addressing and a total of four 4K devices may be

addressed on a single bus system. The A0 pin is a no connect.

The AT24C08 only uses the A2 input for hardwire addressing and a total of two 8K devices may be addressed on a single bus system. The A0 and A1 pins are no connects.

The AT24C16 does not use the device address pins, which limits the number of devices on a single bus to one. The A0, A1 and A2 pins are no connects.

WRITE PROTECT (WP): The AT24C01A/02/04/16 has a Write Protect pin that provides hardware data protection. The Write Protect pin allows normal read/write operations when connected to ground (GND). When the Write Protect pin is connected to V_{CC}, the write protection feature is enabled and operates as shown in the following table.

AT24C01A/02/04/08/16

Bahasa Pemrograman Assembly Yang Digunakan Oleh Mikrokontroller AT89S52

```

    org    00h
    ljmp   init
;

    org    23h
    clr    ES
    jnb    RI,$
    clr    RI
    mov    R7,SBUF
    setb   ES
    reti
;

ISDA    Bit P2.0           ; I2C data
ISCL    Bit P2.1           ; I2C clock
Rest    Bit P2.6
Enbl    Bit P2.7

Ch00    Equ 30h           ; \
Ch01    Equ 31h           ; |
Ch02    Equ 32h           ; |
Ch03    Equ 33h           ; |
Ch04    Equ 34h           ; |
Ch05    Equ 35h           ; |
Ch06    Equ 36h           ; |
Ch07    Equ 37h           ; | nama memory
(character)
Ch08    Equ 38h           ; | untuk baca gps
Ch09    Equ 39h           ; |
Ch0A    Equ 3Ah           ; |
Ch0B    Equ 3Bh           ; |
Ch0C    Equ 3Ch           ; |
Ch0D    Equ 3Dh           ; |
Ch0E    Equ 3Eh           ; |
Ch0F    Equ 3Fh           ; /

Ch10    Equ 40h           ; \
Ch11    Equ 41h           ; |
Ch12    Equ 42h           ; |
Ch13    Equ 43h           ; |
Ch14    Equ 44h           ; |
Ch15    Equ 45h           ; |
Ch16    Equ 46h           ; |
Ch17    Equ 47h           ; | nama memory
(character)
Ch18    Equ 48h           ; | untuk memory gps
Ch19    Equ 49h           ; |
Ch1A    Equ 4Ah           ; |
Ch1B    Equ 4Bh           ; |
Ch1C    Equ 4Ch           ; |
Ch1D    Equ 4Dh           ; |
Ch1E    Equ 4Eh           ; |
Ch1F    Equ 4Fh           ; /

Urut    Equ 50h           ;urut address

```



```

        Adbm      Equ 51h                ; address base
        Admm      Equ 52h                ; address memory
        Dtmm      Equ 53h                ; data memory

        Char      Equ 60h
        Cntr      Equ 61h
        Bufr      Equ 62h
        Tmot      Equ 63h
        Tmo0      Equ 64h
        Tmo1      Equ 65h
        Dly0      Equ 66h
        Dly1      Equ 67h
        Dly2      Equ 68h
        Dly3      Equ 69h

;
init:   lcall    lcd_in
        lcall    srl_in

;
mulai:  mov      DPTR,#tpnama
        lcall    line1
        mov      Char,#20
        lcall    tulis
        mov      DPTR,#tpnims
        lcall    line2
        mov      Char,#20
        lcall    tulis
        mov      DPTR,#tpjurs
        lcall    line3
        mov      Char,#20
        lcall    tulis
        mov      DPTR,#tpuniv
        lcall    line4
        mov      Char,#20
        lcall    tulis
        lcall    line5
        lcall    delay2
        mov      DPTR,#tpjdl0
        lcall    line1
        mov      Char,#20
        lcall    tulis
        mov      DPTR,#tpjdl1
        lcall    line2
        mov      Char,#20
        lcall    tulis
        mov      DPTR,#tpjdl2
        lcall    line3
        mov      Char,#20
        lcall    tulis
        mov      DPTR,#tpjdl3
        lcall    line4
        mov      Char,#20
        lcall    tulis
        lcall    line5
        lcall    delay2

;
measmn: lcall    lcdclr

```

```

        mov     DPTR,#tpkrbm
        lcall  line2
        mov     Char,#20
        lcall  tulis
        lcall  line5
        mov     DPTR,#angka
mesmn0: lcall  bc_rmc           ; baca GPS
        lcall  line3
        lcall  tlgps0       ; tulis GPS
        lcall  line5
        lcall  delay1
        sjmp   mesmn0
;
tnmjrg: mov     Urut,#0
        lcall  lcdclr
        mov     DPTR,#tptnjr
        lcall  line2
        mov     Char,#20
        lcall  tulis
        lcall  line5
        mov     Dly3,#8
        lcall  delay3
        lcall  lcdclr
        mov     Tmot,#00
tnjr00: mov     DPTR,#tpkrbm
        lcall  line1
        mov     Char,#20
        lcall  tulis
        mov     DPTR,#tpkrjr
        lcall  line3
        mov     Char,#20
        lcall  tulis
tnjr01: mov     DPTR,#angka
        mov     P0,#0A5h
        lcall  w_ins
        mov     A,Urut
        inc    A
        lcall  nilai
        mov     DPTR,#lookup
        mov     A,Urut
        mov     B,#2
        mul   AB
        mov     Bufr,A
        movc   A,@A+DPTR
        mov     Adbm,A
        inc   Bufr
        mov     A,Bufr
        movc   A,@A+DPTR
        mov     Admm,A
        lcall  rdmdta
        lcall  line4
        lcall  tlgps1       ; tulis data GPS
        lcall  line5
        mov     Dly3,#1
        lcall  delay3
        ljmp   tnjr03

```

```

;
tnjr02: lcall    bc_rmc           ; baca GPS
        lcall    line2
        lcall    tlgps0         ; tulis memory GPS
        lcall    line5

;
tnjr03: lcall    scnkpdc
        cjne    R0,#11,tnjr04
        mov     SP,#07h
        ljmp    mulai
tnjr04: cjne    R0,#12,tnjr05
        ljmp    svkrdn
tnjr05: cjne    R0,#15,tnjr07
        inc     Urut
        mov     A,Urut
        cjne    A,#100,tnjr06
        mov     Urut,#099
tnjr06: mov     Tmot,#15
        ljmp    tnjr01
tnjr07: cjne    R0,#16,tnjr09
        dec     Urut
        mov     A,Urut
        cjne    A,#255,tnjr08
        mov     Urut,#000
tnjr08: mov     Tmot,#15
        ljmp    tnjr01
tnjr09: mov     A,Tmot
        jz     tnjr02
tnjr0A: djnz    Tmot,tnjr0B
tnjr0B: ljmp    tnjr01

;
svkrdn: mov     Ch10,Ch00
        mov     Ch11,Ch01
        mov     Ch12,Ch02
        mov     Ch13,Ch03
        mov     Ch14,Ch04
        mov     Ch15,Ch05
        mov     Ch16,Ch06
        mov     Ch17,Ch07
        mov     Ch18,Ch08
        mov     Ch19,Ch09
        mov     Ch1A,Ch0A
        mov     Ch1B,Ch0B
        mov     Ch1C,Ch0C
        mov     Ch1D,Ch0D
        mov     Ch1E,Ch0E
        mov     Ch1F,Ch0F
        mov     DPTR,#tpsvng
        lcall    line1
        mov     Char,#20
        lcall    tulis
        mov     DPTR,#lookup
        mov     A,Urut
        mov     B,#2
        mul     AB
        mov     BufR,A

```

```

        movc    A,@A+DPTR
        mov     Adbm,A
        inc    Bufr
        mov     A,Bufr
        movc    A,@A+DPTR
        mov     Admm,A
        lcall  wrmdta
        lcall  tg_lps
        mov     Tmot,#15
        ljmp   tnjr00
;
lpjrg:  mov     Urut,#0
        lcall  lcdclr
        mov     DPTR,#tplpjr
        lcall  line2
        mov     Char,#20
        lcall  tulis
        lcall  line5
        mov     Dly3,#8
        lcall  delay3
        lcall  lcdclr
lpjr00: mov     Tmot,#00
        mov     DPTR,#tpkrbm
        lcall  line1
        mov     Char,#20
        lcall  tulis
        mov     DPTR,#tpkrjr
        lcall  line3
        mov     Char,#20
        lcall  tulis
lpjr01: mov     DPTR,#angka
        mov     P0,#0A5h
        lcall  w_ins
        mov     A,Urut
        inc    A
        lcall  nilai
        mov     DPTR,#lookup
        mov     A,Urut
        mov     B,#2
        mul    AB
        mov     Bufr,A
        movc    A,@A+DPTR
        mov     Adbm,A
        inc    Bufr
        mov     A,Bufr
        movc    A,@A+DPTR
        mov     Admm,A
        lcall  rdmdta
        lcall  line4
        lcall  tlgps1           ; tulis data GPS
        lcall  line5
        mov     Dly3,#1
        lcall  delay3
        ljmp   lpjr03
;
lpjr02: lcall  bc_rmc           ; baca GPS

```

```

        lcall    line2
        lcall    tlgps0           ; tulis memory GPS
        lcall    line5
;
lpjr03: lcall    scnkpd
        cjne    R0,#11,lpjr04
        mov     SP,#07h
        ljmp    mulai
lpjr04: cjne    R0,#12,lpjr05
        ljmp    rskrdn
lpjr05: cjne    R0,#15,lpjr07
        inc     Urut
        mov     A,Urut
        cjne    A,#100,lpjr06
        mov     Urut,#099
lpjr06: mov     Tmot,#15
        ljmp    lpjr01
lpjr07: cjne    R0,#16,lpjr09
        dec     Urut
        mov     A,Urut
        cjne    A,#255,lpjr08
        mov     Urut,#000
lpjr08: mov     Tmot,#15
        ljmp    lpjr01
lpjr09: mov     A,Tmot
        jz     lpjr02
lpjr0A: djnz   Tmot,lpjr0B
lpjr0B: ljmp    lpjr01
;
rskrdn: mov     Ch10,#'- '
        mov     Ch11,#'- '
        mov     Ch12,#'- '
        mov     Ch13,#'- '
        mov     Ch14,#'- '
        mov     Ch15,#'- '
        mov     Ch16,#'- '
        mov     Ch17,#'- '
        mov     Ch18,#'- '
        mov     Ch19,#'- '
        mov     Ch1A,#'- '
        mov     Ch1B,#'- '
        mov     Ch1C,#'- '
        mov     Ch1D,#'- '
        mov     Ch1E,#'- '
        mov     Ch1F,#'- '
        mov     DPTR,#tprsng
        lcall   line1
        mov     Char,#20
        lcall   tulis
        mov     DPTR,#lookup
        mov     A,Urut
        mov     B,#2
        mul    AB
        mov    Bufr,A
        movc   A,@A+DPTR
        mov    Adbm,A

```

```

        inc     Bufr
        mov     A,Bufr
        movc   A,@A+DPTR
        mov     Admm,A
        lcall  wrmdta
        lcall  tg_lps
        mov     Tmot,#15
        ljmp   lpjr00
;
bc_rmc: lcall  bc_fbk           ;\
        cjne   R7,'#R',bc_rmc ; | baca gps
        mov     Cntr,#16       ; | yang diawali character R
bcrmc:  lcall  bc_srl         ; | 17 character dihilangkan
        djnz   Cntr,bcrmc     ;/
        lcall  bc_srl
        mov     Ch00,R7
        lcall  bc_srl
        mov     Ch01,R7
        lcall  bc_srl
        mov     Ch02,R7
        lcall  bc_srl
        mov     Ch03,R7
        lcall  bc_srl
        lcall  bc_srl
        mov     Ch04,R7
        lcall  bc_srl
        mov     Ch05,R7
        lcall  bc_srl
        lcall  bc_srl
        lcall  bc_srl
        lcall  bc_srl
        mov     Ch06,R7       ; asli data lintang (S)
        lcall  bc_srl
        lcall  bc_srl
;
        mov     Ch07,R7
        lcall  bc_srl
        mov     Ch08,R7
        lcall  bc_srl
        mov     Ch09,R7
        lcall  bc_srl
        mov     Ch0A,R7
        lcall  bc_srl
        mov     Ch0B,R7
        lcall  bc_srl
        lcall  bc_srl
        mov     Ch0C,R7
        lcall  bc_srl
        mov     Ch0D,R7
        lcall  bc_srl
        lcall  bc_srl
        lcall  bc_srl
        lcall  bc_srl
        mov     Ch0E,R7       ; asli data bujur (E)
        ret
;

```

```

tlgps0: mov     P0,Ch00
        lcall  w_chr
        mov     P0,Ch01
        lcall  w_chr
        mov     P0,Ch02
        lcall  w_chr
        mov     P0,Ch03
        lcall  w_chr
        mov     P0,#'.'
        lcall  w_chr
        mov     P0,Ch04
        lcall  w_chr
        mov     P0,Ch05
        lcall  w_chr
        mov     P0,#' '
        lcall  w_chr
        mov     P0,Ch06
        lcall  w_chr
        mov     P0,#' '
        lcall  w_chr
        mov     P0,Ch07
        lcall  w_chr
        mov     P0,Ch08
        lcall  w_chr
        mov     P0,Ch09
        lcall  w_chr
        mov     P0,Ch0A
        lcall  w_chr
        mov     P0,Ch0B
        lcall  w_chr
        mov     P0,#'.'
        lcall  w_chr
        mov     P0,Ch0C
        lcall  w_chr
        mov     P0,Ch0D
        lcall  w_chr
        mov     P0,#' '
        lcall  w_chr
        mov     P0,Ch0E
        lcall  w_chr
        ret

```

```

;
tlgps1: mov     P0,Ch10
        lcall  w_chr
        mov     P0,Ch11
        lcall  w_chr
        mov     P0,Ch12
        lcall  w_chr
        mov     P0,Ch13
        lcall  w_chr
        mov     P0,#'.'
        lcall  w_chr
        mov     P0,Ch14
        lcall  w_chr
        mov     P0,Ch15
        lcall  w_chr

```

```

    mov     P0, #' '
    lcall   w_chr
    mov     P0, Ch16
    lcall   w_chr
    mov     P0, #' '
    lcall   w_chr
    mov     P0, Ch17
    lcall   w_chr
    mov     P0, Ch18
    lcall   w_chr
    mov     P0, Ch19
    lcall   w_chr
    mov     P0, Ch1A
    lcall   w_chr
    mov     P0, Ch1B
    lcall   w_chr
    mov     P0, #'.'
    lcall   w_chr
    mov     P0, Ch1C
    lcall   w_chr
    mov     P0, Ch1D
    lcall   w_chr
    mov     P0, #' '
    lcall   w_chr
    mov     P0, Ch1E
    lcall   w_chr
    ret

;
wrmdta: mov     Dtmn, Ch10           ; data memory
    lcall   wr_mem
    inc     Admn
    mov     Dtmn, Ch11           ; data memory
    lcall   wr_mem
    inc     Admn
    mov     Dtmn, Ch12           ; data memory
    lcall   wr_mem
    inc     Admn
    mov     Dtmn, Ch13           ; data memory
    lcall   wr_mem
    inc     Admn
    mov     Dtmn, Ch14           ; data memory
    lcall   wr_mem
    inc     Admn
    mov     Dtmn, Ch15           ; data memory
    lcall   wr_mem
    inc     Admn
    mov     Dtmn, Ch16           ; data memory
    lcall   wr_mem
    inc     Admn
    mov     Dtmn, Ch17           ; data memory
    lcall   wr_mem
    inc     Admn
    mov     Dtmn, Ch18           ; data memory
    lcall   wr_mem
    inc     Admn
    mov     Dtmn, Ch19           ; data memory

```



```

        lcall    wr_mem
        inc     Admm
        mov     Dtmn,Ch1A           ; data memory
        lcall    wr_mem
        inc     Admm
        mov     Dtmn,Ch1B           ; data memory
        lcall    wr_mem
        inc     Admm
        mov     Dtmn,Ch1C           ; data memory
        lcall    wr_mem
        inc     Admm
        mov     Dtmn,Ch1D           ; data memory
        lcall    wr_mem
        inc     Admm
        mov     Dtmn,Ch1E           ; data memory
        lcall    wr_mem
        inc     Admm
        mov     Dtmn,Ch1F           ; data memory
        lcall    wr_mem
        ret
;
rdmdta: mov     A,Adbm              ; AT24C16 write address
        lcall    adrtx
        mov     A,Admm              ; address memory
        lcall    dtatx
        inc     Adbm
        mov     A,Adbm              ; AT24C16 read address
        lcall    adrtx
        lcall    dtarx              ; terima data
        mov     Ch10,A
        lcall    givack             ; beri ack
        lcall    dtarx              ; terima data
        mov     Ch11,A
        lcall    givack             ; beri ack
        lcall    dtarx              ; terima data
        mov     Ch12,A
        lcall    givack             ; beri ack
        lcall    dtarx              ; terima data
        mov     Ch13,A
        lcall    givack             ; beri ack
        lcall    dtarx              ; terima data
        mov     Ch14,A
        lcall    givack             ; beri ack
        lcall    dtarx              ; terima data
        mov     Ch15,A
        lcall    givack             ; beri ack
        lcall    dtarx              ; terima data
        mov     Ch16,A
        lcall    givack             ; beri ack
        lcall    dtarx              ; terima data
        mov     Ch17,A
        lcall    givack             ; beri ack
        lcall    dtarx              ; terima data
        mov     Ch18,A
        lcall    givack             ; beri ack
        lcall    dtarx              ; terima data

```

```

        mov     Ch19,A
        lcall  givack           ; beri ack
        lcall  dtarx           ; terima data
        mov     Ch1A,A
        lcall  givack           ; beri ack
        lcall  dtarx           ; terima data
        mov     Ch1B,A
        lcall  givack           ; beri ack
        lcall  dtarx           ; terima data
        mov     Ch1C,A
        lcall  givack           ; beri ack
        lcall  dtarx           ; terima data
        mov     Ch1D,A
        lcall  givack           ; beri ack
        lcall  dtarx           ; terima data
        mov     Ch1E,A
        lcall  givack           ; beri ack
        lcall  dtarx           ; terima data
        mov     Ch1F,A
        lcall  i2cstp          ; i2c stop
        ret

;
wr_mem: mov     A,Adbm         ; AT24C16 write address
        lcall  adrtrx
        mov     A,Admm         ; address hardware memory
        lcall  dtatx
        mov     A,Dtmm         ; data memory
        lcall  dtatx
        lcall  i2cstp          ; i2c stop
        lcall  wt_wr
        ret

;
wt_wr:  mov     Dly0,#000
        mov     Dly1,#025
wtwr:   lcall  delay0
        djnz   Dly1,wtwr
        ret

;
adrtrx: lcall  i2cstr         ; send address
        lcall  putbit
        ret

;
dtatx:  lcall  putbit         ; send data
        ret

;
dtarx:  lcall  getbit         ; get data
        ret

;
putbit: mov     R6,#8
putbt:  RLC     A
        mov     ISDA,C
        setb   ISCL
        clr    ISCL
        djnz   R6,putbt
        setb   ISDA
        lcall  getack

```

```

        ret
;
getbit: mov     R6,#8
getbt:  setb    ISCL
        mov     C,ISDA
        RLC     A
        clr     ISCL
        djnz   R6,getbt
        setb    ISDA
        ret
;
getack: setb    ISDA                ; tunggu ack
        setb    ISCL                ; \
ackbit: mov     C,ISDA              ; | D=1, C=1
        jc      ackbit              ; | tunggu D=0, C=0
        clr     ISCL                ; /
        ret
;
givack: clr     ISDA                ; kirim ack
        setb    ISCL                ; \
        clr     ISCL                ; | D=0, C=1, C=0, D=1
        setb    ISDA                ; /
        ret
;
i2cstr: setb    ISCL                ; i2c start
        setb    ISDA                ; \
        clr     ISDA                ; | C=1, D=1, D=0, C=0
        clr     ISCL                ; /
        ret
;
i2cstp: clr     ISDA                ; i2c stop
        setb    ISCL                ; \ D=0, C=1, D=1, C=0
        setb    ISDA                ; /
        clr     ISCL
        ret
;
srl_in: mov     Dly3,#1
        lcall   delay3
        mov     TMOD,#25h
        mov     TH1,#0FAh
        mov     SCON,#50h
        setb    TR1
        setb    ES
        setb    EA
        ret
;
bc_fbk: lcall   rstcmd                ; \
bc_fb0: cjne   R7,#0FFh,bc_fb1        ; | tunggu feedback
        ljmp   bc_fb0                ; | data dari GPS
bc_fb1: ret
;
bc_srl: lcall   rstcmd                ; \
        mov     Tmo0,#10              ; |
        mov     Tmo1,#00              ; | baca serial
bc_srl0: cjne  R7,#0FFh,bcsrl1        ; | tunggu data bukan FF
        djnz   Tmo1,bcsrl0           ; | sebelum time out

```

```

        djnz     Tmo0,bcsrl0      ; |
bcsrl1: ret                    ;/
;
rstcmd: mov     R7,#0FFh
        ret
;
nilai:  mov     B,#100
        div     AB
        lcall   wr_chr
        mov     A,B
        mov     B,#10
        div     AB
        lcall   wr_chr
        mov     A,B
        lcall   wr_chr
        ret
;
line1:  mov     P0,#080h
        lcall   w_ins
        ret
;
line2:  mov     P0,#0C0h
        lcall   w_ins
        ret
;
line3:  mov     P0,#094h
        lcall   w_ins
        ret
;
line4:  mov     P0,#0D4h
        lcall   w_ins
        ret
;
line5:  mov     P0,#0F0h
        lcall   w_ins
        ret
;
tulis:  clr     A
        lcall   wr_chr
        inc     DPTR
        djnz   Char,tulis
        ret
;
wr_chr: movc    A,@A+DPTR
        mov     P0,A
        lcall   w_chr
        ret
;
w_ins:  clr     Enbl
        clr     Rest
        setb   Enbl
        clr     Enbl
        mov     Dly0,#100
        lcall   delay0
        ret
;

```

```

w_chr:  clr      Enbl
        setb     Rest
        setb     Enbl
        clr      Enbl
        mov      Dly0,#100
        lcall    delay0
        ret

;
lcd_in: mov      Dly3,#1
        lcall    delay3
        mov      P0,#01h           ; Display Clear
        lcall    w_ins
        mov      P0,#38h           ; Function Set
        lcall    w_ins
        mov      P0,#0Dh           ; Display On, Cursor,
        Blink
        lcall    w_ins
        mov      P0,#06h           ; Entry Mode
        lcall    w_ins
        mov      P0,#02h           ; Cursor Home
        lcall    w_ins
        ret

;
lcdclr: mov      P0,#01h           ; Display Clear
        lcall    w_ins
        lcall    delay0
        lcall    delay0
        lcall    delay0
        ret

;
scnkpd: mov      R0,#10
        lcall    delay0
coll:   mov      P1,#11111110b
        mov      A,P1
c1b1:   cjne     A,#11101110b,c1b2
        mov      R0,#1
c1b2:   cjne     A,#11011110b,c1b3
        mov      R0,#2
c1b3:   cjne     A,#10111110b,c1b4
        mov      R0,#3
c1b4:   cjne     A,#01111110b,col2
        mov      R0,#13

;
col2:   mov      P1,#11111101b
        mov      A,P1
c2b1:   cjne     A,#11101101b,c2b2
        mov      R0,#4
c2b2:   cjne     A,#11011101b,c2b3
        mov      R0,#5
c2b3:   cjne     A,#10111101b,c2b4
        mov      R0,#6
c2b4:   cjne     A,#01111101b,col3
        mov      R0,#14

;
col3:   mov      P1,#11111011b
        mov      A,P1

```

```

c3b1:  cjne    A,#11101011b,c3b2
        mov     R0,#7
c3b2:  cjne    A,#11011011b,c3b3
        mov     R0,#8
c3b3:  cjne    A,#10111011b,c3b4
        mov     R0,#9
c3b4:  cjne    A,#01111011b,col4
        mov     R0,#15
;
col4:  mov     P1,#11110111b
        mov     A,P1
c4b1:  cjne    A,#11100111b,c4b2
        mov     R0,#11
c4b2:  cjne    A,#11010111b,c4b3
        mov     R0,#0
c4b3:  cjne    A,#10110111b,c4b4
        mov     R0,#12
c4b4:  cjne    A,#01110111b,back
        mov     R0,#16
back:  ret
;
tg_tkn: lcall   scnkpdc
        lcall   delay0
tg_tk0: cjne    R0,#16,tg_tk1
        ljmp   tg_tkn
tg_tk1: cjne    R0,#15,tg_tk2
        ljmp   tg_tkn
tg_tk2: cjne    R0,#14,tg_tk3
        ljmp   tg_tkn
tg_tk3: cjne    R0,#13,tg_tk4
        ljmp   tg_tkn
tg_tk4: cjne    R0,#12,tg_tk5
        ljmp   tg_tkn
tg_tk5: cjne    R0,#11,tg_tk6
        ljmp   tg_tkn
tg_tk6: cjne    R0,#10,tg_tk7
        ljmp   tg_tkn
tg_tk7: ret
;
tg_lps: lcall   scnkpdc
        lcall   delay0
        cjne    R0,#10,tg_lps
        ret
;
delay0: djnz   Dly0,delay0
        ret
;
delay1: lcall   scnkpdc
        cjne    R0,#12,dely10
;
dely10: cjne    R0,#13,dely11
        ljmp   tnmjrg
dely11: cjne    R0,#14,dely12
        ljmp   lpsjrg
dely12: cjne    R0,#16,dely13

```

```

dely13: djnz    Dly1,delay1
        ret
;
delay2: mov     Dly2,#20
dely2:  lcall   delay1
        djnz    Dly2,dely2
        ret
;
delay3: lcall   delay0
        djnz    Dly1,delay3
        djnz    Dly3,delay3
        ret
;
tpnama: DB      ' Estefano N Ximenes '
tpnims: DB      '   NIM 07.12.910   '
tpjurs: DB      '   T. Elektro   '
tpuniv: DB      '   ITN Malang   '
tpjdl0: DB      ' Aplikasi GPS pada '
tpjdl1: DB      ' Kapal Nelayan   '
tpjdl2: DB      ' Untuk Penentu   '
tpjdl3: DB      ' Posisi Bubu     '
tpkrbm: DB      ' Koordinat Bumi   '

tptnjr: DB      '   Tanam Jaring   '
tplpjr: DB      '   Lepas Jaring   '
tpkrjr: DB      'Koordinat Jaring 050'
tpkrnl: DB      '0758.96 S 11239.45 E'

tpsvng: DB      '   Saving ...   '
tprsng: DB      '   Resetting .... '
angka:  DB      '0123456789   '
;
-----
-----
lookup: DB
0A0h,000h,0A0h,010h,0A0h,020h,0A0h,030h,0A0h,040h,0A0h,050h,0A0h,
060h,0A0h,07
0h
DB
0A0h,080h,0A0h,090h,0A0h,0A0h,0A0h,0B0h,0A0h,0C0h,0A0h,0D0h,0A0h,
0E0h,0A0h,0F
0h
DB
0A2h,000h,0A2h,010h,0A2h,020h,0A2h,030h,0A2h,040h,0A2h,050h,0A2h,
060h,0A2h,07
0h
DB
0A2h,080h,0A2h,090h,0A2h,0A0h,0A2h,0B0h,0A2h,0C0h,0A2h,0D0h,0A2h,
0E0h,0A2h,0F
0h
DB
0A4h,000h,0A4h,010h,0A4h,020h,0A4h,030h,0A4h,040h,0A4h,050h,0A4h,
060h,0A4h,07
0h
DB

```

```
0A4h,080h,0A4h,090h,0A4h,0A0h,0A4h,0B0h,0A4h,0C0h,0A4h,0D0h,0A4h,
    0E0h,0A4h,0F
0h
    DB
0A6h,000h,0A6h,010h,0A6h,020h,0A6h,030h,0A6h,040h,0A6h,050h,0A6h,
    060h,0A6h,07
0h
    DB
0A6h,080h,0A6h,090h,0A6h,0A0h,0A6h,0B0h,0A6h,0C0h,0A6h,0D0h,0A6h,
    0E0h,0A6h,0F
0h
    DB
0A8h,000h,0A8h,010h,0A8h,020h,0A8h,030h,0A8h,040h,0A8h,050h,0A8h,
    060h,0A8h,07
0h
    DB
0A8h,080h,0A8h,090h,0A8h,0A0h,0A8h,0B0h,0A8h,0C0h,0A8h,0D0h,0A8h,
    0E0h,0A8h,0F
0h
    DB
0AAh,000h,0AAh,010h,0AAh,020h,0AAh,030h,0AAh,040h,0AAh,050h,0AAh,
    060h,0AAh,07
0h
    DB
0AAh,080h,0AAh,090h,0AAh,0A0h,0AAh,0B0h,0AAh,0C0h,0AAh,0D0h,0AAh,
    0E0h,0AAh,0F
0h
    DB
0ACh,000h,0ACh,010h,0ACh,020h,0ACh,030h,0ACh,040h,0ACh,050h,0ACh,
    060h,0ACh,07
0h
    DB
0ACh,080h,0ACh,090h,0ACh,0A0h,0ACh,0B0h,0ACh,0C0h,0ACh,0D0h,0ACh,
    0E0h,0ACh,0F
0h
    DB
0AEh,000h,0AEh,010h,0AEh,020h,0AEh,030h,0AEh,040h,0AEh,050h,0AEh,
    060h,0AEh,07
0h
    DB
0AEh,080h,0AEh,090h,0AEh,0A0h,0AEh,0B0h,0AEh,0C0h,0AEh,0D0h,0AEh,
    0E0h,0AEh,0F
0h
;
```

end



FORMULIR BIMBINGAN SKRIPSI

Nama : ESTEFANO NUNO XIMENES
Nim : 07.12.910
Masa Bimbingan : 3 MEI 2010 s/d 3 NOVEMBER 2010
Judul Skripsi : PERANCANGAN DAN PEMBUATAN SISTEM PEMANTAU POSISI
KAPAL NELAYAN DENGAN GPS UNTUK PENENTU POSISI BUBU
BERBASIS MIKROKONTROLLER AT89S52

No.	Tanggal	Uraian	Paraf Pembimbing
1.	05/25/2010	Revisi BAB II dan BAB III	
2.	06/11/2010	ACC BAB II dan III	
3.	06/22/2010	Maju alat, BAB II dan III	
4.	06/28/210	ACC alat maju BAB IV , revisi BAB IV	
5.	07/05/2010	ACC BAB IV	
6.	07/18/2010	ACC keseluruhan Skripsi	
7.	07/22/2010	Maju makalah Skripsi dan TTD makalah skripsi	

Malang 22 juli 2010

Dosen Pembimbing

I Komang Somawirata, ST. MT.

NIP.P. 1030100361

Form S-4B



PERKUMPULAN PENGELOLAH PENDIDIKAN DAN TEKNOLOGI NASIONAL MALANG

INSTITUT TEKNOLOGI NASIONAL MALANG

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

ERSERO) MALANG
NIAGA MALANG

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

FORMULIR PERBAIKAN SKRIPSI

Dari hasil Komprehensip Jenjang Strata Satu (S-1) Jurusan Teknik Elektro Konsentrasi Teknik Elektronika yang diselenggarakan pada :

Hari : Sabtu
Tanggal : 21 Agustus 2010

Telah dilaksanakan Perbaikan Skripsi oleh Saudara

Nama : Estefano Nuno Ximenes
N.I.M : 07 12 910
Masa Bimbingan : 6 Bulan (03 Mei – 03 November)
Judul Skripsi : “Perancangan dan Pembuatan Sistem Pemantau Posisi Kapal Nelayan

Dengan GPS Untuk Penentu Posisi Bubu Berbasis Mikrokontroller

AT89S52”

Tanggal	Dosen Penguji	Uraian	Paraf
21-8-2010	Penguji I	<ul style="list-style-type: none">- Cantumkan nama penulis buku tentang GPS pada latar belakang.- Port yang digunakan oleh memory external- Cantumkan program baca GPS, I2C protocol yang digunakan pada system perancangan di bab IV.- Perbaiki Gambar Rangkaian dan lampiran.	
21-8-2010	Penguji II	Revisi Kesimpulan	

**Mengetahui
Dosen Pembimbing**

I Komang Somawirata, ST. MT.
NIP.P. 1030100361

Penguji I

Sonny Prasetio, ST. MT

Penguji II

Irmalia Suryani Faradisa, ST. MT

NIP. P 1030100365

THE UNIVERSITY OF CHICAGO
DEPARTMENT OF CHEMISTRY
5700 SOUTH CAMPUS DRIVE
CHICAGO, ILLINOIS 60637

RECEIVED
JAN 15 1964
1964

TO: [Illegible]
FROM: [Illegible]
SUBJECT: [Illegible]

[Illegible text block]

[Illegible signature and stamp]

[Illegible text at bottom left]

[Illegible text at bottom right]