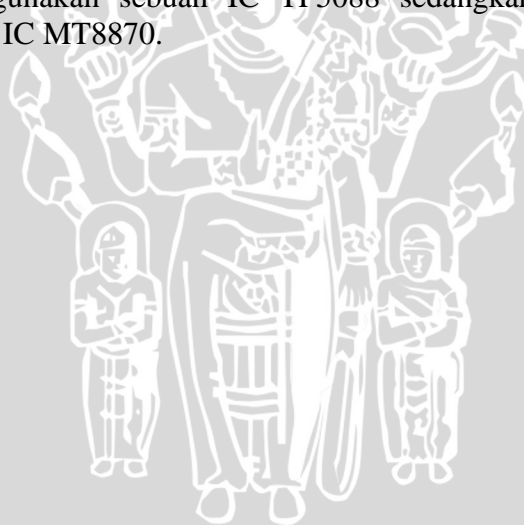


ABSTRAK

IMAM SUMARSONO. Jurusan Teknik Elektro. Fakultas Teknik. Universitas Brawijaya. Februari 2007, *Sistem Pengolah dan Penransmisi Sinyal GPS pada Sistem Informasi Posisi Kereta Api Berbasis GPS (Global Positioning System)*. Dosen Pembimbing : Ir. Ponco Siwindarto, MS dan R. Arif Setiawan, ST, MT.

Peningkatan kualitas pelayanan perlu dilakukan oleh perusahaan kereta api, terutama untuk meningkatkan kualitas ketepatan waktu jadwal kereta. Selama ini para calon penumpang sering dikecewakan dengan keterlambatan jadwal kedatangan kereta api sehingga harus menghabiskan banyak waktu yang untuk menunggu. Dengan memanfaatkan sistem informasi posisi kereta api berbasis GPS (*Global Positioning System*) ini para calon penumpang dapat mengetahui jarak kereta api terhadap stasiun yang diinginkan hanya dengan mengirimkan SMS ke pusat informasi posisi kereta api. Keberadaan kereta diperoleh dari keluaran GPS (*Global Positioning System*) dan jarak kereta dihitung terhadap stasiun yang dikehendaki. Perintah dari pusat informasi ke kereta ataupun informasi data dari kereta ke pusat informasi berupa nada DTMF yang dikirimkan melalui HT (*Handy Talky*). Data lintang dan bujur diperoleh dari GPS *Garmin Etrex*. Pengontrol utama menggunakan sebuah mikrokontroler AT89S52. Untuk mengirimkan DTMF menggunakan sebuah IC TP5088 sedangkan untuk menerima DTMF menggunakan sebuah IC MT8870.



BAB V

PENGUJIAN DAN ANALISIS

Bab ini membahas pengujian dan analisis alat yang telah dirancang dari peralatan yang telah dibuat. Pengujian dilakukan dengan pengukuran tiap-tiap blok dengan tujuan mengamati apakah blok-blok tersebut bekerja sesuai dengan yang diharapkan.

Pengujian dilakukan berdasarkan pada masing-masing komponen pendukung dari alat secara keseluruhan. Pengujian yang dilakukan adalah pengujian terhadap:

1. Pengujian komunikasi serial GPS *receiver*
2. Pengujian rangkaian MAX-232
3. Pengujian Antarmuka GPS dengan Mikrokontroler
4. Pengujian rangkaian pembangkit DTMF
5. Pengujian rangkaian penerima DTMF
6. Pengujian penghitungan jarak
7. Pengujian rangkaian *switching* PTT
8. Pengujian komunikasi antar HT (*Handy Talky*)
9. Pengujian keseluruhan sistem

5.1 Pengujian Komunikasi Serial GPS Receiver

5.1.1 Tujuan Pengujian

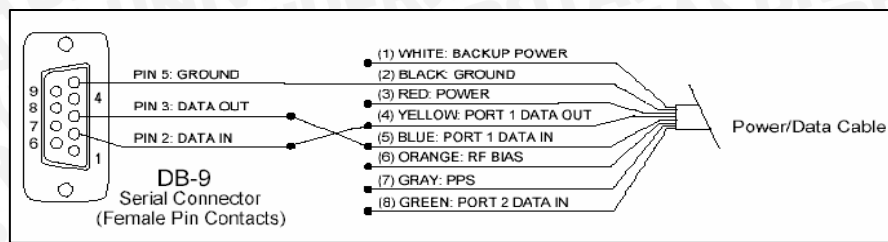
Pengujian ini bertujuan untuk menguji apakah keluaran dari GPS Receiver sesuai dengan yang dibutuhkan pada perancangan sistem, yaitu mengandung kalimat dengan format RMC.

5.1.2 Alat yang Dipergunakan

1. GPS Receiver Garmin
2. Kabel data Garmin Etrex
3. Seperangkat PC.
4. Software aplikasi Hyper Terminal dalam sistem operasi Windows.

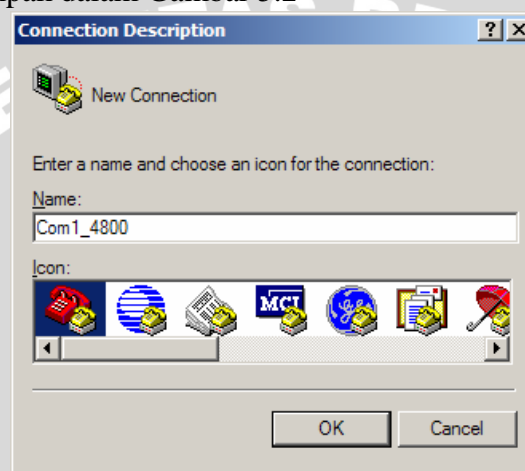
5.1.3 Prosedur Pengujian

1. GPS Receiver dan PC dihubungkan sesuai dengan susunan dalam Gambar 5.1.



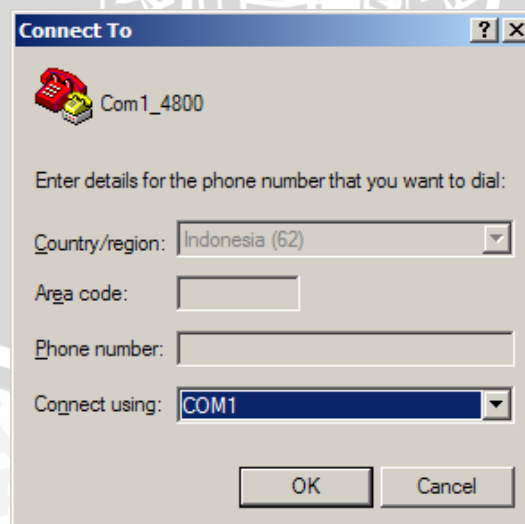
Gambar 5.1. Hubungan Antara GPS Receiver dan PC

2. Menjalankan program Hyper Terminal. Beri nama koneksi yang akan dibentuk, seperti yang tampak dalam Gambar 5.2



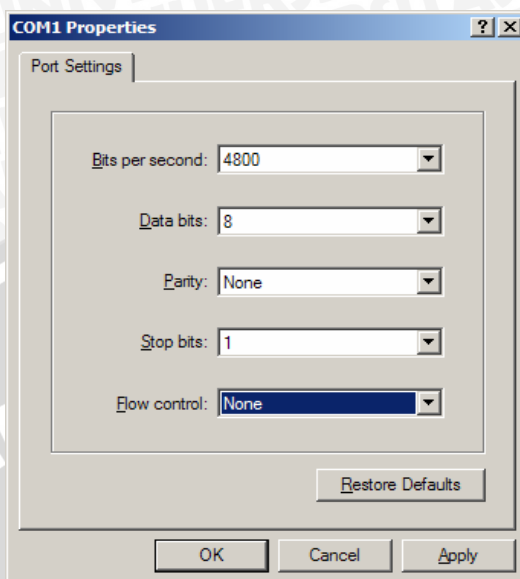
Gambar 5.2. Pemberian nama koneksi

3. Mengatur konfigurasi port serial yang akan digunakan dengan mengisi Connect Using dengan Com1, seperti dalam Gambar 5.3



Gambar 5.3. Pemilihan konfigurasi port serial

4. Menentukan baudrate, lebar data, jenis paritas, jumlah bit stop dan flow control sesuai dengan spesifikasi komunikasi serial GPS seperti pada Gambar 5.4

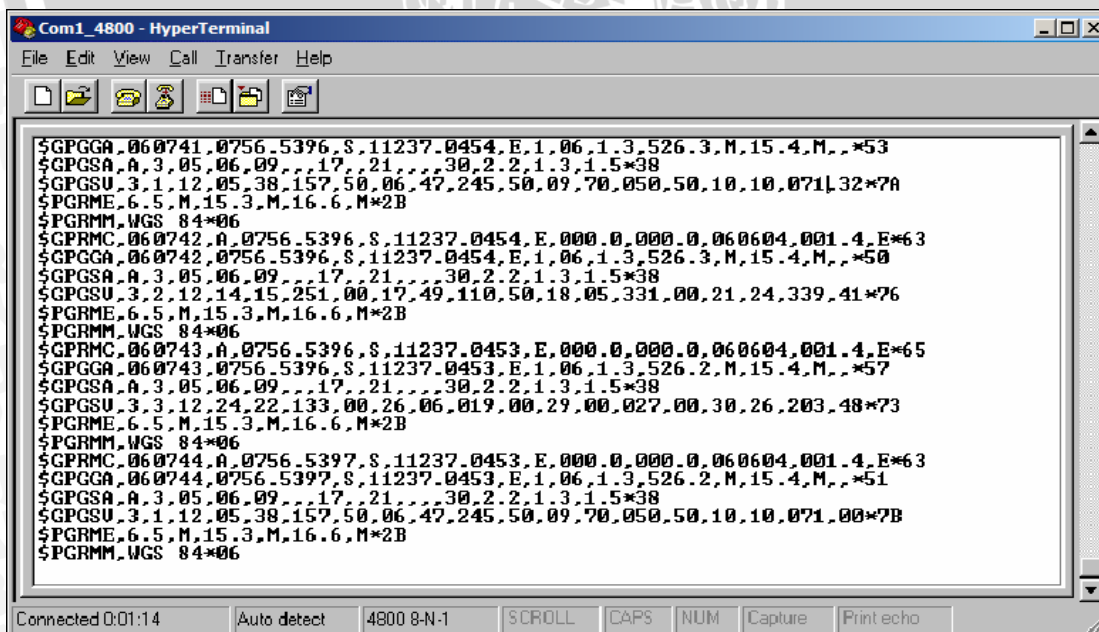


Gambar 5.4. Pengaturan spesifikasi komunikasi serial

5. Menempatkan GPS Receiver di luar ruangan.
6. Mengamati tampilan aplikasi Hyper Terminal.

5.1.4 Hasil Pengujian dan Analisis

GPS Receiver dinyalakan diluar ruangan (outdoor), setelah beberapa saat pada tampilan akan muncul kalimat-kalimat NMEA 0183 seperti pada gambar 5.5.



Gambar 5.5. Tampilan Keluaran Aplikasi Hyper Terminal

Pada tampilan program tampak bahwa GPS Receiver menghasilkan kalimat-kalimat NMEA 0183 yang terdiri dari jenis kalimat GPRMC, GPGGA, GPGSA, GPGSV, PGRME, PGRMM.

GPS Receiver dapat berkomunikasi secara serial dengan baik dan menghasilkan kalimat NMEA 0183 yang diinginkan.

5.2 Pengujian Rangkaian Max-232

5.2.1 Tujuan Pengujian

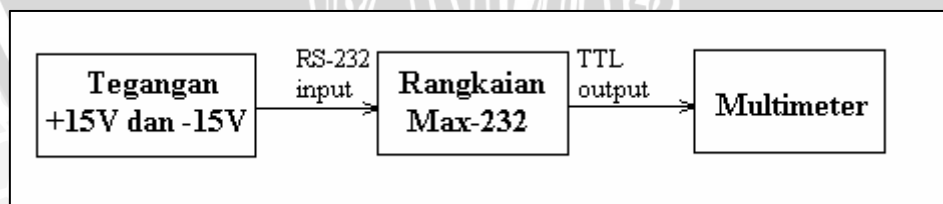
Pengujian ini bertujuan untuk memeriksa apakah rangkaian Max-232 dapat bekerja dengan baik untuk mengkonversi sinyal dari GPS yang bertipe RS-232 menjadi sinyal dengan tipe TTL yang dapat diterima oleh mikrokontroler.

5.2.2 Alat yang dipergunakan

1. IC Max-232
2. Rangkaian pendukung
3. Catu daya +5 volt
4. Multimeter

5.2.3 Prosedur Pengujian

1. Merangkai rangkaian pengujian seperti terlihat dalam Gambar 5.6
2. Memberikan tegangan masukan pada masukan IC Max-232 sebesar +15V dan -15V
3. Memeriksa tegangan keluaran dari IC Max-232 dengan menggunakan multimeter



Gambar 5.6 Blok diagram pengujian Rangkaian Max-232

5.2.4 Hasil Pengujian dan Analisis

Hasil Pengujian dari rangkaian Max-232 dapat dilihat dalam Tabel 5.1.

Tabel 5.1 Hasil Pengujian Rangkaian Max-232

No.	Masukan	Keluaran
1	+15 V	0,01 V
2	-15 V	4,98 V

Dari pengujian didapatkan bahwa ketika tegangan masukan sebesar +15 V, maka keluaran dari rangkaian Max-232 adalah sebesar 0,01 V, dan ketika tegangan masukan sebesar -15 V, maka tegangan keluaran rangkaian Max-232 adalah sebesar 4,98 V, jadi rangkaian Max-232 tersebut dapat bekerja dengan baik.

5.3 Pengujian Antarmuka GPS dengan Mikrokontroler

5.3.1 Tujuan Pengujian

Pengujian antarmuka GPS dengan mikrokontroler dilakukan untuk memeriksa apakah mikrokontroler dapat berkomunikasi serial dengan GPS *receiver*.

5.3.2. Alat yang dipergunakan

1. *Writer* mikrokontroler AT89S52
2. Rangkaian unit mikrokontroler
3. Kabel data GPS
4. Sumber tegangan 5 Volt

5.3.3. Prosedur Pengujian

1. Mengisi memori mikrokontroler dengan program seperti dalam Gambar 5.7
2. Menghubungkan mikrokontroler dengan GPS *receiver* melalui IC MAX-232 seperti dalam Gambar 5.8.
3. Menyalakan GPS *receiver*
4. Menunggu sampai GPS *receiver* siap mengirimkan data
5. Melihat perubahan nyala LED yang terhubung dengan PORT0

5.3.4 Hasil Pengujian dan Analisis

Tabel 5.2 Tabel pengujian antarmuka GPS dengan Mikrokontroler

Tempat pengujian	Keluaran LED	Data Keluaran
Di dalam ruangan	Mati	Tidak Valid
Di luar ruangan	Menyala	Valid

Dari 3 kali pengujian di tempat yang berbeda didapatkan hasil yang sama seperti yang terlihat dalam Tabel 5.2. Sehingga dapat dianalisis bahwa mikrokontroler dapat berkomunikasi serial dengan GPS *receiver* dan mengeluarkan data valid ketika GPS *receiver* berada di luar ruangan, hal ini ditunjukkan dengan nyala LED yang dihubungkan ke Port 0 mikrokontroler, jika data yang diterima valid maka LED menyala.

5.4 Pengujian Rangkaian Pembangkit DTMF

5.4.1 Tujuan Pengujian

Tujuan untuk mengetahui perbedaan antara frekuensi sinyal keluaran pembangkit DTMF melalui pengukuran dengan frekuensi standar.

5.4.2. Alat yang dipergunakan

1. Rangkaian pembangkit DTMF
2. Switch untuk masukan data tone
3. Osiloskop

5.4.3 Prosedur Pengujian

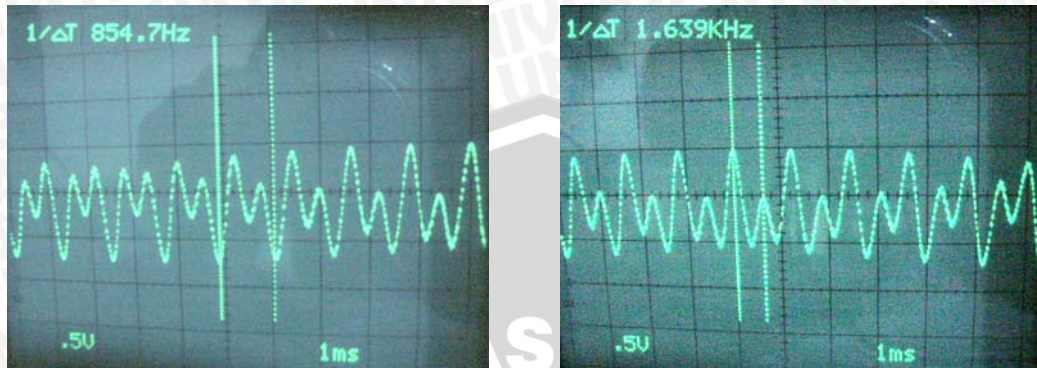
1. Merangkai rangkaian pengujian seperti terlihat dalam Gambar 5.9.
2. Memberikan masukan pada switch masukan
3. Memberikan masukan tepi naik pada pin *Tone Enable* pembangkit DTMF
4. Melihat sinyal DTMF yang dihasilkan melalui Osiloskop
5. Menghitung persen kesalahan dari pembangkit DTMF



Gambar 5.9 Blok Diagram Pengujian Pembangkit DTMF

5.4.4 Hasil Pengujian dan Analisis

Sinyal yang dihasilkan pembangkit DTMF dengan masukan 1111 seperti terlihat dalam Gambar 5.10.



Gambar 5.10 Sinyal keluaran pembangkit DTMF

Dari gambar terlihat Frekuensi rendah sebesar 854,7Hz dan Frekuensi tinggi sebesar 1639 kHz.

Kesalahan dari sinyal keluaran pembangkit DTMF tersebut dapat dihitung dengan rumus:

$$(5.1) \quad \% \text{ Kesalahan} = \frac{f_{\text{pengukuran}} (\text{Hz}) - f_{\text{standar}} (\text{Hz})}{f_{\text{standar}} (\text{Hz})} \times 100 \%$$

Kesalahan untuk Frekuensi tinggi sebesar :

$$\begin{aligned} \% \text{ Kesalahan} &= \frac{1639 - 1633}{1633} \times 100 \% \\ &= \frac{6}{1633} \times 100\% \\ &= 0,37\% \end{aligned}$$

Kesalahan untuk Frekuensi rendah sebesar :

$$\begin{aligned} \% \text{ Kesalahan} &= \frac{854,7 - 852}{852} \times 100 \% \\ &= \frac{2,7}{852} \times 100\% \\ &= 0,32\% \end{aligned}$$

Hasil pengujian dari rangkaian pembangkit DTMF dapat dilihat dalam Tabel 5.3.

Tabel 5.3 Hasil Pengujian Rangkaian Pembangkit DTMF

Data Masukan	Frekuensi Rendah			Frekuensi Tinggi		
	Frekuensi Standar (Hz)	Hasil Pengukuran (Hz)	Error (%)	Frekuensi Standar (Hz)	Hasil Pengukuran (Hz)	Error (%)
0001	697	701,2	0,60	1209	1216	0,58
0010	697	700,8	0,55	1336	1341	0,37
0011	697	702,2	0,75	1477	1481	0,27
0100	770	775,2	0,68	1209	1216	0,58
0101	770	774,8	0,62	1336	1342	0,45
0110	770	774,6	0,60	1477	1483	0,41
0111	852	857,1	0,60	1209	1215	0,50
1000	852	856,7	0,55	1336	1341	0,37
1001	852	856,5	0,53	1477	1482	0,34
1010	941	945,3	0,46	1209	1216	0,58
1011	941	944,9	0,41	1336	1340	0,30
1100	941	946,3	0,56	1477	1484	0,47
1101	697	701,1	0,59	1633	1638	0,31
1110	770	775,2	0,68	1633	1640	0,43
1111	852	854,7	0,32	1633	1639	0,37
0000	941	934,3	0,29	1633	1639	0,39

Dari hasil pengujian didapat adanya perbedaan antara frekuensi standar dengan frekuensi hasil pengukuran, akan tetapi karena kesalahan tersebut nilainya sangat kecil (kurang dari 1 %), sedangkan rangkaian penerima DTMF dapat menerima sinyal dengan kesalahan sampai $\pm 1,5\%$ (Datasheet MT8870), sehingga frekwensi keluaran DTMF tersebut masih dapat diterima oleh rangkaian penerima DTMF.

5.5 Pengujian Rangkaian Penerima DTMF

5.5.3 Tujuan Penelitian

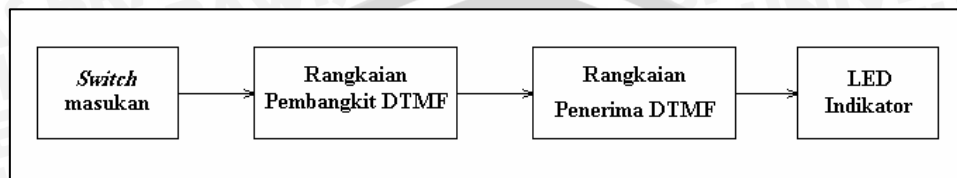
Tujuan pengujian adalah untuk mengetahui apakah rangkaian penerima DTMF dapat menerima nada-nada DTMF dari rangkaian pengirim DTMF dan menterjemahkannya dalam data biner 4 bit.

5.5.2. Alat yang dipergunakan

- Rangkaian penerima DTMF
- Rangkaian pembangkit DTMF yang telah diuji
- Peraga LED (5 buah)
- Catu daya

5.5.3 Prosedur Pengujian

1. Merangkai rangkaian seperti ditunjukkan dalam Gambar 5.11.
2. Memberikan masukan melalui *switch* masukan, berupa data biner 4 bit.
3. Melakukan pemicuan tepi naik pada pin *tone enable* pada pembangkit DTMF
4. Melihat keluaran dari LED indikator.



Gambar 5.11 Blok diagram pengujian penerima DTMF

5.5.4 Hasil Pengujian dan Analisis

Hasil keluaran dicatat dalam Tabel 5.4.

Tabel 5.4. Hasil pengujian rangkaian penerima DTMF

Data Masukan	Pin <i>tone enable</i>	Data keluaran (LED)
0001	Tepi naik	0001
0010	Tepi naik	0010
0011	Tepi naik	0011
0100	Tepi naik	0100
0101	Tepi naik	0101
0110	Tepi naik	0110
0111	Tepi naik	0111
1000	Tepi naik	1000
1001	Tepi naik	1001
1010	Tepi naik	1010
1011	Tepi naik	1011
1100	Tepi naik	1100
1101	Tepi naik	1101
1110	Tepi naik	1110
1111	Tepi naik	1111
0000	Tepi naik	0000

Dari hasil pengujian yang terlihat dalam Tabel 5.4 didapatkan bahwa rangkaian pengirim dan penerima DTMF menunjukkan data biner yang sama, sehingga rangkaian penerima DTMF tersebut dapat berfungsi dengan baik.

5.6 Pengujian Penghitungan Jarak

5.6.1. Tujuan Pengujian

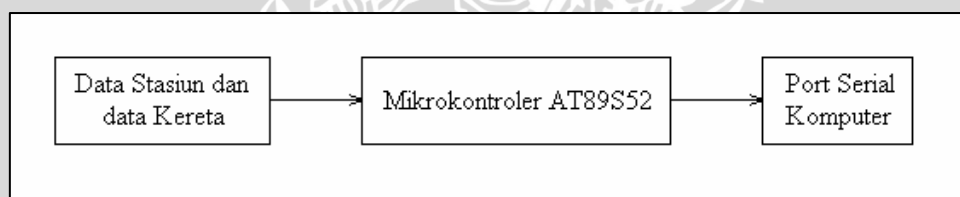
Tujuan Pengujian penghitungan jarak adalah untuk membandingkan antara hasil penghitungan jarak dengan cara perhitungan manual dengan perhitungan jarak oleh mikrokontroler.

5.6.2. Alat yang dipergunakan

1. Mikrokontroler AT89S52
2. Sumber tegangan 5 V
3. IC Max-232
4. Seperangkat PC

5.6.3 Prosedur Pengujian

1. Merangkai rangkaian seperti ditunjukkan dalam Gambar 5.12
2. Mengaktifkan rangkaian
3. Melihat keluaran hasil perhitungan melalui komputer



Gambar 5.12 Blok diagram pengujian penghitungan jarak

5.6.4 Hasil Pengujian dan Analisis

Hasil pengujian ditunjukkan dalam Tabel 5.5

Tabel 5.5. Hasil pengujian penghitungan jarak

KA	Lintang KA	Bujur KA	Stasiun	Lintang Stasiun	Bujur Stasiun	Jarak Perhitungan	Jarak Keluaran
2	07°08,5464'	111°40,6656'	1	07°14,4895'	112°44,6587'	118,4718985	118,5
2	07°08,5464'	111°40,6656'	2	07°09,9485'	111°52,4321'	21,84413405	21,8
2	07°08,5464'	111°40,6656'	3	07°09,4589'	111°35,3254'	9,987469528	10,0
2	07°08,5464'	111°40,6656'	4	06°58,4514'	110°20,1285'	149,6268079	149,6
2	07°08,5464'	111°40,6656'	5	06°53,2578'	109°40,6489'	223,0314758	223,0
2	07°08,5464'	111°40,6656'	6	06°51,4628'	109°31,2849'	240,5767501	240,6
2	07°08,5464'	111°40,6656'	7	06°42,9874'	108°33,2548'	348,6815158	348,7
2	07°08,5464'	111°40,6656'	8	06°24,6512'	107°27,1589'	474,3007986	474,3
2	07°08,5464'	111°40,6656'	9	06°10,4697'	106°49,5648'	547,2465824	547,2
7	06°38,6346'	107°30,4263'	1	07°14,4895'	112°44,6587'	583,0206927	583,0
7	06°38,6346'	107°30,4263'	2	07°09,9485'	111°52,4321'	486,4259624	486,4
7	06°38,6346'	107°30,4263'	3	07°09,4589'	111°35,3254'	455,0181307	455,0
7	06°38,6346'	107°30,4263'	4	06°58,4514'	110°20,1285'	314,9586671	315,0
7	06°38,6346'	107°30,4263'	5	06°53,2578'	109°40,6489'	241,5632813	241,6
7	06°38,6346'	107°30,4263'	6	06°51,4628'	109°31,2849'	224,0431679	224,0
7	06°38,6346'	107°30,4263'	7	06°42,9874'	108°33,2548'	116,0934871	116,1
7	06°38,6346'	107°30,4263'	8	06°24,6512'	107°27,1589'	26,56116673	26,6
7	06°38,6346'	107°30,4263'	9	06°10,4697'	106°49,5648'	91,58736716	91,6
9	06°14,6454'	106°31,9233'	1	07°14,4895'	112°44,6587'	805,3219218	805,3
9	06°14,6454'	106°31,9233'	2	07°09,9485'	111°52,4321'	599,5977888	599,6
9	06°14,6454'	106°31,9233'	3	07°09,4589'	111°35,3254'	568,39005	568,4
9	06°14,6454'	106°31,9233'	4	06°58,4514'	110°20,1285'	428,3935397	428,4
9	06°14,6454'	106°31,9233'	5	06°53,2578'	109°40,6489'	355,1423294	355,1
9	06°14,6454'	106°31,9233'	6	06°51,4628'	109°31,2849'	337,5662622	337,6
9	06°14,6454'	106°31,9233'	7	06°42,9874'	108°33,2548'	229,7182577	229,7
9	06°14,6454'	106°31,9233'	8	06°24,6512'	107°27,1589'	103,4865953	103,5
9	06°14,6454'	106°31,9233'	9	06°10,4697'	106°49,5648'	33,4241292	33,4

Dari Tabel 5.5 terlihat bahwa antara hasil perhitungan manual dengan hasil perhitungan oleh mikrokontroler AT89S52 memiliki nilai yang sesuai. Karena sistem akan diterapkan pada kereta api yang berkecepatan sekitar 60 km / jam, sehingga perubahan jaraknya adalah sekitar 1 km tiap menit, dengan asumsi bahwa proses pengiriman dan penerimaan sms sekitar 15 detik maka tingkat resolusi sebesar 0,1 km sudah mencukupi..

5.7 Pengujian Rangkaian *Switching* PTT

5.7.1. Tujuan Pengujian

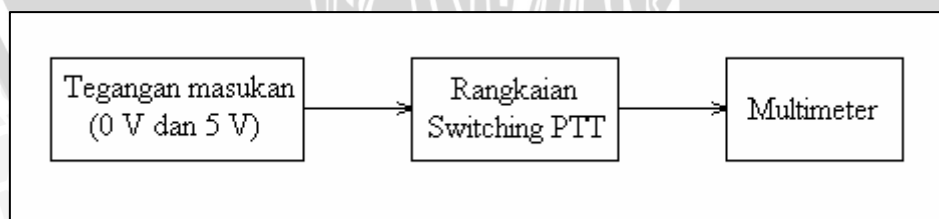
Tujuan Pengujian rangkaian *switching* PTT adalah untuk melihat apakah *switch* kontak relay dapat berfungsi dengan baik.

5.7.2. Alat yang dipergunakan

5. Rangkaian *Switching* PTT
6. Sumber tegangan 5 V
7. Multimeter
8. Catu daya

5.7.3 Prosedur Pengujian

4. Merangkai rangkaian seperti ditunjukkan dalam Gambar 5.13
5. Memberikan masukan berupa tegangan 5 V dan 0 V
6. Mengukur resistansi pada keluaran Relay apakah terbuka (resistansi = \sim), atau tertutup (resistansi = 0).



Gambar 5.13 Blok diagram pengujian rangkaian *switching* PTT

5.7.4 Hasil Pengujian dan Analisis

Hasil pengujian ditunjukkan dalam Tabel 5.6

Tabel 5.6. Hasil pengujian rangkaian *switching PTT*

Tegangan Masukan	Terminal <i>Normaly Open</i>
0 V	Terbuka
5 V	Tertutup

Dari Tabel 5.6 ditunjukkan bahwa pada saat rangkaian *switching PTT* diberi masukan tegangan 0 V maka keluaran *normaly close* pada relay akan terbuka, sedangkan apabila rangkaian *switching PTT* diberi tegangan 5 V maka keluaran *normaly close* pada relay akan tertutup.

5.8 Pengujian Komunikasi Antar HT (*Handy Talky*)

5.8.1 Tujuan Pengujian

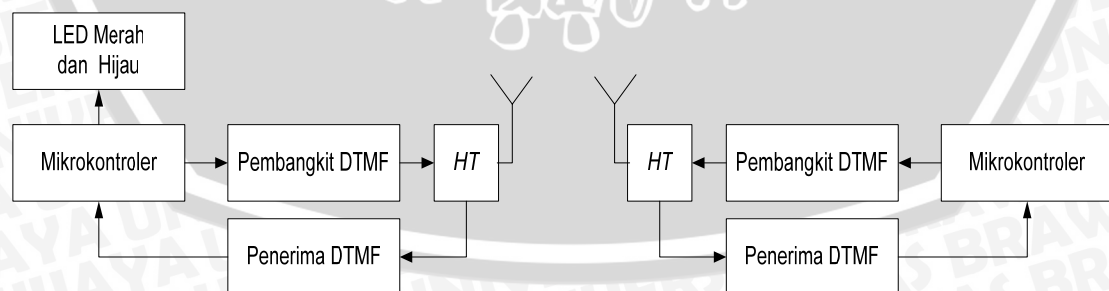
Tujuan pengujian ini adalah untuk mengetahui sistem dapat berkomunikasi dua arah dengan menggunakan perantara HT.

5.8.2. Alat yang dipergunakan

1. Rangkaian Sistem Pengolah dan Pentransmisi Sinyal GPS pada Sistem Informasi Kereta Api Berbasis GPS
2. Rangkaian Penerima dan Pengolah Data Posisi
3. Catu daya 12 V
4. HT (*Handy Talky*)

5.8.3 Prosedur Pengujian

1. Merangkai rangkaian pengujian seperti yang terlihat dalam Gambar 5.14

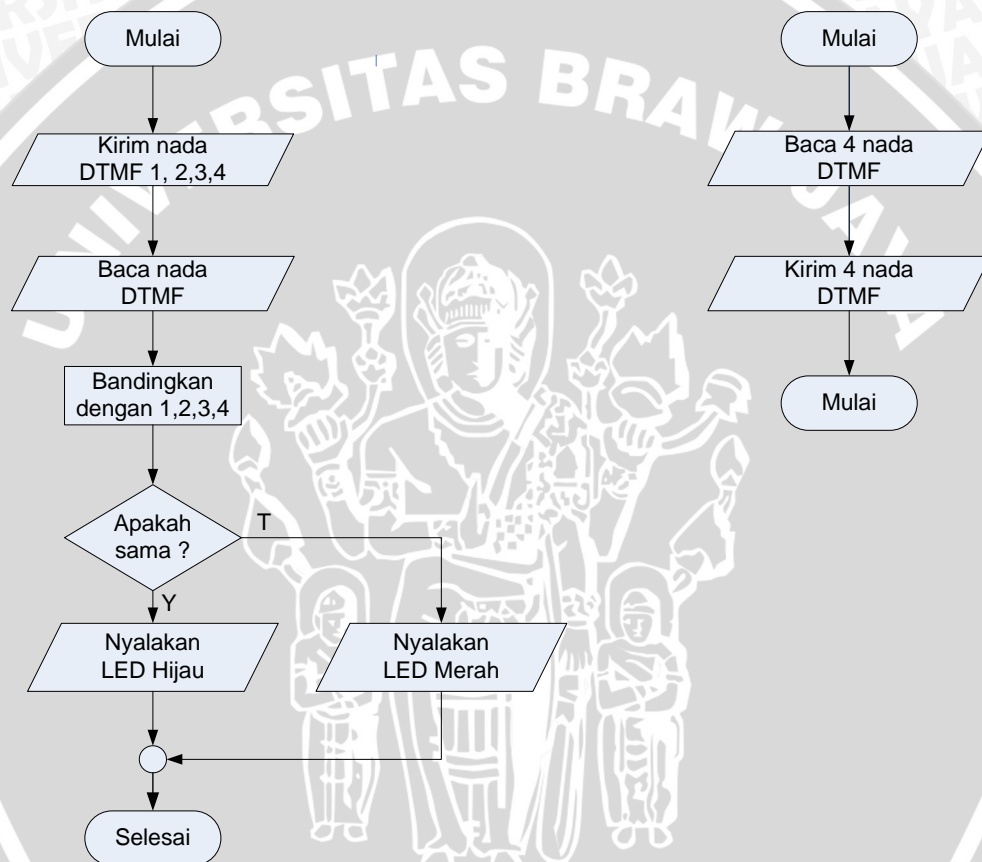


(a) Blok sistem pegolah data GPS

(b) Blok sistem penerima data posisi

Gambar 5.14. Blok diagram pengujian komunikasi antar HT

2. Membuat program sederhana untuk menguji komunikasi antar HT dengan mengirimkan empat nada DTMF yang berbeda yaitu 1,2,3,4 secara berurutan melalui pembangkit DTMF, kemudian menunggu empat nada DTMF yang diterima mikrokontroler melalui penerima DTMF. Nada DTMF yang diterima tersebut kemudian dibandingkan dengan nada yang dikirimkan sebelumnya. Jika nada DTMF yang diterima sama, maka mikrokontroler akan menyalakan LED Hijau, sebaliknya nyalakan LED Merah, seperti dalam flowchart yang ditunjukkan dalam Gambar 5.15.



(a) Flowchart untuk sistem Pengolah data GPS posisi

(b) Flowchart untuk sistem penerima data

Gambar 5.15. Flowchart program pengujian komunikasi antar HT

3. Menyalakan catu daya
4. Melihat keluaran LED

5.8.4 Hasil Pengujian dan Analisis

Tabel 5.7 Tabel pengujian komunikasi antar HT

Kondisi	LED Merah	LED Hijau	Komunikasi
HT penerima dinonaktifkan	Menyala	Mati	Gagal
HT diaktifkan	Mati	Menyala	Berhasil

Dari 3 kali pengujian didapatkan hasil yang sama seperti yang terlihat dalam Tabel 5.7. Jika LED merah menyala menandakan komunikasi terganggu, dan jika LED hijau yang menyala menandakan komunikasi berjalan lancar. Dari hasil pengujian yang terlihat dalam Tabel 5.7 dapat dianalisis bahwa komunikasi antar HT dapat berjalan seperti yang diharapkan.

5.9 Pengujian Keseluruhan Sistem

5.9.1 Tujuan Pengujian

Tujuan pengujian ini adalah untuk mengetahui kerja secara keseluruhan sistem apakah sesuai dengan perencanaan awal yang diharapkan.

5.9.2. Alat yang dipergunakan

1. Rangkaian Sistem Pengolah dan Pentransmisi Sinyal GPS pada Sistem Informasi Kereta Api Berbasis GPS
2. Rangkaian Penerima dan Pengolah Data
3. Catu daya 12 V
4. HT (*Handy Talky*)
5. HP (*Hand Phone*)
6. GPS receiver

5.9.3 Prosedur Pengujian

1. Merangkai keseluruhan sistem Pengolah dan Pentransmisi Sinya GPS dan dipadukan dengan rangkaian sistem Penerima dan pengolah data.
2. Mengirimkan SMS ke pusat informasi dengan format KA (ID kereta),(ID stasiun) misal : KA 2,6 seperti terlihat dalam Gambar 5.16a.
3. Menunggu balasan dari pusat informasi berupa SMS seperti tampak dalam Gambar 5.16b

5.9.4 Hasil Pengujian

Setelah dilakukan pengiriman SMS ke *Handphone* pusat informasi dengan format KA (spasi) ID kereta , ID stasiun seperti tampak dalam Gambar 5.16a kemudian sistem dapat memberikan balasan berupa jarak antara kereta yang dimaksud dengan stasiun yang diinginkan seperti terlihat dalam Gambar 5.16b sehingga keseluruhan sistem dapat bekerja seperti yang diharapkan.



(a) Format sms permintaan

(b) Format sms balasan

Gambar 5.16. Tampilan pesan permintaan jarak kereta dan balasan

BAB VI

KESIMPULAN DAN SARAN

6.1 Kesimpulan

Dari hasil Perancangan dan pembuatan Sistem Pengolah dan Pentransmisi Sinyal Gps Pada Sistem Informasi Posisi Kereta Api Berbasis Gps(*Global Positioning Sistem*) dapat diambil kesimpulan sebagai berikut:

1. Sistem dapat membaca data posisi (Lintang dan Bujur) dari GPS *Receiver* dengan format data NMEA 0183 ver 3.0.
2. Sistem dapat mengolah data posisi menjadi jarak (dalam kilometer) terhadap stasiun yang diinginkan dengan tingkat resolusi sebesar 1 km.
3. Sistem dapat mentransmisikan data hasil pengolahan data dari GPS melalui *Handy Talky*.
4. Sistem dapat membangkitkan dan menerima sinyal DTMF untuk mengirimkan data hasil pengolahan data GPS (Jarak terhadap stasiun yang diinginkan), dengan tingkat kesalahan dari pembangkit DTMF kurang dari 1 %

6.2 Saran

Untuk pengembangan alat ini ada hal-hal yang dapat dilakukan, antara lain dapat dikembangkan suatu system untuk mengatur lalu lintas perjalanan kereta api dan sistem peringatan dini terhadap kecelakaan kereta api.



BAB I PENDAHULUAN

1.1 Latar Belakang Masalah

Pada era globalisasi saat ini, perusahaan jasa transportasi semakin bersaing untuk meningkatkan kualitas armadanya. Khususnya jasa transportasi kereta api. Untuk meningkatkan kualitas pelayanan terhadap masyarakat, perusahaan kereta api perlu menambah jumlah armadanya dan meningkatkan kualitas ketepatan waktu.

Selama ini para calon penumpang sering dikecewakan dengan keterlambatan jadwal kedatangan kereta api. Para penumpang harus menghabiskan waktu yang lama untuk menunggu kedatangan kereta api, sehingga mereka merasa bosan dan banyak waktu berharga terbuang sia-sia. Untuk meningkatkan pelayanan terhadap masyarakat yang ingin menggunakan jasa kereta api maka perlu adanya peningkatan kualitas pelayanan. Dengan bantuan sistem informasi posisi kereta api berbasis GPS (*Global Positioning System*) dapat meningkatkan kualitas pelayanan terhadap para calon penumpang.

Sistem informasi posisi kereta api dengan GPS (*Global Positioning System*) ini, menggunakan fasilitas SMS (*Short Message Service*) untuk sarana komunikasinya. Sehingga para calon penumpang dapat mengetahui dimanakah posisi kereta api saat ini dan berapa jaraknya dari stasiun yang dituju hanya dengan mengirimkan SMS ke pusat informasi posisi kereta api. Dengan demikian para calon penumpang tidak perlu menghabiskan banyak waktu di stasiun tetapi para calon penumpang dapat melakukan aktifitas lain selama menunggu kedatangan kereta api.

Dalam penelitian sebelumnya yang berjudul “Papan Informasi Gerbong Kereta Api dengan Peraga Dot Matrik” oleh M. Choirur Rizqi Anwar, pengiriman informasinya posisi kereta api dilakukan secara manual oleh masinis pada tiap stasiun dengan mendisplaykan nama stasiun, waktu dan pemberitahuan keberangkatan. Hal ini tentunya kurang praktis dan nyaman.

Dalam penelitian berikutnya yang berjudul “Penampil Informasi Posisi Kereta” oleh Romadlon Tri Atmaja, menggunakan sensor inframerah untuk mengetahui posisi kereta yang bertujuan memberikan pemberitahuan kepada penumpang didalam kereta

tentang posisi kereta api. Sehingga hanya data posisi pada tempat tertentu saja yang diketahui.

Dalam penelitian selanjutnya yang berjudul “Pemantauan Posisi Kereta Api dengan Memanfaatkan Teknologi GPS” oleh Devara Dahata, pengiriman datanya menggunakan fasilitas SMS (*Short Message Service*) pada handphone dan sistem tersebut hanya digunakan untuk memantau posisi kereta api dari stasiun. Kelemahan dari sistem ini adalah mahal biaya untuk pengiriman data posisi dari kereta ke penerima yang ada di stasiun.

Berdasarkan masalah diatas maka perlu dibuat sistem informasi posisi kereta api untuk memperbaiki sistem sebelumnya. Penelitian ini menggunakan GPS (*Global Positioning System*) untuk mendapatkan data posisi kereta api dan pengiriman datanya melalui HT (*Handy Talky*). Selain itu sistem ini akan digunakan untuk memberikan informasi posisi kereta api dan jaraknya dari stasiun kepada para calon penumpang.

Dalam skripsi ini akan direncanakan dan dibuat sistem informasi posisi kereta berbasis GPS (*Global Positioning System*) bagian pengolah dan pentransmisi Sinyal GPS. Sistem ini dapat membaca data posisi kereta dari GPS dan mentransmisikannya melalui saluran Handy Talky (HT) dengan menggunakan frekuensi milik PT KAI. Dengan demikian data posisi kereta api dapat dikirimkan ke pusat informasi posisi kereta api.

1.2 Rumusan Masalah

Dalam skripsi ini dibahas rumusan masalah sebagai berikut :

1. Bagaimana membaca data posisi (Lintang dan Bujur) kereta api dari GPS.
2. Bagaimana mengolah data posisi untuk menghitung jarak dalam kilometer.
3. Bagaimana mentransmisikan data hasil pembacaan dari GPS melalui Handy Talky.
4. Bagaimana merancang rangkaian pembangkit dan penerima sinyal DTMF untuk mengirmkan data GPS.

1.3 Ruang Lingkup

Dengan mengacu pada permasalahan yang telah dirumuskan, maka hal-hal yang berkaitan dengan alat yang akan dibuat, diberi batasan sebagai berikut:

1. Sistem dioperasikan untuk kereta api yang bergerak dari stasiun di Surabaya ke stasiun di Jakarta sebagai simulasi
2. Penggunaan alat digunakan untuk di luar ruangan (outdoor) serta dibatasi hanya untuk suatu wilayah terbatas yang telah ditentukan sebagai simulasi
3. Tingkat akurasi informasi posisi yang diberikan alat tergantung pada Akurasi GPS yang digunakan
4. Menitikberatkan pada proses pembacaan dan pentransmisi data serta antarmuka GPS dan HT dengan mikrokontroler

1.4 Tujuan

Tujuan penyusunan skripsi ini adalah Merancang Sistem yang dapat menunjukkan jarak masing-masing kereta terhadap stasiun yang diinginkan sehingga calon penumpang dengan cepat dapat mengetahui melalui SMS (*Short Message Service*).

1.5 Sistematika Penulisan

Sistematika penulisan dalam skripsi ini sebagai berikut:

BAB I Pendahuluan

Memuat latar belakang, rumusan masalah, tujuan, batasan masalah, metodologi pembahasan, dan sistematika pembahasan.

BAB II Teori Penunjang

Membahas teori-teori yang mendukung dalam perencanaan dan pembuatan alat.

BAB III Metodologi

Berisi tentang metode penelitian dan perencanaan alat serta pengujian.

BAB IV Perencanaan dan Pembuatan Alat

Perancangan dan perealisasiian alat yang meliputi spesifikasi, perencanaan blok diagram, prinsip kerja dan realisasi alat.

BAB V Pengujian Alat

Memuat hasil pengujian terhadap alat yang telah direalisasikan.

BAB VI Kesimpulan dan Saran

Memuat kesimpulan dan saran-saran.

BAB II DASAR TEORI

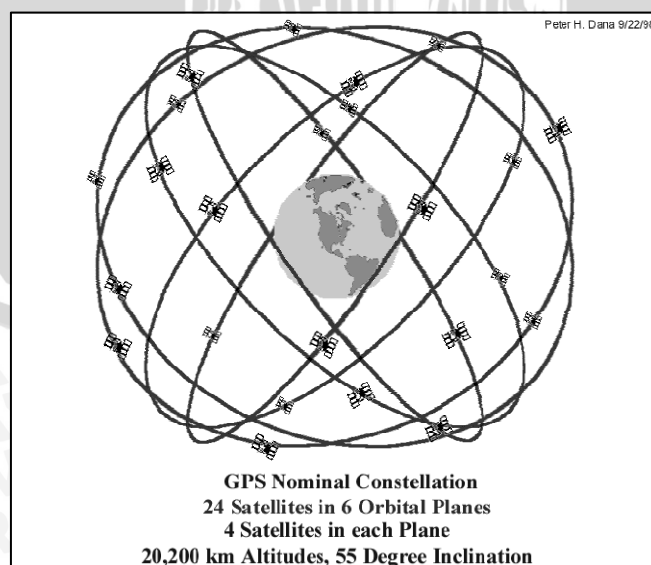
Dalam merencanakan dan merealisasikan Sistem Pengolah dan Pentransmisi sinyal GPS pada Sistem Informasi Posisi Kereta Api Berbasis GPS (*Global Positioning System*) maka dibutuhkan pemahaman tentang berbagai hal yang mendukung. Pemahaman ini akan bermanfaat untuk merancang perangkat keras dan perangkat lunak sistem. Pengetahuan yang mendukung perencanaan dan realisasi alat meliputi *Global Positioning System* (GPS), Max232, Mikrokontroler AT89S52, Komunikasi Sinkron dan Asinkron, serta *Dual Tone Multiple Frequency* (DTMF).

2.1 *Global Positioning System*

GPS adalah sistem navigasi yang berbasiskan satelit yang terdiri dari 24 jaringan orbit satelit yang berada 11.000 mil laut di angkasa dan pada enam jalur orbit yang berbeda. Satelit-satelit ini bergerak secara konstan dan menyelesaikan 2 kali orbit mengelilingi bumi di bawah 24 jam. Satelit GPS adalah satelit Navstar dan setiap satelit memancarkan 3 frekuensi yaitu L1 dengan frekuensi 1575,42 MHz, L2 dengan frekuensi 1227,60 MHz dan L5 dengan frekuensi 1176,45 MHz. Untuk kalangan sipil frekuensi yang digunakan adalah frekuensi L1 yaitu 1575,42 MHz.

[Appleseed, 2004 :1]

Gambar jaringan satelit GPS ditunjukkan dalam Gambar 2.1.



Gambar 2.1 Jaringan satelit GPS
Sumber : Dana, 1998:1

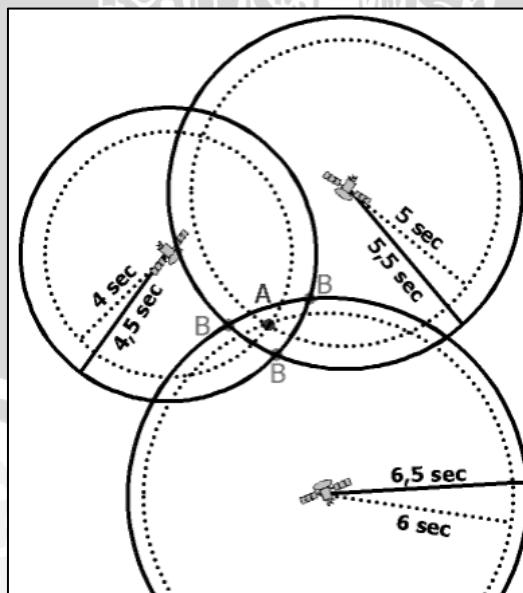
2.1.1 Cara Kerja GPS

Satelit GPS mengirimkan sinyal yang berisi *pseudo-random code* (kode yang berisi identitas satelit), *ephemeris* (berisi informasi status satelit, tanggal dan waktu saat ini) dan *almanac* (berisi lokasi setiap satelit GPS pada setiap waktu). Setiap satelit mengirimkan data *almanac* yang menunjukkan informasi orbit untuk satelit tersebut dan satelit lainnya pada sistem tersebut.

Secara lebih mudah dapat diandaikan setiap satelit mengirimkan informasi berupa “Saya satelit #X, posisi saya Y dan pesan ini dikirim pada waktu Z”. GPS Receiver akan menyimpan informasi ini dan sekaligus mengeset (membetulkan) waktu pada GPS Receiver.

Untuk menentukan posisi di permukaan bumi, GPS Receiver akan membandingkan waktu pada sinyal yang ditransmisikan oleh satelit dengan waktu pada saat sinyal tersebut diterima oleh GPS Receiver. Perbedaan waktu tersebut akan memberitahukan berapa jauh satelit tersebut dari GPS Receiver. Jika menggunakan jarak perhitungan dari beberapa satelit, suatu posisi dapat ditentukan dari bentuk segitiga yang didapat berdasarkan posisi satelit. Dengan tiga atau lebih satelit, GPS Receiver dapat menentukan posisi *latitude/longitude*, yang disebut posisi tetap 2D. Sedangkan dengan 4 atau lebih satelit, GPS Receiver dapat menentukan posisi 3D, yang terdiri dari *latitude* (lintang), *longitude* (bujur), dan *altitude* (ketinggian).

Gambar pengukuran 2D dengan tiga satelit ditunjukkan dalam Gambar 2.2.



Gambar 2.2 Hasil Pengukuran 2D dengan 3 Satelit

Sumber : Woessner, 2005:1

2.1.2 Format Data Lintang dan Bujur posisi

Standar bahasa yang digunakan oleh GPS *receiver* yang dipergunakan adalah NMEA 0813 (*National Marine Electronics Association*). NMEA merupakan standar dari antarmuka elektronik dan protokol data untuk komunikasi data antara alat-alat kelautan.

[Garmin, 2002 : 5]

Format data posisi (bujur dan lintang) yang dikeluarkan oleh GPS *Receiver* memiliki format :

\$GPRMC, <1>, <2>, <3>, <4>, <5>, <6>, <7>, <8>, <9>, <10>, <11>, <12>*hh<CR><LF>

Tabel keterangan dari format data GPS ditunjukkan dalam Tabel 2.1.

Tabel 2.1 Tabel Keterangan Format data GPRMC

<1>	UTC time, hhmmss format
<2>	Status, A = Valid position, V = NAV receiver warning
<3>	Latitude, ddm.mmmmm format
<4>	Latitude Hemisphere, N or S
<5>	Longitude, dddmm.mmmmm format
<6>	Longitude Hemisphere, E or W
<7>	Speed over ground, 000.0 to 999.9 knots
<8>	Course over ground, 000.0 to 359.9 degrees
<9>	UTC date of position fix, ddmmyy format
<10>	Magnetic variation, 000.0 to 180.0 degrees
<11>	Magnetic variation direction, E or W
<12>	Mode Indicator

ddmm.mmmmm = 2 digit derajat, 2 digit menit dan 4 digit menit dibelakang koma

Sumber : Garmin, 2002 : 15

Contoh data yang diperoleh dari GPS *Receiver* adalah

\$GPRMC,060149,A,0756.5394,S,11237.0438,E,000.0,000.0,060606,001.4,E*66

data tersebut berarti :

060149 menunjukkan waktu UTC yaitu jam 06; menit 01; detik 49

A berarti data tersebut valid / benar

0756.5394 menunjukkan koordinat latitudenya pada 07 derajat 56.5394 menit

S menunjukkan lintang selatan

11237.0483 menunjukkan koordinat longitudenya pada 112 derajat 37.0438 menit

E menunjukkan bujur timur

000.0 menunjukkan kecepatan di permukaan

000.0 menunjukkan arah di permukaan

060606 menunjukkan tanggal data tersebut yaitu tanggal 06 Juni 2006

001.4 menunjukkan variasi magnetiknya

E menunjukkan arah variasi magnetiknya

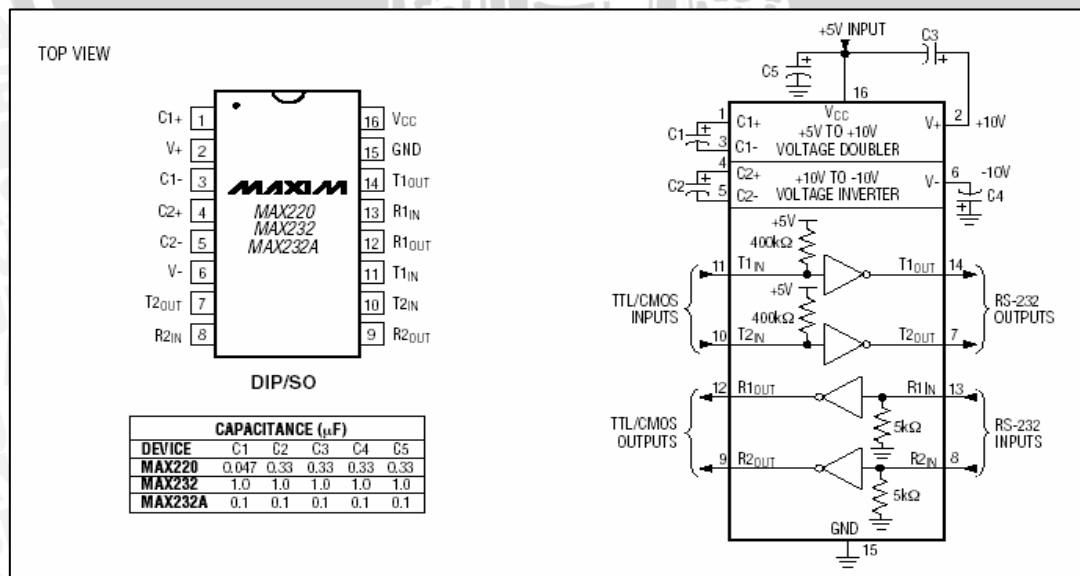
***66** adalah mode indikator

Data yang dikeluarkan oleh GPS *Receiver* tersebut sesuai dengan standar ASCII. Data yang dikeluarkan oleh GPS *Receiver* berupa data serial asinkron dengan level logika RS-232. Level logika tersebut tidak sesuai dengan level logika pada mikrokontroler yang memiliki level logika TTL/CMOS. Oleh karena itu perlu disesuaikan dengan menggunakan converter Max 232.

2.2 Max 232

Level Tegangan dari RS232 adalah +3 sampai +25 volt untuk logika “0” sedangkan -3 sampai -25 volt untuk logika “1”. Level tegangan ini berbeda dengan level tegangan logika dari mikrokontroler yang bertipe TTL/CMOS dengan supply 5 V yang memiliki keluaran untuk logika tinggi minimal 2,4 volt dan logika rendah maksimal 0,4 volt, sehingga dibutuhkan IC MAX 232 yang berfungsi sebagai penyetara level tegangan logika.

Gambar IC MAX 232 dan rangkaian minimum sistemnya ditunjukkan dalam Gambar 2.3.



Gambar 2.3 IC MAX232 dan Rangkaian minimum sistemnya

Sumber : Maxim, 2004 : 17

IC MAX 232 memiliki empat bagian yaitu *dual charge pump*, konverter tegangan, RS 232 *driver* dan RS 232 *receiver*. *Dual charge pump* merubah tegangan masukan +5V menjadi $\pm 10V$ (tak terbebani) pada RS 232 *driver*. Konverter pertama menggunakan kapasitor C1 untuk menggandakan tegangan +5V menjadi +10V di C3 pada keluaran V+. Konverter kedua menggunakan kapasitor C2 untuk membalik +10V menjadi -10V di C4 pada keluaran V-.

Keluaran dari RS 232 *driver* berayun dari $\pm 8V$ ketika dibebani dengan $5k\Omega$ (nominal) dengan Vcc sebesar 5V. *Pull up* resistor yang tersambung dengan Vcc menyebabkan keluaran dari *driver* yang tidak digunakan pada kondisi rendah karena semua *driver* adalah terbalik.

Spesifikasi dari EIA/TIA-232 E dan V.28 menentukan bahwa level tegangan yang lebih dari 3V adalah berlogika 0. Jadi, semua *receiver* adalah terbalik. *Input Threshold* ditentukan pada 0,8V dan 2,4V sehingga keluaran dari *receiver* akan sesuai dengan level tegangan dari TTL.

Keluaran dari Max 232 sudah sesuai dengan level tegangan mikrokontroler AT89S52.

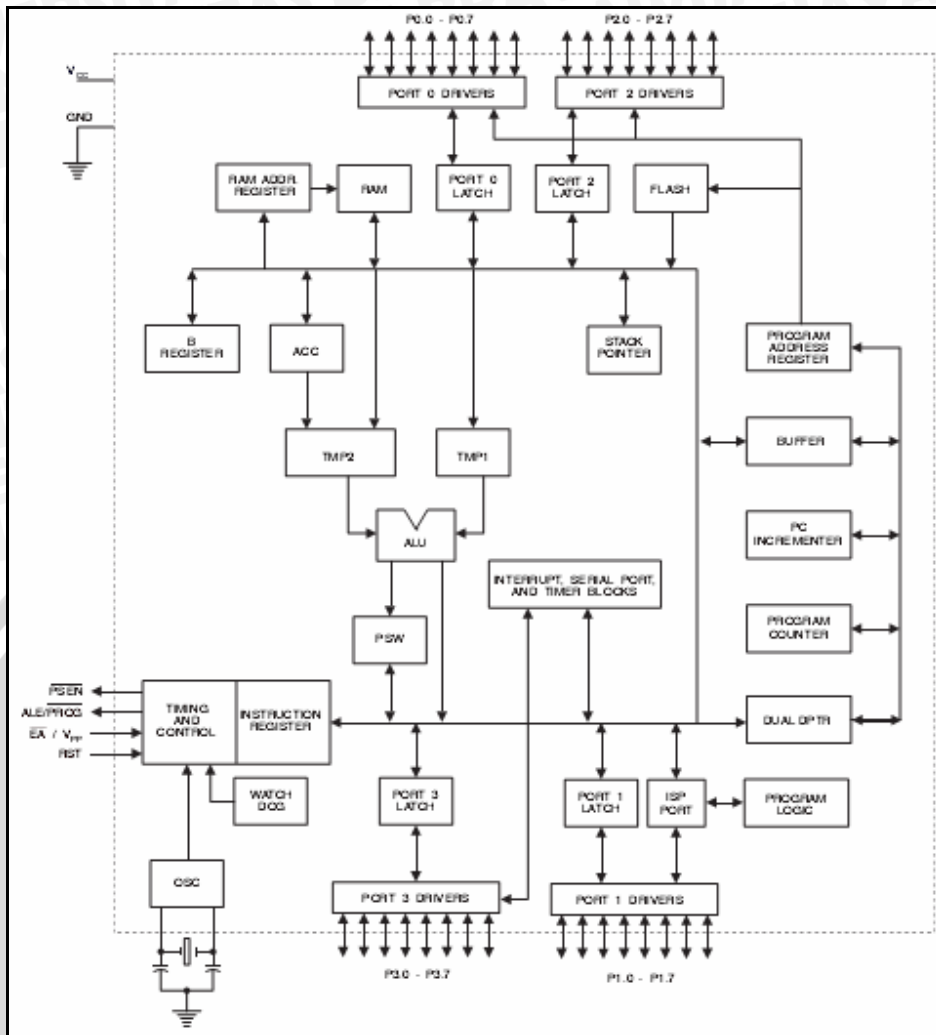
2.3 Mikrokontroler AT89S52

Mikrokontroler AT89S52 yang diproduksi oleh ATMEL *Company* Amerika Serikat merupakan salah satu anggota keluarga dari MCS-51. IC jenis ini berorientasi pada kontroler yang dapat diprogram ulang.

Mikrokontroler AT89S52 mempunyai karakteristik utama sebagai berikut:

- 8 kilo bytes Flash PEROM memori yang bisa diprogram dan dihapus
- 256 X 8 bit RAM internal
- 32 jalur I/O
- Delapan buah *vector interrupt*
- Tiga buah *timer/counter* 16 bit
- Serial UART dengan *full duplex*

Diagram blok mikrokontroler dalam Gambar 2.4 adalah mewakili semua seri dalam keluarga mikrokontroler MCS-51. Perbedaan mendasar terletak pada kapasitas memori RAM dan ROM internal.



Gambar 2.4 Blok diagram AT89S52

Sumber : Atmel ;3, 1997

Mikrokontroler menerima masukan dari GPS Receiver dan penerima DTMF. Antarmuka dengan GPS menggunakan komunikasi serial Asinkron.

2.4 Komunikasi Serial Sinkron dan Asinkron

Ada dua jenis komunikasi pada serial yaitu sinkron dan asinkron. Komunikasi bertipe sinkron dimana pada saat awal dua alat yang akan berkomunikasi melakukan sinkronisasi satu sama lain dan secara berkelanjutan mengirimkan karakter untuk menjaga agar tetap sinkron. Bahkan ketika data tidak benar-benar dikirimkan, aliran bit yang konstan memungkinkan tiap alat untuk tahu dimana alat yang lainnya pada setiap saat. Bit-bit tersebut bisa merupakan data yang sebenarnya maupun karakter *idle*. Komunikasi sinkron memungkinkan untuk transfer data yang lebih cepat daripada komunikasi asinkron karena pada komunikasi sinkron tidak membutuhkan tambahan bit

pada awal dan akhir. Serial port pada IBM PC adalah asinkron karena itu komunikasi yang bisa dilakukan adalah komunikasi serial asinkron.

Komunikasi asinkron artinya tidak ada sinkronisasi dan komunikasi tersebut tidak membutuhkan karakter *idle* yang harus dikirim atau diterima. Akan tetapi pada awal dan akhir dari byte data harus disertakan *start* dan *stop* bit. *Start* bit menandakan bahwa data tersebut akan mulai dikirim sedangkan *stop* bit menandakan bahwa data tersebut telah berakhir. Pengiriman *start* dan *stop* bit ini akan menyebabkan komunikasi asinkron berjalan lebih lambat daripada komunikasi sinkron. Pada jalur komunikasi asinkron jika sedang *idle* ditandakan dengan logika 1 (*mark*). Logika tersebut menunjukkan bahwa tidak ada data yang sedang dikirimkan. Ketika sebuah karakter akan dikirimkan maka *start* bit akan dikirimkan. *Start* bit berlogika 0 (*space*). Jika jalur berubah dari logika 1 menjadi logika 0 maka penerima akan bersiap-siap karena akan ada data yang akan diterima.

[Naiwan, 2003 : 43]

Mikrokontroler juga mengeluarkan data menuju pengirim DTMF dan menerima data dari penerima DTMF (*Dual Tone Multiple Frequency*).

2.5 Dual Tone Multiple Frequency

Dual Tone Multiple Frequency (DTMF) adalah teknik mengirimkan angka-angka pembentuk nomor telpon yang di-kode-kan dengan 2 nada yang dipilih dari 8 buah frekuensi yang sudah ditentukan. 8 frekuensi tersebut adalah 697 Hz, 770 Hz, 852 Hz, 941 Hz, 1209 Hz, 1336 Hz, 1477 Hz dan 1633 Hz. Kombinasi dari 8 frekuensi tersebut bisa dipakai untuk meng-kode-kan 16 tanda. Tabel kebenaran DTMF ditunjukkan dalam Tabel 2.2.

Tabel 2.2. Tabel kebenaran DTMF

Flow(Hz)	Fhigh(Hz)	Digit	D3	D2	D1	D0
697	1209	1	0	0	0	1
697	1336	2	0	0	1	0
697	1477	3	0	0	1	1
770	1209	4	0	1	0	0
770	1336	5	0	1	0	1
770	1477	6	0	1	1	0
852	1209	7	0	1	1	1
852	1336	8	1	0	0	0
852	1477	9	1	0	0	1
941	1209	0	1	0	1	0
941	1336	*	1	0	1	1
941	1477	#	1	1	0	0
697	1633	A	1	1	0	1
770	1633	B	1	1	1	0
852	1633	C	1	1	1	1
941	1633	D	0	0	0	0

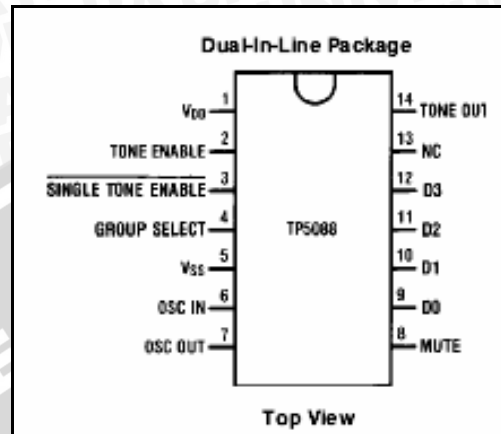
Alat pengirim kode DTMF merupakan 8 rangkaian oscilator yang masing-masing membangkitkan frekuensi ‘aneh’ di atas, ditambah dengan rangkaian pencampur frekuensi untuk mengirimkan 2 nada yang terpilih. Sedangkan penerima kode DTMF lebih rumit lagi, dibentuk dari 8 buah filter yang tidak sederhana dan rangkaian tambahan lainnya.

Beberapa pabrik membuat IC khusus untuk keperluan DTMF, diantaranya yang banyak dijumpai adalah MC145436 buatan Motorola, MT8870, MT8880 dan MT8888 buatan Mitel Semiconductor. MC145436 dan MT8870 merupakan penerima DTMF, menerima sinyal dari saluran telepon kalau ternyata sinyal yang diterima tadi merupakan kombinasi nada yang sesuai dengan ketentuan DTMF, mengeluarkan kode biner sesuai dengan kombinasi nada tersebut.

2.5.1 Pemancar DTMF dengan TP5088

TP5088 *tone encoder* merupakan IC yang didesain untuk membangkitkan sinyal DTMF. IC ini membutuhkan sebuah masukan sinyal sinusoida yang biasanya di suplai

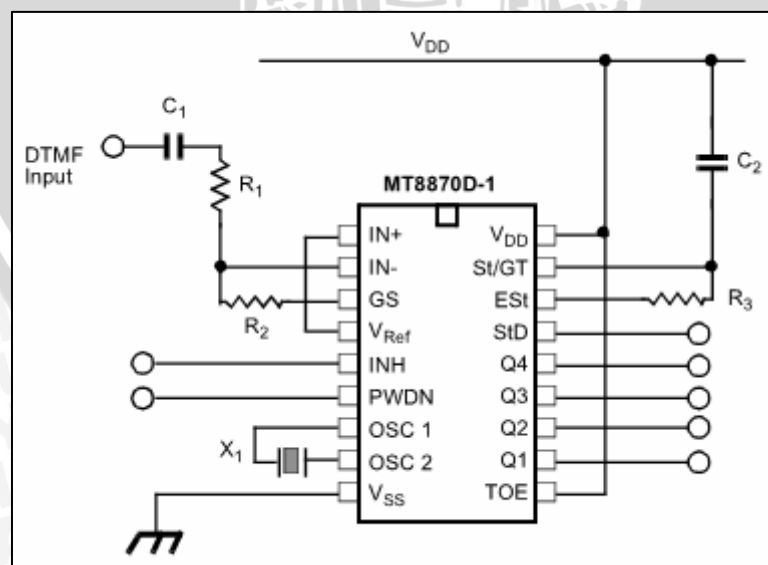
crystal pada frekuensi 3.579545 MHz untuk membangkitkan delapan macam sinyal sinusoida yang frekwensinya berbeda. Ketika membangkitkan sinyal *dual-tone*, encoder membangkitkan satu nada rendah dan satu nada tinggi dan menambahkan keduanya untuk outputnya.



Gambar 2.5 Kaki-kaki IC TP5088
Sumber : National :3, 1991

2.5.2 Penerima DTMF dengan MT8870

MT8870 dilengkapi dengan kristal X1(3.579545 MHz), C2 dan R3 dipakai untuk menentukan waktu minimal untuk mengenali nada DTMF yang diterima, rangkaian penguat sinyal DTMF dibentuk dengan R1, C1 dan R2. Nilai-nilai komponen ini langsung diambil dari lembar data (data sheet) MT8870 yang sudah disesuaikan dengan karakteristik sinyal DTMF pada umumnya.



Gambar 2.6 Rangkaian penerima DTMF dengan MT8870
Sumber : Mitel, 1995

- stD (Delayed Steering - kaki 15 MT8870) merupakan output yang menandakan MT8870 mempunyai data DTMF baru yang bisa diambil. Saat tidak ada nada DTMF kaki $stD=0$, jika sinyal yang masuk MT8870 mengandung nada DTMF dan nada itu lamanya melebihi konstanta waktu yang ditentukan oleh C2 dan R3, stD akan menjadi '1' memberitahu AT89S52 bahwa ada data di $D0..D3$ (kaki 11 sampai dengan 14 MT8870) yang bisa di ambil. Sinyal stD akan tetap bertahan = '1' manakala nada DTMF masih ada. toE (Tristate Output Enable - kaki 10 MT8870) merupakan input untuk mengatur data di $D0..D3$, jika $toE=0$ rangkaian output $D0..D3$ akan mengambang (high impedance state) sehingga data tidak bisa diambil. Jika $D0..D3$ tidak digabungkan dengan jalur data peralatan lainnya, kaki toE bisa saja dihubungkan ke '1'.



BAB III

METODOLOGI PENELITIAN

Penyusunan skripsi ini didasarkan pada masalah yang bersifat aplikatif, yaitu perencanaan dan perealisasiannya agar dapat menampilkan unjuk kerja sesuai dengan yang direncanakan dengan mengacu pada rumusan masalah. Data dan spesifikasi komponen yang digunakan dalam perencanaan merupakan data sekunder yang diambil dari buku data komponen elektronika. Pemilihan komponen berdasarkan perencanaan dan disesuaikan dengan komponen yang ada di pasaran.

Langkah-langkah yang perlu dilakukan untuk merealisasikan alat yang akan dibuat adalah sebagai berikut:

1. Studi Literatur.
2. Perencanaan Alat.
3. Pembuatan Alat dan perangkat lunak.
4. Pengujian Alat

3.1 Studi Literatur

Studi literatur mengacu pada prinsip kerja alat yang dirancang meliputi *Global Positioning System (GPS)*, *Max232*, *Komunikasi Sinkron dan Asinkron*, *Dual Tone Multiple Frequency (DTMF)*, serta Mikrokontroler *AT89S51*.

Hal-hal yang berhubungan adalah sebagai berikut:

- a. Studi tentang *Global Positioning Sistem (GPS)*
 - Teknik pembacaan data posisi dari GPS
 - Konversi koordinat dari GPS ke dalam satuan jarak
 - Protokol NMEA 0813
- b. Studi tentang *Max232*
 - Level logika RS232.
 - Data sheet *Max232*.
- c. Studi tentang komunikasi Sinkron dan Asinkron
 - Prinsip kerja komunikasi Asinkron
 - Teori komunikasi Asinkron pada Mikrokontroler

- d. Studi tentang *Dual Tone Multiple Frequency* (DTMF)
 - Prinsip kerja *Dual Tone Multiple Frequency*
 - Rangkaian DTMF baik pengirim ataupun penerima.
 - Datasheet DTMF
- e. Studi tentang Mikrokontroler
 - Hardware dari AT89S52 minimum system
 - software yang digunakan pada AT89S52
 - Pemrograman C dalam Mikrokontroler AT89S52
 - Data *sheet* AT89S52

3.2 Perencanaan Alat

Hal ini berhubungan dengan perancangan rangkaian dan komponen-komponen yang digunakan serta *lay out* (tata letak) rangkaian.

- Spesifikasi alat
 - a. Tegangan keluaran catu daya adalah 5V diperoleh dari aki 12V.
 - b. Menggunakan Mikrokontroler AT89S52 sebagai pengendali utama.
 - c. Masukan berupa permintaan data posisi kereta api terhadap stasiun yang dikehendaki yang dikirimkan oleh pusat informasi berupa sinyal DTMF.
 - d. Keluaran berupa pengiriman pengiriman sinyal DTMF yang berisi data posisi kereta api berupa jarak kereta api terhadap stasiun yang kehendaki.
- Langkah-langkah perencanaan alat
 - a. Pembuatan diagram blok rangkaian.
 - b. Perencanaan dan pembuatan rangkaian dari masing-masing blok
 - c. Manggabungkan beberapa blok menjadi keseluruhan sistem yang direncanakan
 - d. Mengadakan pengujian per blok
 - e. Pembuatan perangkat lunak

3.3 Pembuatan Alat

Pembuatan Sistem Pengolah dan Pentransmisi sinyal GPS pada Sistem Informasi Posisi Kereta Api Berbasis GPS (*Global Positioning System*) meliputi

pembuatan PCB (pembuatan *lay out*, pengetsaan dan pengeboran). Perakitan komponen dan penyolderan dilakukan pada PCB.

3.4 Pengujian Alat

Pengujian dilakukan dalam 2 bentuk yaitu pengujian perangkat keras dan perangkat lunak.

a. Pengujian Perangkat keras

Untuk pengujian perangkat keras, pengujian dilakukan per blok terlebih dahulu, kemudian dilakukan pengujian sistem. Hal-hal yang dilakukan dalam pengujian adalah sebagai berikut:

1. Pengujian keluaran GPS

Pengujian ini untuk mengetahui sinyal keluaran dari GPS

2. Pengujian IC Max232

Pengujian ini dilakukan dengan memberikan sinyal dengan level tegangan RS232 kemudian dilihat level tegangan keluaran berupa TTL/CMOS

2. Pengujian Pembangkit DTMF

Tujuan pengujian ini adalah untuk melihat sinyal keluaran dari pembangkit DTMF apakah sudah sesuai dengan data yang ada

3. Pengujian Penerima DTMF

Tujuan pengujian ini adalah untuk mengetahui apakah penerima DTMF tersebut dapat menerima sinyal DTMF dan mengkonfersikan ke format BCD dengan benar

4. Pengujian Mikrokontroler AT89S52

Tujuan pengujian ini adalah untuk mengetahui apakah mikrokontroler bekerja dengan baik yaitu dengan memberikan program sederhana kemudian diberi masukan, lalu dilihat apakah keluaran dari mikrokontroler sudah sesuai.

b. Pengujian Perangkat lunak

Untuk pengujian perangkat lunak ini digunakan fasilitas yang ada pada sistem simulasi pada software modul AT89S52 minimum sistem, hal ini digunakan untuk mengetahui kerja program agar sesuai dengan kebutuhan aplikasi ini.

c. Pengujian keseluruhan sistem.

Pengujian keseluruhan sistem dilakukan dengan menyambungkan semua blok selanjutnya mengoperasikan sistem dan dapat diketahui apakah alat ini bekerja sesuai dengan spesifikasi yang diharapkan.



BAB IV

PERANCANGAN DAN PEMBUATAN ALAT

Bab ini membahas tentang perancangan dan pembuatan Sistem Pengolah dan Pentransmisi sinyal GPS pada Sistem Informasi Posisi Kereta Api Berbasis GPS (*Global Positioning System*).

Perancangan sistem ini meliputi perancangan perangkat keras (*hardware*) dan perangkat lunak (*software*). Perancangan perangkat keras (*hardware*) meliputi perancangan rangkaian Pembangkit DTMF, rangkaian penerima DTMF, rangkaian konverter RS232 ke TTL/CMOS, dan rangkaian minimum system AT89S52. Sedangkan perangkat lunak (*software*) meliputi program untuk AT89S52 yang menggunakan bahasa C.

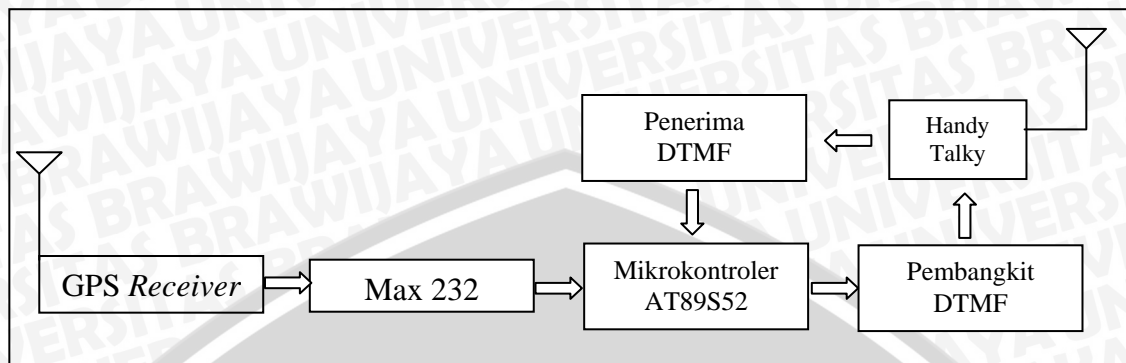
4.1 Penentuan Spesifikasi Alat

Spesifikasi alat secara global ditetapkan terlebih dahulu sebagai acuan dalam perancangan selanjutnya. Spesifikasi alat yang direncanakan adalah sebagai berikut:

- Menggunakan GPS *receiver* jenis Garmin ETREX produksi Garmin Corporation dengan format keluaran standar NMEA 0183 baudrate 4800bps.
- Alat dipasang pada lokomotif kereta api.
- Tegangan keluaran catu daya adalah 5V yang diperoleh dari aki 12V.
- Menggunakan Mikrokontroler AT89S52 sebagai pengendali utama.
- Menggunakan *Handy Talky* (HT) Kenwood tipe TM201 sebagai media pembawa sinyal DTMF.
- Masukan berupa permintaan data posisi kereta api terhadap stasiun yang dikehendaki yang dikirimkan oleh pusat informasi berupa sinyal DTMF.
- Keluaran berupa pengiriman pengiriman sinyal DTMF yang berisi data posisi kereta api berupa jarak dan arah kereta api terhadap stasiun yang kehendaki.

4.2 Diagram Blok Rangkaian

Diagram blok alat pendeteksi kualitas telur ditunjukkan dalam Gambar 4.1



Gambar 4.1. Blok Diagram alat

Keterangan diagram blok yang ditunjukkan dalam Gambar 4.1 adalah sebagai berikut :

- GPS Receiver
Berfungsi menerima posisi dalam lintang dan bujur dari kereta.
- Max 232
IC MAX232 berfungsi menyamakan level tegangan logika antara IC CMOS/TTL dengan RS232.
- Mikrokontroler AT89S52
Digunakan untuk mengatur kerja sistem secara keseluruhan.
- Penerima dan Pemancar DTMF
Berfungsi untuk membangkitkan sinyal DTMF berupa data GPS melalui HT ke pusat informasi.
- Handy Talky
Berfungsi untuk menerima atau mengirimkan sinyal data jarak kereta api yang dipancarkan oleh HT pengirim atau penerima.

Cara kerja Sistem Pengolah dan Pentransmisi sinyal GPS pada Sistem Informasi Posisi Kereta Api Berbasis GPS (*Global Positioning System*) adalah sebagai berikut :

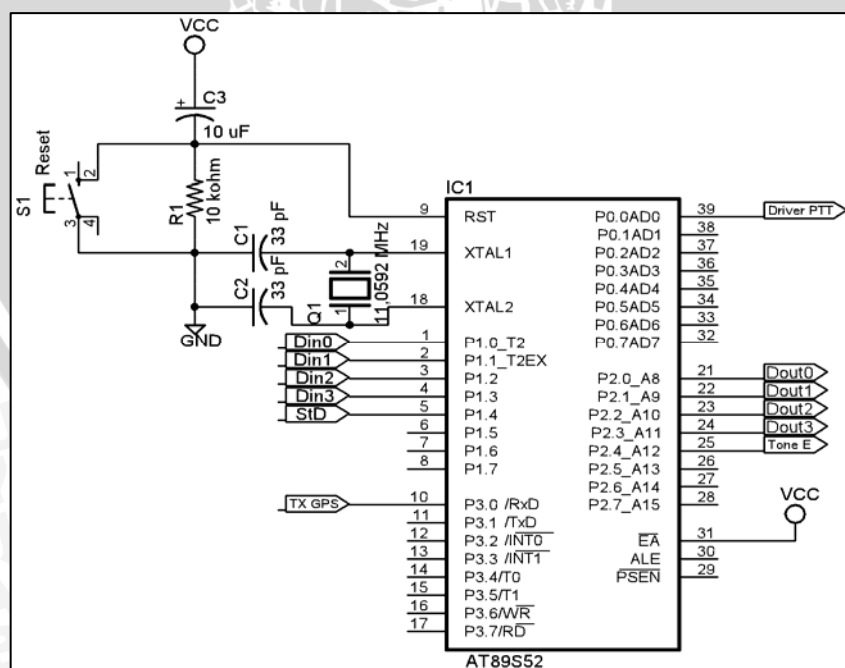
- Mikrokontroler membaca data dari penerima DTMF
- Jika ada data yang diterima dan data tersebut merupakan permintaan jarak dari pusat informasi maka Mikrokontroler akan membaca data posisi saat itu dari GPS
- Data tersebut kemudian diolah untuk kemudian dikonfersi ke jarak, berupa jarak dan arah kereta tersebut ke stasiun yang diinginkan
- Data jarak tersebut kemudian dikirim ke pusat informasi berupa nada DTMF melalui *Handy Talky* (HT).

4.3 Perancangan Perangkat Elektronik

4.3.1 Rangkaian Mikrokontroler AT89S52

Salah satu komponen yang digunakan dalam Alat ini adalah mikrokontroler AT89S52. Jenis mikrokontroler ini dipilih karena pin I/O yang mencukupi untuk penggunaan pada alat ini.

Sebagai pusat dari pengolahan data dan pengontrolan alat, pin-pin AT89S52 dihubungkan pada rangkaian pendukung membentuk suatu sistem minimum seperti dalam Gambar 4.2.

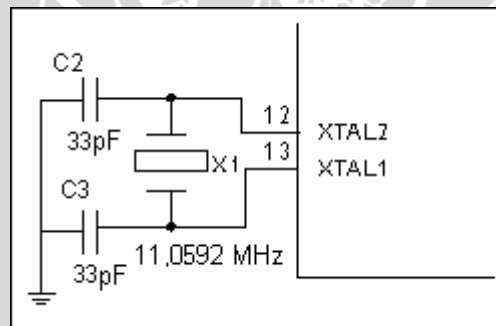


Gambar 4.2 Rangkaian Mikrokontroler

Mikrokontroler AT89S52 mempunyai 4 port yaitu port 0, port 1, port 2 dan port 3, 32 jalur yang dapat diprogram menjadi masukan atau keluaran. Dalam Gambar 4.2 ditunjukkan rangkaian mikrokontroler dimana pin-pin yang digunakan dalam perancangan pendeteksi kualitas telur ini adalah sebagai berikut :

- P0.0 dihubungkan ke masukan *driver* PTT.
- P1.0-P1.3 digunakan untuk data keluaran pada penerima DTMF.
- P1.4 dihubungkan ke kaki *StD* pada penerima DTMF.
- P2.0-P2.3 digunakan untuk menerima data dari pengirim DTMF.
- P2.4 dihubungkan ke pin *Tone Enable* pada pengirim DTMF.
- P3.0 dihubungkan ke keluaran IC MAX-232.
- XTAL1 dan XTAL2 digunakan sebagai input dari rangkaian osilator kristal.
- $\overline{\text{RESET}}$ difungsikan sebagai input pengendali reset.

Untuk dapat bekerja, mikrokontroler membutuhkan rangkaian lain sebagai sumber *clock*. Sistem yang dirancang ini akan menggunakan osilator external yang. Sumber *clock* dari kristal 8 MHz dihubungkan ke XTAL1 dan XTAL2 pada mikrokontroler AT89S52. Rangkaian osilator ditunjukkan dalam Gambar 4.3.

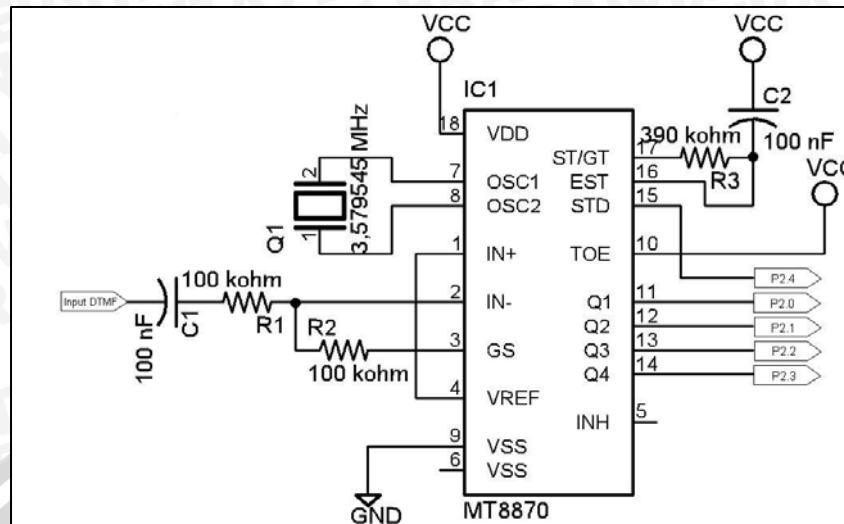


Gambar 4.3. Rangkaian osilator

Nilai C_2 dan C_3 adalah nilai yang direkomendasikan dalam *datasheet* mikrokontroler AT89S52 untuk frekuensi kristal 11,0592 MHz.

4.3.2 Rangkaian Penerima DTMF

Rangkaian penerima DTMF menggunakan komponen utama MT8870 sebagai penerima sinyal DTMF yang kemudian diubah ke format BCD. Rangkaian penerima DTMF digambarkan dalam Gambar 4.4.



Gambar 4.4. Rangkaian penerima DTMF
Sumber : Mitel, 1995:15

MT8870 dilengkapi dengan kristal (3,579545MHz), C_2 dan R_3 dipakai untuk menentukan waktu minimal untuk mengenali nada DTMF yang diterima, rangkaian penguat sinyal DTMF dibentuk dengan R_1 , C_1 , dan R_2 .

StD (*Delayed Steering*) pada kaki nomor 15 merupakan output yang menandakan MT8870 mempunyai data DTMF baru yang bisa diambil. Pada saat tidak ada nada DTMF kaki StD berlogika 0. Jika sinyal yang masuk MT8870 mengandung nada DTMF dan nada itu lamanya melebihi konstanta waktu yang ditentukan oleh C_2 dan R_3 , maka Std akan berubah menjadi logika 1. Sehingga mikrokontroler mengetahui bahwa ada data di D_0 sampai D_3 (kaki 11 sampai dengan 14) yang bisa diambil. Sinyal Std akan tetap bertahan pada logika 1 bilamana nada DTMF masih ada. Dalam Gambar 4.2 StD dipantau lewat kaki P1.4 AT89S52.

TOE (*Tristate Output Enable*) pada kaki 10 merupakan input untuk mengatur data di D_0 sampai D_3 . Jika TOE berlogika 0 rangkaian output D_0 sampai D_3 akan mengandung *high impedance state* sehingga data tidak bisa diambil. Agar data dapat diambil maka kaki TOE harus diberi logika 1. Dalam Gambar 4.2 TOE dihubungkan dengan V_{CC} .

Kapasitor kopling pada masukan IC DTMF digunakan untuk menahan arus DC agar tidak masuk ke rangkaian. Nilai kapasitansi C_1 ditentukan sebesar 100 nF, dan resistansi R_1 ditentukan sebesar 100 k Ω . Resistor R_2 digunakan sebagai umpan

balik untuk menentukan penguatan pada penguat kerja. Nilai R_2 sebesar 100 k Ω . Nilai – nilai tersebut diperoleh dari :

$$A_v = \frac{R_2}{R_1} \quad (4.1)$$

dengan penguatan yang diharapkan adalah sebesar satu.

$$1 = \frac{R_2}{100k\Omega}$$

$$R_2 = 100k\Omega$$

Waktu tunda ditentukan oleh resistor R_3 dan C_2 dari saat kombinasi nada diterima hingga dikeluarkan dalam bentuk biner 4 bit. Di dalam datasheet, besarnya R_3 dan C_2 adalah 390 k Ω , dan 100 nF. Waktu tunda t_{GTP} (*Guard Time Tone Present*) yang diperlukan DTMF untuk menentukan adanya nada baru dapat dihitung melalui persamaan dibawah ini :

$$t_{GTP} = (R_3 \cdot C_2) \ln \left[\frac{V_{DD}}{V_{DD} - V_{TSt}} \right] \quad (4.2)$$

sesuai dengan datasheet, $V_{TSt \min} = 2,2$ volt, $V_{TSt \max} = 2,5$ volt, dan $V_{DD} = 5$ volt.

Maka nilai maksimumnya diperoleh :

$$\begin{aligned} t_{GTP \max} &= (R_3 \cdot C_2) \ln \left[\frac{V_{DD}}{V_{DD} - V_{TSt \max}} \right] \quad (4.3) \\ &= (390 \cdot 10^3 \cdot 100 \cdot 10^{-9}) \ln \left[\frac{5}{5 - 2,5} \right] \\ &= 27,03274 \text{ ms} \end{aligned}$$

dan nilai minimalnya diperoleh :

$$\begin{aligned} t_{GTP \min} &= (R_3 \cdot C_2) \ln \left[\frac{V_{DD}}{V_{DD} - V_{TSt \min}} \right] \quad (4.4) \\ &= (390 \cdot 10^3 \cdot 100 \cdot 10^{-9}) \ln \left[\frac{5}{5 - 2,2} \right] \\ &= 22,61292 \text{ ms} \end{aligned}$$

dan untuk menentukan waktu tunda t_{GTA} (*Guard Time Tone Absent*) yang diperlukan untuk mendeteksi bahwa ada nada DTMF yang masuk ditentukan oleh persamaan :

$$t_{GTA} = (R_3 \cdot C_2) \ln \left[\frac{V_{DD}}{V_{TSI}} \right] \quad (4.5)$$

nilai maksimalnya :

$$t_{GTA \max} = (R_3 \cdot C_2) \ln \left[\frac{V_{DD}}{V_{TSI \min}} \right] \quad (4.6)$$

$$\begin{aligned} &= (390 \cdot 10^3 \cdot 100 \cdot 10^{-9}) \ln \left[\frac{5}{2,2} \right] \\ &= 32,018241 \text{ ms} \end{aligned}$$

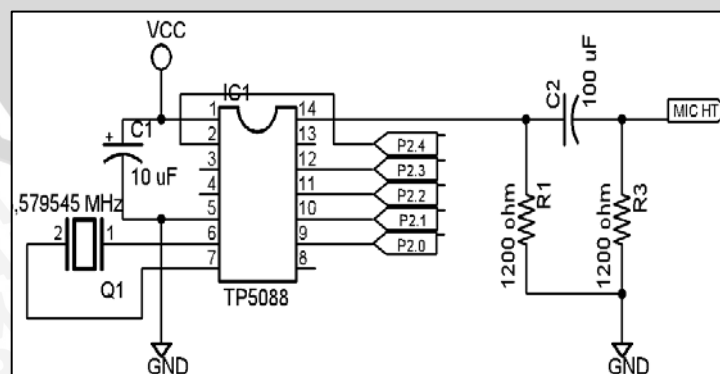
dan nilai minimalnya :

$$t_{GTA \min} = (R_3 \cdot C_2) \ln \left[\frac{V_{DD}}{V_{TSI \max}} \right] \quad (4.7)$$

$$\begin{aligned} &= (390 \cdot 10^3 \cdot 100 \cdot 10^{-9}) \ln \left[\frac{5}{2,5} \right] \\ &= 27,03274 \text{ ms} \end{aligned}$$

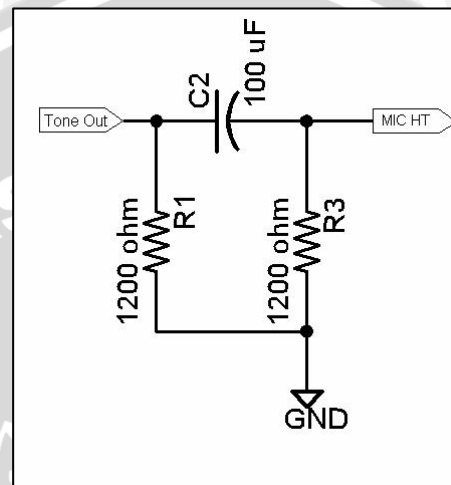
4.3.3 Rangkaian Pembangkit Nada DTMF

Rangkaian pembangkit nada DTMF ini menggunakan komponen utama yaitu TP5088 sebagai pembangkit sinyal DTMF. Masukan dari rangkaian ini berupa data dalam bentuk BCD yang kemudian diubah menjadi sinyal DTMF. Rangkaian Pembangkit nada DTMF digambarkan dalam Gambar 4.5.



Gambar 4.5. Rangkaian pembangkit nada DTMF

Rangkaian pembangkit nada DTMF seperti terlihat dalam Gambar 4.5 tersebut terdapat juga rangkaian penyesuai impedansi keluaran, karena masukan *microphone* dari *handy talky* membutuhkan sinyal masukan dengan impedansi sebesar 600Ω seperti yang tercantum dalam buku manual. Maka pada rangkaian keluaran dihubungkan dengan rangkaian penyesuai impedansi seperti terlihat dalam Gambar 4.6.



Gambar 4.6. Rangkaian penyesuai impedansi

Besarnya impedansi keluaran dalam Gambar 4.6 ditentukan dengan menggunakan penyederhanaan dengan analisis AC, Sehingga Z_C kecil sekali sehingga dianggap nol. Maka besar impedansi keluaran sebesar :

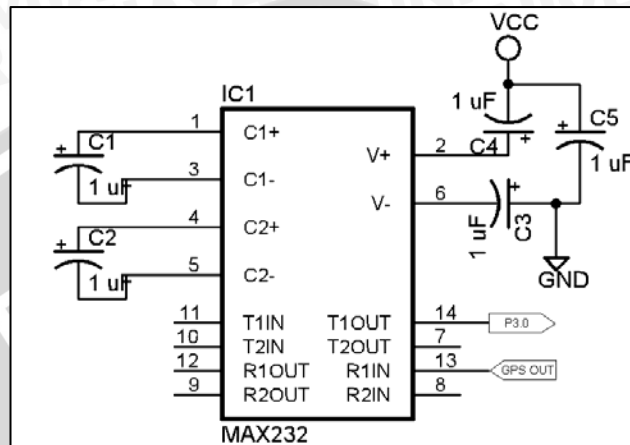
$$\begin{aligned}
 Z_{OUT} &= \frac{R_1 \cdot R_2}{R_1 + R_2} & (4.8) \\
 &= \frac{1200 \cdot 1200}{1200 + 1200} \\
 &= 600 \Omega
 \end{aligned}$$

maka nilai tersebut sesuai dengan impedansi yang diperlukan oleh *handy talky*.

4.3.4 Antarmuka dengan Max-232

Data yang keluar masuk port serial AT89S52 menggunakan level tegangan TTL. Sedangkan keluaran dari GPS Receiver menggunakan standar komunikasi serial RS 232, level tegangan yang digunakan adalah level tegangan yang berkisar antara -3 volt dan -15 volt untuk kondisi logika 1 atau yang disebut dengan keadaan *mark* dan antara $+3$ volt dan $+15$ volt untuk kondisi logika 0 atau disebut dengan

keadaan *space* atau dengan kata lain standar RS 232 menggunakan logika negatif/terbalik. Sehingga diperlukan rangkaian interface MAX232 untuk menyesuaikan level tegangan. Nilai kapasitor yang digunakan telah ditentukan sebesar 1 uF sesuai dengan *datasheet* MAX232. Rangkaian untuk *Interface Unit* RS232 ditunjukkan pada Gambar 4.7.



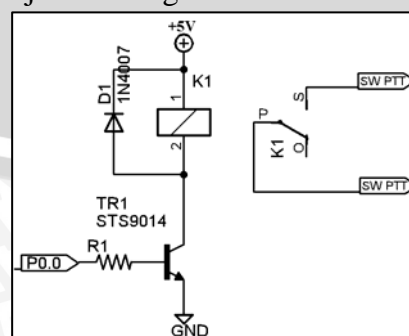
Gambar 4.7. Rangkaian MAX 232

4.3.5 Rangkaian *Switching* PTT (*Push To Talk*)

Rangkaian *switching* PTT berfungsi sebagai saklar yang menghubungkan 2 terminal PTT yang dikendalikan oleh mikrokontroler.

Rangkaian ini tersusun atas sebuah transistor tipe STS9014, dioda 1N4007, resistor basis (R_1), dan relay.

Apabila TP5088 akan mengirimkan data DTMF ke pusat informasi maka *handy talky* harus pada kondisi sebagai pemancar (*transmitter*). Oleh karena itu, dengan cara memberikan logika 1 (*high*) oleh mikrokontroler melalui port P0.0, maka akan menyebabkan TR_1 dan relay aktif, dan saklar akan menghubungkan kedua kutub PTT, sehingga mengakibatkan *handy talky* pada kondisi sebagai pemancar. Gambar 4.8 menunjukkan rangkaian PTT.



Gambar 4.8. Rangkaian *switching* PTT

Dalam perancangan rangkaian diatas, pemasangan dioda berfungsi sebagai pelindung untuk mencegah tegangan lebih pada transistor yang muncul pada kolektor yang disebabkan oleh gaya gerak listrik balik yang dibangkitkan oleh kumparan pada saat transistor mati. Berdasarkan pengukuran, nilai resistansi relay (R_{relay}) sebesar 200Ω , sehingga arus (I_{relay}) yang melewati kumparan pada saat aktif sebesar :

$$\begin{aligned} I_{\text{relay}} &= \frac{V_{CC}}{R_{\text{relay}}} & (4.9) \\ &= \frac{5}{200} \\ &= 25 \text{ mA} \end{aligned}$$

sehingga arus basis (I_B) yang diperlukan untuk menggerakkan relay sebesar :

$$\begin{aligned} I_B &= \frac{I_{\text{relay}}}{h_{FE}} & (4.10) \\ &= \frac{25 \times 10^{-3}}{100} \\ &= 250 \mu\text{A} \end{aligned}$$

jadi nilai resistansi basis (R_1) ditentukan dengan :

$$R_1 = \frac{V_{in} - V_{BE}}{I_B} \quad (4.11)$$

dimana V_{in} adalah tegangan keluaran dari mikrokontroler, jadi

$$\begin{aligned} R_1 &= \frac{(0,75 \times 5) - 0,7}{250 \times 10^{-6}} \\ &= \frac{3,05}{250 \cdot 10^{-6}} \\ &= 12,2 \text{ k}\Omega \end{aligned}$$

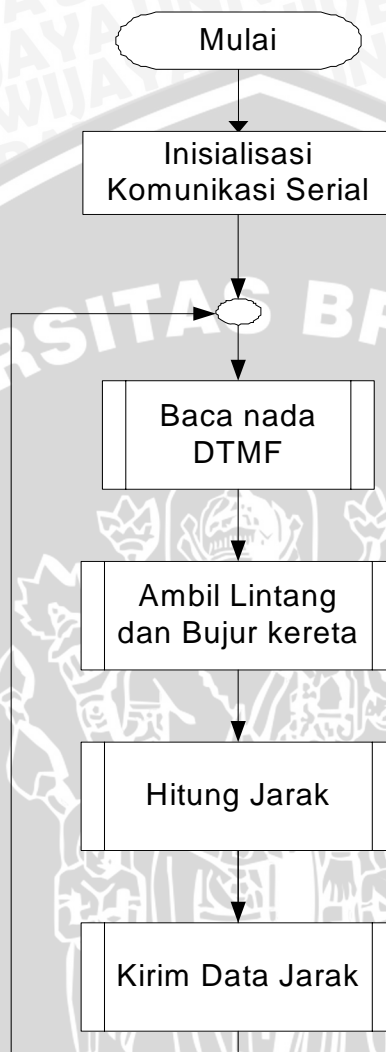
jadi nilai resistansi maksimal dari R_1 sebesar $12,2 \text{ k}\Omega$, sedangkan yang digunakan pada rangkaian ini adalah sebesar $10 \text{ k}\Omega$ yang ada di pasaran.

4.4 Perancangan perangkat lunak

Perangkat lunak yang dibuat menggunakan bahasa C mikrokontroler AT89S52. Program dibuat dan disesuaikan sehingga tujuan akhir perancangan dapat dicapai.

4.4.1 Perancangan program utama

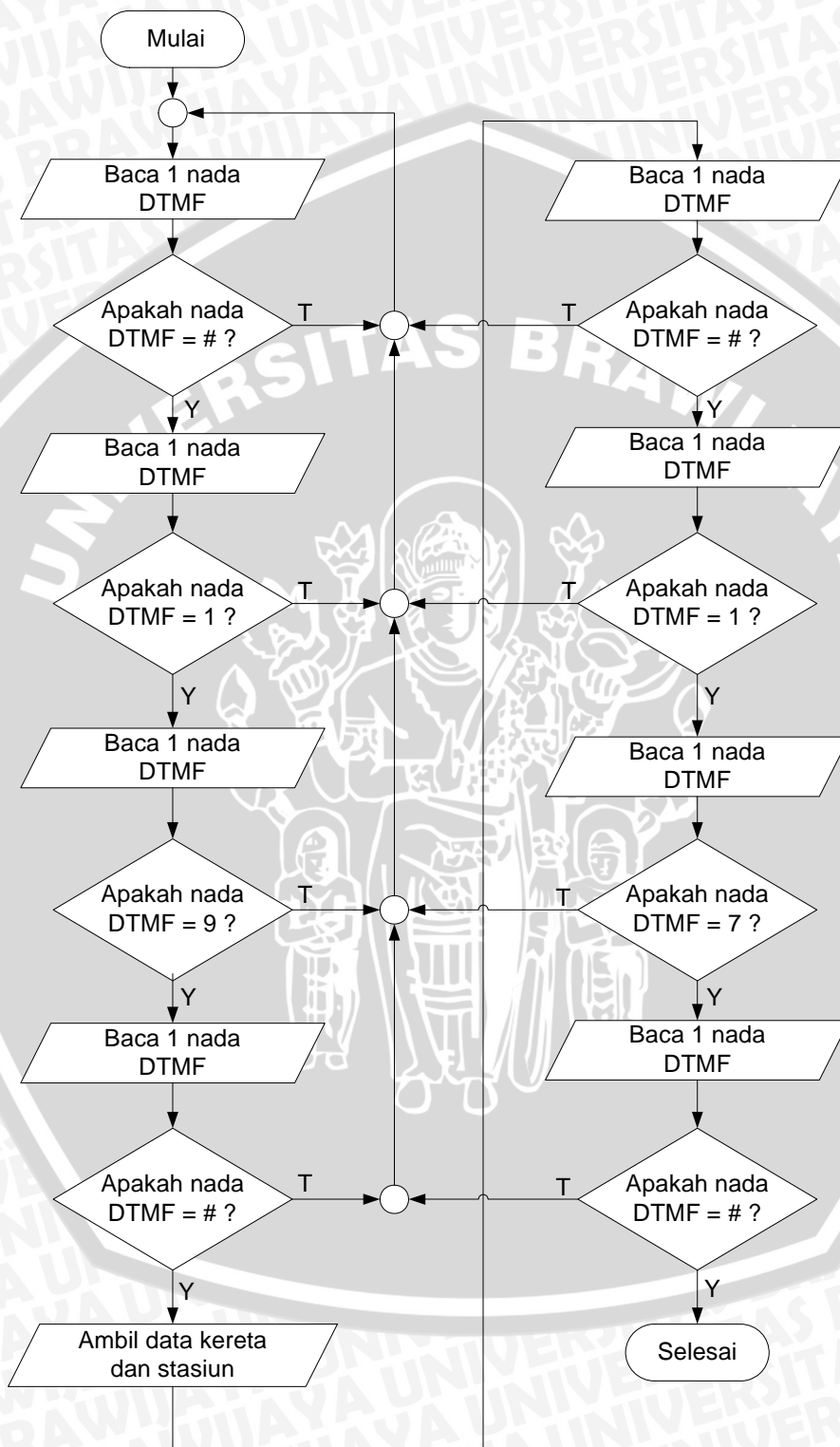
Flowchart program utama terlihat dalam Gambar 4.9.



Gambar 4.9. Flowchart program utama

4.4.2 Sub Program baca nada DTMF

Flowchart Sub Program baca nada DTMF terlihat dalam Gambar 4.10.

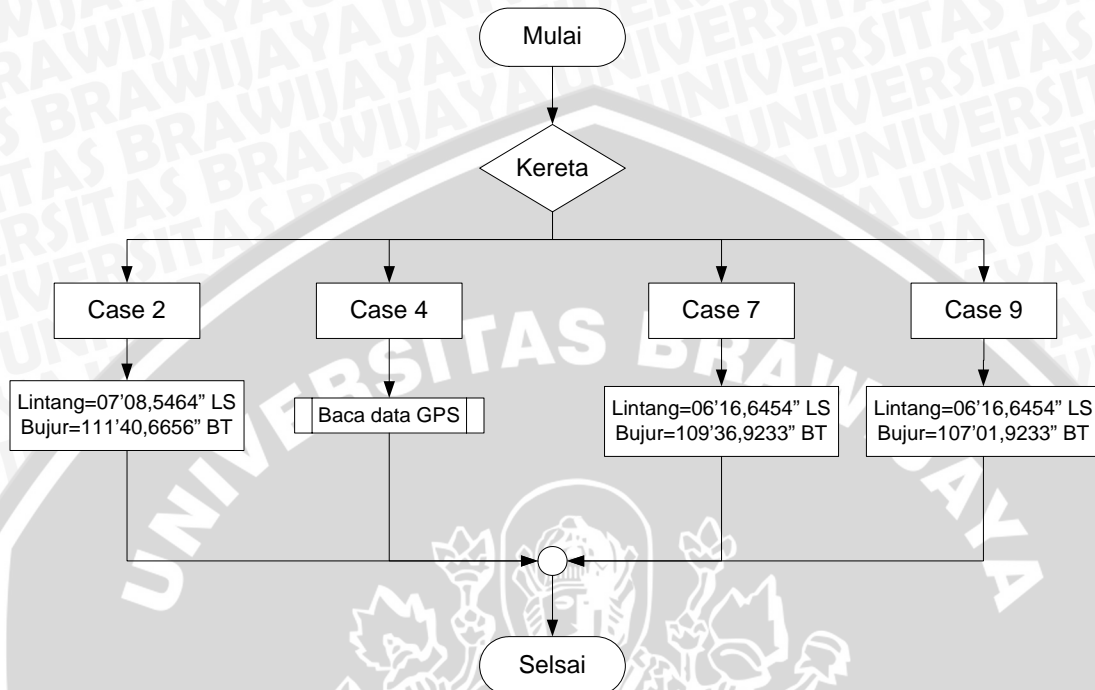


Gambar 4.10. Flowchart Sub Program baca nada DTMF



4.4.3 Sub Program Ambil Lintang dan Bujur Kereta

Flowchart Sub Program Ambil Lintang dan Bujur Kereta terlihat dalam Gambar 4.11

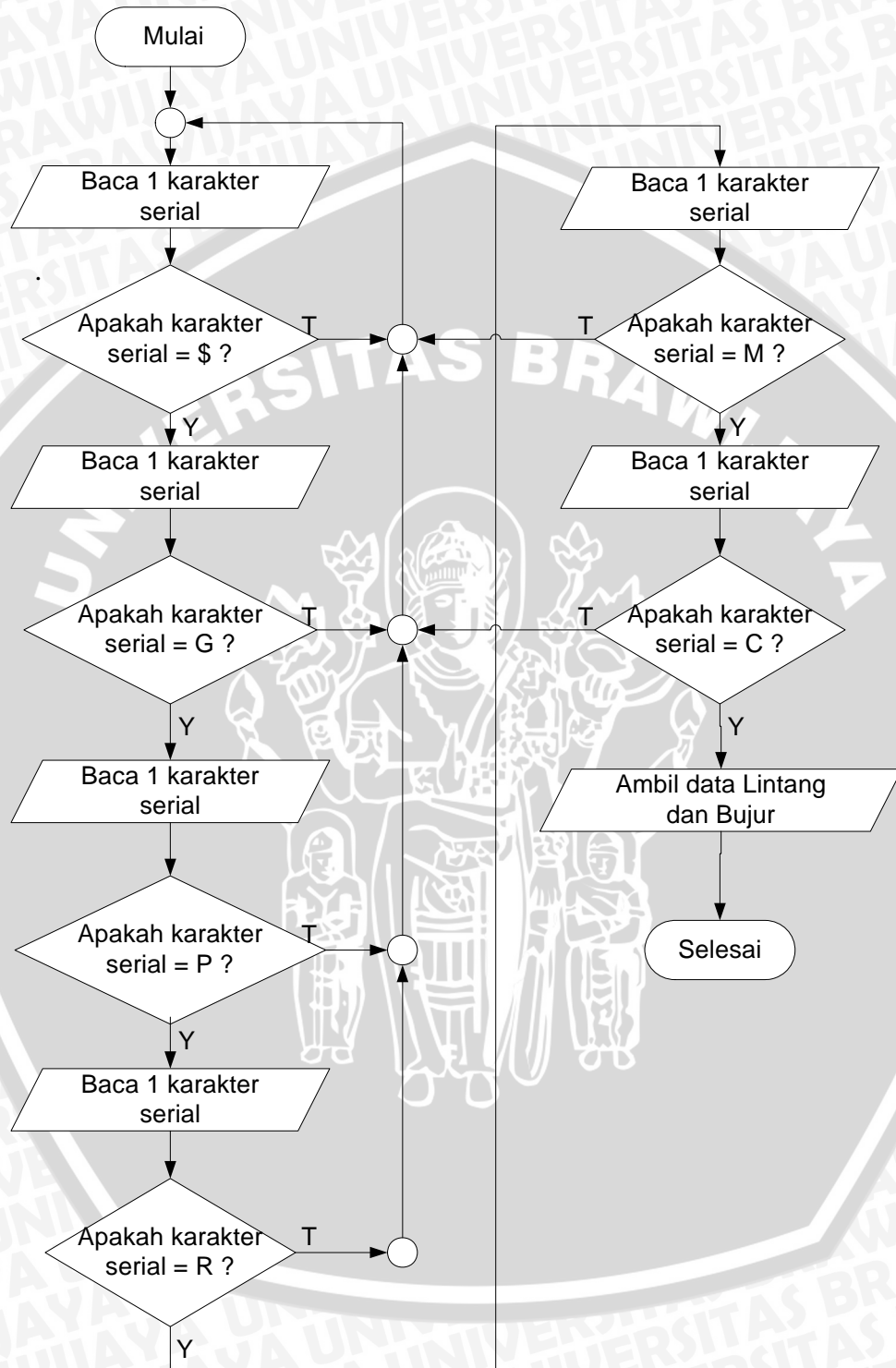


Gambar 4.11. Flowchart Sub Program Ambil Lintang dan Bujur Kereta

Dalam Perancangan alat ini digunakan 3 kereta api simulasi dan 1 kereta api yang menggunakan GPS. Kereta api simulasi antara lain kereta api 2 yang disimulasikan berada pada posisi 7 derajat 8,5464 menit lintang selatan, dan 111 derajat 40,6656 menit bujur timur. Kereta api 7 disimulasikan berada pada posisi 6 derajat 16,6454 menit lintang selatan, dan 109 derajat 36,9233 menit bujur timur. Kereta api 9 disimulasikan berada pada posisi 6 derajat 16,6454 menit lintang selatan dan 107 derajat 1,9233 menit bujur timur. Sedangkan kereta api 4 adalah kereta api yang menggunakan GPS receiver.

4.4.4 Sub Program Baca data GPS

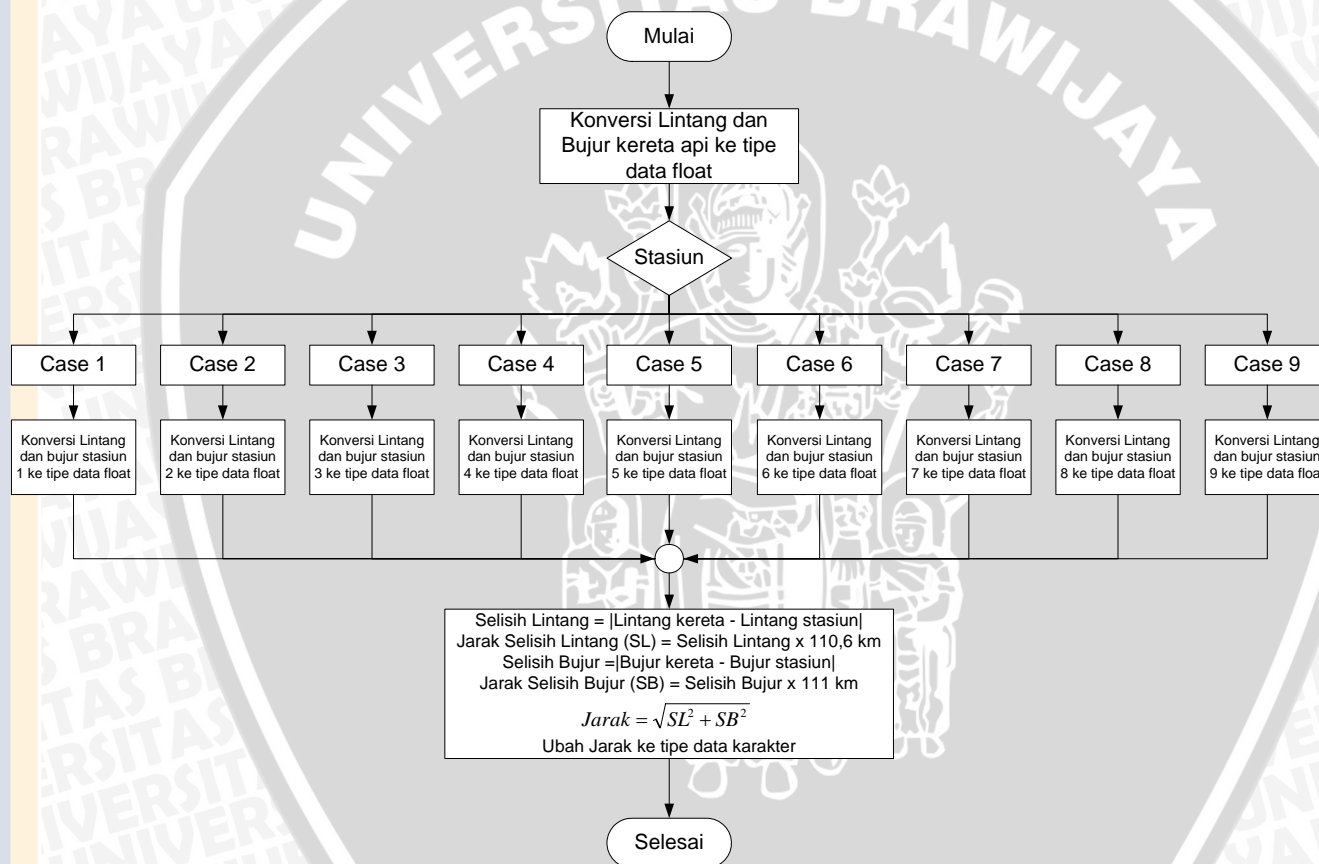
Flowchart Sub Program Baca data GPS terlihat dalam Gambar 4.12



Gambar 4.12. Flowchart Sub Program Baca data GPS

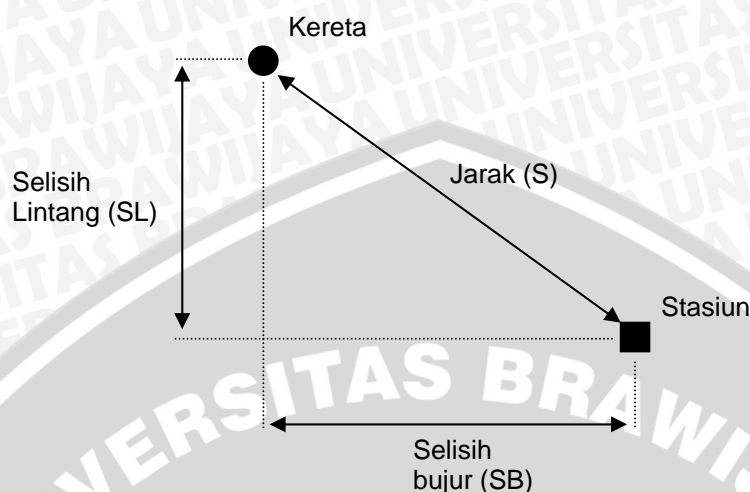
4.4.5 Sub Program Hitung Jarak

Flowchart Sub Program Hitung Jarak terlihat dalam Gambar 4.13



Gambar 4.13. Flowchart Sub Program Hitung Jarak

Proses perhitungan jarak antara kereta dan stasiun yang diinginkan adalah dengan menggunakan rumus *Pythagoras* terlihat dalam Gambar 4.14



Gambar 4.14. Metode penghitungan jarak

Rumus perhitungan jarak adalah sebagai berikut:

$$S = \sqrt{(SL \times KL)^2 + (SB \times KB)^2} \quad (4.12)$$

dengan KL adalah konstanta Lintang sebesar 110,6 km, dan SL adalah konstanta bujur sebesar 111 km. Konstanta Lintang dan Bujur merupakan konfersi 1 derajat Lintang dan Bujur ke satuan kilometer pada daerah sekitar garis equator.

Dalam perancangan kereta api bergerak dari stasiun di Surabaya menuju stasiun di Jakarta sebagai simulasi dengan rute seperti terlihat dalam Gambar 4.15

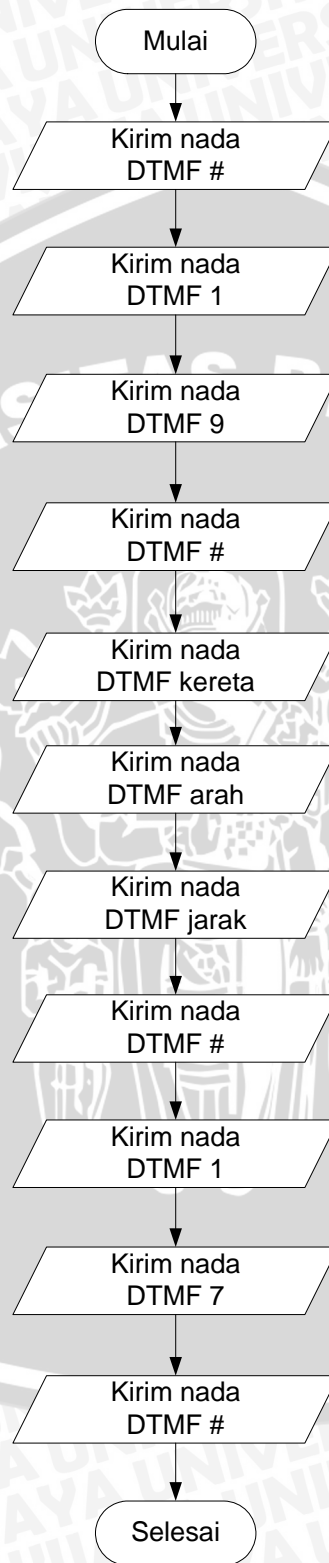


Gambar 4.15. Rute perjalanan kereta api

Kereta api bergerak dari stasiun 1 di Surabaya, stasiun 2 di Bojonegoro, stasiun 3 di Cepu, stasiun 4 di Semarang, stasiun 5 di Pekalongan, stasiun 6 di Tegal, stasiun 7 di Cirebon, stasiun 8 di Cikampek, dan berakhir di stasiun 9 di Jakarta.

4.4.6 Sub Program kirim data Jarak

Flowchart Sub Program kirim data Jarak terlihat dalam Gambar 4.16



Gambar 4.16. Flowchart Sub Program Kirim data Jarak

DAFTAR ISI

PENGANTAR	i
ABSTRAK	iii
DAFTAR ISI	iv
DAFTAR TABEL	v
DAFTAR GAMBAR	vi
I. PENDAHULUAN	
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Ruang Lingkup Pembahasan.....	2
1.4 Tujuan Penulisan	3
1.5 Sistematika Penulisan	3
II. DASAR TEORI	
2.1. <i>Global Positioning System (GPS)</i>	4
2.1.1 Cara Kerja GPS	5
2.1.2 Format Data Lintang dan Bujur Posisi.....	6
2.2 Max 232	7
2.3 Mikrokontroler AT89S52.....	8
2.4 Komunikasi Serial Sinkron dan Asinkron	9
2.5 <i>Dual Tone Multiple Frequency</i>	10
2.5.1 Pemancar DTMF dengan TP5088.....	11
2.5.2 Penerima DTMF dengan MT8870.....	12
III. METODOLOGI PENELITIAN	13
IV. PERANCANGAN DAN PEMBUATAN ALAT	17
4.1. Penentuan Spesifikasi Alat.....	17
4.2. Diagram Blok Rangkaian.....	18
4.3. Perancangan Perangkat Elektronik	19

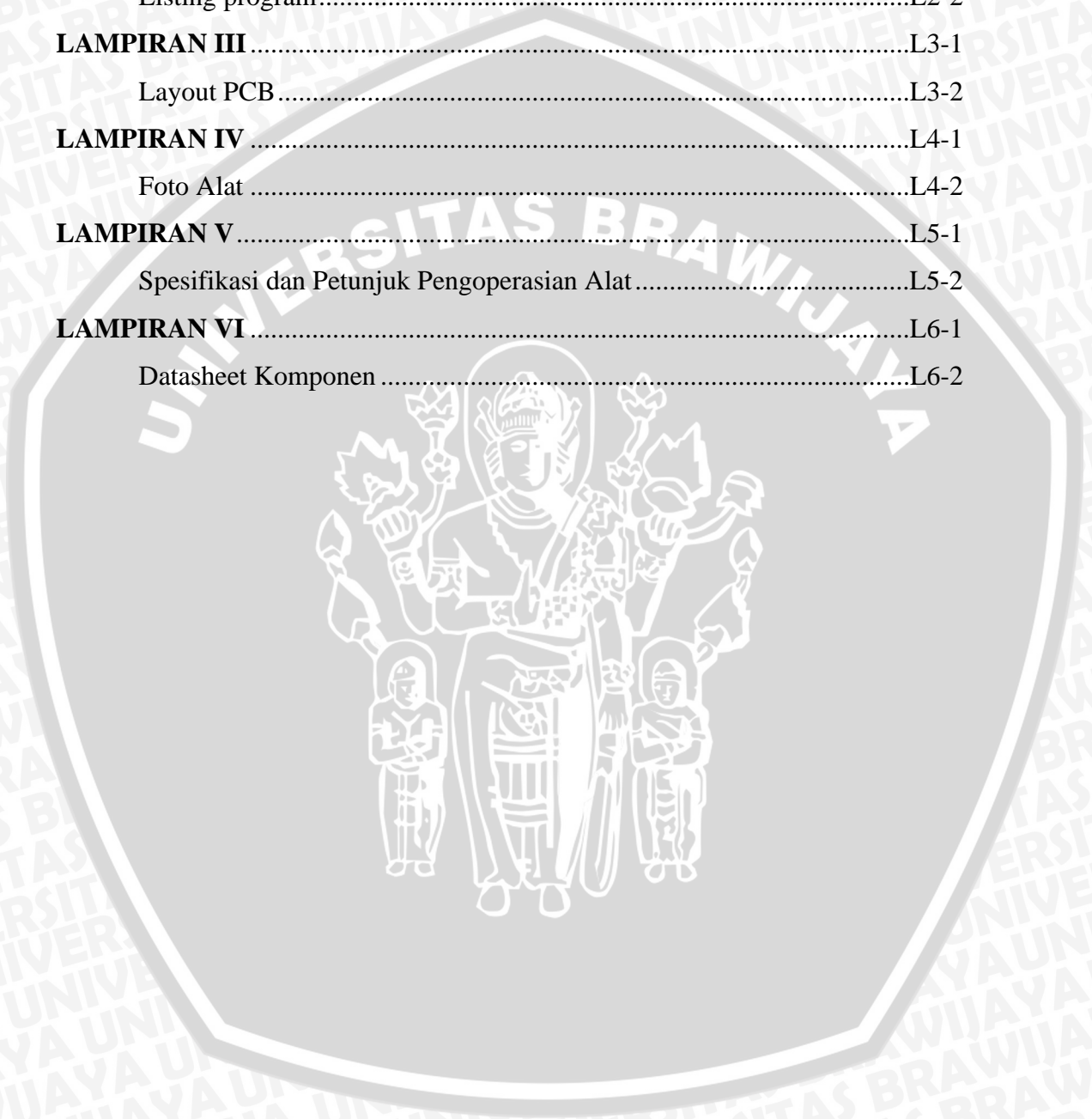


4.3.1	Rangkaian Mikrokontroler AT89S52	19
4.3.2	Rangkaian Penerima DTMF	20
4.3.3	Rangkaian Pembangkit Nada DTMF	23
4.3.4	Antarmuka dengan Max-232	24
4.3.5	Rangkaian <i>Switching</i> PTT (<i>Push To Talk</i>).....	25
4.4.	Perancangan Perangkat Lunak	26
4.4.1	Perancangan Program Utama.....	27
4.4.2	Sub Program Baca Nada DTMF	28
4.4.3	Sub Program Ambil Lintang dan Bujur Kereta.....	29
4.4.4	Sub Program Baca Data GPS.....	30
4.4.5	Sub Program Hitung Jarak	31
4.4.6	Sub Program Kirim data Jarak	33
V.	PENGUJIAN DAN ANALISIS	34
5.1	Pengujian Komunikasi Serial GPS <i>Receiver</i>	34
5.1.1	Tujuan Pengujian	34
5.1.2	Alat yang Dipergunakan	34
5.1.3	Prosedur Pengujian	35
5.1.4	Hasil Pengujian dan Analisis	36
5.2.	Pengujian Rangkaian Max-232	37
5.2.1	Tujuan Pengujian	37
5.2.2	Alat yang Dipergunakan	37
5.2.3	Prosedur Pengujian	37
5.2.4	Hasil Pengujian dan Analisis	37
5.3	Pengujian Antarmuka GPS dengan Mikrokontroler	38
5.3.1	Tujuan Pengujian	38
5.3.2	Alat yang Dipergunakan	38
5.3.3	Prosedur Pengujian	38
5.3.4	Hasil Pengujian dan Analisis	40
5.4	Pengujian Rangkaian Pembangkit DTMF	40
5.4.1	Tujuan Pengujian.....	40
5.4.2	Alat yang Dipergunakan	40

5.4.3	Prosedur Pengujian	40
5.4.4	Hasil Pengujian dan Analisis	41
5.5	Pengujian Rangkaian Penerima DTMF	42
5.5.1	Tujuan Penelitian.....	42
5.5.2	Alat yang Dipergunakan	42
5.5.3	Prosedur Pengujian	43
5.5.4	Hasil Pengujian dan Analisis	43
5.6	Pengujian Penghitungan Jarak	44
5.6.1	Tujuan Penelitian	44
5.6.2	Alat yang Dipergunakan	44
5.6.3	Prosedur Pengujian	44
5.6.4	Hasil Pengujian dan Analisis	45
5.7	Pengujian Rangkaian <i>Switching</i> PTT.....	46
5.7.1	Tujuan Penelitian	46
5.7.2	Alat yang Dipergunakan	46
5.7.3	Prosedur Pengujian	46
5.7.4	Hasil Pengujian dan Analisis	46
5.8	Pengujian Komunikasi Antar HT (<i>Handy Talky</i>).....	47
5.8.1	Tujuan Penelitian	47
5.8.2	Alat yang Dipergunakan	47
5.8.3	Prosedur Pengujian	47
5.8.4	Hasil Pengujian dan Analisis	49
5.8	Pengujian Keseluruhan Sistem.....	49
5.8.1	Tujuan Penelitian	49
5.8.2	Alat yang Dipergunakan	49
5.8.3	Prosedur Pengujian	49
5.8.4	Hasil Pengujian	50

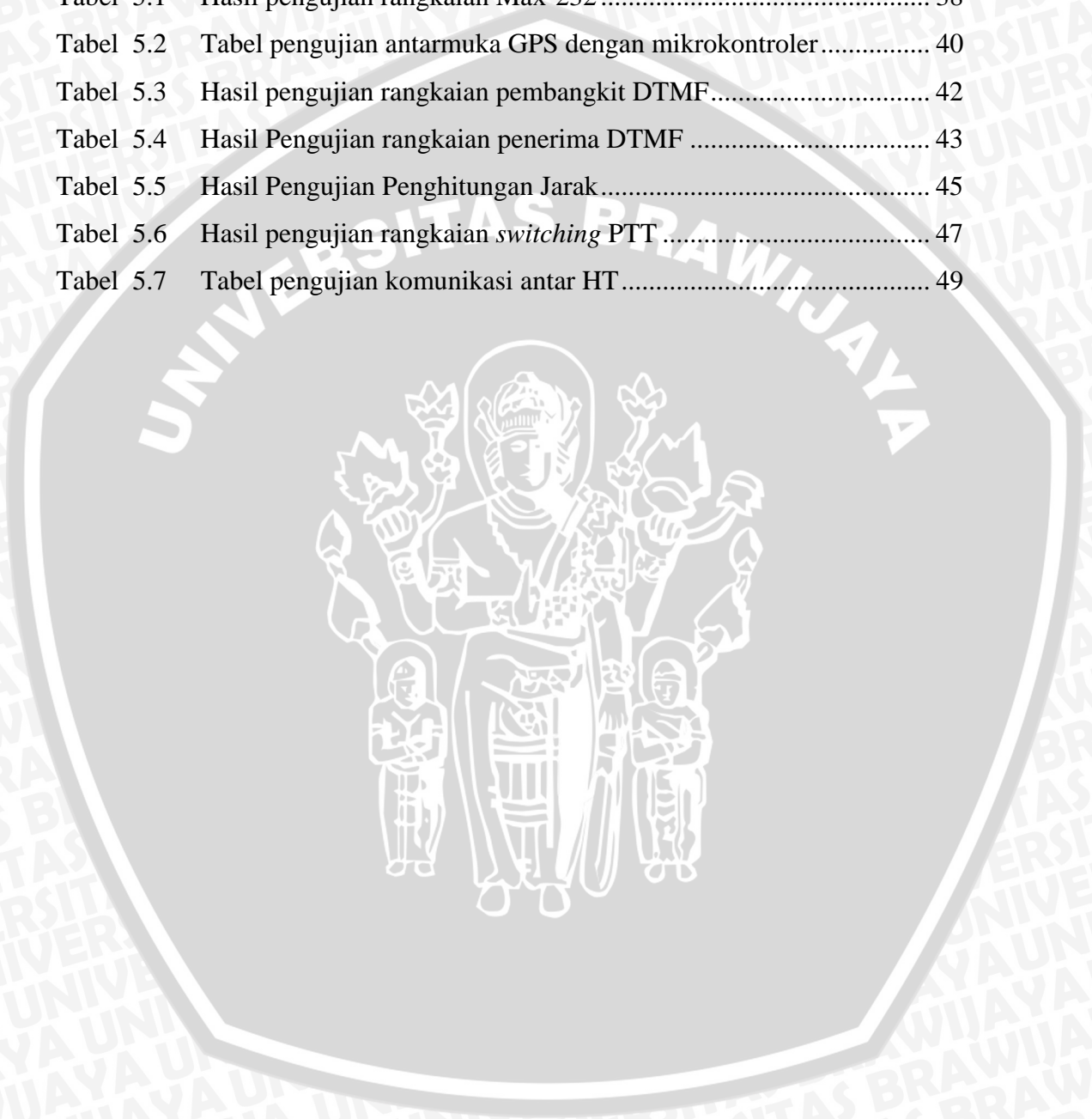
VI.	KESIMPULAN DAN SARAN	51
6.1.	Kesimpulan	51
6.2.	Saran.....	51

DAFTAR PUSTAKA	52
LAMPIRAN I	L1-1
Skema rangkaian	L1-2
LAMPIRAN II	L2-1
Listing program	L2-2
LAMPIRAN III	L3-1
Layout PCB	L3-2
LAMPIRAN IV	L4-1
Foto Alat	L4-2
LAMPIRAN V	L5-1
Spesifikasi dan Petunjuk Pengoperasian Alat	L5-2
LAMPIRAN VI	L6-1
Datasheet Komponen	L6-2



DAFTAR TABEL

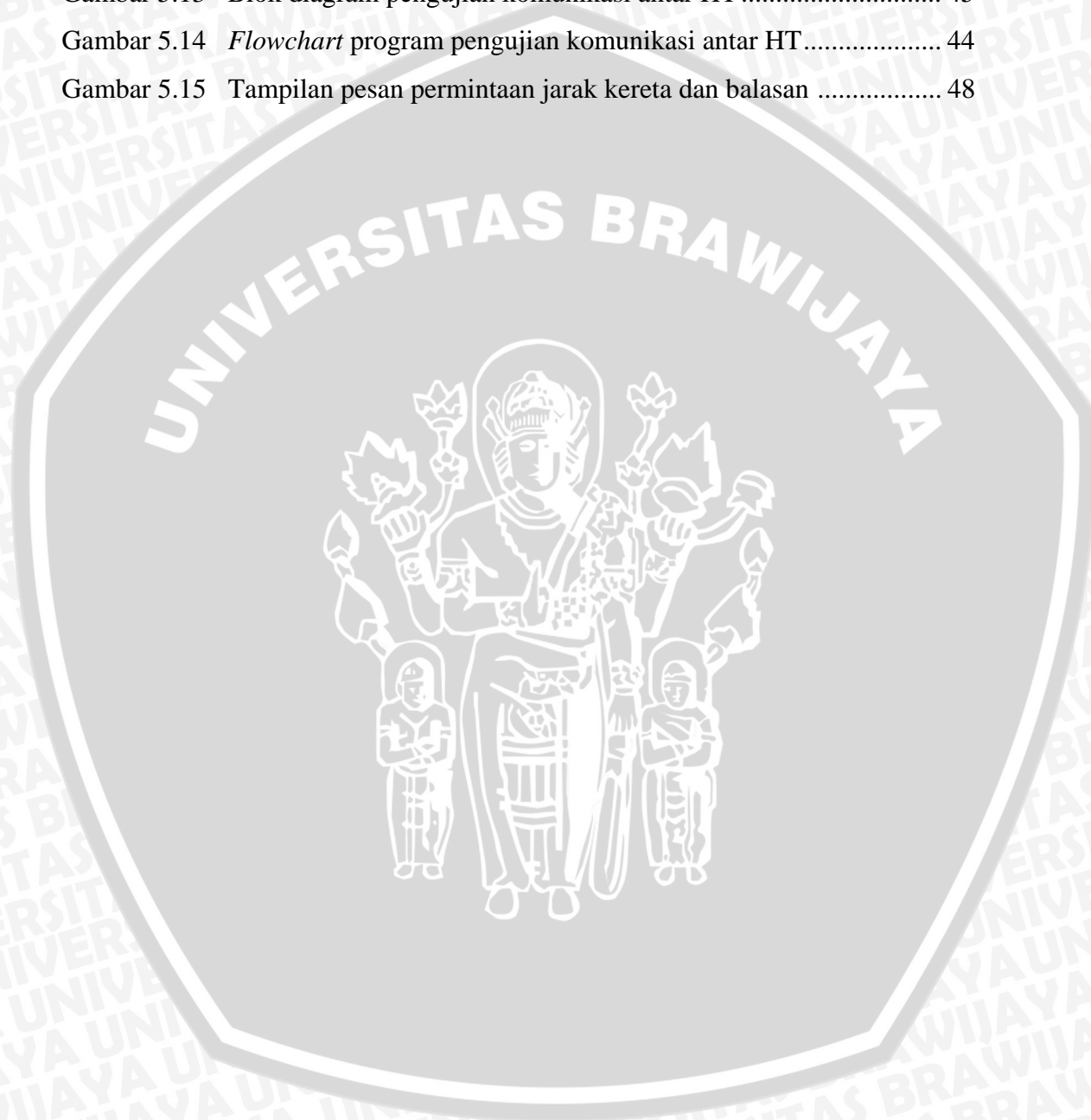
Tabel 2.1	Tabel keterangan format data GPRMC.....	6
Tabel 2.2	Tabel kebenaran DTMF.....	11
Tabel 5.1	Hasil pengujian rangkaian Max-232.....	38
Tabel 5.2	Tabel pengujian antarmuka GPS dengan mikrokontroler.....	40
Tabel 5.3	Hasil pengujian rangkaian pembangkit DTMF.....	42
Tabel 5.4	Hasil Pengujian rangkaian penerima DTMF	43
Tabel 5.5	Hasil Pengujian Penghitungan Jarak.....	45
Tabel 5.6	Hasil pengujian rangkaian <i>switching</i> PTT	47
Tabel 5.7	Tabel pengujian komunikasi antar HT.....	49



DAFTAR GAMBAR

Gambar 2.1	Jaringan Satelit GPS.....	4
Gambar 2.2	Hasil Pengukuran 2D dengan 3 satelit	5
Gambar 2.3	IC MAX232 dan rangkaian minimum sistemnya.....	7
Gambar 2.4	Blok Diagram AT89S52	9
Gambar 2.5	Kaki – kaki IC TP5088	12
Gambar 2.6	Rangkaian Penerima DTMF dengan MT8870.....	12
Gambar 4.1	Blok Diagram alat	18
Gambar 4.2	Rangkaian Mikrokontroler.....	19
Gambar 4.3	Rangkaian Osilator.....	20
Gambar 4.4	Rangkaian penerima DTMF.....	21
Gambar 4.5	Rangkaian pembangkit nada DTMF.....	23
Gambar 4.6	Rangkaian penyesuai impedansi	24
Gambar 4.7	Rangkaian MAX 232	25
Gambar 4.8	Rangkaian switching PTT	25
Gambar 4.9	<i>Flowchart</i> program utama.....	27
Gambar 4.10	<i>Flowchart</i> sub program baca nada DTMF	28
Gambar 4.11	<i>Flowchart</i> sub program ambil lintang dan bujur kereta	29
Gambar 4.12	<i>Flowchart</i> sub program baca data GPS.....	30
Gambar 4.13	<i>Flowchart</i> sub program hitung jarak.....	31
Gambar 4.14	Metode penghitungan jarak.....	32
Gambar 4.15	Rute perjalanan kereta api.....	32
Gambar 4.16	<i>Flowchart</i> sub program kirim data jarak.....	33
Gambar 5.1	Hubungan antara GPS <i>receiver</i> dengan PC	35
Gambar 5.2	Pemberian nama koneksi.....	35
Gambar 5.3	Pemilihan konfigurasi port serial.....	35
Gambar 5.4	Pengaturan spesifikasi komunikasi serial.....	36
Gambar 5.5	Tampilan keluaran aplikasi Hyper Terminal.....	36
Gambar 5.6	Blok diagram pengujian rangkaian Max-232.....	37
Gambar 5.7	<i>Flowchart</i> kirim “AT” ke HP melalui mikrokontroler	39
Gambar 5.8	Rangkaian pengujian antarmuka GPS dengan mikrokontroler.....	39

Gambar 5.9	Blok diagram pengujian pembangkit DTMF	40
Gambar 5.10	Sinyal keluaran pembangkit DTMF	41
Gambar 5.11	Blok diagram pengujian penerima DTMF	43
Gambar 5.12	Blok diagram pengujian rangkaian switching PTT	44
Gambar 5.13	Blok diagram pengujian komunikasi antar HT	45
Gambar 5.14	<i>Flowchart</i> program pengujian komunikasi antar HT	44
Gambar 5.15	Tampilan pesan permintaan jarak kereta dan balasan	48



DAFTAR PUSTAKA

- Anonim. 2000. *Global Positioning System Overview*.
<http://www.colorado.edu/geography/gcraft/notes/gps>, dicuplik pada 12 September 2006
- Anonim. 1998. *GPS Tutorial*. <http://www.mercat.com/QUEST>, dicuplik pada 12 September 2006
- Anonim. 2003. *The National Marine Electronics Association*.
<http://www.marineelectronicsjournal.com/NMEA.htm> dicuplik pada 12 September 2006
- Abidin, Hasanuddin Z, Drs., 1995. *Penentuan Posisi dengan GPS dan Aplikasinya*. Jakarta : PT Pradnya Paramita.
- Appleseed, Johnny, *The Theory and Practice of GPS*, <http://www.ia-gps.com.au/whatisgps.html>, dicuplik pada 12 September 2006
- ATMEL CORP. 1997, *8-bit Microcontroller with 8K Bytes In-System Programmable Flash AT89S52*, ATMEL, www.atmel.com, dicuplik pada 10 September 2006
- Ayala, Kenneth J. 1991. *The 8051 Microcontroller : Architecture, Programming, and Application*. St. Paul : West Publishing Company.
- Barwani M, Tjia M.O, 1994, *Prinsip Prinsip Elektronika*, edisi ketiga, Erlangga, Jakarta.
- Betke, Klaus, 2001, *The NMEA 0183 Standard*, <http://www.nmea.org>, dicuplik pada 12 September 2006
- Dana, Peter. H. 2000, *Global Positioning System Overview*, http://www.colorado.edu/geography/gcraft/notes/gps/gps_f.html, dicuplik pada 12 September 2006
- Gellert, W, 1975, *The VNR Concise Encyclopedia of Mathematics*, VNR Company, New York
- Green, DC. 2002. *Komunikasi Data*. Terjemahan Ir. P. Insap Santosa , Msc. Andi. Yogyakarta.
- Malvino, Albert Paul, 1986, *Prinsip-Prinsip Dasar Elektronika*, Jakarta: Erlangga

MITEL.1995, *MT8870/MT8870D-1 Integrated DTMF Receiver*. MITEL.
www.alldatasheet.com, dicuplik pada 10 September 2006

Naiwan, Paulus Andi. 2003, *Panduan Praktis Teknik Antarmuka dan Pemrograman Mikrikontroler AT89C51*, Jakarta: Elex Media Komputindo

National. 1991, *TP5088 DTMF Generator for Binary Data*, National Semiconductor

Putra, Agfianto Eko. 2002. *Belajar Mikrokontroler AT89C51/52/55*. Gava Media. Jakarta.

Triwisuda, Eko Teguh. 2003. *Alat Pelacak Kendaraan Jarak Jauh Berbasis GPS.* Malang.

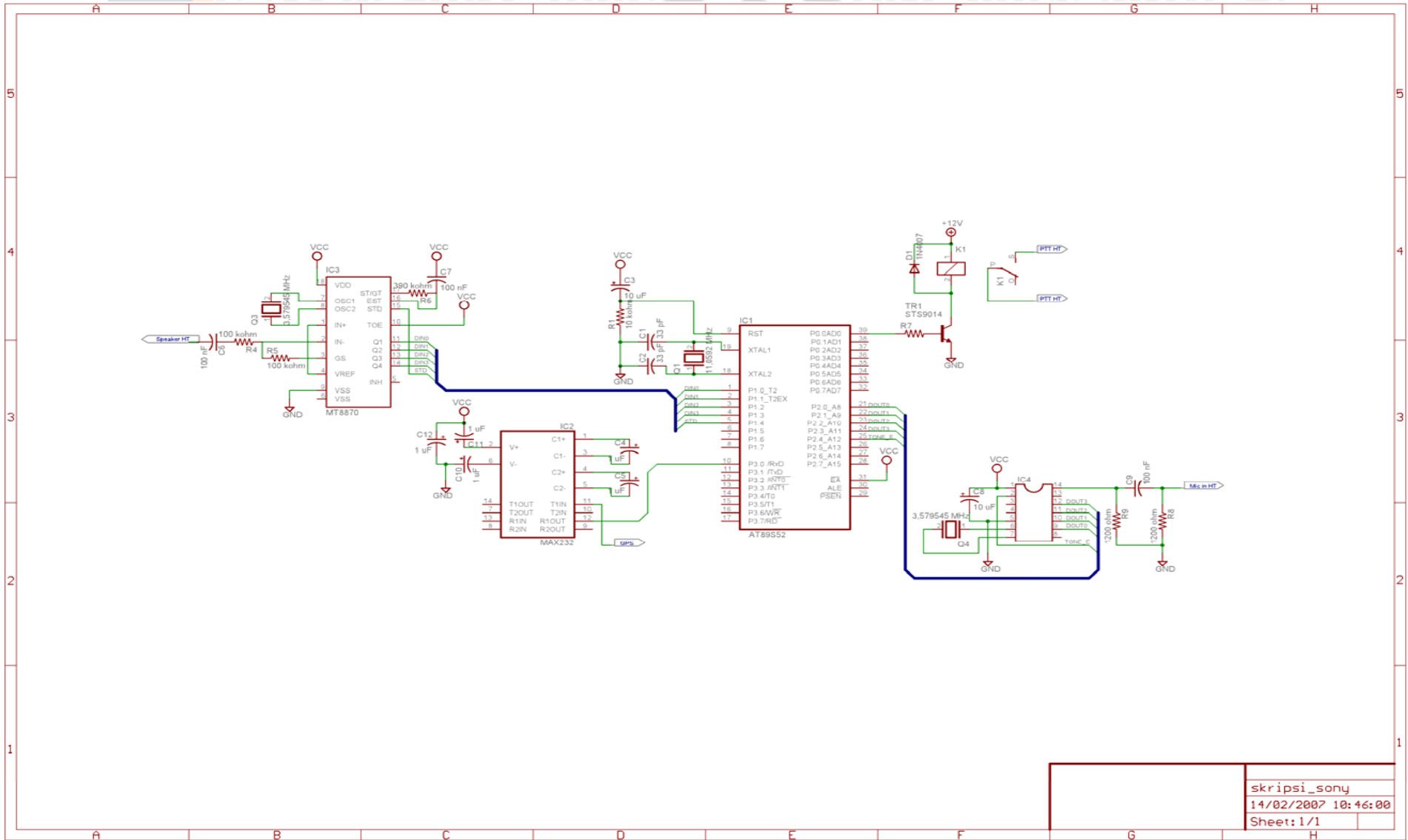
Wasito S. 1997. *Data Sheet Book 1*. PT. Elex Media Komputindo. Jakarta.

Wijaya, Irwan. 1994. *Prinsip-Prinsip dan Penerapan Digital*. Erlangga Jakarta.



LAMPIRAN I
SKEMA RANGKAIAN





LAMPIRAN II
LISTING PROGRAM




```
#include <at89x52.h>
#include <math.h>
#include <stdlib.h>
#include <string.h>
/*
```

```
=====
Definisi Port Mikrokontroler
=====
```

```
-----|
Din0-|1| 40|-Driver PTT
Din1-|2| 39|-
Din2-|3| 38|-
Din3-|4| 37|-
StD-|5| 36|-
-|6| 35|-
-|7| 34|-
-|8| 33|-
-|9| 32|-
TTL out(MAX-232)-|10| 31|-
-|11| 30|-
-|12| 29|-
-|13| 28|-
-|14| 27|-
-|15| 26|-
-|16| 25|-Tone Enable
-|17| 24|-Dout3
-|18| 23|-Dout2
-|19| 22|-Dout1
-|20| 21|-Dout0
-----|
```

```
AT89S52
```

```
=====
Pengesetan Database stasiun
=====
```

```
*/
#define lintang_st1 "0714.4895S" // Surabaya - Pasar turi
#define bujur_st1 "11244.6587E"

#define lintang_st2 "0709.94856" // Bojonegoro
#define bujur_st2 "11152.4321E"

#define lintang_st3 "0709.4589S" // Cepu
#define bujur_st3 "11135.3254E"

#define lintang_st4 "0658.4514S" // Semarang
#define bujur_st4 "11020.1285E"

#define lintang_st5 "0653.2578S" // Pekalongan
#define bujur_st5 "10940.6489E"

#define lintang_st6 "0651.4628S" // Tegal
#define bujur_st6 "10931.2849E"

#define lintang_st7 "0642.9874S" // Cirebon
#define bujur_st7 "10833.2548E"

#define lintang_st8 "0624.6512S" // Cikampek
#define bujur_st8 "10727.1589E"

#define lintang_st9 "0610.4697S" // Jakarta - Gambir
#define bujur_st9 "10649.5648E"
```

/*

```
=====
Pengesetan Porm Mikrokontroler
=====
```

*/

```
#define PortDtmfin P1
#define PortDtmfout P2
```

/*

```
=====
Deklarasi Variabel Global
=====
```

*/

```
float lintang_f1,bujur_f1,lintang_f2,bujur_f2,temp_f1,temp_f2;
unsigned char tanda_='',data_temp[]="0000",data_status=0x03,kereta=0,
                stasiun,jarak[]="0000",lintang[11],null1,bujur[12],null2,hasil=1,tanda=0,datax,
                header[]="GPRMC",data1[]=" ";
int jarak_integer;
```

/*

```
=====
Sub rutin Delay
=====
```

*/

```
void delay(unsigned int lama){
    unsigned int i;
    for (i=0;i<lama;i++){}
```

}

```
void delay2(unsigned char lama){
    unsigned char i;
    for (i=0;i<lama;i++){
        delay(65535);
    }
```

}

/*

```
=====
Sub rutin Inisialisasi Port Serial
=====
```

*/

```
void init_serial(){
    TMOD=0x20; //Timer 1 diaktifkan dengan mode 2 (timer 8 bit dengan TH1 sebagai isi ulang)
    TH1=0xf4;
    TL1=0xf4;
    SCON=0x50; //Aktifkan port serial dengan komunikasi Asinkron
    TR1=1;     // Jalankan Timer 1
}
```

```
void led(char a){
```

```
    unsigned char i;
    for(i=0;i<a;i++){
```

```
        _asm
```

```
            clr P0.0
```

```
        _endasm;
```

```
        delay2(4);
```

```
        _asm
```

```
            setb P0.0
```

```
        _endasm;
```

```
        delay2(4);
```

```
    }
```

```
}
```



```

/*
=====
Sub rutin baca 1 karakter serial
=====
*/
unsigned char baca_karakter(){
    unsigned char datax;
    while (RI==0){} // Tunggu sampai Register RI bernilai1 (Penerimaan data serial lengkap)
    RI=0; // Register RI dinolkan kembali
    datax=SBUF; // Baca data serial yang diterima, dimasukkan ke dalam variabel datax
    return(datax); // Memberikan nilai kembalian
}
void tunggu_koma(){
    unsigned char datax=';';
    while (datax!=';'){ // Tunggu sampai data serial yang diterima adalah koma
        datax=baca_karakter(); // Baca 1 karakter serial
    }
}
/*
=====

```

Sub rutin clearbit

```

/*
void clearbit (unsigned char a,char b){
    if (a==0){
        P0&=~(1<<b);
    }
    else if(a==1){
        P1&=~(1<<b);
    }
    else if(a==2){
        P2&=~(1<<b);
    }
    else{
        P3&=~(1<<b);
    }
}
/*
=====

```

Sub rutin setbit

```

/*
void setbit (unsigned char a,unsigned char b){
    if (a==0){
        P0|=(1<<b);
    }
    else if(a==1){
        P1|=(1<<b);
    }
    else if(a==2){
        P2|=(1<<b);
    }
    else{
        P3|=(1<<b);
    }
}

```




```

/*
=====
Sub rutin baca data GPS
=====
*/
void baca_data_gps(){
    unsigned char i;
    while (tanda==0){
        while (RI==0){} // Tunggu sampai penerimaan data serial lengkap
        RI=0;
        if (SBUF=='$'){
            for (i=0;i<5;i++){
                while (RI==0){}
                RI=0;
                data1[i]=SBUF;
            }
            hasil=strcmp(header,data1); // Bandingkan header dengan data1
            if (hasil==0){ // Apakah keduanya sama?
                tunggu_koma();
                tunggu_koma();
                datax=baca_karakter();
                if (datax=='A'){ // Apakah data GPS valid?
                    tanda=1;
                    datax=baca_karakter();
                    for (i=0;i<9;i++){
                        lintang[i]=baca_karakter(); // Baca lintang
                    }
                    datax=baca_karakter();
                    lintang[9]=baca_karakter();
                    datax=baca_karakter();
                    for(i=0;i<10;i++){
                        bujur[i]=baca_karakter(); // Baca bujur
                    }
                    datax=baca_karakter();
                    bujur[10]=baca_karakter();
                    tanda=1;
                }
            }
        }
        tanda=0;
    }
}
/*
=====
Sub rutin kirim 1 karakter DTMF
=====
*/
void kirim_char_dtmf(unsigned char data2_){
    data2_-=0x30;
    PortDtmfout=data2_; // Mengeluarkan data pada PortDtmfout
    clearbit(2,4);
    delay(2000);
    setbit(2,4);
    setbit(2,5);
    delay(5000);
    clearbit(2,4);
    clearbit(2,5);
}

```

```

/*
=====
Kirim data DTMF
=====
*/
void kirim_dtmf(unsigned char data_[100]){
    unsigned char i=0;
    while(data_[i]!=0){
        kirim_char_dtmf(data_[i]); // Kirim 1 karakter DTMF
        i++;
    }
}
/*
=====
Sub rutin baca 1 karakter DTMF
=====
*/
unsigned char baca_kar_dtmf(){
    unsigned char data_x,tanda_dt=9;
    do{
        data_ =P1;
        data_&=0x10;
    }while (data_==0); // Menunggu sampai pin StD bernilai 1
    data_ = PortDtmfin; // Membaca data dari PortDtmfin
    data_&=0x0f; // Diambil 4 bit rendah
    data_x=data_; // Memasukkan ke variable data_x
    while (tanda_dt==9){ // Melakukan pengecekan kedua
        do{
            data_ =PortDtmfin;
            data_&=0x10;
        }while (data_!=0);
        delay(1000);
        data_ =PortDtmfin;
        data_&=0x10;
        if (data_!=0){ // Jika kedua pengecekan sama maka data DTMF benar
            tanda_dt=9;
        }
        else
            tanda_dt=0;
    }
    return (data_x);
}
/*
=====
Sub rutin membaca masukan DTMF
=====
*/
void baca_nada_DTMF(){
    do{
        data_ =baca_kar_dtmf()+0x30;
        if (data_=='<'){ // Mengecek apakah header dan penutup sesuai
            data_ =baca_kar_dtmf()+0x30;
            if (data_=='1'){
                data_ =baca_kar_dtmf()+0x30;
                if (data_=='9'){
                    data_ =baca_kar_dtmf()+0x30;
                    if(data_=='<'){
                        data_ =baca_kar_dtmf();
                        kereta=data_ ; // Membaca kereta yang dimaksud
                        data_ =baca_kar_dtmf();
                        stasiun=data_ ; // Membaca stasiun yang dimaksud
                        data_ =baca_kar_dtmf()+0x30;
                    }
                }
            }
        }
    }while (data_!=0);
}

```



```

/*
=====
Sub rutin konversi variabel bujur (karakter) ke tipe data float
=====
*/
void konversi_ke_float_bujur(unsigned char data_kar[12]){
    strcpy(data_temp,"0000"); // Menolkan variable data_temp
    data_temp[1]=data_kar[0]; // Mengambil derajat Bujur
    data_temp[2]=data_kar[1];
    data_temp[3]=data_kar[2];
    temp_f2=atoi(data_temp); // Mengubah ke tipe data integer derajat Bujur

    strcpy(data_temp,"0000"); // Menolkan variable data_temp
    data_temp[2]=data_kar[3]; // Mengambil menit Bujur
    data_temp[3]=data_kar[4];
    temp_f1=atoi(data_temp); // Mengubah ke tipe data integer menit Bujur
    temp_f1/=60; // Mengubah ke derajat
    temp_f2+=temp_f1; // Menambahkan ke derajat

    strcpy(data_temp,"0000"); // Menolkan variable data_temp
    data_temp[0]=data_kar[5]; // Mengambil 4 angka di belakang koma menit Bujur
    data_temp[1]=data_kar[6];
    data_temp[2]=data_kar[7];
    data_temp[3]=data_kar[8];
    temp_f1=atoi(data_temp); // Mengubah ke tipe data integer 4 angka di belakang koma
    temp_f1/=600000; // Mengubah ke derajat 1 angka di belakang koma
    temp_f2+=temp_f1; // Menambahkan ke derajat
    temp_f1=temp_f2; // Hasil konversi ditaruh dalam variable temp_f1
}
/*
=====

```

Sub rutin konversi variable jarak (integer) ke tipe data karakter

```

/*
=====
void ubah_ke_string_jarak(int data_int){
    data_=0;
    while(data_int>=1000){ // Ambil Ribuan
        data_int-=1000;
        data_++;
    }
    jarak[0]=data_+0x30;
    data_=0;
    while(data_int>=100){ // Ambil Ratusan
        data_int-=100;
        data_++;
    }
    jarak[1]=data_+0x30;
    data_=0;
    while(data_int>=10){ // Ambil Puluhan
        data_int-=10;
        data_++;
    }
    jarak[2]=data_+0x30;
    jarak[3]=data_int+0x30; // Ambil Satuan
}

```

/*

Sub rutin Penghitungan Jarak

*/

```
void hitung_jarak(){
    konversi_ke_float_lintang(lintang); // Konversi ke tipe data float Lintang dari kereta
    lintang_f1=temp_f1;
    konversi_ke_float_bujur(bujur); // Konversi ke tipe data float Bujur dari kereta
    bujur_f1=temp_f1;
    switch(stasiun){ // Konversi Lintang dan bujur stasiun ke tipe data float
        case 1:
            konversi_ke_float_lintang(lintang_st1);
            lintang_f2=temp_f1;
            konversi_ke_float_bujur(bujur_st1);
            bujur_f2=temp_f1;
            break;
        case 2:
            konversi_ke_float_lintang(lintang_st2);
            lintang_f2=temp_f1;
            konversi_ke_float_bujur(bujur_st2);
            bujur_f2=temp_f1;
            break;
        case 3:
            konversi_ke_float_lintang(lintang_st3);
            lintang_f2=temp_f1;
            konversi_ke_float_bujur(bujur_st3);
            bujur_f2=temp_f1;
            break;
        case 4:
            konversi_ke_float_lintang(lintang_st4);
            lintang_f2=temp_f1;
            konversi_ke_float_bujur(bujur_st4);
            bujur_f2=temp_f1;
            break;
        case 5:
            konversi_ke_float_lintang(lintang_st5);
            lintang_f2=temp_f1;
            konversi_ke_float_bujur(bujur_st5);
            bujur_f2=temp_f1;
            break;
        case 6:
            konversi_ke_float_lintang(lintang_st6);
            lintang_f2=temp_f1;
            konversi_ke_float_bujur(bujur_st6);
            bujur_f2=temp_f1;
            break;
        case 7:
            konversi_ke_float_lintang(lintang_st7);
            lintang_f2=temp_f1;
            konversi_ke_float_bujur(bujur_st7);
            bujur_f2=temp_f1;
            break;
        case 8:
            konversi_ke_float_lintang(lintang_st8);
            lintang_f2=temp_f1;
```

```

        konversi_ke_float_bujur(bujur_st8);
        bujur_f2=temp_f1;
        break;
    case 9:
        konversi_ke_float_lintang(lintang_st9);
        lintang_f2=temp_f1;
        konversi_ke_float_bujur(bujur_st9);
        bujur_f2=temp_f1;
        break;
    default:
    };
    temp_f1=lintang_f1-lintang_f2; // Hitung selisih lintang
    lintang_f1=fabsf(temp_f1)*110.6; // Hitung jarak selisih lintang
    temp_f1=bujur_f1-bujur_f2; // Hitung selisih bujur
    if (temp_f1>=0)
        tanda_='>';
    else
        tanda_='=';
    bujur_f1=fabsf(temp_f1)*111; // Hitung jarak selisih bujur
    lintang_f1*=lintang_f1; // Kuadratkan jarak selisih Lintang
    bujur_f1*=bujur_f1; // Kuadratkan jarak selisih Bujur
    temp_f1=lintang_f1+bujur_f1; // Jumlahkan kuadrat jarak selisih Lintang dan Bujur
    temp_f2=sqrtf(temp_f1); // Mengakarkan hasil penjumlahan
    temp_f2*=10; // Mengalikan hasil perhitungan dengan 10
    temp_f2+=0.5; // Menambahkan 0,5 pada hasil
    jarak_integer=temp_f2; // Hasil ditaruh dalam variable jarak_integer (bertipe integer)
    ubah_ke_string_jarak(jarak_integer); // Mengubah ke tipe data String data jarak
}
/*
=====
Sub rutin baca lintang dan bujur kereta
=====
*/
void ambil_lintang_bujur_kereta(){
    switch(kereta){
        case 2:
            strcpy(lintang,"0708.5464S"); // Ambil Lintang kereta 2
            strcpy(bujur,"11140.6656E"); // Ambil Bujur kereta 2
            break;
        case 4:
            baca_data_gps(); // Baca data Lintang dan Bujur dari GPS
            break;
        case 7:
            strcpy(lintang,"0638.6346S"); // Ambil Lintang kereta 7
            strcpy(bujur,"10730.4263E"); // Ambil Bujur kereta 7
            break;
        case 9:
            strcpy(lintang,"0614.6454S"); // Ambil Lintang kereta 9
            strcpy(bujur,"10631.9233E"); // Ambil Bujur kereta 9
            break;
        default:
            break;
    }
}
void led_kereta(){
    switch (kereta){
        case 2 :
            clearbit(0,4);
            setbit(0,5);
            setbit(0,6);
            setbit(0,7);
            break;

```



```

case 4 :
    setbit(0,4);
    clearbit(0,5);
    setbit(0,6);
    setbit(0,7);
    break;
case 7 :
    setbit(0,4);
    setbit(0,5);
    clearbit(0,6);
    setbit(0,7);
    break;
case 9 :
    setbit(0,4);
    setbit(0,5);
    setbit(0,6);
    clearbit(0,7);
    break;
default:
    setbit(0,4);
    setbit(0,5);
    setbit(0,6);
    setbit(0,7);
    break;
}
}
/*
=====
Sub rutin Mengecek data GPS
=====
*/
void cek_gps_valid(){
    unsigned char i;
    while (tanda==0){
        while (RI==0){} // Menunggu sampai data serial diterima secara lengkap 8 bit
        RI=0;
        if (SBUF=='$'){
            for (i=0;i<5;i++){
                while (RI==0){}
                RI=0;
                data1[i]=SBUF;
            }
            hasil=strcmp(header,data1); // Bandingkan header dengan data1
            if (hasil==0){ // Apakah keduanya sama?
                tunggu_koma();
                tunggu_koma();
                datax=baca_karakter();
                if (datax=='A'){ // Apakah data GPS valid
                    clearbit(0,3);
                    setbit(0,2);
                }
            }
            else{
                setbit(0,3);
                clearbit(0,2);
            }
        }
    }
}
}
}
}

```

/*

Sub Kirim Data Jarak

*/

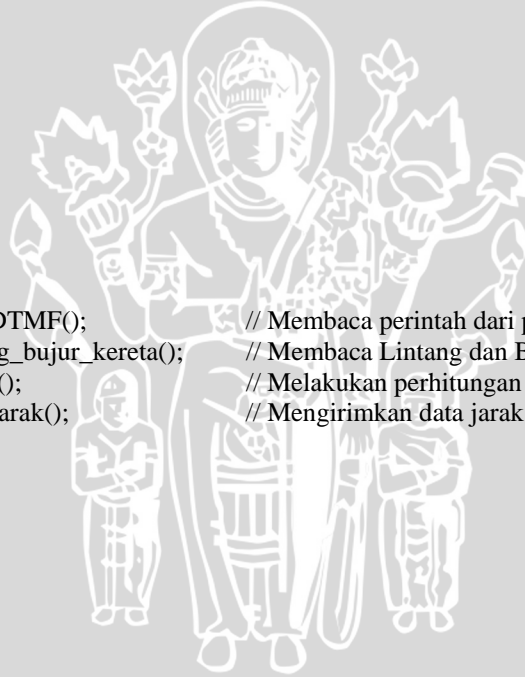
```
void kirim_data_jarak(){
    setbit(0,0); // Aktifkan PTT HT
    delay(2);
    kirim_dtmf("<19<"); // Kirim data DTMF berisi Header
    kirim_char_dtmf(kereta+0x30); // Kirim nada DTMF berisi kereta api yang dimaksud
    kirim_char_dtmf(tanda_); // Kirim nada DTMF tanda sebagai arah kereta
    kirim_dtmf(jarak); // Kirim data DTMF berupa jarak kereta api terhadap stasiun
    kirim_dtmf("<17<"); // Kirim data DTMF berisi Penutup
    clearbit(0,0); // Non aktifkan PTT HT
}
```

/*

Program Utama

*/

```
void main(){
    clearbit(0,0);
    init_serial();
    while(1){
        baca_nada_DTMF(); // Membaca perintah dari pusat informasi
        ambil_lintang_bujur_kereta(); // Membaca Lintang dan Bujur kereta
        hitung_jarak(); // Melakukan perhitungan jarak
        kirim_data_jarak(); // Mengirimkan data jarak ke pusat informasi
    }
}
```



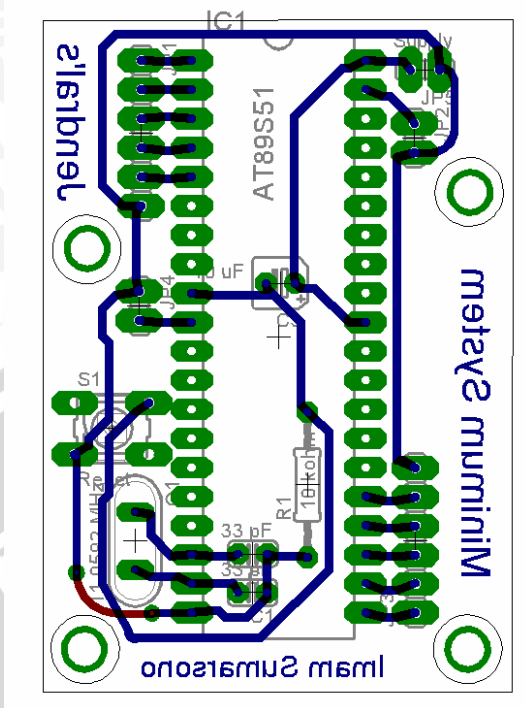
UNIVERSITAS BRAWIJAYA

LAMPIRAN III

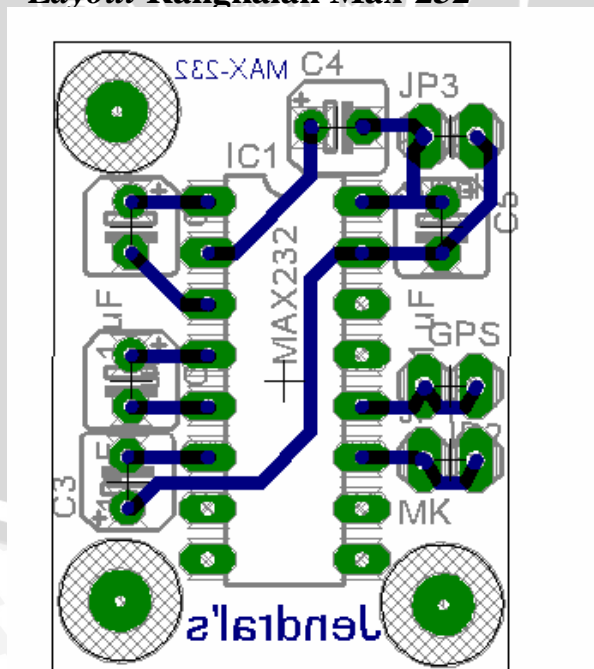
LAYOUT PCB



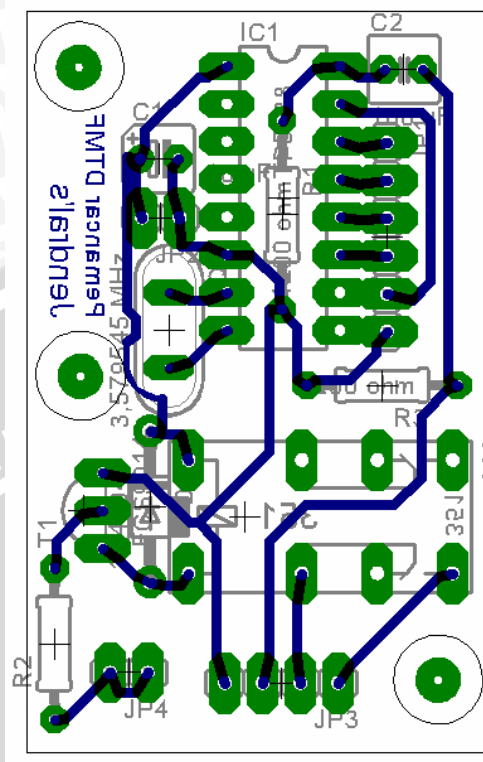
Layout Minimum System Mikrokontroler



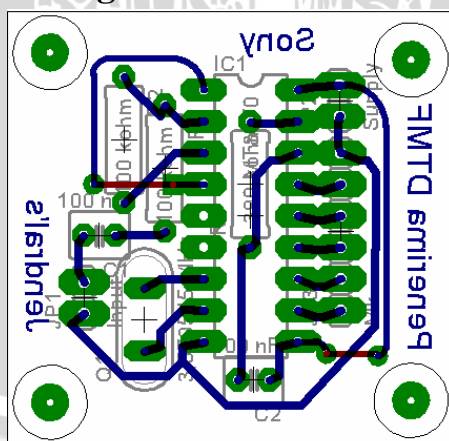
Layout Rangkaian Max-232



Layout Rangkaian Pembangkit DTMF dan Switching PTT



Layout Rangkaian Penerima DTMF



LAMPIRAN IV

FOTO ALAT

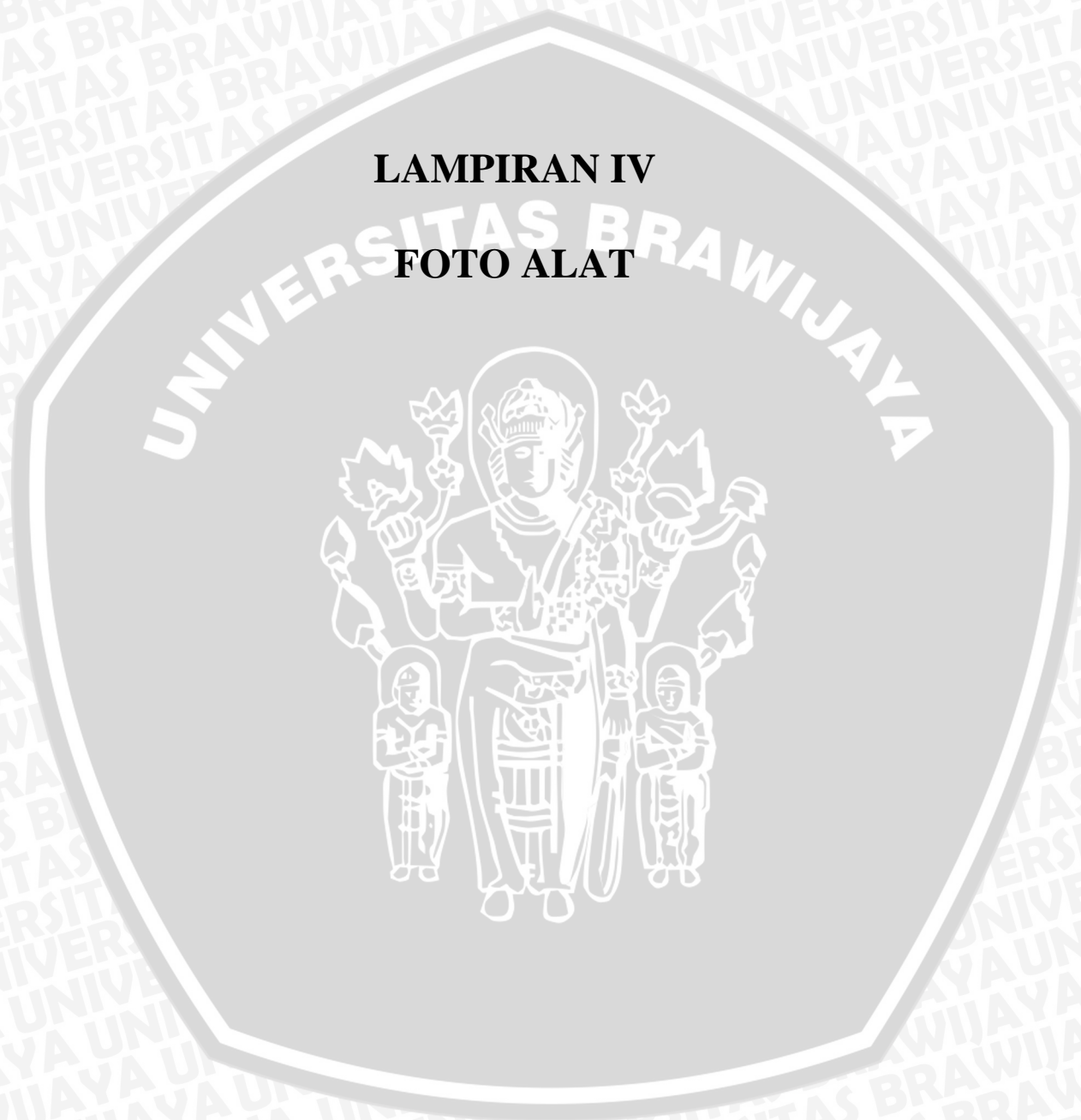


FOTO ALAT TAMPAK LUAR



FOTO ALAT TAMPAK DALAM



LAMPIRAN V
SPEKIFIKASI DAN PETUNJUK
PENGOPERASIAN ALAT



SPEKIFIKASI ALAT

SISTEM PENGOLAH DAN PENTRANSMISI SINYAL GPS PADA SISTEM INFORMASI POSISI KERETA API BERBASIS GPS (GLOBAL POSITIONING SYSTEM)

- Menggunakan GPS *Receiver* jenis ETREX produksi *Garmin Corporation* dengan format keluaran NMEA 0183 *Baudrate* 4800bps.
- Alat dipasang pada Lokomotif kereta Api.
- Data yang diterima dari pusat informasi berupa nada DTMF yang berisi ID kereta dan ID stasiun.
- Data yang dikirimkan ke pusat informasi berupa nada DTMF yang berisi jarak kereta api terhadap stasiun yang diinginkan.
- Menggunakan relay untuk mengaktifkan atau menonaktifkan PTT pada HT(*Handy Talky*).
- Kereta api yang diminta oleh pusat informasi disimulasikan dengan indikator lampu LED.
- Catu daya untuk sistem mikrokontroler dan HT(*Handy Talky*) diperoleh dari aki sebesar 12 V.

PETUNJUK PEMAKAIAN

SISTEM PENGOLAH DAN PENTRANSMISI SINYAL GPS PADA SISTEM INFORMASI POSISI KERETA API BERBASIS GPS (GLOBAL POSITIONING SYSTEM)

- Pasang GPS *Garmin ETREX*, dan HT pada konekor DB-9.
- Menyalakan alat dengan cara memasang catu daya pada tempat yang disediakan dan mengaktifkan tombol power.
- LED valid akan menyala jika data dari GPS *receiver* valid.
- LED tidak valid akan menyala jika data dari GPS *receiver* tidak valid.
- LED kereta merupakan indikator kereta api yang diminta data posisinya oleh pusat informasi.
- LED penerima DTMF akan menyala jika ada nada DTMF yang diterima.
- LED pengirim DTMF akan menyala jika ada nada DTMF yang dikirimkan.
- LED PTT akan menyala jika sistem sedang mengirimkan nada DTMF.
- LED GPS akan berkedip jika ada sinyal yang diterima dari GPS *receiver*.

LAMPIRAN VI
DATASHEET KOMPONEN



PENGANTAR

Dengan sujud syukur ke hadirat Allah SWT yang telah melimpahkan rahmat dan hidayah-Nya kepada Penulis atas terselesaikannya Tugas Akhir ini yang berjudul: “Sistem Pengolah dan Pentransmisi Sinyal GPS pada Sistem Informasi Posisi Kereta Api Berbasis GPS (*Global Positioning System*)”.

Tugas Akhir ini merupakan salah satu syarat menyelesaikan program S-1 (Strata-1) di Jurusan Teknik Elektro Fakultas Teknik Elektro, Universitas Brawijaya Malang.

Seiring selesainya penyusunan skripsi ini, dengan penuh kesungguhan dan kerendahan hati, penulis mengucapkan terima kasih kepada:

1. Bapak Ir. Purwanto, MT selaku Ketua Jurusan Teknik Elektro
2. Bapak Ir. Hery Purnomo selaku Sekretaris Jurusan Teknik Elektro.
3. Bapak Ir. Ponco Siwindarto, MS selaku Ketua Dosen Keahlian Konsentrasi Elektronika.
4. Bapak Ir. Ponco Siwindarto, MS selaku Pembimbing I yang telah memberikan bimbingan serta pengarahan yang berarti selama mengerjakan Tugas Akhir ini.
5. Bapak R. Arif Setiawan, ST, MT selaku Pembimbing II yang telah memberikan bimbingan serta dorongan yang berarti selama mengerjakan Tugas Akhir ini.
6. Seluruh dosen dan karyawan Jurusan Teknik Elektro.
7. Untuk Ibu, Bapak, dan kakakku tercinta atas segala kasih sayang, doa, kesabaran, bantuan dan dorongan baik spiritual maupun material.
8. Untuk mas Mul dan bu Rosa selaku Laboran dan Kepala Laboratorium Elektronika yang telah memberikan kesempatan bagi saya untuk melakukan penelitian di Laboratorium Elektronika.
9. Keluarga besar Laboratorium Elektronika atas seluruh bantuan dan sumbangan pemikiran selama penyelesaian tugas akhir ini.
10. Rekan - rekan Kersen 27 atas bantuan dan dukungannya dalam penyelesaian tugas akhir ini.

11. Semua rekan-rekan dan semua pihak yang tidak dapat disebutkan satu persatu. Semoga Allah Yang Maha Pengasih senantiasa memberikan karunia-Nya atas jasa dan bantuannya yang telah diberikan kepada Penulis

Malang, Februari 2007

Penulis

