SKRIPSI

SISTEM MONITORING POSISI KERETA API BERBASISKAN DATA *GPS* YANG DIKIRIMKAN MELALUI HP (HANDPHONE)

Disusun Oleh :

ATIK RAHMAWATI MUTHOHAROH

NIM : 02.17.120

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

SEPTEMBER 2007

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

LEMBAR PERSETUJUAN

SISTEM MONITORING POSISI KERETA API BERBASISKAN DATA *GPS* YANG DIKIRIMKAN MELALUI HP (HANDPHONE)

SKRIPSI

Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh Gelar Sarjana Teknik Pada Jurusan Teknik Elektro Strata Satu (S-1) Konsentrasi Elektronika

Disusun oleh :

ATIK RAHMAWATI MUTHOHAROH

NIM : 02.17.120

Diperiksa dan Disetujui

Dosen Pembimbing I	Dosen Pembimbing II
Ir. Purwanto, MT	**Sotyohadi, ST, MSc**
NIP.P. 131574847	

Mengetahui

Ketua Jurusan Teknik Elektro S-1

(Ir. F. Yudi Limpraptono, MT)

NIP.Y. 1039500274

KONSENTRASI ELEKTRONIKA
JURUSAN TEKNIK ELEKTRO S-1
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
2007

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

BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI

Nama Mahasiswa : Atik Rahmawati Muthoharoh
NIM : 02.17.120
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Elektronika
Judul Skripsi : **"SISTEM MONITORING POSISI KERETA API BERBASISKAN DATA GPS YANG DIKIRIMKAN MELALUI HP (HANDPHONE)"**

Dipertahankan dihadapan Tim Penguji Skripsi Jenjang Strata Satu (S-1) pada:
Hari : Selasa
Tanggal : 4 September 2007
Dengan Nilai : 73,35 (B+)

Panitia Ujian Skripsi

Ketua
(Ir. Mochtar Asroni, MSME)
NIP. Y. 1018100036

Sekretaris
(Ir.F.Yudi Limpraptono, MT)
NIP. Y. 1039500274

Anggota Penguji

Penguji Pertama
(Ir. Teguh Herbasuki, MT)
NIP. Y. 1038900209

Penguji Kedua
(Ir. Eko Nurcahyo)
NIP. Y. 1028700172

Sistem Monitoring Posisi Kereta Api Berbasiskan Data *GPS* Yang Dikirimkan Melalui HP (Handphone)

Atik Rahmawati Muthoharoh
Institut Teknologi Nasional Malang
Jurusan Teknik Elektro S-1, Konsentrasi Teknik Elektronika
Pembimbing : Ir. Purwanto, MT
Sotyohadi, ST, MSc

ABSTRAKSI

Kata Kunci : *GPS, Mikrokontroler Renesas R8C/Tiny, Handphone, RS 232*

GPS (*Global Positioning System*) adalah sistem navigasi yang berbasiskan satelit. Keberadaan GPS memberikan cara baru dalam menentukan posisi suatu obyek di muka bumi. Salah satu dari aplikasi penggunaan GPS adalah untuk fasilitas *traking* posisi pada kereta api. Fasilitas ini bertujuan untuk meningkatkan mutu pelayanan bagi masyarakat pengguna jasa transportasi kereta api.

Pada tulisan ini dibahas tentang sistem *monitoring* posisi kereta api berbasiskan data GPS yang dikirimkan melalui *handphone*. Pengujian yang dilakukan meliputi : pengujian terhadap fungsi *AT Command*, pengiriman SMS, data In/Out RS 232, dan data GPS. Pengujian dilakukan untuk mengetahui apakah sistem yang dibuat bisa berjalan sesuai dengan perancangan. Hasil menunjukkan bahwa sistem dapat berjalan/berfungsi.

Alhamdulillah.....
Segala Puji Syukur Kehadirat-Mu,
Terima Kasih Yaa Allah, Dzat Yang Maha Agung
Atas Limpahan Rahmat Dan Hidayah-Mu Serta Kekuatan Dari-Mu...
Telah Kucapai Apa Yang Ku Mau
Semua Yang Ku Lakukan, Ku Harap Menjadi Yangterbaik
Kupersembahkan Untuk Orang-Orang Yang Ada Dalam Hidupku...

Segala terima kasihku untuk:

Kedua orang tuaku Bapak Suparni dan Bundaku Kasiyatun tercinta yang telah membesarkanku, mendidikku, menyayangiku, yang selalu menasihati, dan smua yang diberikan agar aku meraih yang terbaik. Maaf, jika yang aku lakukan membekas duka, lara dan air mata. (bapak, ibu' terima kasih.....!!! semoga q bisa jadi anak yang membuat bapak & ibu' bangga). Dan semua keberhasilanku nanti akan aku persembahkan kagem bapak kaliyan ibu'.

Mas eko skluarga (mas, kita bareng-bareng membahagiakan bapak ma ibu' ya.....!!!), buat Mas Arif & Mb' Ut (mas, mb'...adekmu wis ST lho....doain adekmu ya??!!! biar jadi orang sukses, amien!!!), untuk mbak Anik ("jangan anggap keadaan sebagai masalah, tetapi anggaplah keadaan sebagai tantangan". Koyone gampang mbak, tapi kog yo angel yo??? Hehehe...!!!). ponakan-ponakanku, sindu, wisnu, ambar (gak pareng nakal2 nggih???, sekolah yang bener, belajar sing rajin, ngaji-nya juga yang sregep....!!!)

Keluarga besar di Jetis Madiun serta smua sepu2q yang senantiasa mendoakanq dari jauh. Makasih ya....atas doa n support kalian smua.

Bapak Ir. Purwanto, MT & Bapak Sotyohadi, ST, MSc, terima kasih atas bimbingan, doa dan solusi-solusi serta saran-sarannya.

Sobat sekaligus keluargaq Elka-3 '02, terima kasih atas kekompakan, kesolidan dan smua moment indah bareng kalian. "Kenangan itu adalah kado terindah dan abadi dari kalian smua". Makasih sobat,,,Kalian telah menjadikan q bagian dalam hidup kalian. Nama Xan tdk q tulis dsini, namun telah q'ukir dlm sanubariq!!!!!.





Cak Jack "Sang Nyoman Suardana" Nyunk...q gak tau mesti nulis apa??? yang pasti, q gak nyesel udah kenal ma kamu. Makasih ya...atas sayanknya, supportnya, n smuanya buat UnyiL. Klo harus q tulis smua, mungkin akan ngabisin kertas 1 rim. Hehehehehehe.....!!!! Nyunk, jangan pernah lupain UnyiL yach???!!! Ayyo....Kmu jg hrs ttp smangat, sabar, jox sak sik, sak sik!!! n jangan nesuan yo su??!!! *"Senyum, tawa dan air mata pernah terjadi di antara qta, itu smua bukan retorika semata.....namun memang benar adanya".*

Keluargaq selama aq di malang, kalian adalah satu kesatuan yang memberikan arti & kekuatan dalam diriku.....Dulur wedok "jenk city" (jenk, sing sabar yo??? Ttp yakin & percaya smua adalah yang terbaik dari-Nya. Makasih ya ud ikut stress nyari pinjeman GPS. Lemah iku teles, Gusti Allah sing mbales. Sido besanan ta jenk???). Ade'q "ria" (makasih ya atas printernya n atas smua pngertiannya ttg q, ayyo smangat ngrjain TA!!!). Bonek "bolo nekat" Ratih (kpn nyari liek prie lg??? Hehehe...!!!!, makasih ud jd team sukses pncarian "tersangka" mski akhirnya q hrs merelakan smuanya....Hiks,hiks,hiks!!!). Mb'q "Narend" ("andai smua orang punya pkiran yang sm dgn qt, gak akan ada orang yang salah"). Lhie ("menunggu bukan hanya harapan kosong, menunggu adalah kepastian batin untuk mencapai 7an"). Indah ("cuek is the best"). Indrie (jox lali undangane yo jenk???). Mb' iin, Lucy, Koncreng, Eka, Fitri, Mona, Arir, Intan.






Tmen2 SMK GAMALIEL 1 MADIUN, yang selalu kasih smangat n support (G_Penk, bodonk, ngantux, juzto, jago, preman, tessy, eko aw, doyok, kancil, sumi n smua-mua yang belum ksebut, maap yach!!!)

Seluruh Dosen Teknik Elektronika yang telah banyak memberikan ilmunya kepada kami semua...!!! semoga nantinya dapat bermanfaat bagi kami.

Pak Sugeng & Mas Jayeng (makasih ya pak, mas, udah nungguin sampe mo tu2p recordingnya), bu puji serta seluruh staf dan karyawan lainnya. Makasih atas partisipasi yang telah diberikan!!!

ANSA Crew (Mas Irfan, Mas Hendro, Mas Hary, Mas Hadi, Mas Hermawan & yang belum ksebut) terima kasih atas bantuan yang telah diberikan.

BALI Crew (Bli Putu, Grunk, GD, Agus, Kadek), Mas Aris & Jati (melu iki gak pa2 ya, kpn nikah??? Jox lali undangane yo??!!!)

"Masa lalu adalah pengalaman yang harus dikenang, sekarang adalah suatu perjuangan yang harus diraih, dan nanti adalah suatu impian yang harus diwujudkan".

"Motivasi adalah api dari dalam, jika orang lain mencoba menyalakannya untuk anda, kemungkinan apinya hanya menyala sebentar".

KATA PENGANTAR

Alhamdulillah, puji syukur kehadirat-Mu Ya Allah yang telah memberikan Rahmat dan Hidayah-Nya, sehingga penyusun dapat menyelesaikan skripsi yang berjudul "Sistem Monitoring Posisi Kereta Api Berbasiskan Data *GPS* Yang Dikirimkan Melalui HP (*Handphone*)" ini dengan lancar. Skripsi ini merupakan persyaratan kelulusan Studi di Jurusan Teknik Elektro S-1 Konsentrasi Teknik Elektronika ITN Malang dan untuk meraih 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 ITN Malang.
2. Bapak Ir. Mochtar Asroni, MT selaku Dekan FTI.
3. Bapak Ir. F. Yudi Limpraptono, MT selaku Ketua Jurusan Teknik Elektro S-1.
4. Bapak Ir. Purwanto, MT selaku Dosen Pembimbing I.
5. Bapak Sotyohadi, ST, MSc selaku Dosen Pembimbing II.
6. Ayah dan Ibu serta keluarga besar yang telah memberikan do'a, restu, dorongan, semangat dan biaya.
7. Rekan-rekan mahasiswa/i Elektronika 2002 yang telah memberikan semangat dan motivasi dalam penyelesaian skripsi ini.

8. 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 dapat memberikan manfaat bagi perkembangan ilmu pengetahuan dan bagi pembaca.

Malang, September 2007

Penyusun

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

BAB I

PENDAHULUAN

1.1 Latar Belakang

Seringnya terjadi keterlambatan pada pemberangkatan kereta api yang disebabkan adanya gangguan teknis pada manajemen, serta kerusakan pada sistem (kerusakan pada rel, kereta api, mesin lokomotif, listrik) merupakan masalah yang harus segera ditindak lanjuti oleh PT. Kereta Api Indonesia, Persero. Karena selain merugikan negara hal ini juga merugikan masyarakat pada umumnya.

Pada saat ini untuk masalah tersebut diatas diperlukan suatu tindakan guna meningkatkan mutu pelayanan bagi masyarakat pengguna jasa transportasi darat khususnya kereta api yaitu dengan pemasangan fasilitas traking posisi menggunakan *GPS* pada setiap lokomotif kereta api.

Fasilitas traking posisi ini dilengkapi dengan mengetahui posisi kereta api, serta jalur rel yang digunakan dan akan mengirim data secara langsung bila diketahui kereta api mengalami keterlambatan. Hal tersebut dapat meminimalisir kerugian negara dan masyarakat pada umumnya.

Satelit GPS (*Global Positioning System*) pertama kali diluncurkan untuk memenuhi kebutuhan Departemen Pertahanan Amerika Serikat untuk kebutuhan militer, keamanan dan pertahanan negara tersebut. Keberadaan GPS memberikan cara baru dalam menentukan posisi suatu obyek di permukaan bumi yang selama ini memanfaatkan peralatan sederhana semacam kompas, dengan keterbatasan yang ada dan akurasi yang rendah. Selain untuk tujuan pertahanan dan keamanan,

GPS telah dapat dipergunakan oleh kalangan sipil untuk kebutuhannya masing-masing. Salah satu aplikasi GPS di negara-negara maju yaitu pada navigasi perjalanan yang menyediakan informasi peta jalan suatu kota, rute perjalanan yang terbaik/tercepat, informasi posisi kendaraan terhadap lokasi yang dituju atau terhadap titik posisi tertentu, dan berbagai fasilitas lainnya. Sayangnya kemudahan-kemudahan tersebut belum dapat dinikmati di Indonesia karena keterbatasan fasilitas yang tersedia. Padahal dengan memanfaatkan GPS tersebut akan banyak manfaat yang dirasakan, salah satunya adalah memudahkan seseorang dalam mengetahui posisinya setiap saat dan mengetahui letak dan jarak gedung dan fasilitas umum yang terdekat dengan posisinya.

Selama ini *GPS Receiver* yang ada hanya dapat menunjukkan posisi pengguna menurut bujur dan lintang serta ketinggian. Meskipun *GPS Receiver* dapat menyimpan data posisi tertentu tetapi terjadi kesulitan dalam mencari posisi yang akan dituju karena keterbatasan *user interface* pada pesawat *GPS Receiver*. Sehingga dibutuhkan perangkat tambahan untuk memudahkan pengguna dalam menunjukkan posisi yang akan dituju.

Dalam skripsi ini akan direncanakan dan dibuat sistem penelusuran posisi kereta api berbasis GPS (*Global Positioning System*) bagian pengolahan data. Sistem ini dapat menunjukkan dan mendeteksi sejarah posisi (jalan) serta jalur rel yang digunakan pada masing-masing armada kereta api.

1.2 Rumusan Masalah

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

1. Bagaimana melakukan komunikasi dengan lokomotif kereta api melalui SMS untuk memperoleh data posisi dan data kondisi kereta api.
2. Bagaimana merancang *software* untuk mengolah data posisi bujur dan lintang kereta api menjadi data lokasi pada peta yang ada pada komputer.
3. Bagaimana merancang *software* untuk mengetahui kondisi kereta api.

1.3 Batasan Masalah

Dalam perencanaan dan pembuatan alat ini perlu dilakukan pembatasan masalah. Pembatasan masalah yang dibuat dalam skripsi ini antara lain :

1. Penggunaan alat digunakan untuk di luar ruangan (*outdoor*) serta dibatasi hanya untuk wilayah Kampus II ITN Malang sebagai pengujian (eksperimen).
2. *Software* pengolah data lokasi kereta api menggunakan *software* Delphi 7.0.
3. Tingkat akurasi dari lokasi yang ditunjukkan pada peta tergantung pada akurasi GPS yang digunakan.
4. Denah yang dipergunakan hanya untuk pengujian (eksperimen).

5. SMS yang digunakan yang memiliki 160 karakter untuk tiap pengirimannya.

1.4 Tujuan Penulisan

Tujuan penulisan ini adalah merancang sistem yang dapat menunjukkan posisi masing-masing armada kereta api beserta lintasan (rel) yang dilaluinya menurut data bujur dan lintang, mendeteksi waktu serta arah perjalanan kereta api, juga mengirim data pada server bila kereta api diketahui mengalami keterlambatan.

1.5 Sistematika Penulisan

BAB I PENDAHULUAN

Memuat latar belakang, rumusan masalah, ruang lingkup permasalahan, tujuan penulisan serta sistematika penulisan.

BAB II DASAR TEORI

Membahas teori dasar dari komponen serta teori rangkaian yang digunakan dengan mengacu pada perancangan serta pembuatan alat.

BAB III PERANCANGAN DAN PEMBUATAN ALAT

Membahas perancangan alat disertai perhitungan yang sesuai dengan teori yang ada serta membahas pembuatan dan cara kerja dari alat yang dibuat.

BAB IV PENGUJIAN DAN ANALISA ALAT

Membahas pengujian dan analisa alat yang dibuat.

BAB V PENUTUP

Memuat kesimpulan yang diperoleh dari pembuatan dan pengujian alat, serta saran – saran untuk pengembangan lebih lanjut.

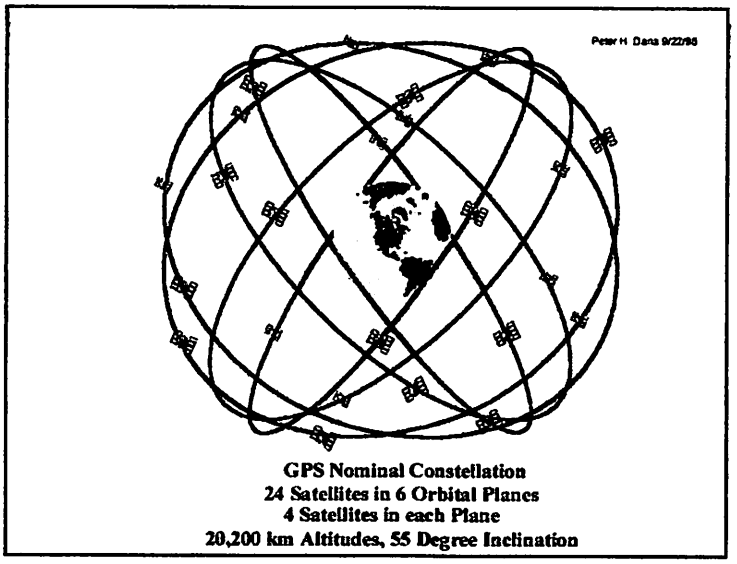
BAB II

DASAR TEORI

2.1 *Global Positioning System* (*GPS*)

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 *(Navigation Satelit and Ranging)* 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 (Johnny Appleseed, 2004 :1)

Gambar jaringan satelit GPS ditunjukkan dalam Gambar 2.1



Gambar 2.1 Jaringan satelit GPS

Sumber : Peter H 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.

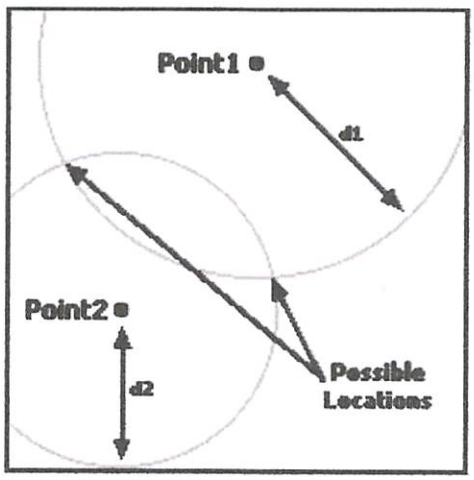
Secara lebih mudah dapat diandaikan setiap satelit mengirimkan informasi berupa "Saya satelit #X, posisi saya Y dan pesan ini dikirim pada waktu Z". GPS *Receiver* akan menyimpan informasi ini dan sekaligus mengeset (membetulkan) waktu pada GPS *Receiver.*

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 (triangulasi) 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). (Johnny Appleseed, 2004 :1)

2.1.2 Penentuan Posisi dengan Triangulasi

GPS *Receiver* dapat membandingkan waktu ketika sinyal dikirimkan oleh satelit tertentu dengan waktu ketika sinyal tersebut diterima oleh GPS *Receiver*. Hal ini dapat digunakan untuk menentukan seberapa jauh satelit tersebut. (Jason, 1999:1).

Gambar penentuan posisi oleh dua satelit ditunjukkan dalam Gambar 2.2.

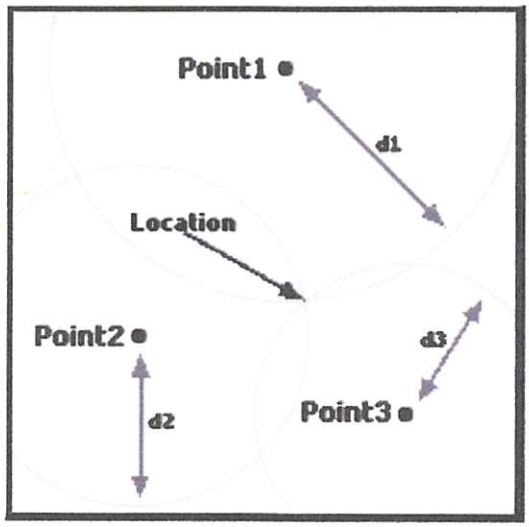


Gambar 2.2 Penentuan Posisi dengan Dua Satelit

Sumber : Jason,1999 : 1

Dalam Gambar 2.2 ditunjukkan bahwa ada dua posisi yang diketahui dari jarak antara satelit 1 dan satelit 2 yaitu posisi 1 dan posisi 2. Dua posisi tersebut adalah posisi dimana GPS *Receiver* mungkin berada. Posisi tersebut ditunjukkan oleh titik pada perpotongan antara lingkaran 1 dan lingkaran 2. Lingkaran pada masing-masing satelit menunjukkan jarak (d1 dan d2) satelit-satelit itu dengan dua lokasi tersebut. Data tersebut adalah data yang diterima oleh GPS *Receiver* dari satelit-satelit yang sedang mengorbit. Untuk memastikan posisi yang benar

diperlukan satelit ketiga. Dengan adanya satelit ketiga maka posisi yang benar dapat ditentukan dari jarak d3. Gambar Penentuan posisi dengan tiga satelit ditunjukkan dalam Gambar 2.3.



Gambar 2.3 Penentuan Posisi dengan Tiga Satelit

Sumber : Jason,1999 : 1

Pada GPS *Receiver* semakin banyak data yang diperoleh dari satelit-satelit tersebut maka akan semakin baik untuk mengurangi ketidaktepatan posisi. (Jason, 1999:1)

2.1.3 Faktor yang Mempengaruhi GPS

Ada beberapa sumber kesalahan yang akan mempengaruhi sinyal GPS secara langsung maupun menyebabkan GPS *Receiver* tidak dapat memberikan hasil yang paling baik. (*Anonymous A*, tanpa tahun : 6)

Faktor tersebut antara lain :

1. Jumlah satelit akan mempengaruhi ketepatan pembacaan posisi GPS *Receiver*. GPS *Receiver* harus mampu melacak empat satelit untuk memperoleh hasil yang baik. Semakin banyak satelit yang digunakan hasil pembacaan akan semakin baik
2. Multipath/pemantulan GPS sinyal di dekat antena seperti fenomena yang hampir sama dengan munculnya bayang-bayang pada TV akibat sinyal yang terpantul oleh suatu permukaan di dekat antena. Hal ini akan menyebabkan kesalahan pada pembacaan posisi dari GPS *Receiver*.
3. Lapisan ionosfer dapat mengubah waktu perjalanan sinyal GPS. Sebelum sinyal GPS mencapai antena GPS *Receiver*, sinyal tersebut melalui daerah yang memiliki muatan yang disebut ionosfer. Daerah ini akan mengubah kecepatan transmisi sinyal GPS yang akan menyebabkan kesalahan ketepatan pembacaan posisi dari GPS *Receiver*. Akan tetapi hal ini tidak akan menjadi masalah jika GPS *Receiver* yang dipergunakan tingkat ketepatannya rendah.
4. Lapisan troposfer dapat mengubah waktu perjalanan sinyal GPS. Troposfer adalah daerah cuaca pada atmosfer bumi. Tetesan hujan dan uap air dapat mempengaruhi kecepatan sinyal GPS.
5. Geometri satelit. Penyebaran satelit di luar angkasa mempengaruhi perhitungan posisi GPS *Receiver*. Hal ini disebut sebagai *Position Dilution of Precission* (PDOP). PDOP disimbolkan dengan angka

dimana semakin kecil angkanya maka akan didapatkan hasil yang semakin bagus. Hasil terbaik dihasilkan ketika PDOP kurang dari tujuh. Ketika satelit menyebar maka PDOP akan rendah dan ketika satelit mengumpul maka PDOP akan tinggi. PDOP ditentukan oleh lokasi geografis, lokasi, waktu dan halangan yang mungkin menutupi satelit.

6. Meskipun satelit adalah sistem yang kuat dan mandiri akan tertapi terkadang satelit tersebut "tidak sehat". Masing-masing satelit mengirimkan status kesehatannya berdasarkan informasi dari Departemen Pertahanan Amerika serikat. GPS *Receiever* telah memiliki pelindung untuk tidak menggunakan data dari satelit yang "tidak sehat".
7. Kekuatan sinyal tergantung pada halangan dan ketinggian satelit di atas horizon. Halangan antara antena GPS dengan satelit harus dihindari dan juga satelit yang dekat dengan horizon karena sinyalnya lemah.
8. Interferensi dari Radio Frekuensi (RF) terkadang menjadi masalah pada penerimaan sinyal GPS. Beberapa sumber interfrensi RF yaitu :
    a. Menara Radio
    b. Pemancar
    c. Piringan satelit
    d. Generator

2.1.4 Format Data Lintang dan Bujur pada GPS *Receiver*

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* pada kereta api 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 pada 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, ddmm.mmmm format
<4>	Latitude Hemisphere, N or S
<5>	Longitude, dddmm.mmmm format
<6>	Logitude 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.mmmm = 2 digit derajat, 2 digit menit dan 4 digit menit dibelakang koma

Sumber : *anonymous*, 2002A : 15

Contoh data yang diperoleh dari GPS *Receiver* adalah

$GPRMC,060149,A,0756.5394,S,11237.0438,E,000.0,000.0,060604,001.4,E*66

data tersebut berarti :

060149 menunjukan 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

060604 menunjukkan tanggal data tersebut yaitu tanggal 06 Juni 2004

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.

2.2 Unit Komunikasi Antar *GPS*, Mikrokontroler dan *HP*

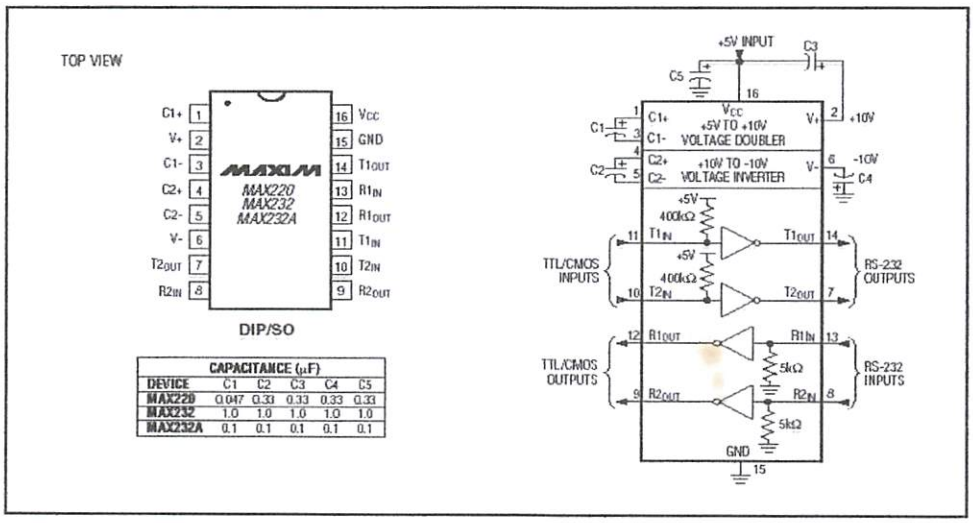
2.2.1 Komunikasi Serial RS 232

RS 232 merupakan salah satu jenis interface dalam proses transfer data antar komputer dalam bentuk serial transfer. RS 232 merupakan kependekan dari **Recommended Standart Number 232**. RS 232 dibuat untuk interface antara peralatan terminal data dengan peralatan komunikasi data, dengan menggunakan data biner serial sebagai data yang ditransmisikan. Serial interface RS 232 memberi ketentuan logic level sebagai berikut :

- Logic 1 disebut " mark " terletak antara -3 Volt hingga -15 Volt.
- Logic 0 disebut " space " terletak antara +3 Volt hingga +15 Volt.

Daerah tegangan antara -3 Volt hingga +3 Volt adalah invaled level, yaitu daerah yang tidak memiliki logic sehingga daerah tersebut harus dihindari. Suatu saluran data RS 232 yang memberi keadaan ini berarti ada kesalahan. Demikian pula saluran daerah lebih negatif dari -15 volt dan daerah lebih positif dari +15 Volt. Pada saat pengiriman data, sebelum mengirim data bit 0 harus diawali dengan start bit terlebih dahulu kemudian baru mengirim bit 0. Setelah mengirim bit 7 masih harus diakhiri dengan stop bit.

Gambar IC MAX 232 dan rangkaian minimum sistemnya ditunjukkan dalam Gambar 2.4



Gambar 2.4 IC MAX232 dan Rangkaian Minimum Sistemnya

Sumber : *anonymous*, 2004 : 17

IC MAX 232 memiliki empat bagian yaitu *dual charge pump*, konverter tegangan, RS 232 *driver* dan RS 232 *receiver*. *Dual charge pump* mengubah tegangan masukan +5V menjadi ±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-.

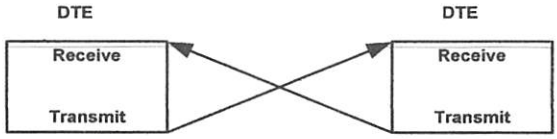
Keluaran dari RS 232 *driver* berayun dari ±8V ketika dibebani dengan 5kΩ (nominal) dengan Vcc sebesar 5V. *Pull up* resistor yang tersambung dengan Vcc menyebabkan keluaran dari *driver* yang tidak digunakan pada kondisi rendah karena semua *driver* adalah terbalik.

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.

2.2.1.1 Dasar-Dasar Serial Interface

Dasar-dasar serial interface telah diuraikan mengenai fungsi dari interface RS 232C. Dalam pembahasan berikut ini, akan dijelaskan bagaimana dasar-dasar interface tersebut. Proses transfer secara serial menggunakan alat RS 232C yang dibuat oleh *Electronic Industry Assosiation* (EIA) antara terminalnya, biasanya menggunakan DTE (*Data Terminal Equipment*) untuk masing-masing terminal.

Data yang ditransfer dari suatu terminal akan diterima oleh terminal lainnya, dan demikian pula sebaliknya melalui seperangkat peralatan di atas. Gambar di bawah menjelaskan konsep transfer antara DTE dengan DTE. Jenis data yang akan ditransfer adalah dalam bentuk biner (bit per bit transfer) dengan satuan baut untuk kecepatan transfernya (bit per detik). Dalam proses transfer ini harus terdapat suatu peralatan yang berfungsi sebagai *hand shake* (jabat tangan) yaitu sebagai pemantau status yang diterima atau ada untuk memberikan respon yng sesuai. Dalam merancang perangkat lunak komunikasi serial, hand shake disempurnakan dengan menambahkan karakter pengendali dalam deretan atau jumlah bit yang ditransfer, biasanya disebut start bit dan stop bit.



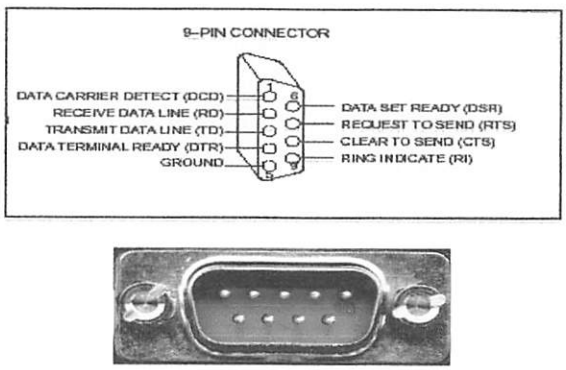
Gambar 2.5 Blok Diagram Transfer Data DTE dengan DTE

Konsep interface antara DTE dengan DTE yang dilakukan berulang-ulang sampai semua data selesai ditransfer, adalah sebagai berikut :

- Pengiriman data dilakukan melalui TxD.
- Penerimaan data dilakukan melalui RxD.

2.2.1.2 Pin -Pin pada EIA RS 232 dan Kegunaannya

Di dalam komputer terdapat fasilitas komunikasi serial yang menggunakan standar RS-232, yaitu terletak pada COM1 dan COM2. Kedua fasilitas ini menggunakan konektor DB9 atau DB25 sebagai penghubung dengan piranti luar. Gambar konektor DB9 seperti terdapat pada Gambar 2.6



Gambar 2.6 Konfigurasi Pin Konektor DB9

Sumber: Dallas Semiconductor,1998:3-9

Fungsi masing-masing pin seperti terdapat dalam Tabel 2.2

Tabel 2.2 Fungsi Pin RS-232 Dalam DB9

Pin	Nama	Fungsi
1	DCD (*Data Carrier Detect*)	Mendeteksi sinyal *carrier* dari modem lain
2	RD (*Receive Data Line*)/ (R_XD)	Pengiriman data serial dari DCE ke DTE
3	TD (*Transmit Data Line*)/(T_XD)	Pengiriman data serial dari DTE ke DCE
4	DTR (*Data Terminal Ready*)	Memberitahu DCE bahwa DTE telah aktif dan siap untuk bekerja
5	*Ground*	Referensi semua tegangan antarmuka
6	DSR (*Data Set Ready*)	Memberitahu DTE bahwa DCE telah aktif dan siap untuk bekerja
7	RTS (*Request To Send*)	Memberitahu DCE bahwa DTE akan mengirim data
8	CTS (*Clear To Send*)	Memberitahu DTE bahwa DCE siap menerima data
9	RI (*Ring Indikator*)	Aktif jika modem menerima sinyal ring pada jalur telepon

Sumber : Ganiadi Gunawan, 1991: 4.

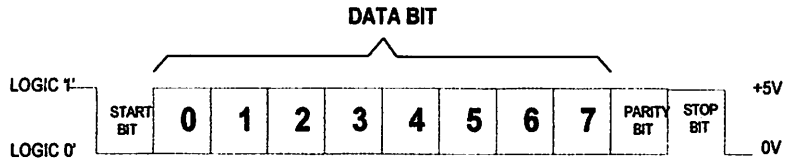
Spesifikasi RS-232 dapat dilihat dalam Tabel 2.3

Tabel 2.3 Spesifikasi RS-232

Keistimewaan	Karakteristik
Jenis operasi	*Single ended* (tak seimbang)
Jenis penggerak dan Penerima per jalur	1 *driver* 1 *receiver*
Data *rate* maksimum	20 kbps
Panjang saluran maksimum	50 ft (15 m)
Tegangan keluaran penggerak	±5 - ±15 volt
Sensitivitas penerima	±3 volt

Sumber : ARC Electronics, 2000

2.2.2 Protokol Komunikasi Pada RS 232



Gambar 2.7 Bentuk Protokol Interface RS232

Sumber : Craig Peacock, June:2005

Beberapa Protocol dalam interface RS 232 adalah:

- Start Bit

  Merupakan sebuah bit dengan logika '0', dimana bit ini yang menandakan bahwa akan ada karakter atau data yang mengikutinya. Bit ini langsung diberikan tanpa harus mensetnya terlebih dahulu.

- Data Bit

  Merupakan bit yang mewakili dari karakter yang diikutinya. Data bit ini dapat diset antara 5 sampai 8 bit.

- Parity Bit

  Merupakan bit yang digunakan sebagai error checking pada receiver, apabila terjadi kesalahan maka receiver akan menset error flag (*parity error*) pada spesial register. Parity bit ini menghitung jumlah data yang berlogika '1' pada data bit. Perhitungan jumlah data bit tersebut tergantung dari jenis parityyang diset. Untuk parity '*event*' jumlah data bit yang berlogika '1' ditambah dengan parity bit akan menghasilkan

jumlah yang ganjil. Sedangkan untuk parity '*mark*' merupakan parity bit yang selalu berlogika '1' demikian pula pada space parity bit selalu berlogika '0' dan parity '*none*' merupakan parity bit yang diabaikan.

- Stop Bit

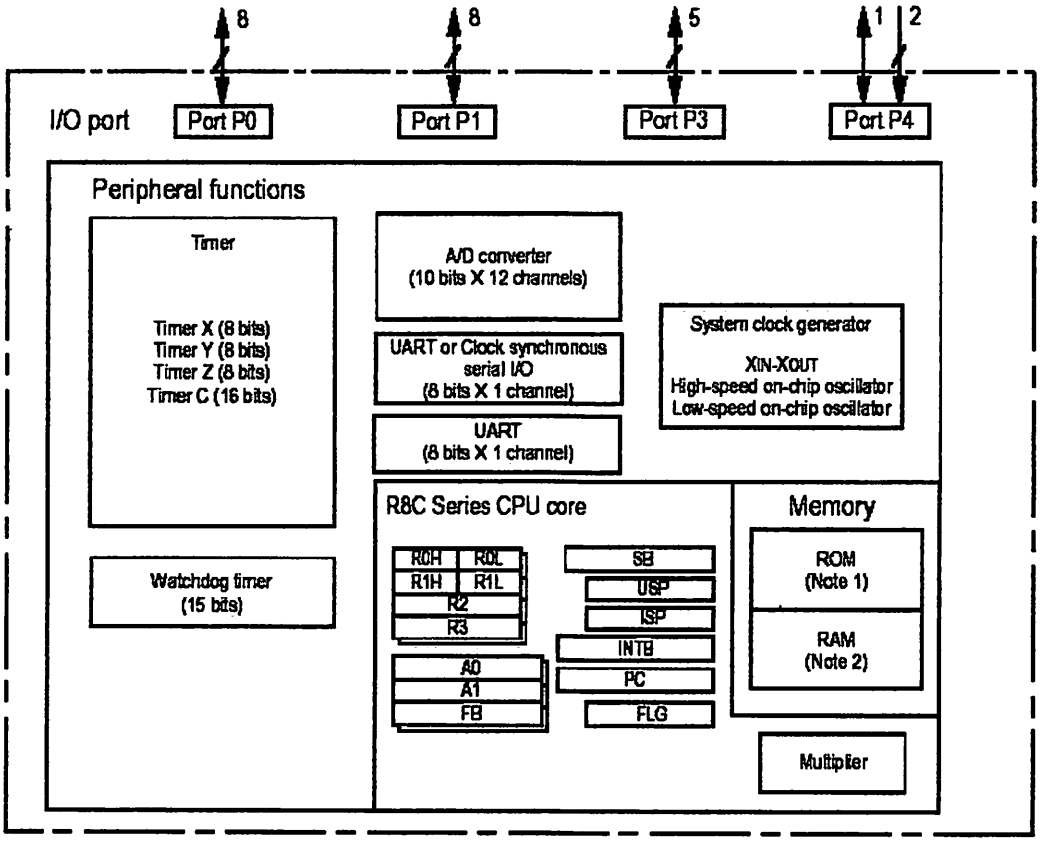
  Merupakan bit yang menandakan akhir dari suatu paket data (biasanya 1 byte data). Seperti pada start bit, bit langsung diberikan pada serial device. Stop bit ini dapat diset panjangnya menjadi satu bit, satu setengah atau dua bit.

- Baud Rate

  Sebenarnya baud rate berarti pergantian kondisi tiap detik, tetapi karena hanya ada dua kondisi pada komunikasi serial (yaitu logic '1' dan '0') maka dapat juga digunakan untuk menunjukkan kecepatan dari transmisi bit (bit persecond).

2.3 Mikrokontroler RENESAS R8C / Tiny

Mikrokontroler RENESAS dibangun menggunakan proses gerbang silicon CMOS dengan kemampuan tinggi menggunakan CPU seri R8C/Tiny dan dikemas dalam modul plastic dengan jumlah pin sebanyak 32. Mikrokontroler ini beroperasi menggunakan perintah canggih khususnya efisiensi perintah dengan level tinggi. Mikrokontroler ini mempunyai 1 Mbytes kapasitas alamat, yang bisa digunakan untuk mengeksekusi perintah dengan kecepatan tinggi. Data flash ROM sebesar 2 KB x 2 blocks.



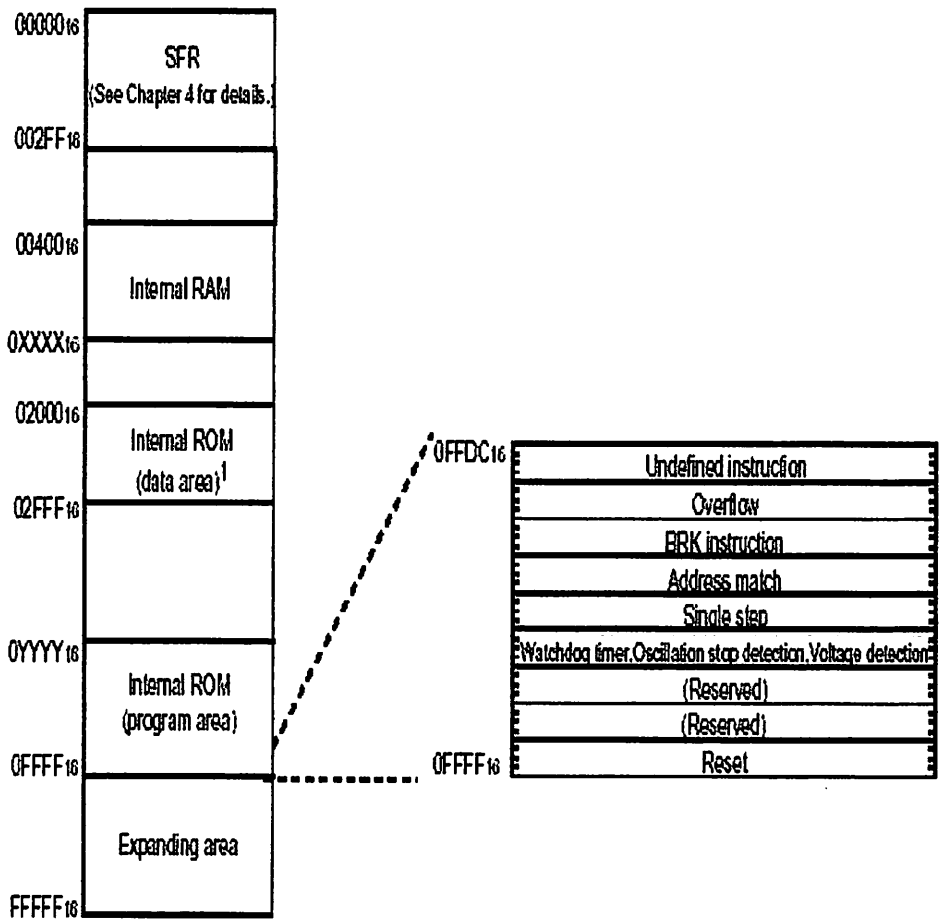
Gambar 2.8 Blok Diagram MCU Renesas

(Sumber : Datasheet Renesas)

Mikrokontroler Renesas R8C/Tiny mempunyai struktur memori yang terdiri atas

- Space alamat hingga 1 Mbytes dari alamat 00000_{16} sampai $FFFFF_{16}$.
- ROM internal (*program area*) dialokasikan pada alamat terendah dimulai dari alamat $0FFFF_{16}$. Misalnya, 16 Kbyte ROM internal dialokasikan pada alamat yang dimulai dari $0C000_{16}$ sampai $0FFFF_{16}$.
- ROM internal untuk *data area* dialokasikan pada alamat 02000_{16} sampai $02FFF_{16}$.

- Sedangkan RAM internal dialokasikan pada arah alamat yang lebih tinggi dimulai dari alamat 00400_{16}.
- *Special Function Register* (SFR) dialokasikan pada alamat mulai dari 00000_{16} sampai $002FF_{16}$. Fungsi register control peripheral digunakan untuk fungsi tertentu yang dispesifikasikan oleh pembuat IC. Misalnya, untuk mengaktifkan mode-mode dan untuk komunikasi serial baud rate diinisialisasi pada alamat.



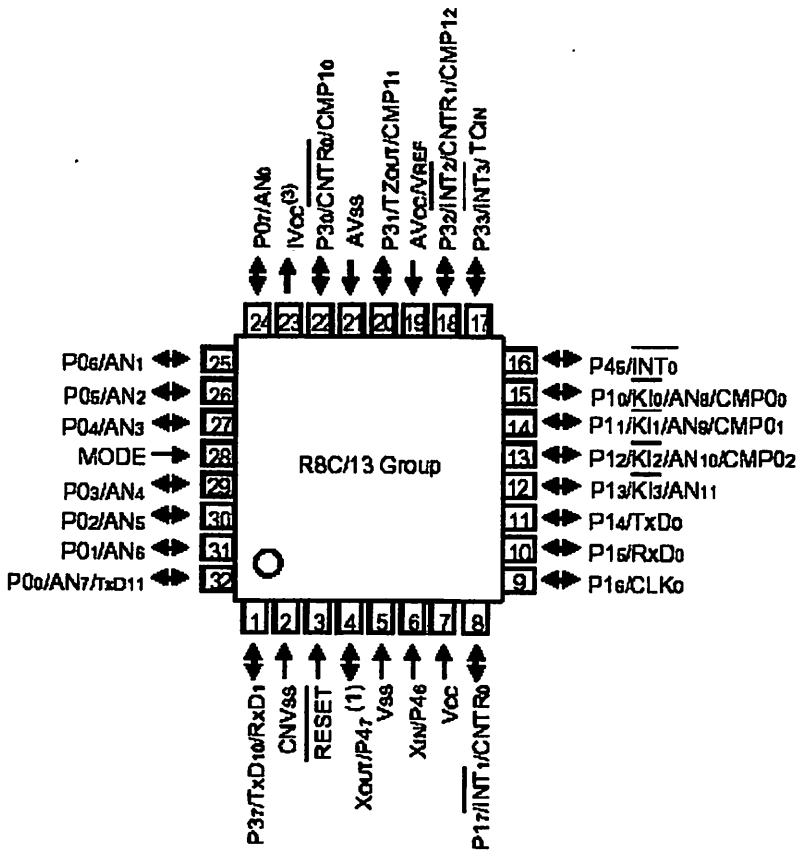
NOTES:

1. The data flash ROM block A (2K bytes) and block B (2K bytes) are shown.
2. Blank spaces are reserved. No access is allowed.

Type name	Internal ROM		Internal RAM	
	Size	Address 0YYYY₁₆	Size	Address 0XXXX₁₆
R5F21134FP, R5F21134DFP	16K bytes	0C000₁₆	1K bytes	007FF₁₆
R5F21133FP, R5F21133DFP	12K bytes	0D000₁₆	768 bytes	006FF₁₆
R5F21132FP, R5F21132DFP	8K bytes	0E000₁₆	512 bytes	005FF₁₆

Gambar 2.9 Memory Map

(Sumber : Datasheet Renesas)



Gambar 2.10 Konfigurasi Pin R8C/Tiny

(Sumber : Datasheet Renesas)

Keterangan fungsi masing-masing pin :

1. V_{CC}

   Digunakan untuk sumber tegangan dengan range nilai antara 2,7 – 5,5 Volt.

2. V_{SS}

   Range tegangan 0 Volt

3. IV_{CC}

   Pin ini digunakan untuk menyetabilkan sumber tegangan internal. Pin ini dihubungkan ke V_{SS} melalui kapasitor 0,1 μF.

4. AV_{CC}, AV_{SS}

   Pin ini merupakan input power supply untuk A/D Converter. Pin ini dihubungkan ke pin V_{CC}, sedangkan AV_{SS} dihubungkan ke V_{SS}. Hubungkan pin AV_{SS} dan AV_{CC} dengan kapasitor.

5. Reset

   Merupakan input reset pada MCU.

6. CNV_{SS}

   Pin ini dihubungkan ke V_{SS} melalui resistor.

7. Mode

   Pin ini dihubungkan ke V_{CC} melalui resistor.

8. X_{IN}, X_{OUT}

   Pin ini disediakan untuk pembangkitan rangkaian I/O pada clock utama. Hubungkan resonator keramik atau osilator kristal antara X_{IN} dan X_{OUT}. Untuk menggunakan clock derived external, masukkan ke pin X_{IN} dan pin X_{OUT} dibiarkan terbuka.

9. $INT_0 - INT_3$

   Merupakan pin input interupt

10. $KI_0 - KI_3$

    Merupakan pin Key Input interrupt.

11. $CNTR_0$ (I/O)

    Merupakan timer pin X I/O

12. $CNTR_0$ (O)

    Merupakan timer pin X output.

13. $CNTR_1$

    Merupakan timer pin Y I/O

14. TZ_{OUT}

    Merupakan timer pin Z output.

15. TC_{IN}

    Merupakan timer pin C input.

16. $CMPO_0 - CMPO_3$ dan $CMPO_{10} - CMPO_{13}$

    Merupakan timer pin C output.

17. CLK_0

    Merupakan transfer clock untuk pin I/O.

18. RxD_0 dan RxD_1

    Pin serial data input.

19. TxD_0, TxD_{10} dan TxD_{11}

    Pin serial data output.

20. V_{REF}

    Referensi pin input tegangan untuk A/D Converter. Hubungkan V_{REF} ke V_{CC}.

21. $AN_0 - AN_{11}$

    Pin input analog untuk A/D Converter.

22. $P0_0 - P0_1$, $P1_0 - P1_7$, $P3_0 - P3_3$, $P3_7$ dan $P4_5$

    Merupakan port 8 bit CMOS I/O. $P1_0 - P1_7$ juga berfungsi sebagai port LED driver.

2.4 Telepon Selular

Telepon selular (*ponsel*) atau yang lebih dikenal dengan HP (*Handphone*) merupakan jenis perangkat telepon bergerak yang memudahkan seseorang untuk berkomunikasi dimanapun dan dalam kondisi apapun asalkan pengguna *handphone* berada dalam *coverage area* dari suatu jaringan *provider* dari *SIM card* yang digunakan. *Handphone* merupakan suatu perangkat yang dapat digunakan untuk mengirim dan menerima data suara. Seiring dengan perkembangan teknologi dalam dunia *mobile system*, *handphone* tidak hanya dapat digunakan untuk mengirimkan data berupa suara namun juga dapat digunakan untuk mengirimkan data berupa data *character* atau yang kini populer dengan sebutan SMS (*Short Message Service*) .

Disamping itu, pesatnya teknologi telekomunikasi menyebabkan *handphone* tidak hanya dapat digunakan untuk berkomunikasi antara sesama *handphone*, tetapi *handphone* juga dapat digunakan untuk berkomunikasi dengan PC (*Personal Computer*) maupun mikrokontroler sehingga pengiriman data dari HP ke PC atau dari HP ke mikrokontroler dan sebaliknya dapat terjadi.

2.4.1 Siemens C35

Siemens merupakan salah satu merek ponsel yang dapat berkomunikasi dengan mikrokontroler melalui port serial, sehingga suatu pentransferan data dapat terjadi antara mikrokontroler dengan ponsel merk Siemens tersebut. Melalui pentransferan data ini, penggguna ponsel dapat mengirim atau menerima suatu pesan singkat (SMS). Untuk dapat berkomunikasi dengan mikrokontroler,

siemens dilengkapi dengan *internal* modem yang dapat mengenali *AT Command* yang merupakan standar bentuk perintah pada telepon selular.

2.5 Format Data SMS

SMS (*Short Message Service*) merupakan sebuah mekanisme pengiriman pesan singkat melalui jaringan bergerak (*mobile network*). Panjang maksimum dari sebuah pesan singkat adalah 160 karakter, fasilitas ini disediakan oleh jaringan telepon seluler. Sebenarnya panjang pesan maksimum yang dapat dikirimkan melalui SMS adalah 140 karakter. Teknik ini bertumpu pada keadaan bahwa kode karakter ASCII alfanumerik yang mempunyai lebar data 7 bit (bit ke-7 selalu bernilai 0 sehingga bisa diabaikan). Teknik kompresi *septet to oktet* dilakukan dengan menyisipkan bit-bit LSB karakter selanjutnya ke dalam bit-bit MSB dari data sebelumnya secara berkesinambungan.

Ada dua cara pengiriman dan penerimaan SMS, yaitu dengan menggunakan mode teks dan mode PDU (*Protocol Data Unit*). Mode teks adalah isi sms berupa kata-kata yang bisa dibaca yaitu yang mempunyai lebar data 8 bit per huruf. Mode teks tidak terdapat pada beberapa telepon seluler, maka dalam tugas akhir ini digunakan pesan SMS dengan mode PDU dan tidak membahas mode yang lain. Mode PDU terdiri dari 7 bit berisi bilangan-bilangan heksadesimal yang terdiri atas: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F. pada mode PDU data akan digeser-geser sampai menjadi 8 bit.

2.5.1 Prinsip Kerja SMS (*Short Message Service*) Pada *Handphone*

Di balik tampilan menu *Messages* pada sebuah *handphone* sebenarnya adalah *AT Command* 2x yang bertugas mengirim/menerima data ke/dari SMS-*center*. *AT command* tiap-tiap SMS *device* bisa berbeda-beda, tetapi pada dasarnya sama berdasarkan fungsinya. Untuk Laporan akhir ini digunakan *handphone* Siemens C35 sebagai SMS *device*-nya dan *SIM Card* yang digunakan akan ditentukan setelah diadakan pengujian sistem secara keseluruhan.

Data yang mengalir ke atau dari SMS *centre* harus berbentuk PDU (*Protocol Data Unit*). PDU berisi bilangan-bilangan *heksadesimal* yang mencerminkan bahasa *I/O*. PDU terdiri dari beberapa *header*. *Header* untuk kirim SMS ke SMS-*centre* berbeda dengan SMS yang diterima dari SMS *centre*. (*Sumber : Khang,Bustam Ir, "Trik Pemrograman Aplikasi Berbasis SMS"*)

2.5.1.1 PDU untuk Kirim SMS ke *SMS-Centre*

PDU untuk mengirim SMS terdiri atas delapan *header*, yaitu sebagai berikut: (*Sumber: Khang,Bustam Ir, "Trik Pemrograman Aplikasi Berbasis SMS"*)

1. **Nomor SMS-*Centre***

   Header pertama ini terbagi atas tiga *subheader*, yaitu :

   a) Jumlah pasangan Heksadesimal SMS-*Centre* dalam bilangan heksa.

   b) *National/International Code*.

      - Untuk *National*, kode *subheader*-nya yaitu 81
      - Untuk *International*, kode *subheader*-nya yaitu 91

c) No SMS-*Centre*-nya sendiri, dalam pasangan heksa dibalik-balik. Jika tertinggal satu angka heksa yang tidak memiliki pasangan, angka tersebut akan dipasangkan dengan huruf 'F' didepannya.

Contoh: untuk nomor SMS-*Centre* Indosat-M3 dapat ditulis dengan dua cara sebagai berikut :

Cara 1:

0855000000 diubah menjadi:

a. 06 artinya total pasangan termasuk kode nasional ada 6 pasang

b. 81 terdiri 1 pasang
c. 80-55-00-00-00 terdiri 5 pasang } Total 6 pasang

Digabung menjadi: **06818055000000**

Cara 2:

62855000000 diubah menjadi:

a. 07 artinya total pasangan termasuk kode internasional ada 7 pasang

b. 91 terdiri 1 pasang
c. 26-58-05-00-00-F0 terdiri 6 pasang } Total 7 pasang

Digabung menjadi: **07912658050000F0**

Berikut ini beberapa nomor SMS-Center operator SMS di Indonesia. Ada dua cara dalam mengkodekan nomor SMS *Centre* ke bentuk kode PDU, yaitu:

Cara 1:

No	Operator SMS	Nomor SMS *Centre*	Kode PDU
1.	Telkomsel	0811000000	06818011000000
2.	Satelindo	0816125	0581806121F5
3.	Excelcomindo	0818445009	06818081440590
4.	Indosat-M3	0855000000	06818055000000
5.	PT.Telkom	0809800000	06818090080000

Cara 2 :

No	Operator SMS	Nomor SMS *Centre*	Kode PDU
1.	Telkomsel	62811000000	07912618010000F0
2.	Satelindo	62816125	059126181652
3.	Excelcomindo	62818445009	07912618485400F9
4.	Indosat-M3	62855000000	07912658050000F0
5.	PT.Telkom	62809800000	07912608890000F0

Tabel 2.4 Pengkodean Nomor SMS *Centre* ke Bentuk Kode PDU

(Sumber : Khang,Bustam Ir,"Trik Pemrograman Aplikasi Berbasis SMS")

2. Tipe SMS

Untuk *SEND* tipe SMS adalah 1. Jadi bilangan heksanya adalah 01.

3. Nomor Referensi SMS

Nomor referensi ini dibiarkan dahulu 0, jadi bilangan heksanya adalah 00. Nanti akan diberikan sebuah nomor referensi otomatis oleh *handphone*.

4. Nomor *Handphone* Penerima

Sama seperti cara menulis PDU *Header* untuk SMS-*Centre*, *header* ini juga terbagi atas tiga bagian, sebagai berikut :

- Jumlah bilangan desimal nomor *handphone* yang dituju dalam bilangan heksa.
- *National/international Code.*
  - Untuk *national*, kode *subheader*-nya : 81
  - Untuk *international*, kode *subheader*-nya : 91
- Nomor *handphone* yang dituju, dalam pasangan heksa dibalik-balik. Jika tertinggal satu angka heksa yang tidak memiliki pasangan, angka tersebut dipasangkan dengan huruf 'F' di depannya.

Contoh :

Untuk nomor *handphone* yang dituju = 628129573337 dapat ditulis dengan dua cara sebagai berikut :

Cara 1 : **08129573337** diubah menjadi :

a. 0B ⟶ ada 11 angka

b. 81

c. 80-21-59-37-33-F7

Digabung menjadi : **0B818021593733F7**

Cara 2 : **628129573337** diubah menjadi :

a. 0C ⟶ ada 12 angka

b. 91

c. 26-18-92-75-33-73

Digabung menjadi : **0C91261892753373**

5. Bentuk SMS, antara lain :

0 → 00 → dikirim sebagai SMS

1 → 01 → dikirim sebagai telex

2 → 02 → dikirim sebagai fax

Dalam hal ini, untuk mengirim dalam bentuk SMS tentu saja kita memakai **00**

6. Skema *Encoding* Data *I/O*

Ada dua skema, yaitu :

a. Skema 7 bit → ditandai dengan angka 0 → 00

b. Skema 8 bit → ditandai dengan angka lebih besar dari 0, diubah ke heksa

Kebanyakan *handphone*/SMS *gateway* yang ada di pasaran sekarang menggunakan skema 7 bit sehingga menggunakan kode **00.**

7. Jangka Waktu Sebelum SMS *Expired*

Jika bagian ini di-*skip*, itu berarti tidak dibatasi waktu berlakunya SMS. Sedangkan jika diisi dengan suatu bilangan *integer* yang kemudian diubah ke pasangan heksa tertentu, bilangan yang di berikan tersebut akan mewakili jumlah waktu *validitas* SMS tersebut.

Rumus untuk menghitung jangka waktu *validitas* SMS adalah sebagai berikut :

Tabel 2.5 Jangka Waktu *Validitas SMS*

Integer (INT)	Jangka Waktu Validitas SMS
0 – 143	(INT+1) x 5 Menit (Berarti : 5 menit s/d 12 jam)
144 – 167	12 jam + ((INT – 143) x 30)
168 – 196	(INT – 166) x 1 Hari
197 - 255	(INT – 192) x 1 Minggu

(Sumber : Khang,Bustam Ir,"Trik Pemrograman Aplikasi Berbasis SMS")

Agar SMS pasti terkirim ke *handphone* penerima, sebaiknya tidak diberikan batasan waktu *valid*-nya.

8. Isi SMS

Header ini terdiri atas dua *subheader*, yaitu :

a. Panjang isi (jumlah huruf dari isi)

Misalnya : untuk kata "*hello*" → ada 5 huruf → 05

b. Isi berupa pasangan bilangan heksa

Untuk *handphone*/SMS *gateway* berskema *encoding* 7 bit, jika diketikkan suatu huruf dari *keypad*-nya, berarti telah dibuat 7 angka I/O berturutan. Skema 7 bit tersebut diperlihatkan pada Tabel 2.8.

Ada dua langkah yang harus kita lakukan untuk mengkonversikan isi SMS, yaitu:

1. Mengubahnya menjadi kode 7 bit.
2. Mengubah kode 7 bit menjadi 8 bit, yang diwakili oleh pasangan heksa.

Contoh: untuk kata "hello"

Langkah Pertama:

	Bit ke						
Isi	**7**	**6**	**5**	**4**	**3**	**2**	**1**
h	1	1	0	1	0	0	0
e	1	1	0	0	1	0	1
l	1	1	0	1	1	0	0
l	1	1	0	1	1	0	0
o	1	1	0	1	1	1	1

Langkah Kedua:

Mengubah kode 7 bit menjadi 8 bit, yaitu oleh karena total 7 bit x 5 huruf = 35 bit, sedangkan yang diperlukan adalah 8 bit x 5 huruf = 40 bit, maka diperlukan 5 bit *dummy* yang diisi dengan bilangan 0, ditambahkan pada MSB huruf terakhir. Setiap 8 bit mewakili suatu pasangan heksa. Tiap 4 bit mewakili suatu angka heksa, tentu saja karena secara logika $2^4 = 16$

Susunan menjadi :

$$\underbrace{00000}_{\text{Bit dummy}}\ \underbrace{1101111}_{o}\ \underbrace{1101100}_{l}\ \underbrace{1101100}_{l}\ \underbrace{1100101}_{e}\ \underbrace{1101000}_{h}$$

Susunan 4 bit dimulai dari MSB menjadi :

$$\underbrace{0000\ 0110}_{o}\ \underbrace{1111\ 1101}_{l}\ \underbrace{1001\ 1011}_{l}\ \underbrace{0011\ 0010}_{e}\ \underbrace{1110\ 1000}_{h}$$

	Bit ke							
isi	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**
h	**E**				**8**			
	1	1	1	0	1	0	0	0
e	**3**				**2**			
	0	0	1	1	0	0	1	0
l	**9**				**B**			
	1	0	0	1	1	0	1	1
l	**F**				**D**			
	1	1	1	1	1	1	0	1
o	**0**				**6**			
	0	0	0	0	0	1	1	0

Dengan demikian kata "hello" hasil konversinya menjadi: **E8329BFD06**.

Tabel 2.6 Skema 7 Bit

				b7	0	0	0	0	1	1	1	1
				b6	0	0	1	1	0	0	1	1
				b5	0	1	0	1	0	1	0	1
b4	**b3**	**b2**	**B1**		0	1	2	3	4	5	6	7
0	0	0	0	0	@	Δ	SP	0	-	P	¨	p
0	0	0	1	1			!	1	A	Q	a	q
0	0	1	0	2	$	Φ	"	2	B	R	b	r
0	0	1	1	3		Γ	#	3	C	S	c	s
0	1	0	0	4		Λ		4	D	T	d	t
0	1	0	1	5		Ω	%	5	E	U	e	u
0	1	1	0	6		∏	&	6	F	V	f	v
0	1	1	1	7		Ψ	.	7	G	W	g	w
1	0	0	0	8		Σ	(	8	H	X	h	x
1	0	0	1	9		Θ	)	9	I	Y	i	y
1	0	1	0	10	LF	Ξ	*	:	J	Z	j	z
1	0	1	1	11			+	;	K	Ä	k	ä
1	1	0	0	12			,	<	L	Ö	l	ö
1	1	0	1	13	CR		-	=	M		m	
1	1	1	0	14		ß	.	>	N	Ü	n	ü
1	1	1	1	15			/	?	O		o	

(Sumber : Khang,Bustam Ir,"Trik Pemrograman Aplikasi Berbasis SMS")

Cara penggunaan tabel :

Misalnya untuk karakter 'N', yaitu :

1. Cari posisi karakter 'N' pada tabel.
2. Baca nilai biner B1 – B4 pada sisi kiri tabel dan B5 – B7 pada sisi atas tabel.

3. Pembacaan nilai biner dimulai dari kiri ke kanan yaitu B7 – B1, untuk karakter 'N' bernilai 1001110.

Setelah mempelajari masing-masing *header* maupun *sub-header* untuk mengirim SMS maka untuk menggabungkan kedelapan *header* tersebut menjadi PDU yang lengkap adalah sebagai berikut :

Contoh: Untuk mengirimkan kata "*hello*" ke *handphone* nomor 628129573337 lewat SMS-*center* Excelcom , tanpa membatasi jangka waktu *valid*, maka PDU lengkapnya adalah:

Penjelasan dari format tersebut dapat dijelaskan pada tabel berikut:

Tabel 2.7 Format SMS Kirim Dalam PDU

Heksa	**2 Penjelasan**
07	Panjang pasangan nomer SMS Center termasuk tipe kode
91	Tipe kode Nasional atau Internasional (91 = kode internasional)
2618485400F9	Nomer Service Center (62818445009)
01	Tipe SMS (01 = tipe untuk pengiriman)
00	Nomer Referensi
0C	Panjang nomer HP penerima
91	Type kode Nasional atau Internasional
261892753373	Nomer HP penerima (628129573337)
00	Type bentuk SMS (00 = dikirim sebagai SMS)

00	Type data Coding
05	Panjang pasangan dari isi SMS
E8329BFD06	Isi SMS

(Sumber : Khang,Bustam Ir,"Trik Pemrograman Aplikasi Berbasis SMS"

2.5.1.2 PDU untuk Terima SMS dari *SMS-Centre*

Kebanyakan *header* di bawah ini telah dibahas sebelumnya, kecuali beberapa yang berbeda, dijelaskan di bawah ini : (*Sumber : Khang,Bustam Ir,"Trik Pemrograman Aplikasi Berbasis SMS"*)

- No SMS-*Center*
- Tipe SMS → untuk SMS-terima = 4 → **04**
- No *handphone* pengirim
- Bentuk SMS
- Skema *Encoding*
- Tanggal dan waktu di-stamp di SMS-*Center* diwakili oleh 12 bilangan heksa (6 pasangan) : yy/mm/dd:mm:ss
- Batas waktu *validitas*, jika tidak dibatasi dilambangkan dengan **00**
- Isi SMS

Isi SMS yang berupa tulisan disini nantinya dilakukan pengkonversian yaitu mengubahnya menjadi 8 bit dan kemudian menjadi 7 bit.

Contoh Format SMS Terima dalam PDU:

07	91	2658050000F0	04	0C	91	265816107398	00	00	207022512380	00	05	E8329BFD06

Penjelasan dari format tersebut dapat dijelaskan pada tabel berikut:

Tabel 2.8 Format SMS Terima Dalam PDU

3 Heksa	4 Penjelasan
07	Panjang pasangan nomer SMS Center
91	Tipe kode Nasional atau Internasional (91 = kode internasional)
2658050000F0	Nomor Service Center (62855000000)
04	Tipe SMS (04 = tipe untuk terima)
0C	Panjang dari nomer HP pengirim
91	Tipe kode Nasional atau Internasional
265816107398	Nomer Hp pengirim (628561013789)
00	Tipe bentuk SMS (00 = dikirim sebagai SMS)
00	Tipe data Coding
207022512380	Waktu SMS sampai di SMS Center yaitu tanggal 22-07-02, pukul 15:32:08 WIB
00	Jangka waktu SMS expired (00 = tidak memiliki batas)
05	Panjang pasangan dari isi SMS
E8329BFD06	Isi SMS

(Sumber : Khang,Bustam Ir,"Trik Pemrograman Aplikasi Berbasis SMS")

Setelah mengupas satu demi satu header untuk SMS-Terima ini, maka untuk PDU dibawah ini:

07912658050000F0,04,0C91265816107398,00,00,207022512380,00,

05E8329BFD06

Dapat kita artikan sebagai berikut :

1. SMS tersebut dikirim lewat SMS-center 62855000000
2. SMS tersebut merupakan SMS terima
3. SMS tersebut dikirim dari *handphone* no. 628561013789
4. SMS tersebut diterima dalam bentuk SMS
5. SMS tersebut memiliki skema encoding 7 bit
6. SMS tersebut sampai di SMS-*Centre* pada tanggal: 22-07-02, pukul: 15:32:08 WIB
7. SMS tersebut tidak memiliki batas waktu valid
8. SMS tersebut isinya adalah "hello"

2.5.2 Jaringan GSM (*Global System Mobile*) Untuk Layanan SMS

Short Message Service (SMS) adalah salah satu jasa layanan dari perusahaan operator telepon selular GSM. Dengan sarana ini maka telepon selular dapat menerima dan mengirimkan pesan-pesan pendek dengan bentuk teks dengan panjang maksimal sebanyak 160 karakter untuk alfabet latin dan 70 karakter untuk alfabet non latin, seperti : alfabet Arab atau Cina. Ada satu hal yang sangat menarik dari layanan ini, yaitu tawaran tarif yang relatif murah untuk setiap kali pengiriman pesan.

Sedangkan yang dimaksud dengan *SMS Gateway* adalah program aplikasi yang menghubungkan antara semua SMS yang dikirim dan diterima ke sebuah PC dengan menggunakan jaringan GSM. Bagian ini berfungsi untuk membaca SMS dari MS, kemudian mengirimkan kembali SMS balasan ke MS penanya. Dalam operator GSM setiap SMSC akan selalu berbeda dengan yang lainnya. Berikut daftar SMSC berdasarkan tiap-tiap operatornya: (*Sumber: Kang,Bustam Ir,"Trik Pemrograman Aplikasi Berbasis SMS"*)

Tabel 2.9 Daftar SMSC

OPERATOR GSM	NOMOR SMSC
Telkomsel	+62811000000
Satelindo	+62816124
Excelcom	+62818445009
Indosat	+62855000000

(***Sumber : Khang,Bustam Ir,"Trik Pemrograman Aplikasi Berbasis SMS"***)

SMS Gateway bekerja dengan cara menghubungkan *handphone* yang memiliki fasilitas SMS dengan komputer (PC) selaku operator otomatisnya. Keduanya akan dihubungkan oleh suatu kabel data dari *handphone* yang sesuai dengan serial port yang ada di PC. Komunikasi antara *handphone* dengan PC menggunakan bahasa pemrograman serial port.

2.5.3 *AT Command*

Komunikasi data antara telepon seluler dengan periperal lain seperti mikrokontroler dilakukan secara serial menggunakan perintah-perintah AT (*AT Command*). Dengan mengirimkan perintah-perintah AT yang spesifik dapat memerintahkan telepon seluler untuk melakukan apa yang kita inginkan.

Tabel berikut merupakan beberapa perintah-perintah AT yang digunakan untuk mengoperasikan SMS.

Tabel 2.10 *AT Command* pada SMS

Perintah	Fungsi
AT+CMGC	Mengirim sebuah perintah SMS
AT+CMGD	Menghapus sebuah SMS dalam SMS memori
AT+CMGF	SMS Format
AT+CMGL	Daftar SMS
AT+CMGR	Membaca dalam sebuah SMS
AT+CMGS	Mengirim sebuah SMS
AT+CSCA	Alamat dari pusat SMS servis

Sumber: www.my-siemens.com

BAB III

PERANCANGAN DAN PEMBUATAN ALAT

3.1 Perancangan Alat

Pada skripsi ini dirancang dan dibuat suatu sistem untuk menerima dan mengolah data lintang dan bujur yang dikirimkan oleh GPS Receiver, mengirimkan data posisi sekaligus status kereta api melalui SMS. SMS tersebut akan diterima oleh HP kemudian dibaca oleh PC melalui *serial port* dengan kecepatan transfer sebesar 19200 bps.

Pada skripsi ini juga dirancang dan dibuat *software* untuk mengolah data lintang dan bujur menjadi data posisi pada peta. Sistem pengolah dan penampil lokasi kereta api merupakan sebagian dari "Sistem Penelusuran Posisi Kereta Api Berbasis GPS" yang terdapat pada kereta api.

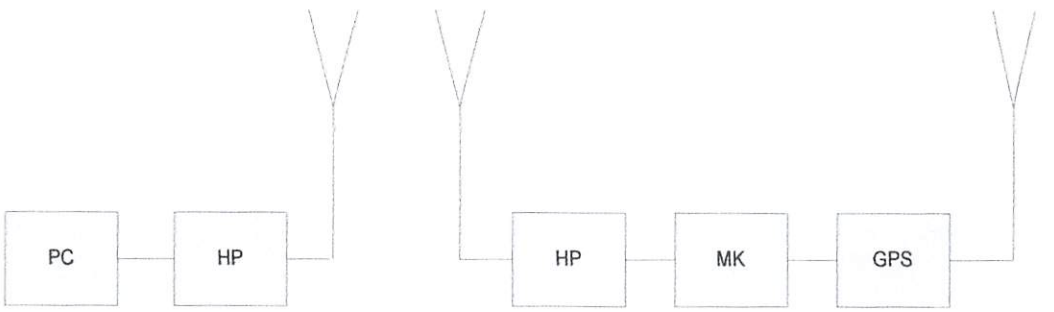
3.2 Spesifikasi Alat

Spesifikasi alat yang direncanakan adalah sebagai berikut :

- Alat ini berbasis pada data sebuah *GPS Receiver* jenis GPS AN1 produksi CV.An-Nashr SOLUSITAMA dengan format keluaran standar NMEA 0183 Baudrate 4800bps dengan menggunakan sistem mikrokontroler RENESAS R8C/Tiny.
- Alat dipasangkan pada kereta api dengan kondisi kereta api berada pada luar ruangan.
- SMS yang diterima oleh HP memiliki kecepatan transfer sebesar 19200 bps.

- Data yang dikirim ke *Base Station* melalui SMS berupa informasi ID kereta api, posisi *Lattitude* (lintang) dan *Longitude* (bujur) kereta api.
- Lokasi yang ditunjukkan memiliki ketelitian berdasarkan tingkat ketelitian dari *GPS Receiver* yang dipergunakan sebesar <15 meter.
- Catu daya untuk sistem mikrokontroler dan *GPS Receiver*, diperoleh dari kereta api sebesar 12 volt.

3.3 Diagram Blok Alat



Keterangan masing-masing blok adalah sebagai berikut :

1. **GPS (*Global Positioning Sistem*)**
   - GPS yang digunakan adalah *GPS Receiver* jenis GPS AN1 produksi CV.An-Nashr SOLUSITAMA dengan format keluaran standar NMEA 0183 dan kecepatan Baudrate 4800bps (bit/second).
   - GPS menerima data lintang dan bujur dari satelit.
   - Keluaran GPS salah satu diambil dari data GPRMC (lintang dan bujur).
   - Protokol yang digunakan antara GPS dengan mikrokontroler adalah komunikasi serial asinkron dengan baudrate 4800bps.

- Tidak ada hand shaking, karena datanya searah.

2. **Mikrokontroller**

- Mikrokontroler yang digunakan adalah Renesas R8C/Tiny.
- Menyeleksi data kemudian diambil yang berupa data lintang dan bujur dari GPS.
- Merubah format data lintang atau bujur tadi (dari GPS) menjadi data yang siap di kirim melalui SMS.

3. **HP (*Handphone*)**

- HP yang digunakan adalah tipe siemens C35 dengan *baudrate* 19200bps.
- Data yang akan dikirim diubah ke data PDU, kemudian dikirim ke HP di *base station.*
- Berfungsi mengirim dan menerima SMS.

4. **PC (*Personal Computer*)**

- Berfungsi untuk mengubah SMS menjadi data lintang dan bujur dalam tampilan peta.
- *Data Base* yang digunakan adalah Paradox 7.0.

3.3.1 Cara Kerja Sistem Pada Alat Secara Keseluruhan

Pada waktu pesawat penerima GPS (GPS *Receiver*) dihidupkan, pesawat tersebut akan mencari sinyal dari satelit terdekat yang dapat diterima. Jika sudah ditemukan, receiver akan mengunci sinyal tersebut dan mencari data pada almanac satelit (informasi yang berisi posisi satelit-satelit dalam satu harinya) yang berada dekat dengan *receiver* untuk membandingkan waktu pada saat sinyal

yang ditransmisikan oleh satelit dengan waktu pada saat sinyal tersebut diterima oleh GPS penerima. Data yang diterima oleh GPS berupa data lintang dan bujur.

Mikrokontroller menyeleksi data kemudian diambil data yang berupa lintang dan bujur. Mikrokontroller akan merubah format data lintang dan bujur tadi menjadi data yang siap dikirim menjadi SMS (*Short Message Service*). Panjang maksimum dari sebuah pesan singkat adalah 160 karakter, fasilitas ini disediakan oleh jaringan telepon selular.

Setelah data yang berupa SMS diterima oleh HP pada *base station*, data tersebut akan dideteksi dan diolah oleh komputer melalui *software* penampil lokasi kereta api yang ditulis dengan bahasa pemrograman Delphi 7.0.

Software penampil lokasi kereta api akan mengolah data posisi lintang dan bujur kereta api menjadi lokasi posisi koordinat x dan y pada layar monitor. Sehingga pada layar monitor status serta lokasi kereta api akan tampak pada peta.

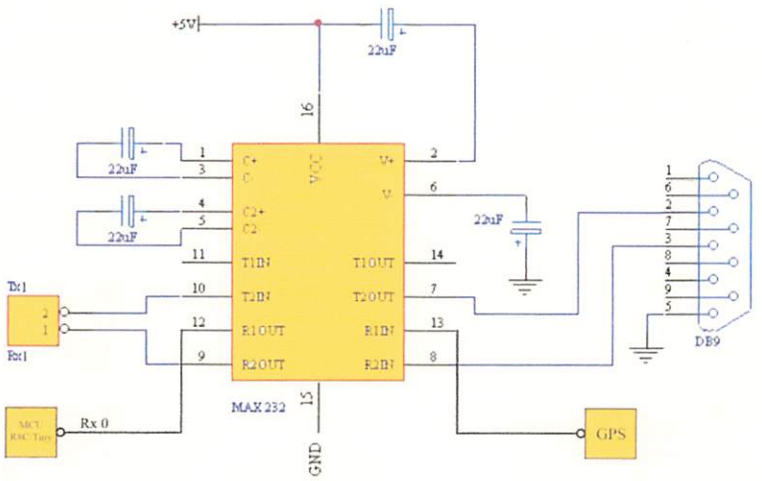
3.4 Perancangan serta Pembuatan Perangkat Keras (*Hardware*)

Informasi merupakan SMS yang berasal dari HP dan berisi data berupa PDU (data 7 bit). Oleh karena itu diperlukan *software* pembaca SMS sehingga data dapat terbaca dan dapat diolah menjadi gambar yang kemudian disimpan.

3.4.1 Perencanaan Rangkaian *Interface*

Untuk dapat berkomunikasi antara MCU dengan HP Siemens C35, maka perlu disesuaikan *signal* yang dipakai yaitu dengan menggunakan *interface* RS

232. Dengan IC yang dipakai adalah MAX232. Berikut ini adalah rangkaian *interface* dari MCU ke HP Siemens C35.



Gambar 3.1 Rangkaian RS 232

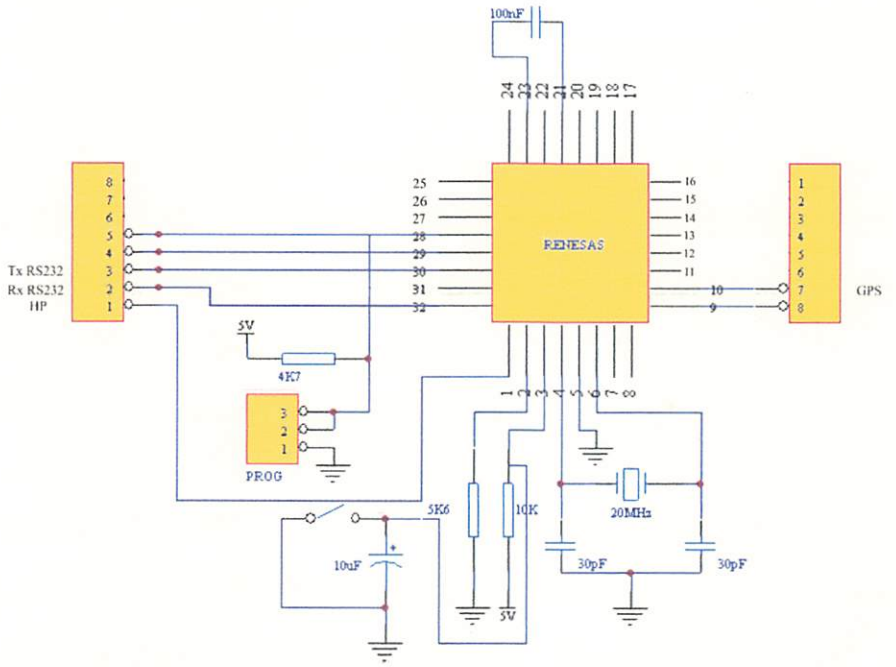
(Sumber : Perencanaan)

Pada MAX232, port yang digunakan :

- Port 1 (C1+) dengan port 3 (C1-) disambungkan dengan capasitor 22μF.
- Port 2 (V+) dengan port 16 (Vcc) disambungkan dengan capasitor 22μF.
- Port 4 (C2+) dengan port 5 (C2-) disambungkan dengan capasitor 22μF.
- Port 6 (V-) dengan GND (ground) disambungkan dengan capasitor 22μF.
- Port 7 (T2 OUT) disambungkan ke DB 9 port 2 (Rx).
- Port 8 (R2 IN) disambungkan ke DB 9 port 3 (Tx).
- Port 9 (R2 OUT) disambungkan ke Rx1 (MCU Renesas R8C/Tiny).
- Port 10 (T2 IN) disambungkan ke Tx1 (MCU Renesas R8C/Tiny).
- Port 12 (R1 OUT) disambungkan ke Rx0 (MCU Renesas R8C/Tiny).
- Port 13 (R1 IN) disambungkan ke GPS

3.4.2 Perencanaan MCU Renesas R8C/Tiny

MCU bekerja jika GPS aktif. Alasan penggunaan MCU Renesas R8C/Tiny adalah rangkaiannya yang praktis (bentuk fisik IC-nya yang sangat kecil) karena tersusun dalam satu modul yang sangat mudah untuk kita gunakan, sederhana dan tidak memakan tempat. Jumlah Tx (Transmitter) dan Rx (Receiver) ada 2, yaitu untuk Tx pada kaki ke-11 (port 1.4) dan kaki ke-32 (port 0.0). Sedangkan untuk Rx pada kaki ke-1 (port 3.7) dan kaki ke-10 (port 1.5). Pada MCU Renesas R8C/Tiny rangkaian ADC (*Analog to Digital Converter*) sudah ada sehingga kita tidak perlu membuat rangkaian ADC baru. Mikrokontroler RENESAS dibangun menggunakan proses gerbang *silicon* CMOS dengan kemampuan tinggi menggunakan CPU seri R8C/Tiny dan dikemas dalam modul plastik dengan jumlah pin sebanyak 32. Mikrokontroler ini beroperasi menggunakan perintah canggih khususnya efisiensi perintah dengan level tinggi. Mikrokontroler ini mempunyai 1 Mbytes kapasitas alamat, yang bisa digunakan untuk mengeksekusi perintah dengan kecepatan tinggi. Data flash ROM sebesar 2 KB x 2 blocks.



Gambar 3.2 Port Pada MCU Renesas R8C/Tiny Yang Dipakai Pada Sistem (Sumber : Perencanaan)

Fungsi port pada MCU yang digunakan pada alat adalah sebagai berikut :

- Port 0 memiliki fungsi sebagai berikut :
  - Port 0.0 digunakan sebagai output ke RS 232 (Tx)
- Port 1 memiliki fungsi sebagai berikut :
  - Port 1.5 digunakan sebagai input data dari GPS.
- Port 3 memiliki fungsi sebagai berikut :
  - Port 3.7 digunakan sebagai input dari RS 232 (Rx)
- Port 4 memilki fungsi sebagai berikut :
  - Port 4.7 dengan port 4.6 disambungkan dengan rangkaian osilator.

3.4.3 Rangkaian Reset

Untuk me-reset mikrokontroler Renesas R8C/Tiny, maka pin RST diberi logika tinggi selama sekurangnya dua siklus mesin (24 periode osilator). Untuk membangkitkan sinyal reset kapasitor dihubungkan dengan V_{CC} dan sebuah resistor yang dihubungkan ke ground.

Karena kristal yang digunakan mempunyai frekuensi sebesar 20 MHz, maka satu periode dapat dihitung dari persamaan :

$$T = \frac{1}{f_{XTAL}} = \frac{1}{20MHz} s = 5 \times 10^{-8} \text{ s} \quad \text{.....(3.1)}$$

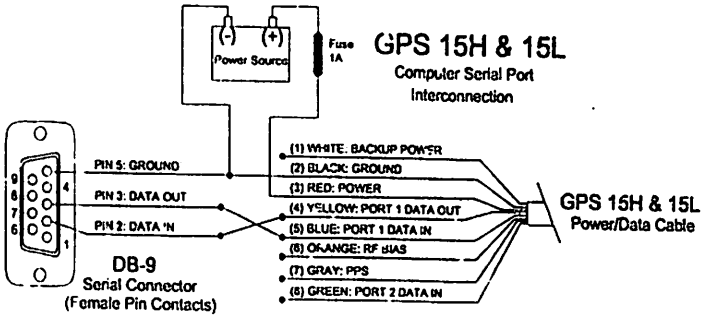
Sehingga waktu minimal logika tinggi yang dibutuhkan untuk mereset mikrokontroler dapat dihitung menggunakan persamaan :

$$t_{reset(min)} = T \times \text{periode yang dibutuhkan} \quad \text{.....(3.2)}$$

Jadi mikrokontroler membutuhkan waktu minimal 1.2 μs untuk mereset. Waktu minimal inilah yang dijadikan pedoman untuk menentukan nilai R dan C. Dari Persamaan (3.3) dengan menentukan nilai R = 22 kΩ, dan C = 10 μF, maka:

$$t = 0{,}357 \text{ R.C} \quad \text{.....(3.3)}$$

3.4.4 Perancangan GPS Dengan DB9



Gambar 3.3 Perancangan GPS Dengan DB9

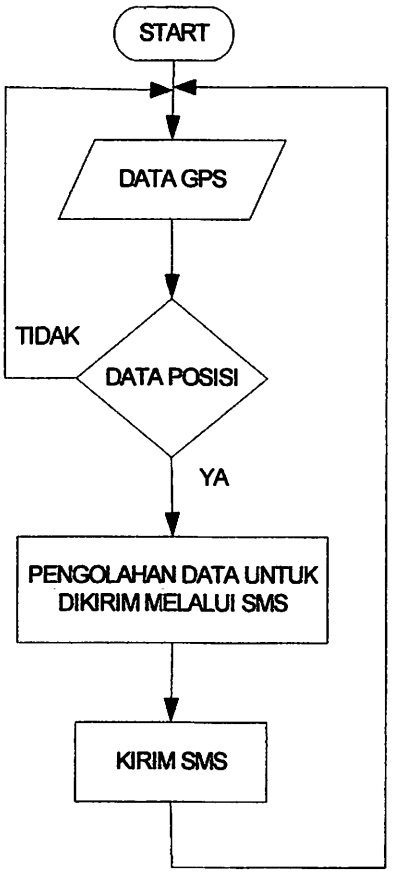
(Sumber : Perancangan)

3.5 Perancangan serta Pembuatan Perangkat Lunak *(Software)*

Software untuk mengolah dan menampilkan posisi kereta api dibuat dengan *software* Borland Delphi 7.0 Untuk mengirim dan menerima data dari *serial port* digunakan komponen Delphi yaitu Cport2.6.

Software ini terdiri dari program utama dan subrutin-subrutin yang digunakan pada program utama tersebut.

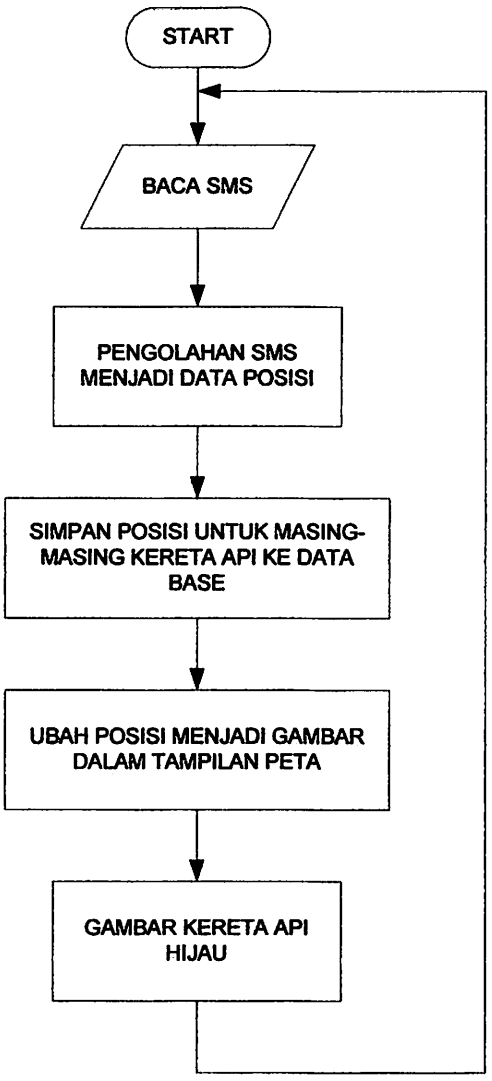
3.5.1 *Flowchart* Program Pada Mikrokontroller



Keterangan *Flowchart*:

- Data GPS dibaca kemudian dibandingkan apakah data tersebut berupa data posisi atau tidak.
- Jika ya, data posisi diolah menjadi data PDU kemudian dikirim melalui SMS.
- Proses akan diulang secara terus menerus.

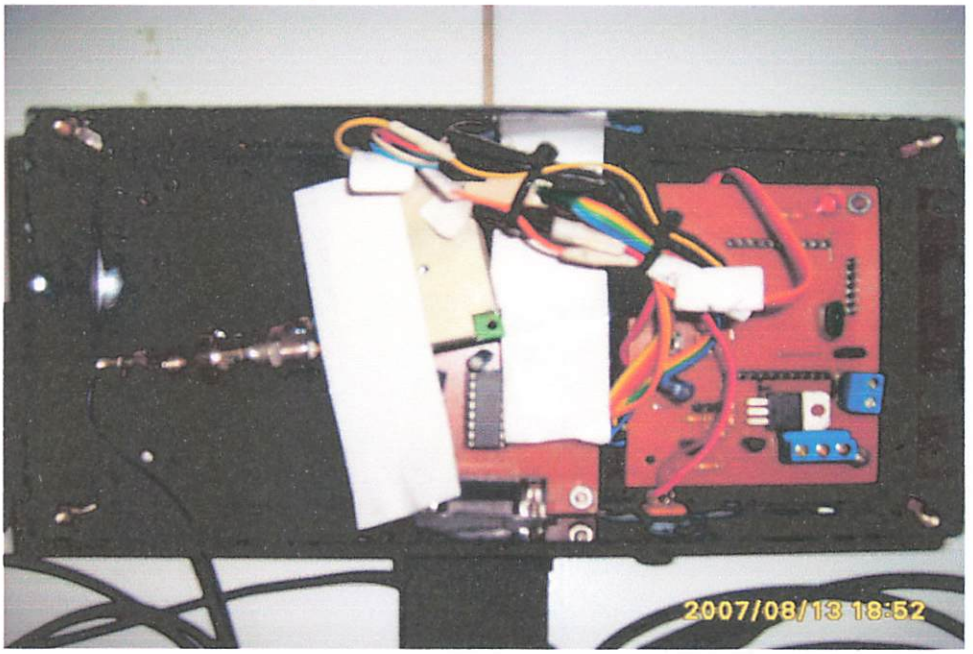
3.5.2 *Flowchart* Program Pada PC



Keterangan *Flowchart*:

- SMS yang diterima kemudian diolah menjadi data posisi.
- Data posisi untuk masing-masing kereta api kemudian disimpan ke *data base.*
- Kemudian posisi diubah menjadi gambar dalam tampilan peta dan ditandai dengan warna hijau yang menunjukkan posisi kereta api.

3.6 Hasil Perancangan Sistem



Gambar 3.4 Rangkaian Hasil Perancangan Sistem

(Sumber : Perancangan)

BAB IV

PENGUJIAN DAN ANALISA ALAT

Pada bab ini dibahas tentang pengujian dan analisa alat yang dirancang, yang meliputi perangkat keras (*hardware*) dan perangkat lunak (*software*). Untuk mengetahui sistem yang dirancang sesuai dengan fungsi yang diharapkan, dilakukan pengujian terhadap sistem tersebut baik secara keseluruhan maupun subsistemnya. Berikut merupakan penjelasan mengenai prosedur pengukuran dan data hasil pengujian.

Bagian-bagian yang diuji adalah:

a) Bagian telepon seluler pada rangkaian sistem penelusuran posisi kereta api dengan menguji fungsi *AT Command* dan menguji format data SMS.
b) Pengiriman SMS.
c) Data In/Data Out RS232
d) Data GPS (*Global Positioning System*).

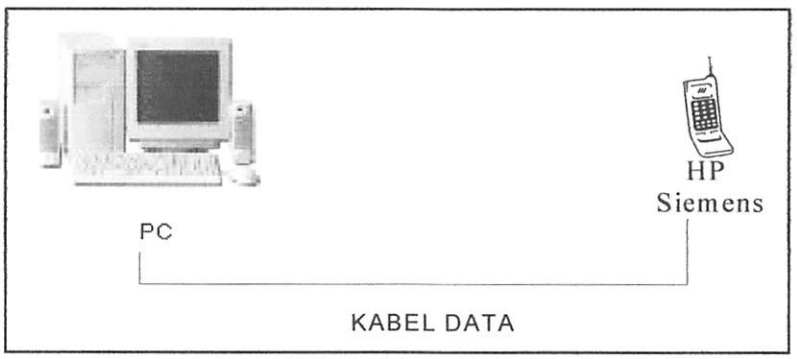
Alat-alat bantu yang digunakan dalam pengujian adalah sebagai berikut:

1. *Handphone* Siemens C35.
2. Kabel data Siemens C35.
3. Komputer.
4. Osiloskop

4.1 Pengujian *AT Command* dan Format Data SMS Pada Telepon Seluler

❖ **Tujuan**

Untuk menguji fungsi *AT Command* pada telepon seluler dan mengetahui data PDU (*Protocol Data Unit*) yang dikirim dari telepon seluler ke PC (*Personal Computer*).

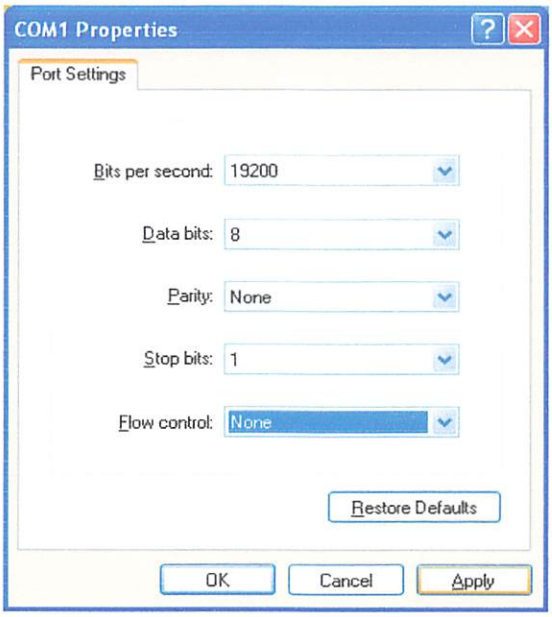


Gambar 4.1 Blok Diagram Pengujian Telepon Seluler
Sumber : Pengujian

❖ **Langkah pengujiannya adalah sebagai berikut :**

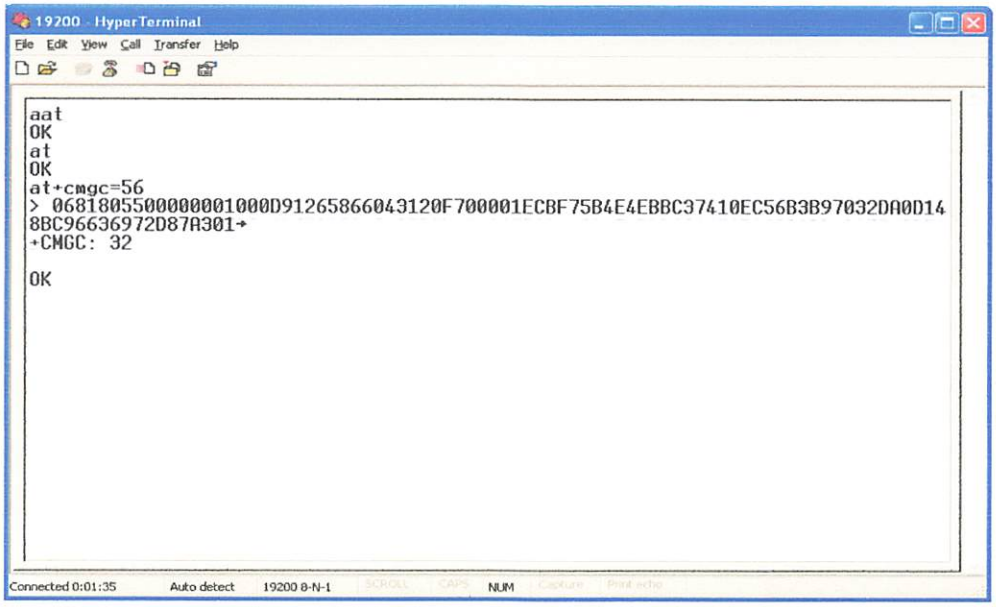
- Hubungkan telepon seluler dengan komputer menggunakan kabel data serial
- Menjalankan program *Hyper Terminal*
- Melakukan *setting port* serial pada program *Hyper Terminal*
- Mengetik instruksi AT+CMGC = 56 untuk mengirim SMS
- 56 = jumlah pasangan heksa PDU SMS dimulai setelah nomor *SMS-centre* (maksimal 140)

❖ Hasil Pengujian



Gambar 4.2 Penyesuaian *Baudrate* dan *Flow Control*

Sumber : Pengujian



Gambar 4.3 Hasil Pengujian Dengan *AT Command*

Sumber : Pengujian

❖ **Analisa Hasil Pengujian**

Berdasarkan pengujian ini menunjukkan bahwa komunikasi dengan telepon seluler dapat dilakukan dengan menggunakan instruksi *AT Command* dan *AT Command* merupakan bahasa yang digunakan untuk menjalankan menu-menu pada telepon seluler.

Dari format data PDU yang diterima dapat dilihat ada delapan header didalamnya yaitu:

1. Nomor *SMS-Centre*, terdapat tiga *subheader*:
   - 06 = Jumlah pasangan heksa *SMS-Centre*
   - 81 = Kode internasional
   - 80-55-00-00-00 = Nomor *SMS-Centre* Indosat-IM3: 0855000000
2. Tipe SMS:
   - Untuk *SEND* tipe *SMS* = 1. Jadi bilangan heksanya adalah 01
3. Nomor Referensi

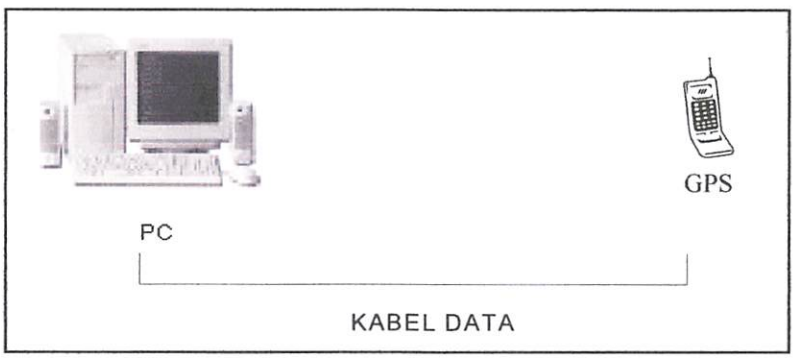
   Nomor referensi ini dibiarkan dulu 0, jadi bilangan heksanya adalah 00.
4. Nomor ponsel penerima, terdapat tiga *subheader*:
   - 0D = Jumlah bilangan desimal nomor pengirim (13 angka)
   - 91 = Kode Internasional
   - 26-58-66-04-31-20-F7 = Nomor penerima (6285664013027)
5. Bentuk SMS:
   - 00 = Menandakan data dikirim sebagai SMS.

6. Skema *Encoding*:
   - 00 = Menandakan skema *encoding* menggunakan skema 7 bit.
7. Batas waktu validitas:
   - Agar SMS kita pasti terkirim sampai ke ponsel penerima, sebaiknya kita tidak memberikan batasan waktu validnya.
8. Isi SMS:
   - 1E = Jumlah karakter dari data yang dikirm (contoh : Koordinat 0756.8286 11236.7030 = 30 karakter)
   - CBF75B4E4EBBC37410EC56B3B97032DA0D148BC96636972D87A30 1 (bentuk PDU).

4.2 Pengujian Data GPS

❖ **Tujuan**

Untuk menguji fungsi GPS apakah masih bisa digunakan atau tidak.



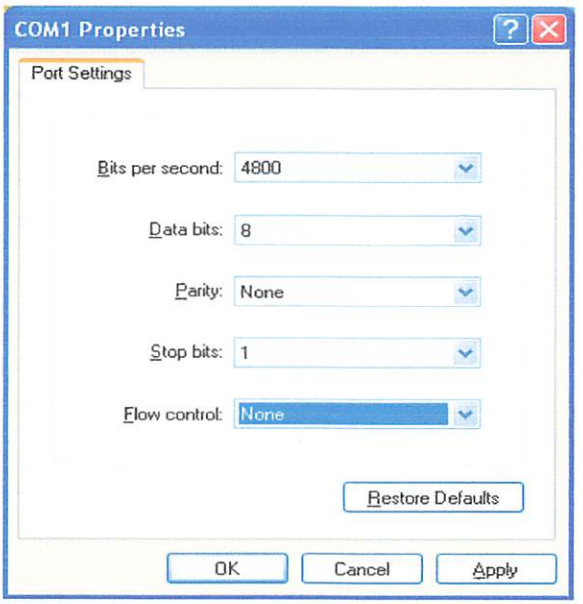
Gambar 4.4 Blok Diagram Pengujian Data GPS

Sumber : Pengujian

❖ **Langkah pengujiannya adalah sebagai berikut :**

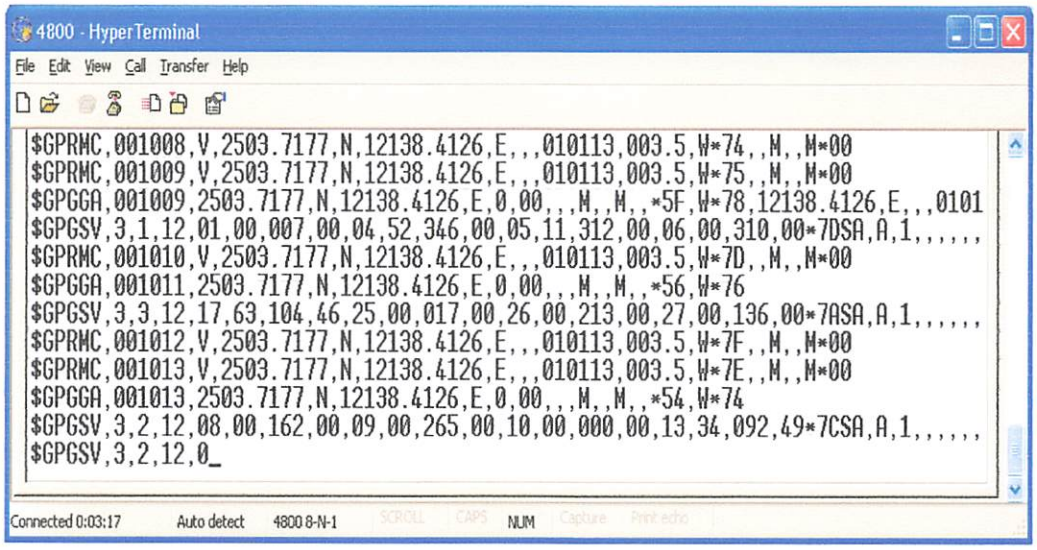
- Hubungkan GPS dengan komputer menggunakan kabel data serial
- Menjalankan program *Hyper Terminal*
- Melakukan *setting port* serial pada program *Hyper Terminal*

❖ **Hasil Pengujian**



Gambar 4.5 Penyesuaian *Baudrate* dan *Flow Control*

Sumber : Pengujian



Gambar 4.6 Hasil Pengujian Data GPS

Sumber : Pengujian

❖ **Analisa Hasil Pengujian**

Data yang diambil adalah data GPRMC yang berupa data *lattitude* (lintang) dan data *longitude* (bujur). Contoh data GPRMC:

$GPRMC,060149,A,**0756.5394**,S,**11237.0438**,E,000.0,000.0,060604,001.4,E*66

0756.5394 = data *latitude* (lintang)

11237.0438 = data *longitude* (bujur)

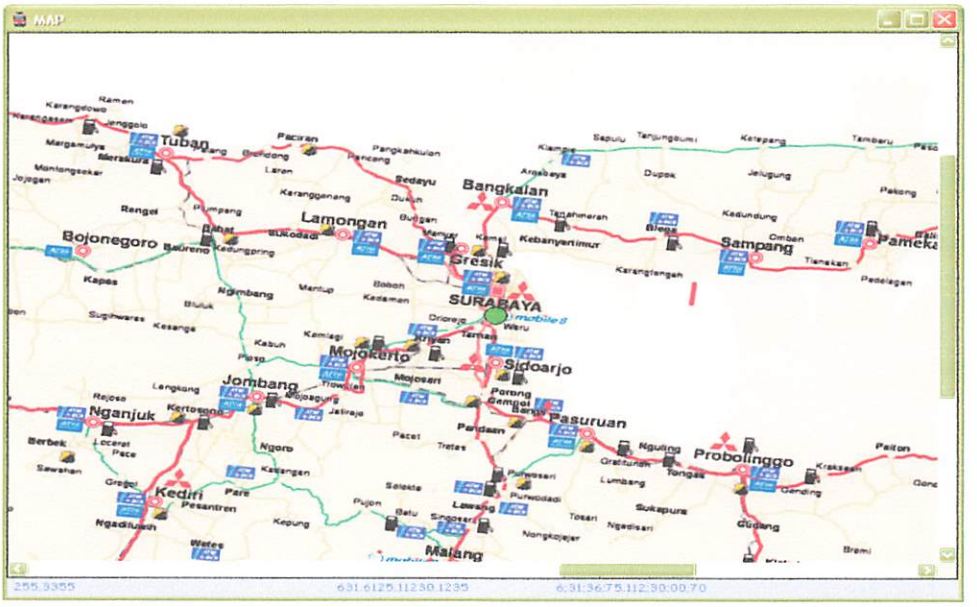
❖ Contoh Tampilan Peta Pada PC

Gambar 4.7 Posisi Kereta Api pada Koordinat (754.9900 S, 11238.1000 E)

Sumber : Pengujian

Gambar 4.8 Posisi Kereta Api pada Koordinat (753.8138 S, 11239.9636 E)

Sumber : Pengujian



Gambar 4.9 Posisi Kereta Api pada Koordinat (717.3661 S, 11244.1426 E)

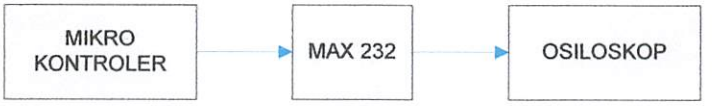
Sumber : Pengujian

4.3 Pengujian RS 232

❖ **Tujuan**

Pengujian RS 232 ini dilakukan untuk melihat adanya sinyal pada transmitter RS 232. Untuk melihat bentuk gelombang ini dilakukan dengan menggunakan osiloskop.

❖ **Langkah pengujiannya adalah sebagai berikut :**

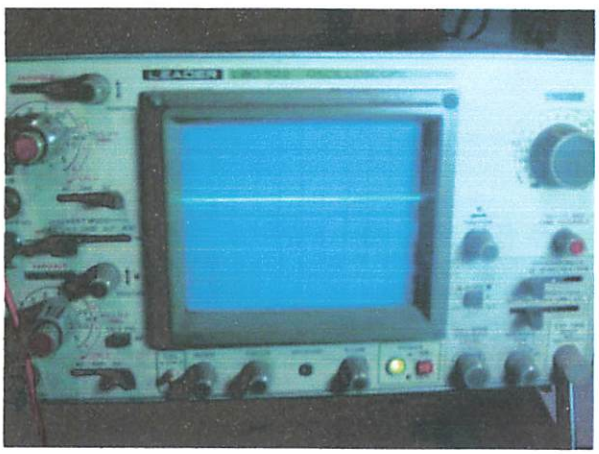


Gambar 4.10 Blok Diagram Pengujian RS 232

Sumber : Pengujian

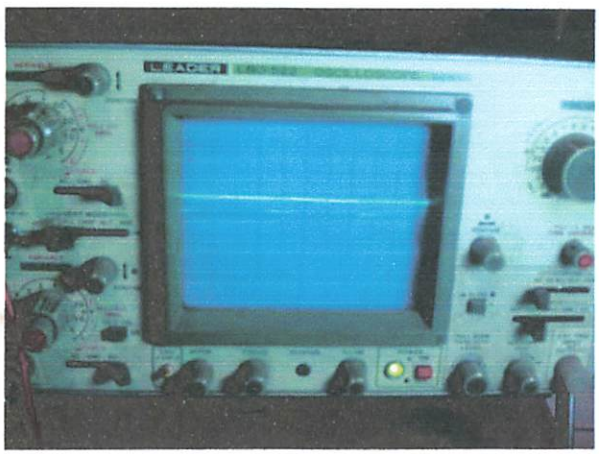
- Pin Tx mikrokontroler dihubungkan ke Max232, keluaran Max232 dihubungkan ke osiloskop.

❖ **Hasil Pengujian :**



Gambar 4.11 Gambar Sinyal Ketika Belum Ada Pengiriman Data

Sumber : Pengujian



Gambar 4.12 Gambar Sinyal Ketika Sudah Ada Pengiriman Data

Sumber : Pengujian

❖ **Analisa Hasil Pengujian**

- Sebelum ada pengiriman data tidak ada gelombang kotak yang keluar pada osiloskop, hal ini menunjukkan tidak ada data *AT-Command* yang dikirim.
- Setelah ada pengiriman data ada gelombang kotak yang keluar pada osiloskop, hal ini berarti sudah ada data *AT-Command* yang dikirim.

4.4 Pengujian SMS

❖ **Tujuan**

Pengujian SMS ini bertujuan untuk mengetahui error dari setiap SMS yang diterima. Serta untuk mengetahui apakah selang waktu setiap pengiriman SMS adalah dalam 2 menit.

❖ **Hasil Pengujian**

Tabel 4.1 Hasil Pengujian SMS

No.	Waktu SMS Terima	Lokasi Pengujian	Isi SMS
1.	22-9-07 (08:12 WIB)	Jln. Bendungan Darma 16	Koordinat 0754,9747 11238,1053
2.	22-9-07 (08:15 WIB)	Jln. Bendungan Darma 16	Koordinat 0754,9747 11238,1053
3.	22-9-07 (10:24 WIB)	Jln. Sigura-gura / pertigaan Kampus 1 ITN Malang	Koordinat 0757,4771 11236,6903
4.	22-9-07 (10:26 WIB)	Jln. Sigura-gura / pertigaan Kampus 1 ITN Malang	Koordinat 0757,4756 11236,6900
5.	22-9-07 (11:03 WIB)	Jln. Soekarno Hatta / depan Kampus Polinema Malang	Koordinat 0756,8306 11237,0152

6.	22-9-07 (11:05 WIB)	Jln. Soekarno Hatta / depan Kampus Polinema Malang	Koordinat 0756,8306 11237,0152
7.	22-9-07 (13:51 WIB)	Jln. Terusan Surabaya / depan Kampus UM	Koordinat 0757,9161 11236,9907
8.	22-9-07 (13:53 WIB)	Jln. Terusan Surabaya / depan Kampus UM	Koordinat 0757,9161 11236,9907
9.	23-9-07 (10:53 WIB)	Jln. Kertorahayu 69	Koordinat 0756,8257 11236,6992
10.	23-9-07 (10:55 WIB)	Jln. Kertorahayu 69	Koordinat 0756.8261 11236,6995

❖ **Analisa Hasil Pengujian**

Tabel 4.2 Perhitungan Error

No.	Lokasi Pengujian	Perhitungan E (%)	Hasil Perhitungan E (%)
1.	Jln. Bendungan Darma 16	$E\ \% = \frac{0754{,}9750 - 0754{,}9747}{0754{,}9750} \times 100\%$	3.97×10^{-5}
	Jln. Bendungan Darma 16	$E\ \% = \frac{11238{,}1153 - 11238{,}1053}{11238{,}1153} \times 100\%$	$8{,}89 \times 10^{-5}$
2.	Jln. Bendungan Darma 16	$E\ \% = \frac{0754{,}9750 - 0754{,}9747}{0754{,}9750} \times 100\%$	3.97×10^{-5}
	Jln. Bendungan Darma 16	$E\ \% = \frac{11238{,}1153 - 11238{,}1053}{11238{,}1153} \times 100\%$	$8{,}89 \times 10^{-5}$
3.	Jln. Sigura-gura / pertigaan Kampus 1 ITN Malang	$E\ \% = \frac{0757{,}4802 - 0757{,}4771}{0757{,}4802} \times 100\%$	$4{,}09 \times 10^{-4}$
	Jln. Sigura-gura / pertigaan Kampus 1 ITN Malang	$E\ \% = \frac{11236{,}7001 - 11236{,}6903}{11236{,}7001} \times 100\%$	$8{,}72 \times 10^{-5}$
4.	Jln. Sigura-gura / pertigaan Kampus 1 ITN Malang	$E\ \% = \frac{0757{,}4786 - 0757{,}4756}{0757{,}4786} \times 100\%$	$3{,}96 \times 10^{-4}$

	Jln. Sigura-gura / pertigaan Kampus 1 ITN Malang	E % = $\frac{11236,6906 - 11236,6900}{11236,6906}$ x 100%	$5,34 \times 10^{-6}$
5.	Jln. Soekarno Hatta / depan Kampus Polinema Malang	E % = $\frac{0756,8310 - 0756,8306}{0756,8310}$ x 100%	$5,28 \times 10^{-5}$
	Jln. Soekarno Hatta / depan Kampus Polinema Malang	E % = $\frac{11237,0160 - 11237,0152}{11237,0160}$ x 100%	$7,12 \times 10^{-6}$
6.	Jln.Soekarno Hatta / depan Kampus Polinema Malang	E % = $\frac{0756,8310 - 0756,8306}{0756,8310}$ x 100%	$5,28 \times 10^{-5}$
	Jln. Soekarno Hatta / depan Kampus Polinema Malang	E % = $\frac{11237,0160 - 11237,0152}{11237,0160}$ x 100%	$7,12 \times 10^{-6}$
7.	Jln. Terusan Surabaya / depan Kampus UM	E % = $\frac{0757,9180 - 0757,9161}{0757,9180}$ x 100%	$2,5 \times 10^{-4}$
	Jln. Terusan Surabaya / depan Kampus UM	E % = $\frac{11236,9914 - 11236,9907}{11236,9914}$ x 100%	$6,23 \times 10^{-6}$
8.	Jln. Terusan Surabaya / depan Kampus UM	E % = $\frac{0757,9180 - 0757,9161}{0757,9180}$ x 100%	$2,5 \times 10^{-4}$
	Jln. Terusan Surabaya / depan Kampus UM	E % = $\frac{11236,9914 - 11236,9907}{11236,9914}$ x 100%	$6,23 \times 10^{-6}$
9.	Jln. Kertorahayu 69	E % = $\frac{0756,8260 - 0756,8257}{0756,8260}$ x 100%	$3,96 \times 10^{-5}$
	Jln. Kertorahayu 69	E % = $\frac{11236,7001 - 11236,6992}{11236,7001}$ x 100%	8×10^{-6}
10.	Jln. Kertorahayu 69	E % = $\frac{0756,8301 - 0756,8261}{0756,8301}$ x 100%	$5,24 \times 10^{-4}$
	Jln. Kertorahayu 69	E % = $\frac{11236,7003 - 11236,6995}{11236,7003}$ x 100%	$7,12 \times 10^{-6}$

❖ **Perhitungan Error Rata-rata**

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

$$E\ \%\ (\text{total}) = (3{,}97\text{x}10^{-5} + 8{,}89\text{x}10^{-5}) + (3{,}97\text{x}10^{-5} + 8{,}89\text{x}10^{-5}) + (4{,}09\text{x}10^{-4} + 8{,}72\text{x}10^{-5}) + (3{,}96\text{x}10^{-4} + 5{,}34\text{x}10^{-6}) + (5{,}28\text{x}10^{-5} + 7{,}12\text{x}10^{-6}) + (5{,}28\text{x}10^{-5} + 7{,}12\text{x}10^{-6}) + (2{,}5\text{x}10^{-4} + 6{,}23\text{x}10^{-6}) + (2{,}5\text{x}10^{-4} + 6{,}23\text{x}10^{-6}) + (3{,}96\text{x}10^{-5} + 8\text{x}10^{-6}) + (5{,}24\text{x}10^{-4} + 7{,}12\text{x}10^{-6})$$

$$= 2{,}366\text{x}10^{-3}$$

$$\text{Jadi, } E\ \%\ (\text{rata-rata}) = \frac{2{,}366\text{x}10^{-3}}{10} \times 100\ \%$$

$$= 0{,}02\ \%$$

BAB V

PENUTUP

5.1 Kesimpulan

Dari hasil perancangan, pembuatan dan pengujian alat "Sistem Monitoring Posisi Kereta Api Berbasiskan Data *GPS* Yang Dikirimkan Melalui HP (Handphone)" secara keseluruhan, maka dapat diambil beberapa kesimpulan sebagai berikut:

1. Alat akan mengirim SMS setiap 2 menit yang berisi data lintang dan bujur dari GPS dan dapat diketahui arah serta waktu perjalanan kereta api.
2. Setelah dilakukan perbandingan pengujian dengan GARMIN GPSMAP 76CSx, selisih pengukuran hanya sedikit karena pada perancangan alat juga menggunakan modul GPS Garmin.
3. Error rata-rata dari pengujian pengiriman SMS sebesar 0,02 %.
4. Informasi dari *Handphone* Siemens C35 dan informasi dari GPS, tidak dapat dibaca langsung oleh MCU, sehingga dibutuhkan *interface* RS 232 dengan menggunakan data biner serial sebagai data yang ditransmisikan.
5. MCU Renesas R8C/Tiny bekerja secara otomatis jika *GPS* yang ada pada alat ini aktif sehingga MCU akan mengirim SMS.
6. Data lintang dan bujur dapat diolah menjadi tampilan peta pada *PC* jika ada SMS yang diterima.

5.2 Saran

1. Perangkat yang dibuat akan lebih akurat dan lebih presisi jika menggunakan komponen-komponen yang kualitasnya lebih baik.
2. Diharapkan untuk penentuan posisi kereta api akan lebih akurat dengan menggunakan server yang bisa mengakses dengan peta digital.
3. Dalam perkembangan selanjutnya dapat menggunakan HP yang menggunakan teknologi 3G, bukan menggunakan HP Siemens C35 lagi.
4. Supaya sistem lebih stabil, maka diharapkan dalam perkembangannya nanti untuk HP diganti dengan GSM *engine*.

DAFTAR PUSTAKA

[1]. Khang, Bustam, *Trik Pemrograman Aplikasi Berbasis SMS,* PT. Elex Media Komputindo, Jakarta, 2002

[2]. Abidin, Z Hasanuddin, *Penentuan Posisi Dengan GPS Dan Aplikasinya,* PT. Pradhya Paramita, Jakarta, 1995

[3]. Anonim, Datasheet RENESAS R8C/Tiny

[4]. Anonim, Datasheet GARMIN-GPS

[5]. Anonim, Datasheet MAX 232

LAMPIRAN A

(LISTING PROGRAM)

```
/*********************************************************************/
/*                                                                   */
/*  FILE        :tes.c                                               */
/*  DATE        :Mon, Aug 6, 2007                                    */
/*  DESCRIPTION :Main Program                                        */
/*  CPU TYPE    :Other                                               */
/*                                                                   */
/*  This file is generated by Renesas Project Generator (Ver.4.0).   */
/*                                                                   */
/*********************************************************************/

#include <stdio.h>
#include "sfr_r813.h"
#include "timer.h"
#include "lcdku.h"
#include "serial.h"


//#define tgps          p1_5
#define resetgps  p0_3

//#define tujuan  "08153036301,"
#define tujuan  "085664013027,"
#define senter  "62855000000,"
#define sinyal    "$GPRMC"


unsigned char b,a,i,j,c,ib,as,tsms;
unsigned char pdu[72],string[72],d[100],simpan[30];
unsigned int jl,il,adc;
long x,y,itimer;

void initADC(void)                              //rutin inisialisasi adc
{
adcon0 = 0x01;    //prescaler fAD/2 (invalid), A/D conv disabled, P0
group selected, oneshot mode, pin AN1 selected
adcon1 = 0x38; //0x38;  //Vref  connected, freq select at fAD valid, 10
bit mode
adcon2 = 0x00;    //0;//without sample and hold
//adst = 1;       //start conversion
}

unsigned char kon(unsigned char n)
  {
   if (n <10) return(n+0x30);
   else return(n+0x37);

  }

void isisenter()
{
i=0;
while(senter[i]!=',') i++;
b=1+(i+(i%2))/2;
sua=(b/0x10);sua=kon(sua);kirim_serial(sua);
sua=(b%0x10);sua=kon(sua);kirim_serial(sua);
sua='9';kirim_serial('9');sua='1';kirim_serial('1');
```

```
if(i%2==1)
  {
  ib=0;
while(b!=0)
    {
   ib++;
   if (i==ib) {b=0;sua='F';kirim_serial(sua);}
   else {sua=senter[ib];kirim_serial(sua);}
   ib--;
   sua=senter[ib];kirim_serial(sua);
   ib++;ib++;
    }
  }
else
  {
  ib=0;
while(b!=0)
    {
   if(i==ib) b=0;
   ib++;
   if (b==0) ;
   else {sua=senter[ib];kirim_serial(sua);}
   ib--;
   if (b==0) ;
   else {sua=senter[ib];kirim_serial(sua);}
   ib++;ib++;
    }
  }
}
void isitujuan()
{
i=0;
while(tujuan[i]!=',') i++;
b=i;
sua=(b/0x10);sua=kon(sua);kirim_serial(sua);
sua=(b%0x10);sua=kon(sua);kirim_serial(sua);
sua='8';kirim_serial('8');sua='1';kirim_serial('1');
if(i%2==1)
  {
  ib=0;
while(b!=0)
    {
   ib++;
   if (i==ib) {b=0;sua='F';kirim_serial(sua);}
   else {sua=tujuan[ib];kirim_serial(sua);}
   ib--;
   sua=tujuan[ib];kirim_serial(sua);
   ib++;ib++;
    }
  }
else
  {
  ib=0;
while(b!=0)
    {
   if(i==ib) b=0;
   ib++;
```

```
   if (b==0) ;
   else {sua=tujuan[ib];kirim_serial(sua);}
   ib--;
   if (b==0) ;
   else {sua=tujuan[ib];kirim_serial(sua);}
   ib++;ib++;
    }
  }

}


void isispasi()
{
for(i=0;i<72;i++)
string[i]=' ';
string[71]='#';
}

void ubahkepdu()
{
i=0;j=0;b=1;
while(b!=0)
  {
b=string[j+1];
if(b=='#') b=0;
pdu[i]=(string[j]>>(i%7))+(b<<(7-(i%7)));

sua=pdu[i]/0x10;sua=kon(sua);kirim_serial(sua);//dataout();
sua=pdu[i]%0x10;sua=kon(sua);kirim_serial(sua);//dataout();

i++;
if(((j+1)-(j+1)/8)%7==0) j=j+2;
else j++;
  }
}
void detek()
{
      a=0;c=0;
      as=jan[a];
      while (a<5)
      {
            while (as==jan[a])
            {
            as=GetChar();
            }
            if (as==jan[a+1])
            a++;
            else
            {
            a=0;
            as=jan[a];
            }
      }
      b=0;
      while (b<3)
      {
```

```
        while (as!=',')
        {
                as=GetChar();
        }
        as='l';
        b++;
}
simpan[c]='D';
c++;
while (as!=',')
{
        as='l';
        while (as=='l')
        {
                as=GetChar();
        }
        if (as=='.')
        simpan[c]='A';
        else
        simpan[c]=as;
        c++;

}
simpan[c-1]='B';
b=0;
while (b<2)
{
        while (as!=',')
        {
                as=GetChar();
        }
        as='l';
        b++;
}
while (as!=',')
{
        as='l';
        while (as=='l')
        {
                as=GetChar();
        }
        if (as=='.')
        simpan[c]='A';
        else
        simpan[c]=as;
        c++;
}
        simpan[c-1]='C';
b=0;
while (b<2)
{
        while (as!=',')
        {
                as=GetChar();
        }
        as='l';
        b++;
```

```
        }
        while (as!=',')
        {
                as='l';
                while (as=='l')
                {
                        as=GetChar();
                }
                if (as=='.')
                simpan[c]='A';
                else
                simpan[c]=as;
                c++;
        }
        simpan[c]=',';
        a=0;c--;
}
void Send_SMS(unsigned char *CommandSMS, unsigned char *DataPDU)
{
        //send command SMS
        Send_Text(CommandSMS);
        delay(20)           ;
        enter()                         ;
        delay(100)          ;

        //send DataPDU
        Send_Text(DataPDU);
        delay(20)           ;
      kirim_serial(0x1A)          ;
        delay(500)          ;

        return;
}
void sms()
{
        //send command SMS
        Send_Text("at+cmgc=54");
        delay(20)           ;
        enter()                         ;
        delay(100)          ;

        //send DataPDU
        //Send_Text("07912658050000F001000B808065630851F2000014EAB29B5C6687
41EBB43C0D A297E5E2FA3A0C");
        isisenter();
        Send_Text("0100");
        isitujuan();
        Send_Text("000047");
        //Send_Text("000014EAB29B5C668741EBB43C0DA297E5E2FA3A0C");
        ubahkepdu();

        delay(20)           ;
      kirim_serial(0x1a)          ;
        delay(500)          ;

        return;
}
```

```
void cetakpdu(unsigned char *text)
{
while(*text)
      {
      string[j]=(*text++);
      j++;
      }
}
void cetaksms(unsigned char *text)
{
isispasi();j=0;
while(*text)
      {
      string[j]=(*text++);
      j++;
      }
sms();
}
void deteksms()
{
a=0;
      as='l';
      while (!(as=='O'))
//while(a<50)
      {
            as='l';
            while (as=='l')
            {
            as=getsms();
        }
            d[a]=as;
            a++;
      }
      if(a>20) tsms=1;
      else ;
}


void posisi()
{
unsigned char j1,j2,j3,j4,j5,j6,j7,j8,j9;
//cetak(1,1,"detek posisi    ");
detek();
pos(1,1);
a=0;
while(simpan[a]!=',')
{asu=simpan[a];dataout(sua);a++;}
pos(2,1);
a=11;
while(simpan[a]!=',')
{sua=simpan[a];dataout(sua);a++;}
delay(2000);
      {isispasi();j=0;
      cetakpdu("Koordinat ");
      string[j]=simpan[1];j++;
      string[j]=simpan[2];j++;
      string[j]=simpan[3];j++;
```

```
      string[j]=simpan[4];j++;
      string[j]='.';j++;
      string[j]=simpan[6];j++;
      string[j]=simpan[7];j++;
      string[j]=simpan[8];j++;
      string[j]=simpan[9];j++;
      string[j]=' ';j++;
      string[j]=simpan[11];j++;
      string[j]=simpan[12];j++;
      string[j]=simpan[13];j++;
      string[j]=simpan[14];j++;
      string[j]=simpan[15];j++;
      string[j]='.';j++;
      string[j]=simpan[17];j++;
      string[j]=simpan[18];j++;
      string[j]=simpan[19];j++;
      string[j]=simpan[20];j++;
      sms();
      }
delay(3000);
}


#pragma INTERRUPT rx
void rx(void)
{
while(ri_u1c1 == 0);
      ri_u1c1 =0;

      u1c1 |= 1;
      }

#pragma INTERRUPT rx0
void rx0(void)
{
while(ri_u0c1 == 0);
      ri_u0c1 =0;
      //txept_u1c0 = 0;
      //u1rb = data;
      //dataout(u0rb);
      //kirim_serial(u0rb);
      u0c1 |= 1;
      }

#pragma INTERRUPT tmx
void tmx(void)
{
      txund=0;
      itimer++;//kelip++;
      prex = 99;
      tx = 199;

      }

void main(void)
{
/* Inisialisasi Variable*/
```

```
  /* Inisialisasi Awal MK */
  asm("FCLR I");     // Interrupt disable
  prcr = 1;          // Protect off
  cm13 = 1;          // X-in X-out = Clock External
  cm15 = 1;          // XCIN-XCOUT drivecapacity select bit : HIGH
  cm05 = 0;          // X-in on
  cm16 = 0;          // Main clock = No division mode
  cm17 = 0;
  cm06 = 0;          // CM16 and CM17 enable
  asm("nop");
  asm("nop");
  asm("nop");
  asm("nop");
  ocd2 = 0;          // Main clock change (x-tal)
  prcr = 0;          // Protect on
  //s1ric=2;s0ric=0;txic=1;
  txic=1;


  /* p1 sebagai keluaran dan p0 sebagai masukan */
  pd1=0x80;
  pd1_4=0;
  //prc2=1;
  pd0=0;
  prc2=1;pd0_5=1;
  prc2=1;pd0_4=1;
  //pd0=0x7f;pd4_5=1;
  pd3=0xff;
  //pd3_0=0;
  pd3_7=0;

  asm("FSET I");
  //pd1=0x0f;
  /* Proses memasukkan dan mengeluarkan data */
  //asm("FCLR I");
  initlcd();
  init_timer_X();txs = 1;//mesin=1;
initSerial();
  initADC();
  cetak(1,1,"itn");
  delay(1000);
  //atcmg("atd0341579799;");
  cetak(1,1,"atik");
  delay(2000);

  if(ri_u0c1 == 0){
  while(ri_u0c1 == 0)
  {resetgps=1;delay(500);resetgps=0;delay(1000);resetgps=1;}
  ri_u0c1 =0;
  }
  else ;

  cetak(1,1,"atik");
  delay(1000);

  itimer=0;
  //detek();a=0;
```

```
        //while(simpan[a]!=',')
        //{sua=simpan[a];dataout(sua);a++;}

        //07912658050000F0040C912658368610250000704092618044820141
        /*
        atcmg("at+cmgl=1");
        deteksms();
        tampilsms();
        tampilnomor();
        */
        while (1) ////////////////////////////////////////////////////
        {

        if(itimer>120000) {posisi();itimer=0;}


        }

}
```

Software Pada PC

```
unit Unit1;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, Text2pdu, StdCtrls, Buttons, CPort, ExtCtrls, ComCtrls, jpeg,
  DB, DBTables, Mask, DBCtrls, FMTBcd, DBXpress, SqlExpr, RpDefine, RpCon,
  RpConDS, DBClient, Provider, Grids, DBGrids,DateUtils, Menus;

type
  TForm1 = class(TForm)
    ComPort1: TComPort;
    BHP: TSpeedButton;
    BSMS: TButton;
    memo1: TMemo;
    Timer3: TTimer;
    LSMS: TLabel;
    Text2pdu1: TText2pdu;
    DBGrid1: TDBGrid;
    DBGrid2: TDBGrid;
    DSMember: TDataSource;
    DSPosisi: TDataSource;
    TbMember: TTable;
    TbSlave: TTable;
    DBNavigator1: TDBNavigator;
    DBNavigator2: TDBNavigator;
    TbPosisi: TTable;
    MainMenu1: TMainMenu;
    Lihat1: TMenuItem;
    Peta1: TMenuItem;
    Print1: TMenuItem;
    Laporan1: TMenuItem;
    BSPosisi: TButton;
    Memo2: TMemo;
    Label1: TLabel;
    Label2: TLabel;
    Button1: TButton;
    Button2: TButton;
    Edit1: TEdit;
    Label3: TLabel;
    procedure BHPClick(Sender: TObject);
    procedure ComPort1RxChar(Sender: TObject; Count: Integer);
    procedure KirimSms(No:String ;Isi:String );
```

```
    procedure IsiSms(Isi:String);
    procedure Timer3Timer(Sender: TObject);
    procedure PecahPosisi(body:String;pengirim:String;waktu:TDateTime);
    procedure FormCreate(Sender: TObject);
    procedure Peta1Click(Sender: TObject);
    procedure BSPosisiClick(Sender: TObject);
    function PilDaerah(L:string; B:string): string;
    function Derajat2toDesimal(S:string): longint;
    procedure BSMSClick(Sender: TObject);
    procedure Button1Click(Sender: TObject);
    procedure Button2Click(Sender: TObject);
  private
    { Private declarations }
    function baliktext(text:string):string;
    function HexToInt(s: string): Longword;
  public
    { Public declarations }
  end;
Const CRLF = #13+#10;
      CTRZ = #26;
      JarakL2=20.7483;
      JarakB2=21.906;
      JarakL3=12.513;
      JarakB3=12.6396;

var
  Form1: TForm1;
  pducmgc,data_sms,no,sms_del: string;
  pdusms: TPDU;
  dtsms,sms: tsms;
  no_sms_del : integer;

implementation

uses Math, Unit2, Unit3;

{$R *.dfm}

procedure TForm1.KirimSms(No:String; Isi:String);
var Nonya, Isinya: string;
begin
  Nonya:='0'+Copy(No,3,length(No)-1);
  Isinya:=Isi;
  pdusms:=Text2pdu1.texttosms('',Nonya,Isinya);
  ComPort1.WriteStr('AT+CMGC='+IntToStr(pdusms.Panjang)+CRLF);
  Sleep(500);
```

```
  pducmgc:='00'+Copy(pdusms.PDU,5,length(pdusms.PDU));
  ComPort1.WriteStr(pducmgc+CTRZ);
 end;

 function TForm1.PilDaerah(L:string; B:string): string;
 var kondisi:Boolean;
 begin
 kondisi:=false;
  while not ((Form3.TbJalanSlave.Eof) or kondisi) do
  begin
  end;
  repeat
  Form3.TbJalanSlave.First;
    repeat

    until (Form3.TbJalanSlave.Eof) or kondisi;
  until (Form3.TbAreaSlave.Eof) or kondisi;
 end;

 function TForm1.Derajat2toDesimal(S:string): longint;
 begin
 result:=round(((strtoint(copy(s,length(s)-3,4))+
              (strtoint(copy(s,length(s)-6,2))*10000)+
              (strtoint(copy(s,1,length(s)-7))))));
 end;

 function TForm1.baliktext(text:string):string;
 var
  i,len:integer;
  res : string;
  ins : string;
 begin
  if(text <> '') then
  begin
  len := StrLen(pChar(text));
  ins := text;
  res := '';
  for i:=1 to len do
  if(Odd(i)) then
  begin
     res := res + ins[i+1] + ins[i];
  end;
  result := res;
  end
  else result := '';
 end;
```

```
function TForm1.HexToInt(s: string): Longword;
var
  b: Byte;
  c: Char;
begin
  Result := 0;
  s := UpperCase(s);
  for b := 1 to Length(s) do
  begin
    Result := Result * 16;
    c := s[b];
    case c of
      '0'..'9': Inc(Result, Ord(c) - Ord('0'));
      'A'..'F': Inc(Result, Ord(c) - Ord('A') + 10);
      else
        raise EConvertError.Create('No Hex-Number');
    end;
  end;
end;


procedure TForm1.IsiSms(Isi:String );
var ins,temp,dt,pengirim,body : string;
    len : integer;
    waktu:TDateTime;
begin
//  ShowMessage(IntToStr(j));
//  j:=j+1;
  ins:=Isi;
  ins := StrLower(pchar(ins));
  temp := ins[1] + ins[2];

  if temp<>'' then len := HexToInt(temp);
  //delete sms center dari pdu
  Delete(ins,1,2*len+2);
  //delete tipe sms
  Delete(ins,1,2);

  temp := ins[1] + ins[2];
//  ShowMessage(temp);
//  ShowMessage();
  if temp<>'' then  len := HexToInt(temp);
  if(Odd(len)) then len := len +1; //tambah buat yang ganjil yaitu di tambah f wkt di kirim
  len := len +2; //tambah buat tipe sender
  pengirim :=copy(ins,3,len);
```

```
  if(Copy(pengirim,1,2)='91') then  Delete(pengirim,1,2);
  pengirim := baliktext(pengirim);
  //cek apakah stelah di balik ada angka f
  if(pengirim[StrLen(pchar(pengirim))]='f') then
 Delete(pengirim,StrLen(pchar(pengirim)),1);

  //delete information sender
  Delete(ins,1,len+2);
  //delete PID(Protocol Identifier) dan DCS(Data Coding Scheme)
  Delete(ins,1,4);

  //tanggal
  dt := copy(ins,1,12);
  dt := baliktext(dt);
  waktu := EncodeDateTime(StrToInt(copy(FormatDateTime('yyyy',now),1,2)
+dt[1]+dt[2]),StrToInt(dt[3]+dt[4]),StrToInt(dt[5]+dt[6]),
            StrToInt(dt[7]+dt[8]),StrToInt(dt[9]+dt[10]),StrToInt(dt[11]+dt[12]),0);
  //delete tanggal
  Delete(ins,1,12);
//  ShowMessage(DateToStr(waktu));
//  ShowMessage(TimeToStr(waktu));
  //delete expire validated
  Delete(ins,1,2);

  temp := copy(ins,1,2);
  len := HexToInt(temp);
  delete(ins,1,2);
  body := ins;
//  ShowMessage(PENGIRIM);
//  Memo3.Clear;
//  Memo3.Text:=body;
//  ShowMessage(body+'A');
  PecahPosisi(body,pengirim,waktu);

end;

procedure TForm1.PecahPosisi(body:String;pengirim:String;waktu:TDateTime);
var lintang,bujur,kec,jam:String;
   p,no,menit,menit_awal,waktu_awal,jam_awal:integer;
begin
TbPosisi.Last;
if TbPosisi.FieldByName('no').AsString='' then no:=1
else no:=TbPosisi.FieldByName('no').AsInteger+1;

menit:=round(24*60/TbMember.FieldByName('paket').AsInteger/10);
waktu_awal:=menit*9;
```

```
if (copy(body,length(body),1)='f') and  (copy(body,1,1)='d') then
begin
 if length(body)<50 then
   begin
    p:=Pos('a',body);
    lintang:=copy(body,2,p-2);
    Delete(body,1,p);
    p:=Pos('b',body);
    lintang:=lintang+'.'+copy(body,1,p-1);
    Delete(body,1,p);
    p:=Pos('a',body);
    bujur:=copy(body,1,p-1);
    Delete(body,1,p);
    p:=Pos('c',body);
    bujur:=bujur+'.'+copy(body,1,p-1);
    Delete(body,1,p);
    p:=Pos('a',body);
    kec:=copy(body,1,p-1);
    Delete(body,1,p);
    p:=Pos('d',body);
    kec:=kec+'.'+copy(body,1,p-1);

    TbMember.FindKey([pengirim]);
    TbPosisi.Append;
    TbPosisi.FieldByName('No').AsInteger:=no;
    TbPosisi.FieldByName('Mobile').AsString:=pengirim;
    TbPosisi.FieldByName('Lintang').AsString:=lintang;
    TbPosisi.FieldByName('Bujur').AsString:=bujur;
    TbPosisi.FieldByName('Tanggal').AsString:=DateToStr(waktu);
    TbPosisi.FieldByName('jam').AsString:=TimeToStr(waktu);
    TbPosisi.Post;
  end
 else
 begin
 repeat
    p:=Pos('a',body);
    lintang:=copy(body,2,p-2);
    Delete(body,1,p);
    p:=Pos('b',body);
    lintang:=lintang+'.'+copy(body,1,p-1);
    Delete(body,1,p);
    p:=Pos('a',body);
    bujur:=copy(body,1,p-1);
    Delete(body,1,p);
    p:=Pos('c',body);
```

```
    bujur:=bujur+'.'+copy(body,1,p-1);
    Delete(body,1,p);
    p:=Pos('a',body);
    kec:=copy(body,1,p-1);
    Delete(body,1,p);
    p:=Pos('d',body);
    kec:=kec+'.'+copy(body,1,p-1);
    Delete(body,1,p-1);

    jam_awal:=trunc(waktu_awal/60);
    menit_awal:=waktu_awal mod 60;
    jam:=TimeToStr(waktu-
StrToTime(IntToStr(jam_awal)+':'+IntToStr(menit_awal)+':0'));

    TbMember.FindKey([pengirim]);
    TbPosisi.Append;
    TbPosisi.FieldByName('No').AsInteger:=no;
    TbPosisi.FieldByName('Mobile').AsString:=pengirim;
    TbPosisi.FieldByName('Lintang').AsString:=lintang;
    TbPosisi.FieldByName('Bujur').AsString:=bujur;
    TbPosisi.FieldByName('Tanggal').AsString:=DateToStr(waktu);
    TbPosisi.FieldByName('jam').AsString:=jam;
    TbPosisi.FieldByName('kecepatan').AsString:=kec;
    TbPosisi.FieldByName('daerah').AsString:='';
    TbPosisi.Post;

    waktu_awal:=waktu_awal-menit;
    no:=no+1;
  until copy(body,2,1)='f';
  end;
end;
TbSlave.Close;
TbSlave.Open;
end;

procedure TForm1.BHPClick(Sender: TObject);
begin
  ComPort1.ShowSetupDialog;
end;

procedure TForm1.ComPort1RxChar(Sender: TObject; Count: Integer);
var datasl,isi_smssl,sms_delsl:TStringList;
  p,no:Integer;
  isi_sms,data,data_ok:String;

  kondisi:Boolean;
```

```
begin

ComPort1.ReadStr(data,count);
data_sms:=data_sms+data;
Memo1.Text:=data_sms;

 if copy(data_sms,length(data_sms)-3,2)='OK' then
 begin
    repeat  begin

      datasl:=TStringList.Create;
      datasl.Clear;
      datasl.Delimiter:=#10;
      datasl.DelimitedText:=data_sms;

      isi_sms:=datasl[0];
      Memo2.Text:=isi_sms;
//      ShowMessage('1');
      if copy(data_sms,1,1)='+' then
       begin
       sms_delsl:=TStringList.Create;
       sms_delsl.Clear;
       sms_delsl.Delimiter:=',';
       sms_delsl.DelimitedText:=copy(data_sms,8,4);

       sms_del:=sms_delsl[0];
//       ShowMessage('2');
//       StatusBar1.SimpleText:='data del';
//       ComPort1.WriteStr('at+cmgd='+sms_del+CRLF);
       Sleep(500);

       end;
      if Length(Memo2.text)>10 then
       begin
       sms := Text2pdu1.smstotext(isi_sms);
//       StatusBar1.SimpleText:='PDU kan';
//       Edit1.Text:=sms.Body;
//       Edit2.Text:=sms.Sender;
//       Edit3.Text:=IntToStr(length(sms.Body));
//       Edit4.Text:=copy(sms.Body,1,9);
//       ShowMessage(DateTimeToStr(sms.Tanggal));
      if TbPosisi.RecordCount=0 then no:=1
      else
       begin
       TbPosisi.Last;
       no:=TbPosisi.FieldByName('no').AsInteger+1;
```

```
      end;
     isi_smssl:=TStringList.Create;
     isi_smssl.Clear;
     isi_smssl.Delimiter:=#10;
     isi_smssl.DelimitedText:=sms.Body;
     TbPosisi.Append;
     TbPosisi.FieldByName('no').AsInteger:=no;
     TbPosisi.FieldByName('mobile').AsString:=sms.Sender;
     TbPosisi.FieldByName('lintang').AsString:=isi_smssl[1];
     TbPosisi.FieldByName('bujur').AsString:=isi_smssl[2];
     TbPosisi.FieldByName('tanggal').AsString:=DateToStr(sms.Tanggal);
     TbPosisi.FieldByName('jam').AsString:=TimeToStr(sms.Tanggal);
     TbPosisi.Post;
     TbSlave.Refresh;
      p:=Pos(#10,Memo1.Text);
      data_sms:=Memo1.Text;
      Delete(data_sms,1,p);
      Memo1.Text:=data_sms;
      end
     else
      begin
      p:=Pos(#10,Memo1.Text);
      data_sms:=Memo1.Text;
      Delete(data_sms,1,p);
      Memo1.Text:=data_sms;
      end;

     end until Memo2.Text='OK';
  end;
end;

procedure TForm1.Timer3Timer(Sender: TObject);
begin
  BSMS.Click;
end;

procedure TForm1.FormCreate(Sender: TObject);
begin
Form1.Left:=0;
Form1.Top:=0;
end;

procedure TForm1.Peta1Click(Sender: TObject);
begin
Form2.Awal('7;13;26;98','112;39;47;35');
Form2.Show;
```

```
Form1.Visible:=false;
end;

procedure TForm1.BSPosisiClick(Sender: TObject);
begin
Form2.Awal('5;54;0;0','101;0;0;0');
Form2.Show;
Form2.Gambar(1,TbSlave.fieldbyname('lintang').AsString,TbSlave.fieldbyname('bujur').
AsString);
Form1.Visible:=false;
end;

procedure TForm1.BSMSClick(Sender: TObject);
begin
if LSMS.Caption='SMS BARU' then
  begin
  LSMS.Caption:='SMS LAMA';
  memo1.Clear;
  memo2.Clear;
  data_sms:='';
// StatusBar1.SimpleText:='Membaca SMS';
  sleep(500);
  ComPort1.ClearBuffer(True,False);
  ComPort1.WriteStr('AT+CMGL=0'+#13+#10);
  end
else
  begin
  LSMS.Caption:='SMS BARU';
  memo1.Clear;
  memo2.Clear;
  data_sms:='';
// StatusBar1.SimpleText:='Membaca SMS';
  sleep(500);
  ComPort1.ClearBuffer(True,False);
  ComPort1.WriteStr('AT+CMGL=1'+#13+#10);
  end;

end;

procedure TForm1.Button1Click(Sender: TObject);
begin
IF TbPosisi.RecordCount>0 THEN
begin
  repeat
  TbPosisi.Delete;
  until TbPosisi.RecordCount=0;
```

```
TbSlave.Refresh;
end;
end;

procedure TForm1.Button2Click(Sender: TObject);
begin

IF Button2.Caption='START' THEN
 BEGIN
 Timer3.Interval:=StrToInt(Edit1.Text);
 Timer3.Enabled:=TRUE;
 Button2.Caption:='STOP';
 END
ELSE
  BEGIN
   Timer3.Enabled:=FALSE;
  Button2.Caption:='START';
  END;
end;

end.
```

```
unit Unit2;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, jpeg, ExtCtrls, StdCtrls;

type
  TForm2 = class(TForm)
    ScrollBox1: TScrollBox;
    ImgPeta: TImage;
    Lkoor: TLabel;
    LKoorDerajat2: TLabel;
    LKoorDerajat3: TLabel;
    procedure ImgPetaMouseMove(Sender: TObject; Shift: TShiftState; X,
      Y: Integer);
    procedure FormCreate(Sender: TObject);
    procedure FormClose(Sender: TObject; var Action: TCloseAction);
    function PixeltoDerajat(Awal:longint; Posisi:Integer; konsJenis:real;
konsDerajat:integer):string;
    function Derajat3toDesimal(s: string; konstanta:integer): Longint;
    function Derajat2toPixel(s:String; jenis:string):longint;
    procedure Awal(L: string; B:string);
    procedure Gambar(no:integer;y:string;x:string);
    procedure ImgPetaClick(Sender: TObject);

  private
    { Private declarations }
  public
    { Public declarations }
    Cursor: TPoint;
  end;

Const
//     JarakL3=809;
//     JarakB3=1000.5;
//     JarakL2=1348.5;
//     JarakB2=1700.5;
     JarakL3=885;
     JarakB3=1234;
     JarakL2=1475;
     JarakB2=2057;
     K2=100;
```

```
    K3=60;

var
  Form2: TForm2;
  KoorY,KoorX,iPeta,jPeta:integer;
  LAwal3,BAwal3,LAwal2,BAwal2:longInt;
  gbrmbl: array [0..10] of TShape;
  GbrPeta: array [0..1000] of TShape;

implementation

uses Unit1, Unit3;

{$R *.dfm}

procedure TForm2.Awal(L:String; B:String);
var i:integer;
begin
LAwal3:=Derajat3toDesimal(L,K3);
BAwal3:=Derajat3toDesimal(B,K3);
LAwal2:=Derajat3toDesimal(L,K2);
BAwal2:=Derajat3toDesimal(B,K2);
      for i:=0 to 4 do begin
      gbrmbl[i]:=TShape.Create(self);
      gbrmbl[i].Parent:=ScrollBox1;
      gbrmbl[i].Width:=20;
      gbrmbl[i].Height:=20;
      gbrmbl[i].Shape:=stCircle;
      gbrmbl[i].Brush.Color:=clBlue;
      gbrmbl[i].Hint:='KERETA'+inttostr(i);
      gbrmbl[i].ShowHint:=True;
      gbrmbl[i].Visible:=False;
      end;
end;

procedure TForm2.Gambar(no:integer;y:string;x:string);
begin
      ScrollBox1.HorzScrollBar.Position:=Derajat2toPixel(x,'x')-410;
      ScrollBox1.VertScrollBar.Position:=Derajat2toPixel(y,'y')-310;
      gbrmbl[no].Left:=400;
      gbrmbl[no].Top:=300;
      gbrmbl[no].Brush.Color:=clLime;
      gbrmbl[no].Show;
end;
```

```
function TForm2.Derajat3toDesimal(s:String; konstanta:integer):longint;
var sl:TStringList;
   a1,a2,a3,a4:integer;
Begin
      sl:=TStringList.Create;
      sl.Clear;
      sl.Delimiter:=';';
      sl.DelimitedText:=s;

      a1:=StrToInt(sl[0]);
      a2:=StrToInt(sl[1]);
      a3:=StrToInt(sl[2]);
      a4:=StrToInt(sl[3]);


result:=(a1*6000*konstanta)+round(a2*6000*konstanta/60)+round(a3*100*konstanta/60
)+a4
end;

function TForm2.Derajat2toPixel(s:String; jenis:string):integer;
Begin
if jenis='y' then result:=round(((strtoint(copy(s,length(s)-3,4))+
            (strtoint(copy(s,length(s)-6,2))*10000)+
            (strtoint(copy(s,1,length(s)-7))*600000)-LAwal2))/JarakL2)
else          result:=round(((strtoint(copy(s,length(s)-3,4))+
            (strtoint(copy(s,length(s)-6,2))*10000)+
            (strtoint(copy(s,1,length(s)-7))*600000)-BAwal2))/JarakB2);
end;

function TForm2.PixeltoDerajat(Awal:longint; Posisi:Integer; konsJenis:real;
konsDerajat:integer):string;
var Derajat:longint;
   a1,a2,a3,a4:integer;
   a1s,a2s,a3s,a4s:string;
begin
      Derajat:=Awal+round(Posisi*konsJenis);

      a1:=trunc(Derajat/(6000*konsDerajat));
      if Derajat<0 then Derajat:=-1*Derajat;
      a2:=trunc((round(Derajat) mod (6000*konsDerajat))*60/(6000*konsDerajat));
      a3:=trunc((round(Derajat) mod (100*konsDerajat))/100);
      a4:=round(Derajat) mod 100;

      a2s:=IntToStr(a2);
      a3s:=IntToStr(a3);
      a4s:=IntToStr(a4);
```

```
    if length(a2s)=1 then a2s:='0'+IntToStr(a2);
    if length(a3s)=1 then a3s:='0'+IntToStr(a3);
    if length(a4s)=1 then a4s:='0'+IntToStr(a4);

    if konsDerajat=100 then result:=intToStr(a1)+''+a2s+'.'+a3s+a4s
    else result:=intToStr(a1)+';'+a2s+';'+a3s+';'+a4s;
end;

procedure TForm2.ImgPetaMouseMove(Sender: TObject; Shift: TShiftState; X,Y:
Integer);
begin
    Cursor:= point(X,Y);
    KoorY:=Y;
    KoorX:=X;
    Lkoor.Caption:=IntToStr(Y)+','+IntToStr(X);

LKoorDerajat3.Caption:=PixeltoDerajat(LAwal3,y,JarakL3,K3)+','+PixeltoDerajat(BAw
al3,x,JarakB3,K3);

LKoorDerajat2.Caption:=PixeltoDerajat(LAwal2,y,JarakL2,K2)+','+PixeltoDerajat(BAw
al2,x,JarakB2,K2);
end;

procedure TForm2.FormCreate(Sender: TObject);
begin
ImgPeta.Left:=0;
ImgPeta.Top:=0;
Form2.Left:=0;
Form2.Top:=0;
ScrollBox1.Left:=0;
ScrollBox1.Top:=0;
end;

procedure TForm2.FormClose(Sender: TObject; var Action: TCloseAction);
begin
Form1.Visible:=true;
end;

procedure TForm2.ImgPetaClick(Sender: TObject);
var sl:Tstringlist;
begin
if Form3.CBArea.Checked or Form3.CBJalan.Checked then
 begin
    sl:=TStringList.Create;
    sl.Clear;
    sl.Delimiter:=',';
```

```
    sl.DelimitedText:=LKoorDerajat2.Caption;

    Form3.ELintang.Text:=sl[0];
    Form3.EBujur.Text:=sl[1];

    sl.Clear;
    sl.Delimiter:=',';
    sl.DelimitedText:=Lkoor.Caption;

    Form3.Ey.Text:=sl[0];
    Form3.Ex.Text:=sl[1];
    Form3.EKota.Text:=Form3.TbKota.fieldbyname('kota').AsString;

    GbrPeta[iPeta].Left:=KoorX-ScrollBox1.HorzScrollBar.Position-round(jPeta/2);
    GbrPeta[iPeta].Top:=KoorY-ScrollBox1.VertScrollBar.Position-round(jPeta/2);
    GbrPeta[iPeta].show;
  end;
end;

end.
```

LAMPIRAN B

(DATA SHEET)

GPS 18 TECHNICAL SPECIFICATIONS

GARMIN®

Garmin International, Inc.
1200 E. 151st Street
Olathe, KS 66062 USA

190-00307-00, Revision D

June 2005

4.2 TRANSMITTED NMEA 0183 SENTENCES

The subsequent paragraphs define the sentences that can be transmitted by the GPS 18 PC and LVC.

4.2.1 Sentence Transmission Rate

Sentences are transmitted with respect to the user selected baud rate.

The GPS sensor will transmit each sentence (except where noted in particular transmitted sentence descriptions) at a periodic rate based on the user selected baud rate and user selected output sentences. The GPS sensor will transmit the selected sentences contiguously. The length of the transmission can be determined by the following equation and Tables 2 and 3:

$$\text{length of transmission} = \frac{\text{total characters to be transmitted}}{\text{characters transmitted per second}}$$

Sentence	Output by Default?	Maximum Characters
GPRMC	✓	74
GPGGA	✓	82
GPGSA	✓	66
GPGSV	✓ (PC and LVC only)	70
PGRME	✓ (PC and LVC only)	35
GPGLL		44
GPVTG	✓ (18-5Hz only)	42
PGRMV		32
PGRMF		82
PGRMB	✓ (PC and LVC only)	40
PGRMT	Once per minute	50

Table 2: NMEA 0183 Output Sentence Order and Size

Baud	Characters per Second
300	30
600	60
1200	120
2400	240
4800	480
9600	960
19200	1920
38400	3840

Table 3: Characters per Second for Available Baud Rates

The maximum number of fields allowed in a single sentence is 82 characters including delimiters. Values in the table include the sentence start delimiter character "$" and the termination delimiter <CR><LF>. For the GPS 18 PC and LVC, the factory set defaults result in a once per second transmission at the NMEA 0183 specification transmission rate of 4800 baud. For the GPS 18-5Hz, the factory set defaults will result in a five times per second transmission at 19200 baud.

For the GPS 18 LVC: Regardless of the selected baud rate, the information transmitted by the GPS sensor is referenced to the one-pulse-per-second output pulse immediately preceding the GPRMC sentence, or whichever sentence is output first in the burst (see Table 2 above).

For the GPS 18-5Hz: Regardless of the selected baud rate, the information transmitted by the GPS sensor is referenced to the preceding five times per second output pulse.

4.2.7 Recommended Minimum Specific GPS/TRANSIT Data (RMC)

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

<1>	UTC time of position fix, hhmmss format for GPS 18 PC/LVC; hhmmss.s format for GPS 18-5Hz
<2>	Status, A = Valid position, V = NAV receiver warning
<3>	Latitude, ddmm.mmm format for GPS 18 PC/LVC; ddmm.mmmmm format for GPS 18-5Hz (leading zeros must be transmitted)
<4>	Latitude hemisphere, N or S
<5>	Longitude, ddmm.mmm format for GPS 18 PC/LVC; ddmm.mmmmm format for GPS 18-5Hz (leading zeros must be transmitted)
<6>	Longitude hemisphere, E or W
<7>	Speed over ground, GPS 18 PC and LVC: 000.0 to 999.9 knots, GPS 18-5Hz: 000.00 to 999.99 knots (leading zeros will be transmitted)
<8>	Course over ground, 000.0 to 359.9 degrees, true (leading zeros will be transmitted)
<9>	UTC date of position fix, ddmmyy format
<10>	Magnetic variation, 000.0 to 180.0 degrees (leading zeros will be transmitted)
<11>	Magnetic variation direction, E or W (westerly variation adds to true course)
<12>	Mode indicator (only output if NMEA 0183 version 2.30 active), A = Autonomous, D = Differential, E = Estimated, N = Data not valid

4.2.8 Track Made Good and Ground Speed (VTG)

$GPVTG,<1>,T,<2>,M,<3>,N,<4>,K,<5>*hh<CR><LF>

<1>	True course over ground, GPS 18 PC and LVC: 000 to 359 degrees, GPS 18-5Hz: 000.0 to 359.0 degrees(leading zeros will be transmitted)
<2>	Magnetic course over ground, 000 to 359 degrees, GPS 18-5Hz: 000.0 to 359.0 degrees(leading zeros will be transmitted)
<3>	Speed over ground, GPS 18 PC and LVC: 000.0 to 999.9 knots, GPS 18-5Hz: 000.00 to 999.99 knots (leading zeros will be transmitted)
<4>	Speed over ground, GPS 18 PC and LVC: 0000.0 to 1851.8 kilometers per hour, GPS 18-5Hz: 0000.00 to 1851.89 (leading zeros will be transmitted)
<5>	Mode indicator (only output if NMEA 0183 version 2.30 active), A = Autonomous, D = Differential, E = Estimated, N = Data not valid

4.2.9 Geographic Position (GLL)

$GPGLL,<1>,<2>,<3>,<4>,<5>,<6>,<7>*hh<CR><LF>

<1>	Latitude, ddmm.mmm format for GPS 18 PC/LVC; ddmm.mmmmm format for GPS 18-5Hz (leading zeros must be transmitted)
<2>	Latitude hemisphere, N or S
<3>	Longitude, ddmm.mmm format for GPS 18 PC/LVC; ddmm.mmmmm format for GPS 18-5Hz (leading zeros must be transmitted)
<4>	Longitude hemisphere, E or W
<5>	UTC time of position fix, hhmmss format for GPS 18 PC/LVC; hhmmss.s format for GPS 18-5Hz
<6>	Status, A = Valid position, V = NAV receiver warning
<7>	Mode indicator (only output if NMEA 0183 version 2.30 active), A = Autonomous, D = Differential (WAAS), E = Estimated, N = Data not valid

19-0175; Rev 3; 5/96

MAXIM

±15kV ESD-Protected, +5V RS-232 Transceivers

MAX202E–MAX213E, MAX232E/MAX241E

General Description

The MAX202E–MAX213E, MAX232E/MAX241E line drivers/receivers are designed for RS-232 and V.28 communications in harsh environments. Each transmitter output and receiver input is protected against ±15kV electrostatic discharge (ESD) shocks, without latchup. The various combinations of features are outlined in the *Selection Guide*. The drivers and receivers for all ten devices meet all EIA/TIA-232E and CCITT V.28 specifications at data rates up to 120kbps, when loaded in accordance with the EIA/TIA-232E specification.

The MAX211E/MAX213E/MAX241E are available in 28-pin SO packages, as well as a 28-pin SSOP that uses 60% less board space. The MAX202E/MAX232E come in 16-pin narrow SO, wide SO, and DIP packages. The MAX203E comes in a 20-pin DIP/SO package, and needs no external charge-pump capacitors. The MAX205E comes in a 24-pin wide DIP package, and also eliminates external charge-pump capacitors. The MAX206E/MAX207E/MAX208E come in 24-pin SO, SSOP, and narrow DIP packages. The MAX232E/MAX241E operate with four 1µF capacitors, while the MAX202E/MAX206E/MAX207E/MAX208E/MAX211E/MAX213E operate with four 0.1µF capacitors, further reducing cost and board space.

Applications

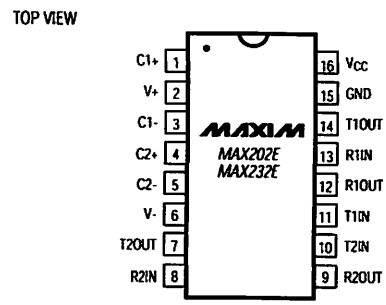
Notebook, Subnotebook, and Palmtop Computers
Battery-Powered Equipment
Hand-Held Equipment

Ordering Information appears at end of data sheet.

Features

- **ESD Protection for RS-232 I/O Pins:**
  ±15kV—Human Body Model
  ±8kV—IEC1000-4-2, Contact Discharge
  ±15kV—IEC1000-4-2, Air-Gap Discharge
- **Latchup Free (unlike bipolar equivalents)**
- **Guaranteed 120kbps Data Rate—LapLink™ Compatible**
- **Guaranteed 3V/µs Min Slew Rate**
- **Operate from a Single +5V Power Supply**

Pin Configurations



DIP/SO

Pin Configurations and Typical Operating Circuits continued at end of data sheet.

Selection Guide

PART	No. of RS-232 DRIVERS	No. of RS-232 RECEIVERS	RECEIVERS ACTIVE IN SHUTDOWN	No. of EXTERNAL CAPACITORS	LOW-POWER SHUTDOWN	TTL THREE-STATE
MAX202E	2	2	0	4 (0.1µF)	No	No
MAX203E	2	2	0	None	No	No
MAX205E	5	5	0	None	Yes	Yes
MAX206E	4	3	0	4 (0.1µF)	Yes	Yes
MAX207E	5	3	0	4 (0.1µF)	No	No
MAX208E	4	4	0	4 (0.1µF)	No	No
MAX211E	4	5	0	4 (0.1µF)	Yes	Yes
MAX213E	4	5	2	4 (0.1µF)	Yes	Yes
MAX232E	2	2	0	4 (1µF)	No	No
MAX241E	4	5	0	4 (1µF)	Yes	Yes

LapLink is a registered trademark of Traveling Software, Inc.

Applications Information

Capacitor Selection

The capacitor type used for C1–C4 is not critical for proper operation. The MAX202E, MAX206–MAX208E, MAX211E, and MAX213E require 0.1µF capacitors, and the MAX232E and MAX241E require 1µF capacitors, although in all cases capacitors up to 10µF can be used without harm. Ceramic, aluminum-electrolytic, or tantalum capacitors are suggested for the 1µF capacitors, and ceramic dielectrics are suggested for the 0.1µF capacitors. When using the minimum recommended capacitor values, make sure the capacitance value does not degrade excessively as the operating temperature varies. If in doubt, use capacitors with a larger (e.g., 2x) nominal value. The capacitors' effective series resistance (ESR), which usually rises at low temperatures, influences the amount of ripple on V+ and V-.

Use larger capacitors (up to 10µF) to reduce the output impedance at V+ and V-. This can be useful when "stealing" power from V+ or from V-. The MAX203E and MAX205E have internal charge-pump capacitors.

Bypass V_{CC} to ground with at least 0.1µF. In applications sensitive to power-supply noise generated by the charge pumps, decouple V_{CC} to ground with a capacitor the same size as (or larger than) the charge-pump capacitors (C1–C4).

V+ and V- as Power Supplies

A small amount of power can be drawn from V+ and V-, although this will reduce both driver output swing and noise margins. Increasing the value of the charge-pump capacitors (up to 10µF) helps maintain performance when power is drawn from V+ or V-.

Driving Multiple Receivers

Each transmitter is designed to drive a single receiver. Transmitters can be paralleled to drive multiple receivers.

Driver Outputs when Exiting Shutdown

The driver outputs display no ringing or undesirable transients as they come out of shutdown.

High Data Rates

These transceivers maintain the RS-232 ±5.0V minimum driver output voltages at data rates of over 120kbps. For data rates above 120kbps, refer to the Transmitter Output Voltage vs. Load Capacitance graphs in the *Typical Operating Characteristics*. Communication at these high rates is easier if the capacitive loads on the transmitters are small; i.e., short cables are best.

Table 2. Summary of EIA/TIA-232E, V.28 Specifications

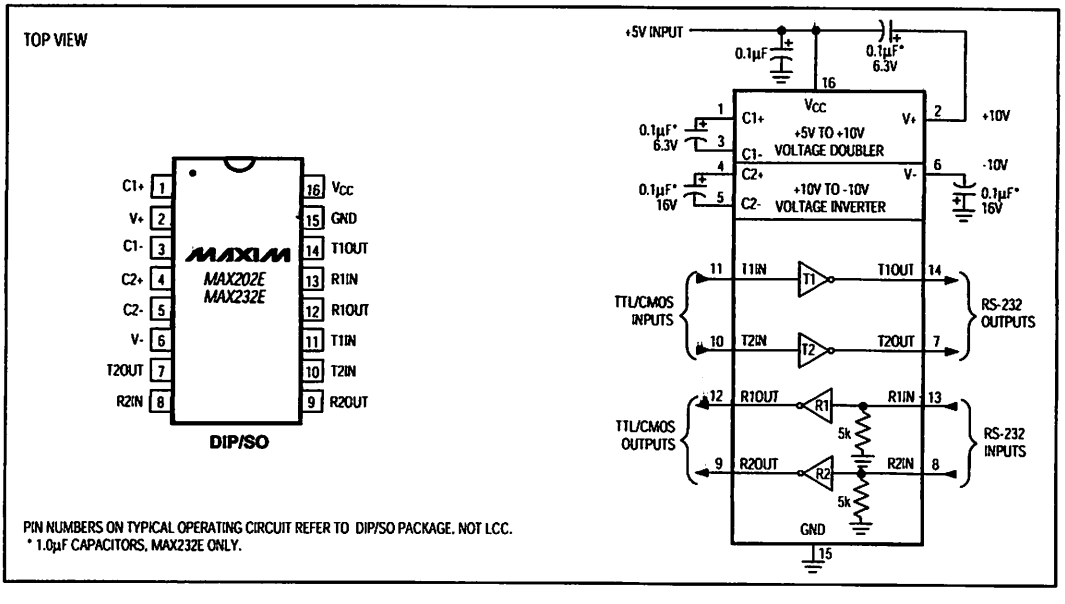
PARAMETER		CONDITIONS	EIA/TIA-232E, V.28 SPECIFICATIONS
Driver Output Voltage	0 Level	3kΩ to 7kΩ load	+5V to +15V
	1 Level	3kΩ to 7kΩ load	-5V to -15V
Driver Output Level, Max		No load	±25V
Data Rate		$3k\Omega \le R_L \le 7k\Omega$, $C_L \le 2500pF$	Up to 20kbps
Receiver Input Voltage	0 Level		+3V to +15V
	1 Level		-3V to -15V
Receiver Input Level			±25V
Instantaneous Slew Rate, Max		$3k\Omega \le R_L \le 7k\Omega$, $C_L \le 2500pF$	30V/µs
Driver Output Short-Circuit Current, Max			100mA
Transition Rate on Driver Output		V.28	1ms or 3% of the period
		EIA/TIA-232E	4% of the period
Driver Output Resistance		$-2V < V_{OUT} < +2V$	300Ω

±15kV ESD-Protected, +5V RS-232 Transceivers

Table 3. DB9 Cable Connections Commonly Used for EIA/TIAE-232E and V.24 Asynchronous Interfaces

PIN	CONNECTION	
1	Received Line Signal Detector (sometimes called Carrier Detect, DCD)	Handshake from DCE
2	Receive Data (RD)	Data from DCE
3	Transmit Data (TD)	Data from DTE
4	Data Terminal Ready	Handshake from DTE
5	Signal Ground	Reference point for signals
6	Data Set Ready (DSR)	Handshake from DCE
7	Request to Send (RTS)	Handshake from DTE
8	Clear to Send (CTS)	Handshake from DCE
9	Ring Indicator	Handshake from DCE

Pin Configurations and Typical Operating Circuits (continued)



PIN NUMBERS ON TYPICAL OPERATING CIRCUIT REFER TO DIP/SO PACKAGE. NOT LCC.
* 1.0µF CAPACITORS, MAX232E ONLY.

B0111-0110

Everywhere you imagine. RENESAS

R8C/13 Group

Hardware Manual

RENESAS 16-BIT SINGLE-CHIP MICROCOMPUTER

M16C FAMILY/R8C/Tiny SERIES

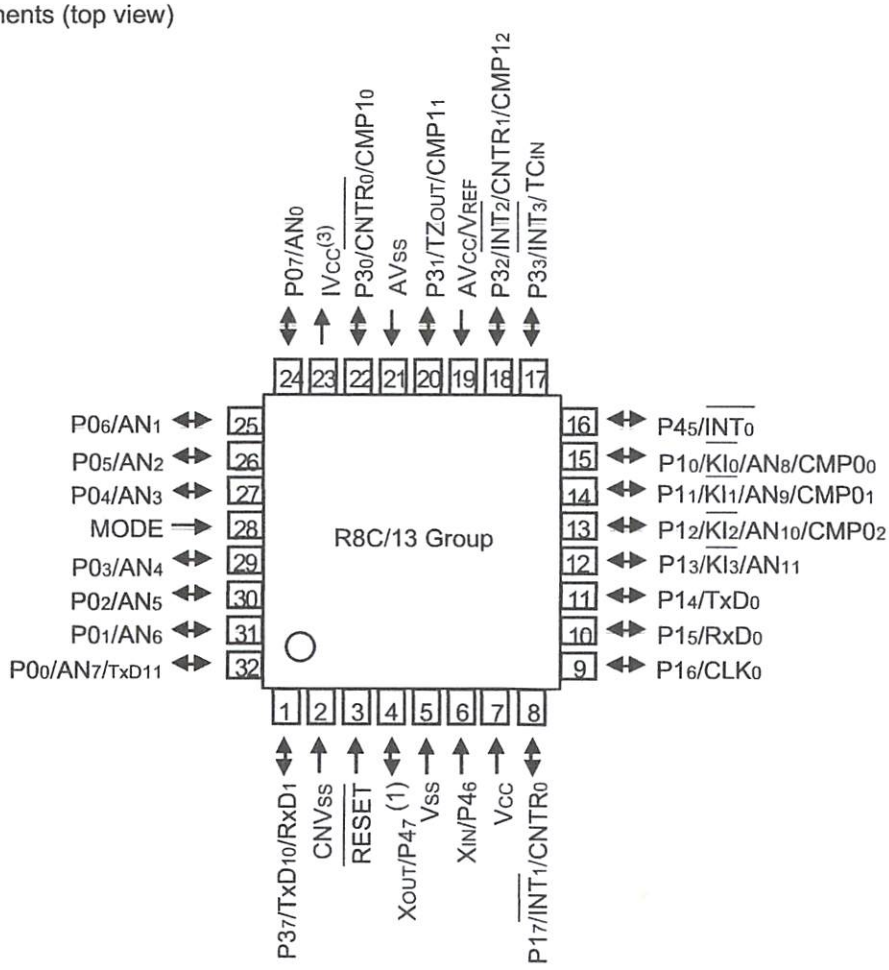
Rev. 1.10
Revision date: Apr 27, 2005

RenesasTechnology
www.renesas.com

.5 Pin Assignments

Figure 1.3 shows the pin configuration (top view).

PIN Assignments (top view)



NOTES:

1. P4₇ functions only as an input port.
2. When using On-chip debugger, do not use P0₀/AN₇/TxD11 and P3₇/TxD10/RxD1 pins.
3. Do not connect IVcc to Vcc.

Package: PLQP0032GB-A (32P6U-A)

Figure 1.3 Pin Assignments (Top View)

Memory

Figure 3.1 is a memory map of this MCU. The address space extends the 1M bytes from address 0000016 to FFFFF16.

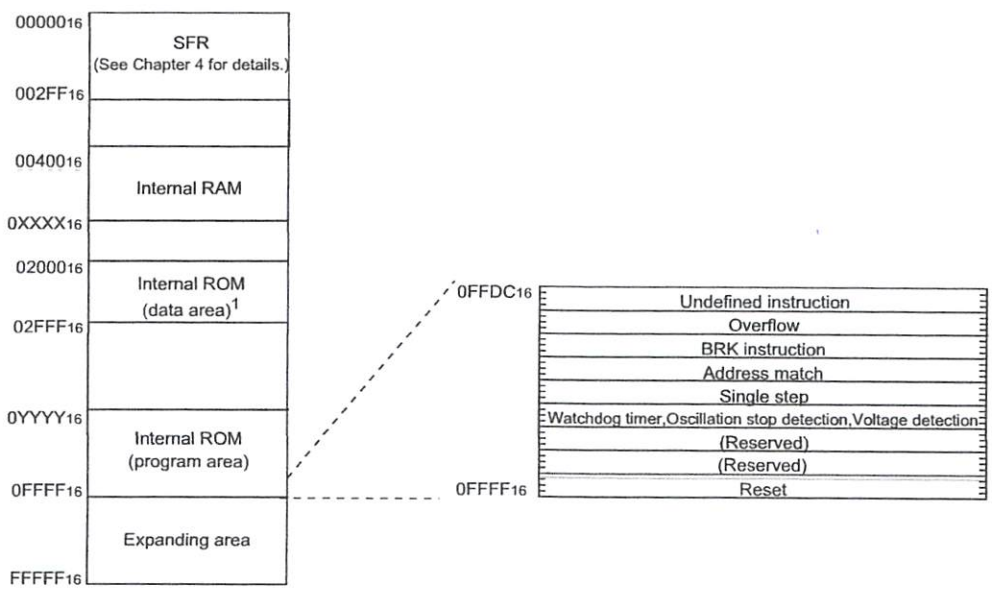
The internal ROM (program area) is allocated in a lower address direction beginning with address 0FFFF16. For example, a 16-Kbyte internal ROM is allocated to the addresses from 0C00016 to 0FFFF16.

The fixed interrupt vector table is allocated to the addresses from 0FFDC16 to 0FFFF16. Therefore, store the start address of each interrupt routine here.

The internal ROM (data area) is allocated to the addresses from 0200016 to 02FFF16.

The internal RAM is allocated in an upper address direction beginning with address 0040016. For example, a 1-Kbyte internal RAM is allocated to the addresses from 0040016 to 007FF16. In addition to storing data, the internal RAM also stores the stack used when calling subroutines and when interrupts are generated.

Special function registers (SFR) are allocated to the addresses from 0000016 to 002FF16. Peripheral function control registers are located here. Of the SFR, any space which has no functions allocated is reserved for future use and cannot be used by users.



NOTES:
1. The data flash ROM block A (2K bytes) and block B (2K bytes) are shown.
2. Blank spaces are reserved. No access is allowed.

Type name	Internal ROM		Internal RAM	
	Size	Address 0YYYY16	Size	Address 0XXXX16
R5F21134FP, R5F21134DFP	16K bytes	0C00016	1K bytes	007FF16
R5F21133FP, R5F21133DFP	12K bytes	0D00016	768 bytes	006FF16
R5F21132FP, R5F21132DFP	8K bytes	0E00016	512 bytes	005FF16

gure 3.1 Memory Map

13.2.3 Bit Rate

Divided-by-16 of frequency by the UiBRG (i=0 to 1) register in UART mode is a bit rate.

<UART Mode>

• When selecting internal clock

$$\text{Setting value to the UiBRG register} = \frac{fj}{\text{Bit Rate} \times 16} - 1$$

fj : Count source frequency of the UiBRG register (f1SIO, f8SIO and f32SIO)

• When selecting external clock

$$\text{Setting value to the UiBRG register} = \frac{fEXT}{\text{Bit Rate} \times 16} - 1$$

fEXT : Count source frequency of the UiBRG register (external clock)

Figure 13.11 Calculation Formula of UiBRG (i=0 to 1) Register Setting Value

Table 13.7 Bit Rate Setting Example in UART Mode

Bit Rate	BRG	System Clock = 20MHz			System Clock = 8MHz		
(bps)	Count Source	BRG Setting Value	Actual Time(bps)	Error(%)	BRG Setting Value	Actual Time(bps)	Error(%)
1200	f8	129 (81_{16})	1201.92	0.16	51 (33_{16})	1201.92	0.16
2400	f8	64 (40_{16})	2403.85	0.16	25 (19_{16})	2403.85	0.16
4800	f8	32 (20_{16})	4734.85	–1.36	12 ($0C_{16}$)	4807.69	0.16
9600	f1	129 (81_{16})	9615.38	0.16	51 (33_{16})	9615.38	0.16
14400	f1	86 (56_{16})	14367.82	–0.22	34 (22_{16})	14285.71	–0.79
19200	f1	64 (40_{16})	19230.77	0.16	25 (19_{16})	19230.77	0.16
28800	f1	42 ($2A_{16}$)	29069.77	0.94	16 (10_{16})	29411.76	2.12
31250	f1	39 (27_{16})	31250.00	0.00	15 ($0F_{16}$)	31250.00	0.00
38400	f1	32 (20_{16})	37878.79	–1.36	12 ($0C_{16}$)	38461.54	0.16
51200	f1	23 (17_{16})	52083.33	1.73	9 (09_{16})	50000.00	–2.34

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

LEMBAR PERBAIKAN SKRIPSI

Nama : Atik Rahmawati Muthoharoh

Nim : 02.17.120

Jurusan : Teknik Elektro S1

Konsentrasi : Teknik Elektronika

Judul :**"SISTEM MONITORING POSISI KERETA API BERBASISKAN DATA GPS YANG DIKIRIMKAN MELALUI HP (HANDPHONE)"**

Hari/Tgl Skripsi : Selasa, 4 September 2007

No.	Materi Perbaikan	Paraf
1.	Masalah pengujian, harap ditampilkan errornya kemudian dimasukkan dalam kesimpulan.	
2.	Seyogyanya kesimpulan disesuaikan tujuan.	

Diperiksa / Disetujui

Penguji

Ir. Teguh Herbasuki, MT
NIP. Y. 1038900209

Mengetahui

Dosen Pembimbing I

Ir. Purwanto, MT
NIP. P. 131574847

Dosen Pembimbing II

Sotyohadi, ST, MSc

INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO

FORMULIR BIMBINGAN SKRIPSI

NAMA : ATIK RAHMAWATI MUTHOHAROH
NIM : 02.17.120
Masa Bimbingan : 14 Februari 2007 – 14 Agustus 2007
Judul Skripsi : Sistem Monitoring Posisi Kereta Api Berbasiskan Data *GPS* Yang Dikirimkan Melalui HP (Handphone)

No.	Tanggal	Uraian	Paraf Pembimbing
1.	21-05-07	Bab I.	
2.		– Latar Belang	
3.		– Batasan Masalah.	
4.	05-06-07.	Rencana Daftar isi	
5.		Bab I sd V.	
6.		Bab II. Ref. di	
7.		Pemahaman	
8.	—	Lengkapi Bab II dst	
9.		Acc Semua hasil.	
10.	—	Acc Bab III & IV → + Pen	

– Acc Ujian.

Malang, 2007
Dosen Pembimbing I

Ir. Purwanto, MT
NIP: P.131574847

Form S- 4A

278