



**UNIVERSITAS INDONESIA**

**RANCANG BANGUN PENGIRIMAN DAN PENERIMAAN  
DATA GPS MELALUI LAYANAN TELEKOMUNIKASI  
SATELIT GARUDA UNTUK DAERAH *REMOTE***

**SKRIPSI**

**EMAN SUSAKA**

**0405030338**

**FAKULTAS TEKNIK  
PROGRAM STUDI TEKNIK ELEKTRO  
DEPOK  
JUNI 2009**



**UNIVERSITAS INDONESIA**

**RANCANG BANGUN PENGIRIMAN DAN PENERIMAAN  
DATA GPS MELALUI LAYANAN TELEKOMUNIKASI  
SATELIT GARUDA UNTUK DAERAH *REMOTE***

**SKRIPSI**

**Diajukan sebagai salah satu syarat untuk memperoleh gelar  
Sarjana**

**EMAN SUSAKA  
0405030338**

**FAKULTAS TEKNIK  
PROGRAM STUDI TEKNIK ELEKTRO  
DEPOK  
JUNI 2009**

## HALAMAN PERNYATAAN ORISINALITAS

Skripsi ini adalah hasil karya saya sendiri, dan semua sumber baik yang dikutip maupun dirujuk telah saya nyatakan dengan benar.

**Nama** : EMAN SUSAKA  
**NPM** : 0405030338

**Tanda Tangan** :

**Tanggal** : Depok, 16 Juni 2009

## HALAMAN PENGESAHAN

Skripsi ini diajukan oleh

Nama : Eman Susaka  
NPM : 0405030338  
Program Studi : Teknik Elektro  
Judul Skripsi : Rancang Bangun Pengiriman Dan Penerimaan  
Data GPS Melalui Layanan Telekomunikasi Satelit  
Garuda Untuk Daerah *Remote*

**Telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Teknik pada Program Studi Elektro, Fakultas Teknik Universitas Indonesia**

### DEWAN PENGUJI

Pembimbing : Ir. Arifin Djauhari, MT ( )

Penguji : Ir. Rochmah N. Sukardi Ny, M.Sc ( )

Penguji : Dr. Fitri Yuli Zulkifli, ST, M.Sc ( )

Ditetapkan di : Depok

Tanggal : 16 Juni 2009

## KATA PENGANTAR

Puji syukur saya panjatkan kepada Allah SWT, karena atas berkat dan rahmat-Nya, saya dapat menyelesaikan skripsi ini. Penulisan skripsi ini dilakukan dalam rangka memenuhi salah satu syarat untuk mencapai gelar Sarjana Teknik Jurusan Teknik Elektro pada Fakultas Teknik Universitas Indonesia. Saya menyadari bahwa tanpa bantuan dan bimbingan dari berbagai pihak, dari masa perkuliahan sampai pada penyusunan skripsi ini, sangatlah sulit bagi saya untuk menyelesaikan skripsi ini. Oleh karena itu, saya mengucapkan terima kasih kepada:

- (1) Ir. Arifin Djauhari, MT dan Dr. Ir. Arman D. Diponegoro, selaku dosen pembimbing 1 dan 2 yang telah menyediakan waktu, tenaga, dan pikiran untuk mengarahkan saya dalam penyusunan tugas akhir ini;
- (2) Pihak PT. PSN yang telah banyak membantu dalam usaha memperoleh data dan peralatan yang saya perlukan;
- (3) Orang tua saya, ibunda Aliyah, ibu yang mengetahui diriku seutuhnya, dahaga dari letihku, ayahanda alm. Soesilo Soemohardjo, ayah adalah cambuk dari semangatku, warisan kemauan keras kudapat darinya;
- (4) Semua pihak yang telah membantu dalam pembuatan dan penyusunan tugas akhir ini.

Akhir kata, saya berharap Allah SWT berkenan membalas segala kebaikan semua pihak yang telah membantu. Semoga skripsi ini dapat membawa manfaat bagi pengembangan ilmu.

Jakarta, 16 Juni 2009

Penulis

EMAN SUSAKA

## LEMBAR PERNYATAAN PERSETUJUAN PUBLIKASI SKRIPSI UNTUK KEPENTINGAN AKADEMIS

---

---

Sebagai sivitas akademika Universitas Indonesia, saya yang bertanda tangan di bawah ini:

Nama : EMAN SUSAKA  
NPM : 0405030338  
Program Studi : Teknik Elektro  
Departemen : Teknik Elektro  
Fakultas : Teknik  
Jenis karya : Skripsi

demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Indonesia **Hak Bebas Royalti Noneklusif (*Non-exclusive Royalty-FreeRight*)** atas karya ilmiah saya yang berjudul:

RANCANG BANGUN PENGIRIMAN DAN PENERIMAAN DATA GPS  
MELALUI LAYANAN TELEKOMUNIKASI SATELIT GARUDA UNTUK  
DAERAH *REMOTE*

beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneklusif ini, Universitas Indonesia berhak menyimpan, mengalihmedia/memformat-kan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan mempublikasikan skripsi saya tanpa meminta izin dari saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Jakarta  
Pada Tanggal : 16 Juni 2009

Yang menyatakan

(EMAN SUSAKA)

## ABSTRAK

Nama : EMAN SUSAKA  
Program Studi : Teknik Elektro  
Judul : RANCANG BANGUN PENGIRIMAN DAN PENERIMAAN  
DATA GPS MELALUI LAYANAN TELEKOMUNIKASI  
SATELIT GARUDA UNTUK DAERAH *REMOTE*

Skripsi ini merancang dan membangun sistem pengambilan dan pengiriman data GPS dengan menggunakan mikrokontroler sebagai pengolah data GPS. Data yang diperoleh dari GPS berupa format NMEA 0183. Data tersebut disimpan di dalam memori mikrokontroler. Mikrokontroler akan memproses data dengan keluaran berupa data waktu dan data posisi. Informasi ini dikirimkan melalui perangkat PASTI/*Byru Marine* ke satelit ACeS Garuda 1. Proses ini berlangsung terus-menerus, sehingga informasi posisi bersifat *real time*. Dari satelit ACeS Garuda 1, informasi tersebut ditransmisikan dan diterima oleh perangkat PASTI/*Byru Marine* yang lain. Data yang telah sampai pada perangkat PASTI/*Byru Marine* pada sisi penerima tersebut, selanjutnya akan ditampilkan pada komputer.

Dalam skripsi ini kami menggunakan mikrokontroler untuk mengambil, menyimpan, memproses, dan mengirimkan data. Kami menggunakan bahasa pemrograman assembler yang sesuai dengan spesifikasi mikrokontroler. RS232 digunakan sebagai antarmuka antara GPS, mikrokontroler, dan perangkat PASTI/*Byru Marine* untuk mengambil dan mengirimkan data.

Kata kunci:

GPS, NMEA 0183, Satelit ACeS Garuda 1, Mikrokontroler, Perangkat PASTI/*Byru Marine*, Konverter RS232, dan Daerah *remote*.

## ABSTRACT

Name : EMAN SUSAKA  
Study Program : Electrical Engineering  
Title : DESIGN AND BUILD OF GPS DATA SENDING AND RECEIVING THROUGH TELECOMMUNICATION SERVICES OF GARUDA SATELLITE FOR REMOTE AREA

This thesis designs and builds the system for retrieving and sending the GPS data using microcontroller. The data which retrieved from GPS in NMEA 0183 format. The data will be saved in the microcontroller. The microcontroller will process the data to set the time and position information. This information then will be sent through PASTI/Byru Marine user terminal to ACeS Garuda 1 satellite. This process will run continuously so the position will be real time. From ACeS Garuda 1 satellite, the information is transmitted and received by other PASTI/Byru Marine user terminal. The information which arrive on PASTI/Byru Marine user terminal in the receiving position, then will be displayed on the computer.

In this thesis we use the microcontroller for retrieving, saving, processing, and sending the data. We build this functionality using assembly programming language, which suitable with microcontroller specification. We use RS232 converter as interface between GPS, microcontroller, and PASTI/Byru Marine terminal to retrieving and sending the data.

Keyword:

GPS, NMEA 0183, ACeS Garuda 1 Satellite, Microcontroller, PASTI/Byru Marine User Terminal, RS232 Converter, and remote area.

## DAFTAR ISI

	Halaman
HALAMAN JUDUL .....	i
HALAMAN PERNYATAAN ORISINALITAS .....	ii
LEMBAR PENGESAHAN .....	iii
KATA PENGANTAR .....	iv
LEMBAR PERSETUJUAN PUBLIKASI KARYA ILMIAH .....	v
ABSTRAK .....	vi
ABSTRACT .....	vii
DAFTAR ISI .....	viii
DAFTAR GAMBAR .....	x
DAFTAR TABEL .....	xi
DAFTAR LAMPIRAN .....	xii
DAFTAR SINGKATAN.....	xiii
<b>BAB 1 PENDAHULUAN .....</b>	<b>1</b>
1.1 LATAR BELAKANG .....	1
1.2 PERUMUSAN MASALAH .....	2
1.3 TUJUAN .....	2
1.4 BATASAN MASALAH .....	3
1.5 SISTEMATIKA PENULISAN .....	3
<b>BAB 2 SISTEM PENGIRIMAN DAN PENERIMAAN DATA GPS</b>	
<b>MELALUI LAYANAN SATELIT .....</b>	<b>4</b>
2.1 KONFIGURASI SISTEM .....	4
2.2 PERANGKAT KERAS .....	6
2.2.1 <i>Global Positioning System</i> .....	6
2.2.2 Konverter RS232.....	8
2.2.3 Mikrokontroler AT89S51.....	11
2.2.4 Perangkat PASTI/ <i>Byry Marine</i> .....	13
2.3 PERANGKAT LUNAK.....	17
<b>BAB 3 APLIKASI PENGIRIMAN DAN PENERIMAAN DATA GPS</b>	
<b>UNTUK DAERAH <i>REMOTE</i> .....</b>	<b>22</b>
3.1 PENGERTIAN DAERAH <i>REMOTE</i> .....	22

3.2 KENDALA PENGIRIMAN DAN PENERIMAAN DATA GPS DI DAERAH <i>REMOTE</i> .....	22
3.3 CONTOH APLIKASI PENGIRIMAN DAN PENERIMAAN DATA GPS MELALUI SATELIT UNTUK DAERAH <i>REMOTE</i> .....	23
<b>BAB 4 PENGUJIAN DAN ANALISIS SISTEM.....</b>	<b>25</b>
4.1 HASIL RANCANG BANGUN.....	25
4.2 PENGUJIAN SISTEM.....	27
4.2.1 Menampilkan Data Keluaran GPS.....	27
4.2.2 Menampilkan Data Keluaran Kontroler.....	29
4.2.3 Menampilkan Data Yang Telah Diterima Pada Sisi Penerima.....	31
4.3 ANALISIS SISTEM.....	32
4.3.1 Analisis Pengujian Pertama.....	32
4.3.2 Analisis Pengujian Kedua.....	32
4.3.3 Analisis Pengujian Ketiga.....	34
<b>BAB 5 KESIMPULAN .....</b>	<b>35</b>
<b>DAFTAR REFERENSI .....</b>	<b>36</b>
<b>DAFTAR PUSTAKA .....</b>	<b>37</b>
<b>LAMPIRAN .....</b>	<b>38</b>

## DAFTAR GAMBAR

	Halaman
Gambar 2.1 Blok diagram sistem.....	5
Gambar 2.2 Tiga segmen pada GPS.....	7
Gambar 2.3 Data keluaran GPS ditampilkan pada computer dengan menggunakan <i>hyperterminal</i> .....	9
Gambar 2.4 Spesifikasi level logika konverter RS232.....	10
Gambar 2.5 Konektor DB9 .....	11
Gambar 2.6 Blok digram fungsional AT89S51.....	14
Gambar 2.7 Pengiriman data melalui perangkat PASTI/ <i>Byru Marine</i> dengan menggunakan <i>hyper terminal</i> .....	15
Gambar 2.8 Penerimaan data melalui perangkat PASTI/ <i>Byru Marine</i> dengan menggunakan <i>hyper terminal</i> .....	17
Gambar 2.9 Diagram alir program mikrokontroler.....	17
Gambar 2.10 Diagram alir program mikrokontroler (lanjutan).....	18
Gambar 4.1(a) Perangkat keras sistem keseluruhan .....	25
Gambar 4.1(b) Perangkat keras sistem pada sisi pengirim.....	26
Gambar 4.1(c) Perangkat keras sistem pada sisi pengirim.....	26
Gambar 4.2 Blok diagram tempat pengujian sistem .....	27
Gambar 4.3 Data keluaran GPS pada kondisi aktif.....	28
Gambar 4.4 Data keluaran GPS pada kondisi tanpa sinyal ( <i>void</i> ).....	29
Gambar 4.5 Tampilan data dari mikrokontroler pada <i>hyper terminal</i> .....	30
Gambar 4.6 Tampilan data yang diterima di sisi penerima.....	31

## DAFTAR TABEL

	Halaman
Tabel 2.1 Format Data GPS.....	8
Tabel 2.2 Spesifikasi RS-232C.....	11
Tabel 2.3 Konverter RS232C pada DB9.....	11



## DAFTAR LAMPIRAN

	Halaman
LAMPIRAN 1 Program Pengolahan Data GPS .....	36
LAMPIRAN 2 Tabel Karakter ASCII .....	40



## DAFTAR SINGKATAN

ACeS	Asia Cellular Satellite
CISC	Complex Instruction Set Computing
CPU	Central Processing Unit
CR	Carriage Return
CTS	Clear To Send
DCD	Data Carrier Detect
DCE	Communication Equipment
DPH	Pointer High
DPL	Data Pointer Low
DOP	Dilution of Precision
DPTR	Data Pointer
DSR	Data Set Ready
DTE	Data Terminal Equipment
DTR	Data Terminal Ready
EA	External Access Enable
EIA	Electronic Industries Association
GND	Signal Ground
GNSS	Global Navigation Satellite System
GPS	Global Positioning System
I/O	Input/Output
ISP	In System Programmable
LF	Line Feed
NAVSTAR	NAVigation System with Timing And Ranging
NCC	Network Control Center
NMEA	National Marine Electronics Association
PASTI	Pasang Telepon Sendiri
PC	Program Counter
PCON	Power Mode Control
RD	Receive Data
RI	Ring Indicator

RTS	Ready To Send
SCON	Serial Port Control
SDO	Serial Data Output
SP	Stack Pointer
TCON	Timer Control
TD	Transmit Data
TMOD	Timer Mode Control
UART	Universal Asynchronous Receiver Transmitter
USART	Universal Synchronous and Asynchronous Receiver Transmitter
UTC	Universal Time Coordinate

# **BAB 1**

## **PENDAHULUAN**

### **1.1 LATAR BELAKANG**

Beberapa tahun belakangan, Indonesia sering ditimpa musibah dalam bidang transportasi seperti kecelakaan transportasi laut dan udara. Sayangnya, lokasi kecelakaan tersebut tidak diketahui secara cepat dan tepat seperti di tengah lautan yang luas atau hutan belantara. Dengan demikian, waktu yang diperlukan untuk proses pencarian korban kecelakaan menjadi lebih lama dan mengurangi harapan hidup bagi korban yang selamat.

Sehubungan dengan fakta tersebut, maka diperlukan usaha untuk mengantisipasi bahaya kecelakaan transportasi, terutama untuk moda transportasi laut dan udara. Untuk itu, diperlukan suatu sistem pemantauan sarana transportasi untuk pemantauan sarana transportasi di wilayah Indonesia. Pemantauan ini berupa pengiriman data lokasi sarana transportasi yang bersangkutan ke stasiun pemantau. Data lokasi sarana transportasi diperoleh dengan menggunakan sistem penentuan posisi global atau GPS (*Global Positioning System*). Data lokasi ini selanjutnya dikirimkan ke stasiun pemantau melalui sistem komunikasi satelit.

Sistem penentuan posisi global atau GPS merupakan sistem navigasi dan penentuan posisi global dengan menggunakan satelit. Sistem ini dapat memberikan informasi mengenai posisi, kelajuan, dan waktu secara cepat, akurat, dan dimana saja di bumi ini setiap saat tanpa tergantung cuaca. Layanan untuk dapat menggunakan sistem ini tersedia bagi semua orang dan tidak ditarik biaya atas pemakaian sistem ini. Hal inilah yang menarik sebagai bahan pertimbangan untuk dapat dijadikan sebagai bagian dalam sistem pemantauan sarana transportasi ini.

Sistem komunikasi satelit merupakan sistem komunikasi yang handal karena memiliki cakupan area yang luas dibandingkan dengan sistem komunikasi yang lain. Sistem komunikasi satelit ini memungkinkan manusia untuk dapat berkomunikasi dimana saja, meskipun di tengah laut atau hutan belantara sekalipun. Sementara itu, teknologi komunikasi dengan menggunakan satelit dewasa ini mengalami perkembangan yang ditandai dengan penggunaan jasa

telekomunikasi satelit yang juga semakin meningkat. Dampak perkembangan ini mengakibatkan penurunan biaya operasional sistem teknologi komunikasi ini yang selanjutnya akan mengakibatkan penurunan tarif untuk komunikasi satelit ini. Dengan kelebihan yang dimiliki oleh sistem komunikasi satelit ini, maka dapat dijadikan sebagai bahan pertimbangan untuk dapat dijadikan sebagai bagian dalam sistem pemantauan sarana transportasi ini.

Satelit sebagai alat komunikasi memiliki spesifikasi dalam segi pemanfaatan. Salah satunya adalah jasa pelayanan data. Sistem pemantauan sarana transportasi dapat memanfaatkan jasa layanan data yang disediakan oleh perusahaan jasa komunikasi satelit. Salah satu satelit yang beredar di wilayah udara Indonesia dan yang menyediakan layanan seperti yang disebutkan adalah satelit ACeS Garuda 1.

Dengan menggabungkan fungsi GPS sebagai alat penentuan posisi global dengan fungsi satelit sebagai alat komunikasi, kami akan merancang dan membuat sistem pemantauan sarana transportasi pada wilayah Indonesia dengan memanfaatkan perangkat yang dapat mengolah data GPS dan dapat berkomunikasi dengan satelit.

## **1.2 PERUMUSAN MASALAH**

Berdasarkan latar belakang tersebut, dapat dirumuskan masalah yaitu bagaimana merancang perangkat sistem pemantauan lokasi sarana transportasi yang handal dan ekonomis dengan memanfaatkan jaringan telepon satelit. Rumusan masalah dapat diperinci menjadi tiga pertanyaan sebagai berikut:

1. Bagaimana merancang dan membuat sistem pengiriman data GPS melalui jaringan telepon satelit dengan memanfaatkan mikrokontroler?
2. Bagaimana mengolah data GPS agar data navigasi tertentu saja yang dikirimkan oleh telepon satelit dengan memanfaatkan mikrokontroler?
3. Bagaimana menampilkan data yang telah dikirimkan ke telepon satelit penerima melalui komputer?

## **1.3 TUJUAN**

Tujuan tugas akhir ini adalah merancang dan membuat sistem pengiriman dan penerimaan data GPS melalui layanan satelit ACeS Garuda 1 dan sistem

komunikasi antara mikrokontroler AT89S51 dengan perangkat telepon satelit PASTI (Pasang Telepon Sendiri)/*Byru Marine* dan GPS.

#### 1.4 BATASAN MASALAH

Permasalahan yang akan dibahas dalam perancangan sistem ini dibatasi pada:

1. Proses pengolahan data GPS dengan menggunakan mikrokontroler AT89S51.
2. Proses pengiriman data hasil pengolahan tersebut melalui layanan satelit ACeS Garuda 1.
3. Proses penerimaan data oleh telepon satelit dan proses penampilan data yang diterima melalui komputer.

#### 1.5 SISTEMATIKA PENULISAN

Penulisan tugas akhir ini disusun berdasarkan sistematika sebagai berikut.

##### Bab 1 Pendahuluan

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

##### Bab 2 Sistem pengiriman dan penerimaan data GPS melalui layanan satelit

Bagian pertama menjelaskan konfigurasi sistem, bagian kedua menjelaskan tentang perangkat keras, dan bagian ketiga menjelaskan tentang perangkat lunak.

##### Bab 3 Aplikasi pengiriman dan penerimaan data gps untuk daerah *remote*

Bagian pertama menjelaskan tentang pengertian daerah *remote*, bagian kedua menjelaskan tentang kendala pengiriman dan penerimaan data GPS di daerah *remote*, dan bagian ketiga menjelaskan tentang contoh aplikasi pengiriman dan penerimaan data GPS melalui satelit untuk daerah *remote*.

##### Bab 4 Pengujian dan Analisis Sistem

Berisikan pengujian sistem dan analisis pada setiap pengujian.

##### Bab 5 Kesimpulan

Berisikan kesimpulan yang didapat dari tugas akhir ini.

## BAB 2

### SISTEM PENGIRIMAN DAN PENERIMAAN DATA GPS MELALUI LAYANAN SATELIT

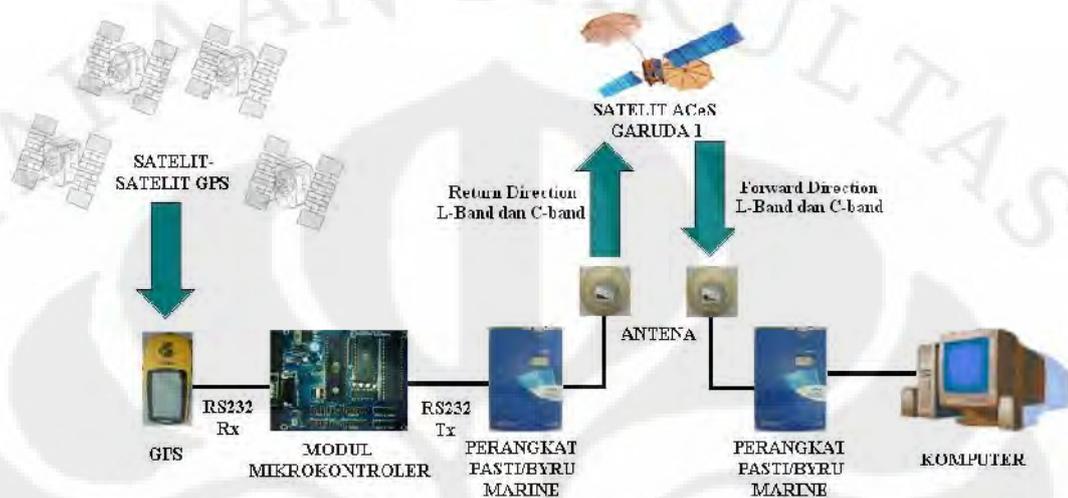
#### 2.1 KONFIGURASI SISTEM

Prinsip kerja sistem secara keseluruhan dimulai dari menempatkan sistem penerimaan GPS pada suatu tempat dan menunggu sampai GPS menerima data dengan baik (minimal menerima sinyal dari empat buah satelit GPS). Selanjutnya, mikrokontroler mulai mengolah data tersebut dengan cara mengambil data dari GPS, kemudian menempatkannya di suatu alamat memori RAM. Data mentah GPS tersebut *dimasking* untuk mendapatkan data navigasi (data posisi lintang dan bujur) dan data waktunya saja yang diwakili oleh data GPRMC. Data GPRMC tersebut selanjutnya akan disimpan mulai dari karakter setelah karakter C sampai akhir data GPRMC yang diwakili oleh *Line Feed/LF* (data 10d). Selanjutnya, data *header* ATDT dan data nomor yang disimpan pada mikrokontroler dikirimkan melalui antarmuka konverter RS232C ke satelit ACeS Garuda 1 melalui perangkat PASTI/*Byru Marine*. Untuk mengakhiri data nomor yang dikirimkan, program akan mengirimkan subrutin akhir data yang tersimpan pada mikrokontroler. Setelah terjadinya proses *handshaking*, maka barulah data GPS yang telah *dimasking* tersebut dapat dikirimkan mulai dari jam UTC sampai dengan akhir data *longitude*.

Data GPS yang dikirimkan merupakan data GPS yang di dalamnya terdapat data kecepatan yang nilainya tidak sama dengan 0. Jika data kecepatan yang terkandung di dalamnya sama dengan 0, maka program akan mengambil data GPS yang baru lagi. Data GPS yang telah *dimasking*, baru akan dikirimkan ketika data kecepatan yang terkandung di dalamnya bernilai tidak sama dengan 0. Ini artinya bahwa jika armada yang dipasang perangkat GPS tidak dalam kondisi bergerak, maka program tidak akan mengirimkan data ke perangkat PASTI/*Byru Marine*. Sebaliknya, jika armada yang dimaksud dalam kondisi bergerak, maka program akan selanjutnya mengirimkan data hasil *masking* ke perangkat PASTI/*Byru Marine* setelah terjadinya proses pemanggilan berhasil dan setelah

terjadinya proses *handshaking*. Data yang dikirimkan akan sampai di perangkat PASTI/Byru Marine penerima dan akan ditampilkan pada peta di komputer.

Sistem terdiri atas beberapa bagian yang dapat digambarkan menjadi blok diagram seperti yang terlihat pada Gambar 2.1.



Gambar 2.1 Blok Diagram Sistem Pengiriman Dan Penerimaan Data GPS Melalui Layanan Komunikasi Satelit Aces Garuda 1

Sistem yang dirancang dibagi menjadi empat bagian, yaitu perangkat penerima GPS, mikrokontroler (kontroler), perangkat PASTI/Byru Marine, dan komputer.

#### 1. Perangkat penerima GPS

Perangkat penerima GPS digunakan untuk menerima data dari satelit GPS.

#### 2. Mikrokontroler

Mikrokontroler yang digunakan adalah mikrokontroler AT89S51 yang akan melakukan proses pengolahan data termasuk *masking* data GPS dan mengirimkannya ke satelit melalui perangkat PASTI/Byru Marine.

#### 3. Perangkat PASTI/Byru Marine

Perangkat PASTI/Byru Marine pada sisi pengirim akan mengirimkan data GPS yang telah diolah mikrokontroler ke satelit ACeS Garuda 1 dan perangkat PASTI/Byru Marine pada sisi penerima akan menerima data yang telah dikirimkan sistem.

#### 4. Komputer

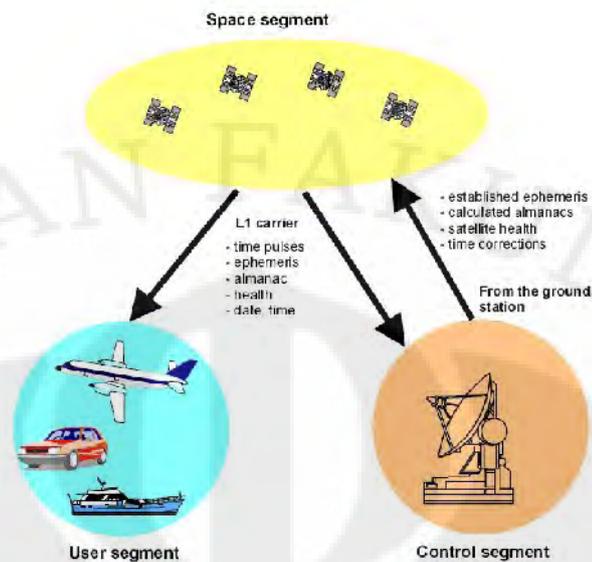
Komputer akan menerima data yang telah dikirimkan sistem dari perangkat PASTI/*Byru Marine* pada sisi penerima dan menampilkannya melalui program *hyperterminal* di komputer.

### 2.2 PERANGKAT KERAS

Perangkat keras yang digunakan sesuai dengan fungsi blok-blok sistem yang digambarkan pada Gambar 2.1. Penerima GPS yang digunakan adalah GPS merk Garmin seri *Etrex-H*; kontroler yang digunakan adalah modul mikrokontroler *AT89S51*; dan perangkat *PASTI/Byru Marine* untuk berkomunikasi dengan satelit Garuda 1. Konverter *RS232C* dihubungkan dengan perangkat penerima GPS dan *PASTI/Byru Marine*. Perangkat navigasi GPS difungsikan sebagai masukan data dan perangkat *PASTI/Byru Marine* difungsikan untuk mengirimkan data ke satelit *ACeS Garuda 1*.

#### 2.2.1 *Global Positioning System (GPS)*

GPS adalah sistem navigasi satelit dan penentuan posisi menggunakan satelit. Istilah GPS juga bisa disebut dengan *NAVigation System with Timing And Ranging Global Positioning System (NAVSTAR-GPS)*. Sebagai sistem navigasi, GPS bekerja dengan melibatkan beberapa segmen yang terkait satu dengan lainnya. Segmen-segmen yang mendukung kinerja GPS adalah segmen angkasa (*space segment*), segmen sistem kontrol (*control segment*), dan segmen pengguna (*user segment*). Di antara segmen angkasa dan segmen sistem kontrol terjadi komunikasi dua arah. Sementara itu, antara segmen pengguna dan segmen angkasa hanya terjadi komunikasi satu arah, yaitu dari segmen angkasa ke segmen pengguna.



Gambar 2.2 Tiga Segmen Pada GPS[1]

Untuk menampilkan variabel GPS seperti waktu dan kecepatan ke perangkat, GPS *receiver* menggunakan perangkat antarmuka konverter RS232C. Akan tetapi, yang paling penting pada saat GPS *receiver* menerima informasi adalah variabel tersebut ditampilkan dalam format data khusus. Format data GPS ini ditentukan oleh *National Marine Electronics Association* (NMEA). Saat ini data yang ditampilkan GPS sesuai dengan format NMEA-0183. Pada GPS sendiri terdapat beberapa macam data yang dapat ditampilkan[2], yaitu:

1. GPRTE adalah data mengenai rute.
2. GPRMC adalah data mengenai informasi navigasi minimum yang direkomendasikan.
3. GPRMB adalah data mengenai informasi navigasi minimum yang direkomendasikan.
4. GPGGA adalah data tetap sistem pemosisian global.
5. GPGSA adalah data mengenai GNSS DOP dan satelit yang aktif, yaitu penurunan akurasi dan jumlah satelit yang aktif pada GNSS DOP.
6. GPGSV adalah data mengenai satelit GNSS yang berada dalam jangkauan.
7. GPGLL adalah data mengenai posisi geografis, yaitu latitude/longitude.
8. GPBOD adalah data mengenai *bearing – waypoint to waypoint*.
9. PGRME adalah data mengenai *error* yang diestimasi Garmin.
10. PGRMZ adalah data mengenai informasi *altitude*/ketinggian.

Format data GPS berdasarkan standar NMEA-0183 adalah sebagai berikut:

```
$GPDTS,Inf_1,Inf_2,Inf_3,Inf_4,Inf_5,Inf_6,Inf_n*CS<CR><LF>[3]
```

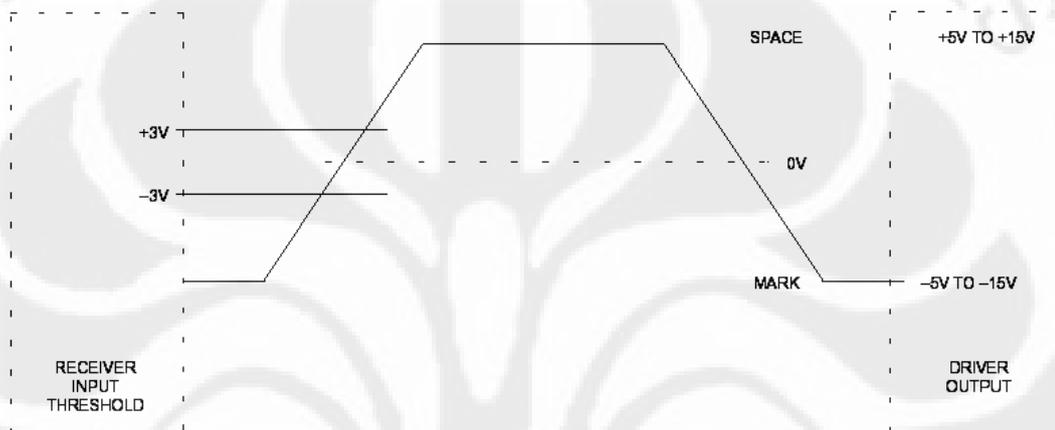
Keterangan simbol-simbol dalam format data GPS tersebut dapat dilihat dalam Tabel 2.1 di bawah ini. Sementara itu keluaran data GPS berdasarkan format NMEA-0183 dapat dilihat pada Gambar 2.3.

Tabel 2.1 Format Data GPS

Simbol	Keterangan
\$	Awal data
GP	Informasi peralatan navigasi
DTS	Jenis informasi yang terkandung
Inf_1 bis Inf_n	Data navigasi
,	Pemisahan antara informasi
*	Pemisahan <i>checksum</i>
CS	Pengecekan kesalahan pada kalimat
<CR><LF>	Akhir dari data: <i>carriage return</i> (<CR>), dan <i>line feed</i> (<LF>)



Karakteristik transmisi data serial konverter versi RS232C pada level logika 1 (*Mark*) dinyatakan dengan tegangan antara -3 sampai -15 Volt dengan beban (-25 Volt tanpa beban), dan level logika 0 (*Space*) dinyatakan dengan tegangan antara +3 sampai +15 Volt dengan beban (+25 Volt tanpa beban). Untuk daerah +3 sampai -3 Volt tidak didefinisikan. Sedangkan paralel mentransmisikan level logika 0 dalam tegangan 0 sampai 0,8 Volt dan level logika 1 dalam tegangan 3,8 sampai 5 Volt.



Gambar 2.4 Spesifikasi Level Logika Konverter RS232[4]

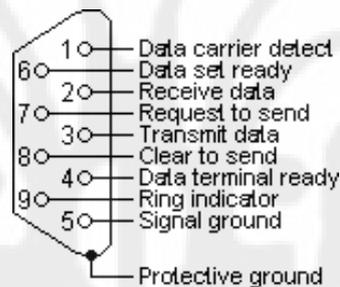
Gambar 2.4 mengilustrasikan spesifikasi level logik konverter RS232C sebagaimana diuraikan dalam paragraf-paragraf sebelumnya.

Kecepatan komunikasi konverter RS232 dinyatakan dalam *baudrate*. Sesuai dengan standar yang berlaku, kecepatannya mencapai 20 kbps dalam jarak kurang dari 15 meter. Beban impedansi pada *driver* harus berada di antara 3000 dan 7000 Ohm serta tidak melebihi 2500 pF.

Pada komunikasi serial konverter RS232C, panjang maksimal kabel antara *transmitter* dan *receiver* tidak boleh melebihi 100 kaki (sekitar 30,48 meter). Spesifikasi konverter RS232C dapat dilihat pada Tabel 2.2 dan perangkat konverter RS232C tipe konektor DB9 dapat dilihat pada Gambar 2.5.

Tabel 2.2 Spesifikasi RS-232C

<b>Spesifikasi</b>	<b>Keterangan</b>
Mode operasi	<i>Single ended</i>
Jumlah Tx dan Rx yang diijinkan	1 Tx, 1 Rx ( <i>point to point</i> )
Panjang kabel maksimum	Kurang dari 100 kaki
<i>Data rate</i> maksimum	20kbps
<i>Driver output range</i> minimum	$\pm 5$ Volt sampai $\pm 15$ Volt
<i>Driver output range</i> maksimum	$\pm 25$ Volt
Impedansi pada Tx	3k $\Omega$ sampai 7k $\Omega$
Sensitivitas masukan Rx	$\pm 3$ Volt
<i>Range</i> tegangan masukan Rx	$\pm 15$ Volt
Tahanan <i>input</i> Rx maksimum	3k $\Omega$ sampai 7k $\Omega$
<i>Receiver threshold</i>	$\pm 3$ Volt



Gambar 2.5 Konektor DB9

Tabel 2.3 memperlihatkan posisi dan deskripsi pin dari konverter RS232C jenis DB9.

Tabel 2.3 Konverter RS232C Pada DB9

<b>Nomor Pin</b>	<b>Sinyal</b>	<b>Deskripsi</b>
1	DCD	<i>Data Carrier Detect</i>
2	RD	<i>Receive Data</i>
3	TD	<i>Transmit Data</i>
4	DTR	<i>Data Terminal Ready</i>
5	GND	<i>Signal Ground</i>
6	DSR	<i>Data Set Ready</i>
7	RTS	<i>Request To Send</i>
8	CTS	<i>Clear To Send</i>
9	RI	<i>Ring Indicator</i>

Sinyal	Fungsi
DCD	Saat modem mendeteksi sinyal <i>carrier</i> dari modem ujung yang lain pada jalur telepon, maka jalur ini akan aktif.
RD	Sinyal ini dihasilkan DCE dan diterima oleh DTE
TD	Sinyal ini dihasilkan DTE dan diterima oleh DCE
DTR	Mengindikasikan kesiapan dari DTE. Sinyal ini <i>ON</i> oleh DTE saat siap untuk mengirim atau menerima data.
DSR	Mengindikasikan kesiapan dari DCE. Sinyal ini <i>ON</i> oleh DCE saat siap untuk mengirim atau menerima data.
RTS	Saat DTE siap untuk mengirim data ke DCE, RTS akan <i>ON</i> . Pada sistem <i>simplex</i> dan <i>duplex</i> , kondisi ini menunjukkan DCE pada mode <i>receive</i> . Pada sistem <i>half-duplex</i> , kondisi ini menunjukkan DCE pada mode pengiriman. Setelah RTS diaktifkan, DCE harus mengaktifkan CTS sebelum memulai komunikasi.
CTS	Digunakan bersama dengan RTS untuk mengadakan <i>handshaking</i> antara DTE dan DCE. Setelah DCE menerima perintah RTS, hal ini menyebabkan CTS <i>ON</i> saat siap untuk memulai komunikasi.

### 2.2.3 Mikrokontroler AT89S51

AT89S51 merupakan satu dari keluarga mikrokontroler dengan kode seri MCS51 yang mempunyai teknologi *Complex Instruction Set Computing* (CISC). Mikrokontroler ini memiliki instruksi yang dikemas dalam kode 16 bit (16 bit *word*). AT89S51 mempunyai arsitektur seperti yang diperlihatkan pada Gambar 2.6. Dari Gambar 2.6 tersebut dapat dilihat bahwa AT89S51 memiliki bagian sebagai berikut:

1. 32 buah saluran I/O (*Input/Output* 'Masukan/Keluaran')  
Jumlah saluran I/O pada mikrokontroler AT89S51 adalah sebanyak 32 saluran yang dibagi menjadi 4 keluaran *Port*, yaitu *Port 0*, *Port 1*, *Port 2*, *Port 3*.
2. 8 bit *Central Processing Unit* (CPU)  
Mikrokontroler AT89S51 terdiri dari 8 bit CPU yang berisi register A (akumulator) dan register B.

3. *Program Counter (PC) dan Data Pointer (DPTR)*

PC dan DPTR memiliki 16 bit data. DPTR terbagi dua, yaitu 8 bit *Data Pointer High (DPH)* dan 8 bit *Data Pointer Low (DPL)*.

4. *Stack Pointer (SP)*

Register SP pada mikrokontroler AT89S51 adalah sebesar 8 bit. Register ini akan bertambah tiap kali data disimpan pada saat program *PUSH* dan pada saat pemanggilan subrutin.

5. *In System Programmable (ISP) flash memory*

Besarnya memori ISP pada tiap tipe mikrokontroler berbeda-beda. Pada tipe AT89S51 besarnya adalah 4K Byte.

6. RAM Internal

Mikrokontroler AT89S51 memiliki RAM internal sebesar 128 byte.

7. *Timer/Counter*

Mikrokontroler AT89S51 memiliki 2 x 16 bit *timer/counter*

8. *Port Serial*

*Port serial* mikrokontroler bekerja pada mode *full duplex*.

9. *Control Register*

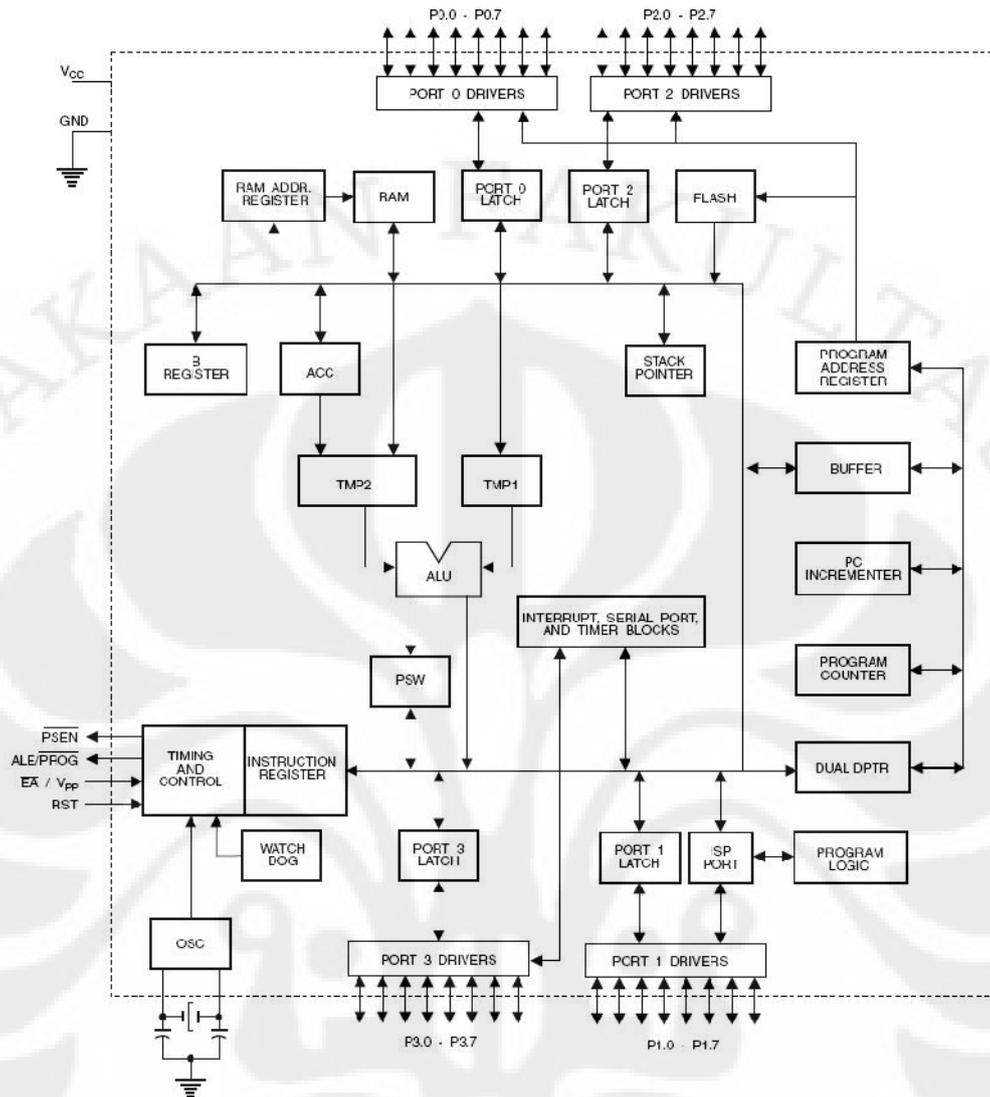
Mikrokontroler dalam bekerja didukung oleh register-register yang berfungsi untuk mengontrol masukan atau keluaran yang diberikan. Register-register ini adalah: TCON, TMOD, SCON, PCON, IP, dan IE.

10. Rangkaian osilator dan *clock*

Rangkaian osilator dan *clock* dihasilkan oleh rangkaian osilasi kristal.

11. Lima unit interupsi

Interupsi pada mikrokontroler terdiri dari: dua interupsi eksternal, dua interupsi *timer*, dan interupsi *port serial*.



Gambar 2.6 Blok Diagram Fungsional AT89S51[5]

## 2.2.4 Perangkat PASTI/Byru Marine

Perangkat PASTI/Byru Marine adalah suatu perangkat telekomunikasi bergerak berbasis satelit yang dapat digunakan untuk layanan suara, data, dan juga *tracking*. Perangkat PASTI/Byru Marine dirancang khusus untuk kebutuhan telekomunikasi yang fleksibel bagi industry kelautan. Perangkat PASTI/Byru Marine ini menggunakan jaringan ACeS (*Asia Cellular Satellite*).

### 2.2.4.1 Metode Pengiriman Data pada Perangkat PASTI/Byru Marine

Untuk mengirimkan data ke satelit melalui perangkat PASTI/Byru Marine diperlukan format tertentu sehingga satelit dapat mengenali data yang dikirimkan. Metode yang dapat dipakai dalam pengiriman data yang dikenali oleh satelit adalah dengan menghubungi nomor yang teregistrasi.

Metode ini dilakukan dengan menghubungi nomor data yang dikenali oleh satelit. Program akan memanggil nomor data perangkat PASTI/*Byru Marine* lain. Setelah berhasil berkomunikasi, barulah data GPS dikirimkan. Pada pemanggilan ini, *header ATDT* disisipkan di depan nomor yang dituju. Setelah terhubung, *header ATDT* dapat tidak disertakan dalam pengiriman data. Dengan metode ini, jumlah data yang dikirim tidak dibatasi. Data tersebut juga bisa dalam bentuk huruf atau karakter ASCII lainnya.

Tampilan untuk pengiriman data berupa karakter ASCII ke satelit melalui perangkat PASTI/*Byru Marine* dapat dilihat pada Gambar 2.7.



```

Skripsi Eman - HyperTerminal
File Edit View Call Transfer Help
AT
NO CARRIER
AT
OK
ATDT0868190
NO CARRIER
AT
OK
AT
OK
ATDT086813862666
NO CARRIER
AT
OK
ATDT086813862666
CONNECT 2400
EFDSDSDFSDSDEEEEEEE
NO CARRIER
AT
OK
ATDT086813862666
CONNECT 2400
TT
OK
AT
OK
Disconnected Auto Detect 2400 8-N-1

```

Gambar 2.7 Pengiriman Data Melalui Perangkat PASTI/*Byru Marine* Dengan Menggunakan *Hyper Terminal*

Sebelum melakukan pemanggilan melalui perangkat PASTI/*Byru Marine*, program akan mengirimkan pesan AT kepada perangkat PASTI/*Byru Marine*. Perangkat PASTI/*Byru Marine* kemudian akan memberikan konfirmasi *OK* atau tidak memberikan konfirmasi apa pun. Konfirmasi *OK* menyatakan bahwa perangkat PASTI/*Byru Marine* telah siap digunakan untuk melakukan pemanggilan dan jika perangkat PASTI/*Byru Marine* tidak memberikan konfirmasi apa pun, maka perangkat PASTI/*Byru Marine* tersebut belum siap digunakan untuk melakukan pemanggilan.

Data pada baris kelima merupakan pemanggilan dengan menggunakan nomor yang tidak terdaftar atau tidak sempurna. Pada baris di bawahnya terlihat pesan *NO CARRIER* yang menunjukkan bahwa tidak dapat dideteksinya nomor tujuan. Akan tetapi, data yang dikirimkan pada saat pemanggilan akan tersimpan di *server*. Di pihak lain, data pada baris kesebelas merupakan pemanggilan dengan menggunakan nomor tujuan yang teregistrasi. Dua belas digit angka di belakang *header ATDT* merupakan nomor yang aktif dan terdaftar. Namun, pada baris di bawahnya terlihat pesan *NO CARRIER*. Hal ini disebabkan bukan karena nomor yang dituju tidak dapat dideteksi, melainkan disebabkan karena adanya gangguan sinyal, seperti sinyal yang lemah atau tidak adanya sinyal sama sekali pada saat terjadinya pemanggilan.

Data pada baris kelima belas merupakan pemanggilan dengan menggunakan nomor tujuan yang sama dengan nomor tujuan yang digunakan pada baris kesebelas, yaitu nomor yang aktif dan terdaftar. Dengan demikian, pada baris berikutnya terlihat pesan *CONNECT 2400* yang berarti koneksi berjalan pada kecepatan 2400 baud. Pesan *CONNECT 2400* ini akan muncul ketika perangkat *PASTI/Byru Marine* menerima sinyal yang cukup atau dengan kata lain tidak terdapat gangguan sinyal dan juga setelah perangkat *PASTI/Byru Marine* yang dituju memberikan konfirmasi positif penerimaan panggilan atau dengan kata lain telah terjadi *handshaking*. Dengan metode ini, perangkat *PASTI/Byru Marine* bebas mengirimkan data apapun dengan format karakter ASCII. Selain dapat mengirimkan data, perangkat *PASTI/Byru Marine* pemanggil juga dapat menerima data apapun dengan format karakter ASCII. Dengan kata lain, komunikasi dapat berjalan, baik searah maupun dua arah.

#### **2.2.4.2 Metode Penerimaan Data pada Perangkat PASTI/Byru Marine**

Tampilan untuk penerimaan data berupa karakter ASCII dari satelit melalui perangkat *PASTI/Byru Marine* dapat dilihat pada Gambar 2.8.

```

RING
RING
RING
ATA
CONNECT 2400
DFSFSEFDFHGRFHGFHGFHGFHWKERWKURVKUWEGRGTRG
NO CARRIER
AT
NO CARRIER
AT
OK
ATDT0868190
NO CARRIER
AT
OK
AT
OK
ATDT086813862666
NO CARRIER
AT
OK
ATDT086813862666
CONNECT 2400
EFDSEFSDFSDFSEEEEEEE

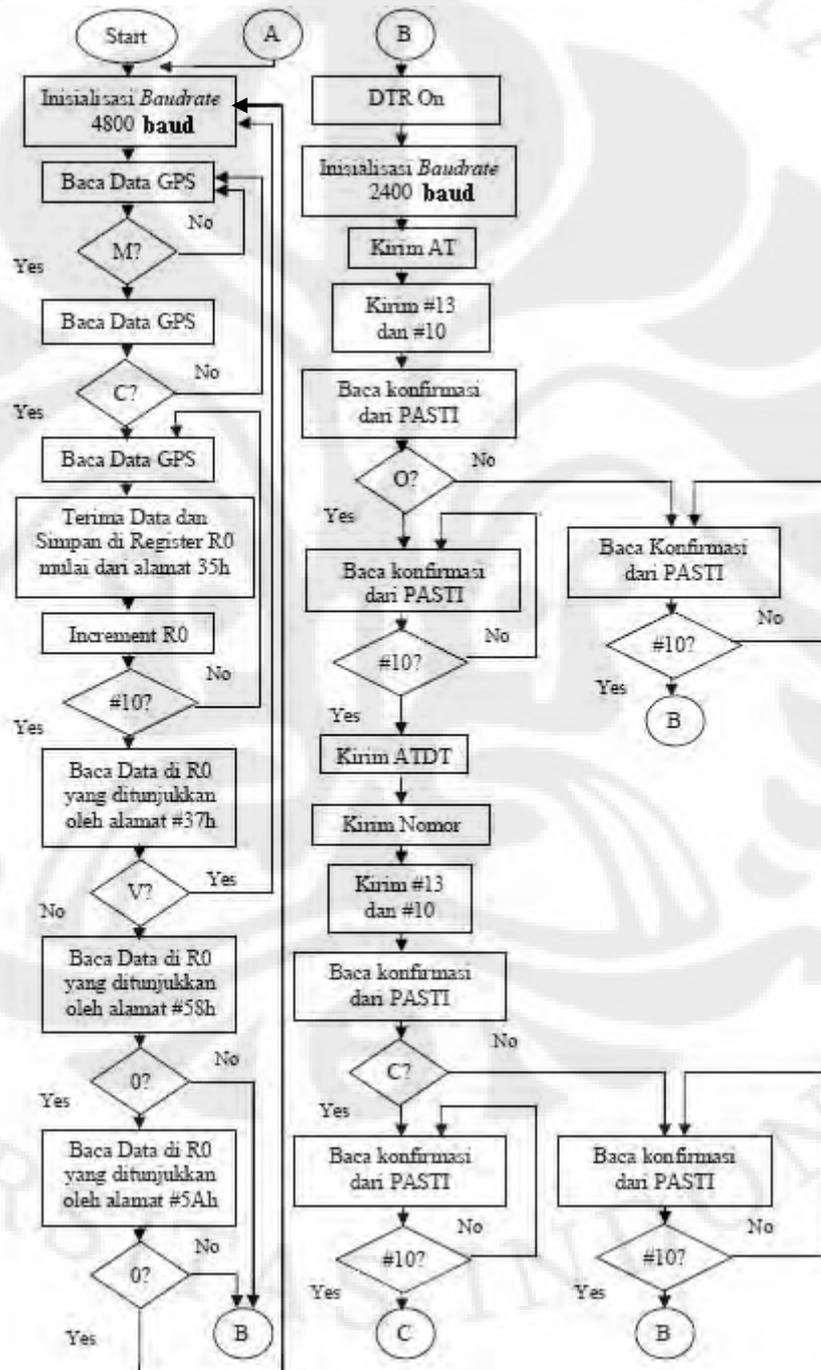
```

Gambar 2.8 Penerimaan Data Melalui Perangkat PASTI/*Byru Marine* Dengan Menggunakan *Hyper Terminal*

Pada saat proses pemanggilan terhadap perangkat PASTI/*Byru Marine* berhasil dilakukan, maka pada perangkat PASTI/*Byru Marine* yang dituju akan mengirimkan pesan *RING* sebanyak tiga kali. Hal ini dapat dilihat pada baris pertama, kedua, dan ketiga pada Gambar 2.8. Untuk dapat berkomunikasi, maka program harus mengirimkan konfirmasi kepada perangkat PASTI/*Byru Marine* pada sisi penerima. Konfirmasi yang dikirimkan kepada perangkat PASTI/*Byru Marine* yaitu berupa pesan *ATA*. Setelah pesan *ATA* ini dikirimkan kepada perangkat PASTI/*Byru Marine* pada sisi penerima, maka pada baris berikutnya akan terlihat pesan *CONNECT 2400* yang berarti koneksi berjalan pada kecepatan 2400 baud dan telah terjadi *handshaking*. Setelah itu, perangkat PASTI/*Byru Marine* bebas menerima data apapun yang dikirimkan dengan format karakter ASCII. Selain dapat menerima data yang dikirimkan, perangkat PASTI/*Byru Marine* yang dituju juga dapat mengirimkan data apapun dengan format karakter ASCII. Dengan kata lain, komunikasi dapat berjalan, baik searah maupun dua arah.

### 2.3 PERANGKAT LUNAK

Perangkat lunak merupakan pemrograman pada mikrokontroler AT89S51. Pemrograman yang dipakai adalah dengan menggunakan bahasa standar pemrograman mikrokontroler, yaitu bahasa assembler. Algoritma pemrosesan data secara keseluruhan yang menggunakan mikrokontroler dapat digambarkan pada Gambar 2.9 dan 2.10 berikut.



Gambar 2.9 Diagram Alir Program Mikrokontroler



Gambar 2.10 Diagram Alir Program Mikrokontroler (lanjutan)

Pemrograman pada mikrokontroler dilakukan dengan membuat subrutin-subrutin sebagai berikut: inisialisasi *port* serial, pengambilan data GPS, *masking* data, penyimpanan data pada RAM, dan pengiriman data ke perangkat PASTI/Byru Marine.

Pada awal diagram alir, mikrokontroler akan melakukan inisialisasi *port* serial yang digunakan; berupa kecepatan baud, mode serial yang digunakan, dan besarnya nilai *timer* yang dipakai.

Pada perancangan ini digunakan dua kecepatan *baud* yang berbeda, yang pertama sebesar 4800 baud dan 2400 baud. Kecepatan 4800 baud difungsikan untuk mengambil data dari GPS. Kecepatan 2400 baud difungsikan untuk mengirimkan data ke satelit. Besaran kecepatan 4800 dan 2400 baud didapat dengan menentukan besarnya *timer* yang digunakan oleh mikrokontroler. Untuk menghitung nilai *timer* kedua kecepatan dapat dipergunakan rumus[6]:

1. Untuk kecepatan 4800 baud, digunakan *timer* sebesar

$$TH1 = 256d - \left( \frac{2^{SMOD}}{32d} \times \frac{frekuensi OSC}{12 \times 4800d} \right) \dots\dots\dots (2.1)$$

Dalam perencanaan dipergunakan SMOD bit 0 dan kristal sebesar 11,0592 MHz. Maka didapatkan.

$$TH1 = 256d - \left( \frac{2^0}{32d} \times \frac{11,0592 \times 10^6}{12 \times 4800d} \right)$$

$$TH1 = 250d$$

$$TH1 = FAh$$

2. Untuk kecepatan 2400 baud, digunakan *timer* sebesar

$$TH1 = 256d - \left( \frac{2^{SMOD}}{32d} \times \frac{frekuensi OSC}{12 \times 2400d} \right) \dots\dots\dots (2.2)$$

Besarnya Parameter dan nilai yang digunakan pada kecepatan 2400 baud sama dengan parameter dan nilai pada kecepatan 4800 baud. Maka didapatkan.

$$TH1 = 256d - \left( \frac{2^0}{32d} \times \frac{11,0592 \times 10^6}{12 \times 2400d} \right)$$

$$TH1 = 244d$$

$$TH1 = F4h$$

Pada Saat melakukan pengambilan data, mikrokontroler akan menset kecepatannya pada kecepatan 4800 baud. Hal ini dikarenakan kecepatan data yang dipakai GPS adalah 4800 baud. Perlu dua kecepatan yang sama dalam melakukan tranmisi data antar perangkat, agar dapat melakukan komunikasi dan menghindari adanya kesalahan pada saat penerimaan data. Sedangkan pada saat mengirim data ke satelit Garuda 1, kecepatan data yang digunakan perangkat PASTI/*Byru Marine* untuk berkomunikasi dengan satelit Garuda 1 adalah 2400 baud. Untuk sinkronisasi, mikrokontroler harus menset kecepatannya juga pada 2400 baud.

Mikrokontroler mengambil data dari GPS yang telah aktif (menerima sinyal sekurang-kurangnya dari empat satelit GPS). Mikrokontroler akan mengambil data mentah GPS. Data yang masuk ke mikrokontroler adalah data NMEA-0183 (sepuluh macam data). Tiap data yang diambil mikrokontroler, disimpan di register *R0*. Pada register ini diberikan *loop* untuk mencari data yang mempunyai karakter M dan C di dalamnya. Karakter M dan C ini melambangkan format data GPRMC.

Setelah ditemukan, tiap karakter setelah karakter M dan C akan disimpan pada memori internal RAM, mulai alamat 35H. Data yang disimpan ini akan dibatasi dengan data LF/*Line Feed* (data 10d). Kemudian, mikrokontroler akan melihat isi dari register *R0* yang ditunjukkan oleh alamat 58h dan 5Ah. Jika isi dari register yang ditunjukkan oleh alamat 58h bernilai 0, maka mikrokontroler akan melihat isi dari register *R0* yang ditunjukkan oleh alamat 5Ah. Jika isi dari register yang ditunjukkan oleh alamat 5Ah ini bernilai 0 juga, maka program akan mengambil data GPS yang baru. Jika isi dari register yang ditunjukkan oleh alamat 58h atau 5Ah tidak bernilai 0, maka program akan melanjutkan ke proses berikutnya, yaitu inisialisasi *baudrate* 2400 baud. Hal ini berarti program akan melakukan komunikasi dengan perangkat PASTI/*Byru Marine*.

Setelah data *masking* GPRMC disimpan, mikrokontroler diubah kecepatannya menjadi 2400 baud. Data yang dikirim pertama kali adalah data AT. Setelah data AT ini dikirimkan ke perangkat PASTI/*Byru Marine*, maka selanjutnya mikrokontroler akan menunggu konfirmasi dari perangkat PASTI/*Byru Marine*. Jika konfirmasi yang diterima oleh mikrokontroler adalah OK, maka mikrokontroler akan mengirimkan data selanjutnya. Dan jika

mikrokontroler belum mendapatkan konfirmasi dari perangkat PASTI/*Byru Marine*, maka mikrokontroler akan mengirimkan kembali data AT ke perangkat PASTI/*Byru Marine*.

Data yang dikirimkan selanjutnya adalah data *header* ATDT. Data tersebut dapat diambil dari subrutin yang telah disediakan. Setelah dikirimkan ke perangkat PASTI/*Byru Marine*, data nomor perangkat PASTI/*Byru Marine* yang teregistrasi disiapkan untuk dikirimkan setelah data *header* ATDT dikirimkan.

Setelah data *header* ATDT dan data nomor perangkat PASTI/*Byru Marine* yang teregistrasi dikirimkan, maka program akan membaca konfirmasi dari perangkat PASTI/*Byru Marine*. Jika konfirmasi dari perangkat PASTI/*Byru Marine* adalah *NO CARRIER*, maka program akan melakukan inisialisasi *baudrate* 2400 baud kembali dan akan melakukan proses berikutnya. Jika konfirmasi dari perangkat PASTI/*Byru Marine* adalah *CONNECT 2400*, maka program akan mengirimkan data GPS hasil *masking* kepada perangkat PASTI/*Byru Marine*, yaitu data dari jam UTC sampai dengan data akhir *longitude*. Program akan mengulang proses pada alamat internal RAM sampai didapatkan akhir data *longitude* yang dimaksud. Untuk mengakhiri pengiriman satu siklus data GPS yang telah diolah, subrutin yang mengandung data *<CR><LF>* dikirimkan ke perangkat PASTI/*Byru Marine*.

Mikrokontroler akan *delay* sistemnya selama 36,4032 detik sebelum mengambil data baru dari GPS.

## **BAB 3**

### **APLIKASI PENGIRIMAN DAN PENERIMAAN DATA GPS UNTUK DAERAH *REMOTE***

#### **3.1 PENGERTIAN DAERAH *REMOTE***

Daerah *remote* adalah daerah terpencil yang jaraknya relatif jauh dalam suatu wilayah tertentu. Sebagai negara kepulauan sepanjang 1/8 bentangan dunia dengan luas 1,9 juta km persegi yang memiliki 18.000 pulau, Indonesia memiliki sejumlah daerah *remote*. Di Indonesia, mayoritas daerah terpencil berada di Indonesia bagian timur seperti Maluku dan Papua yang belum memiliki jaringan telekomunikasi.

Daerah *remote* didefinisikan berdasarkan kondisi geografis wilayah tersebut atau berdasarkan jarak jangkauannya dari suatu wilayah tertentu. Kondisi geografis yang masih berupa pegunungan, terpisah hutan belantara atau lautan mengakibatkan akses komunikasi di wilayah tersebut menjadi sulit.

#### **3.2 KENDALA PENGIRIMAN DAN PENERIMAAN DATA GPS DI DAERAH *REMOTE***

Ada beberapa wilayah yang sulit dijangkau layanan telekomunikasi karena letak geografis dan terbatasnya fasilitas infrastruktur. Akibat letak geografis dan terbatasnya fasilitas infrastruktur, maka akses telekomunikasi di daerah terpencil menjadi terkendala. Akibatnya, bukan hal yang aneh lagi kalau masih banyak daerah di Indonesia yang belum tersentuh sama sekali dengan jaringan telekomunikasi (telepon konvensional ataupun selular). Apalagi sebagian besar wilayah di Indonesia merupakan wilayah laut. Adalah suatu hal yang sangat sulit dan tidak ekonomis untuk membangun jaringan telekomunikasi, seperti telepon konvensional ataupun selular di daerah *remote*. Hal ini disebabkan karena tingkat kepadatan telekomunikasi di daerah *remote* tidak sama dengan daerah kota. Padahal ketersediaan jaringan telekomunikasi merupakan suatu kebutuhan yang mutlak pada saat ini untuk memungkinkan terjadinya komunikasi.

Untuk melakukan pengiriman data GPS suatu lokasi yang bergerak/*mobile* di daerah *remote* dan juga dengan cakupan wilayah yang luas, maka akan mustahil jika dilakukan dengan mengandalkan jaringan telekomunikasi seperti telepon konvensional ataupun selular. Untuk itu, diperlukan suatu sistem telekomunikasi

dengan cakupan wilayah yang luas, memiliki kemampuan layanan yang bersifat *mobile/wireless*, dan juga bersifat ekonomis. Dan sistem telekomunikasi berbasis satelit dapat menjawab semua tantangan itu.

### **3.3 CONTOH APLIKASI PENGIRIMAN DAN PENERIMAAN DATA GPS MELALUI SATELIT UNTUK DAERAH *REMOTE***

Solusi layanan telekomunikasi dan informasi di daerah terpencil seperti pedesaan dan pulau-pulau terluar di Indonesia termasuk wilayah laut adalah dengan sistem telekomunikasi berbasis satelit. Keuntungan sistem telekomunikasi berbasis satelit adalah sebagai berikut:

1. Cakupan yang luas: satu negara, *region*, ataupun satu benua. Untuk mencakup suatu daerah (misalnya Indonesia) hanya diperlukan satu stasiun pengulang alias satu satelit.
2. Pengembangan jaringan telekomunikasi bisa berjalan dengan cepat serta mudah dalam instalasinya.
3. Mempunyai spektrum frekuensi yang lebar.
4. Layanan *mobile/wireless* yang independen terhadap lokasi.
5. Biaya relatif rendah per *site*.

Dengan adanya sistem telekomunikasi berbasis satelit, maka akan dengan mudah untuk melakukan komunikasi dimana saja. Sistem telekomunikasi ini dibuat untuk wilayah yang secara geografis sulit dijangkau dan juga dibuat untuk layanan *mobile/wireless* yang independen terhadap lokasi. Untuk itu, sistem telekomunikasi ini cocok untuk komunikasi bagi armada laut dan udara mengingat kedua armada ini beroperasi di daerah *remote* dan juga membutuhkan layanan *mobile/wireless* yang independen terhadap lokasi.

Salah satu contoh aplikasi sistem pengiriman dan penerimaan data GPS melalui sistem telekomunikasi berbasis satelit ini adalah untuk pemantauan posisi armada, khususnya armada laut dan udara. Dengan rancangan sistem seperti yang telah dijelaskan pada bab sebelumnya, maka lokasi armada yang bersangkutan dapat diketahui keberadaannya secara cepat, tepat dan *real time*. Dengan rancangan sistem seperti ini, maka akan menggantikan atau melengkapi fungsi dari RADAR (*Radio Detection and Ranging*) sebagai alat pemantau armada yang telah ada dan telah digunakan sebelumnya.

Armada laut dan udara yang sering beroperasi atau bahkan selalu beroperasi di daerah remote harus selalu dipantau keberadaannya untuk alasan keamanan. Dengan adanya sistem pengiriman dan penerimaan data GPS melalui sistem telekomunikasi berbasis satelit untuk armada laut dan udara yang beroperasi di daerah *remote*, maka keberadaan dari kedua armada tersebut akan dengan mudah diketahui secara cepat, tepat, dan *real time*.

Selain digunakan untuk armada laut dan udara, sistem pengiriman dan penerimaan data GPS untuk fungsi pemantauan melalui sistem telekomunikasi berbasis satelit ini juga bisa digunakan untuk armada darat. Salah satu contohnya adalah untuk armada kereta api. Mengingat kondisi geografis di wilayah Indonesia, maka sistem ini dapat diterapkan untuk armada kereta api. Meskipun sudah ada sistem pemantauan kereta api yang bersifat konvensional, namun terdapat beberapa kelemahan dari sistem yang sudah ada tersebut. Dengan adanya sistem pengiriman dan penerimaan data GPS untuk fungsi pemantauan melalui sistem telekomunikasi berbasis satelit ini, maka akan menggantikan atau melengkapi fungsi dari sistem yang sudah ada sebelumnya, yaitu sistem *track circuit*.

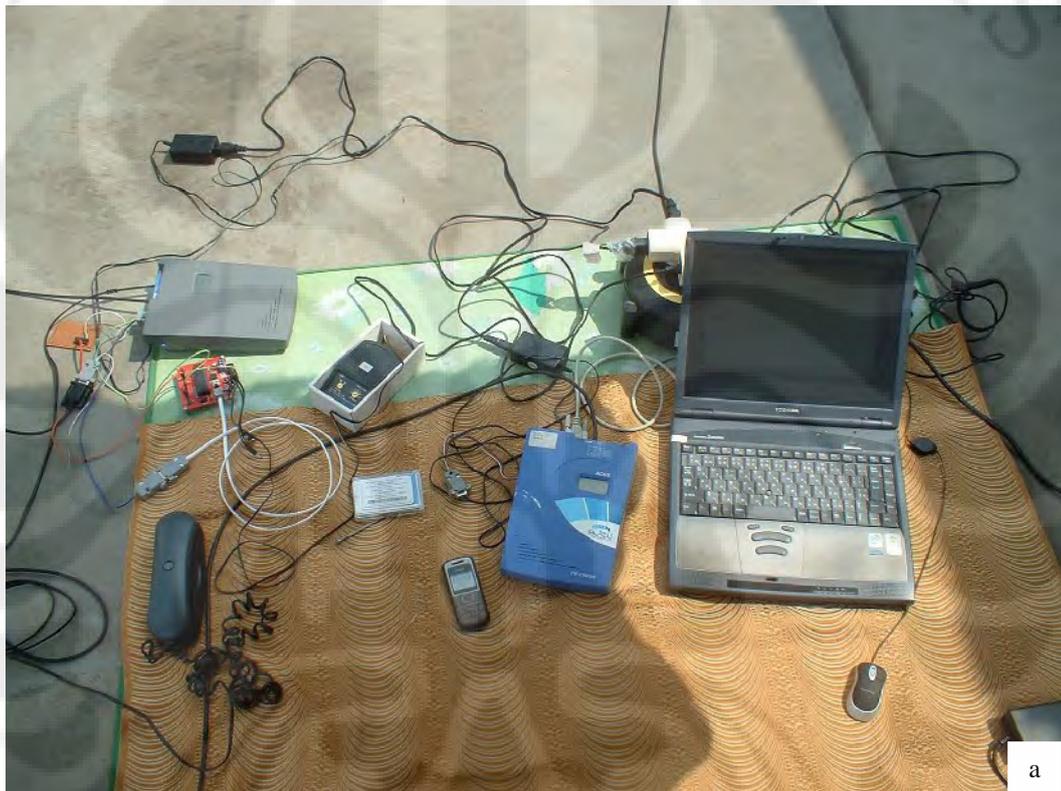
Selain sistem ini dapat digunakan untuk fungsi pemantauan armada, sistem ini juga dapat digunakan untuk fungsi pemantauan yang lain.

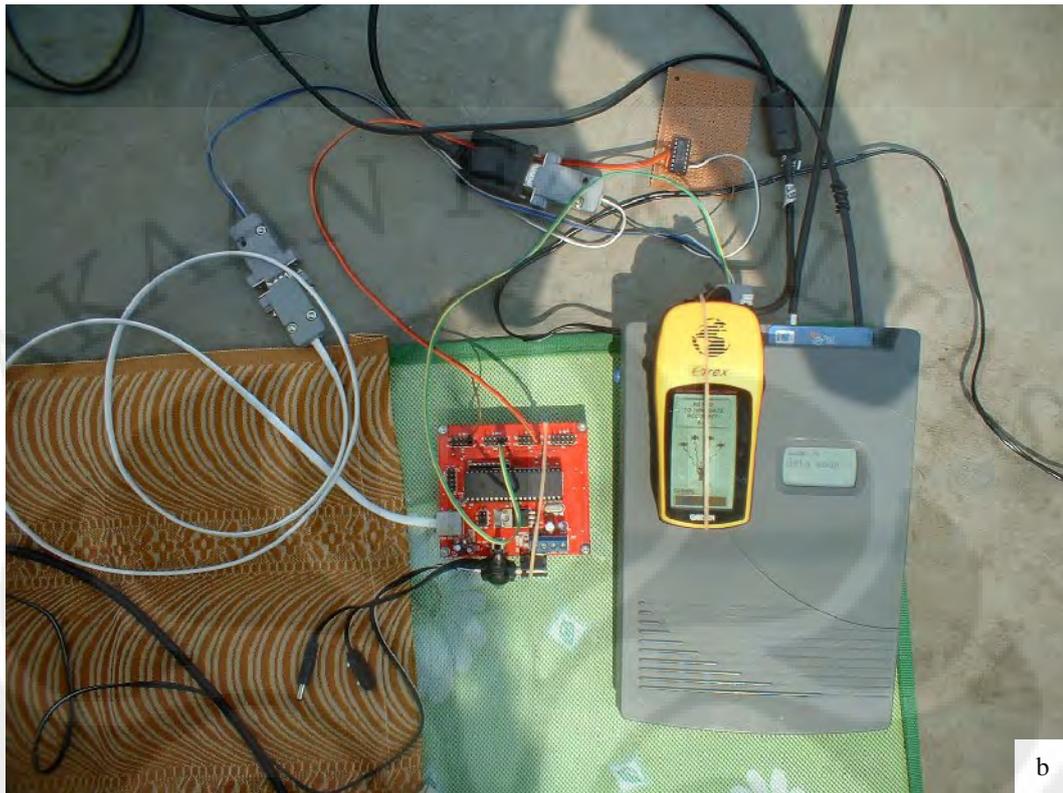
## BAB 4

### PENGUJIAN DAN ANALISIS SISTEM

#### 4.1 HASIL RANCANG BANGUN

Berdasarkan rancang bangun pada Bab 3, dibuatlah sistem secara keseluruhan. Gambar 4.1 memperlihatkan perangkat keras yang digunakan pada sistem yang dirancang pada Bab sebelumnya.





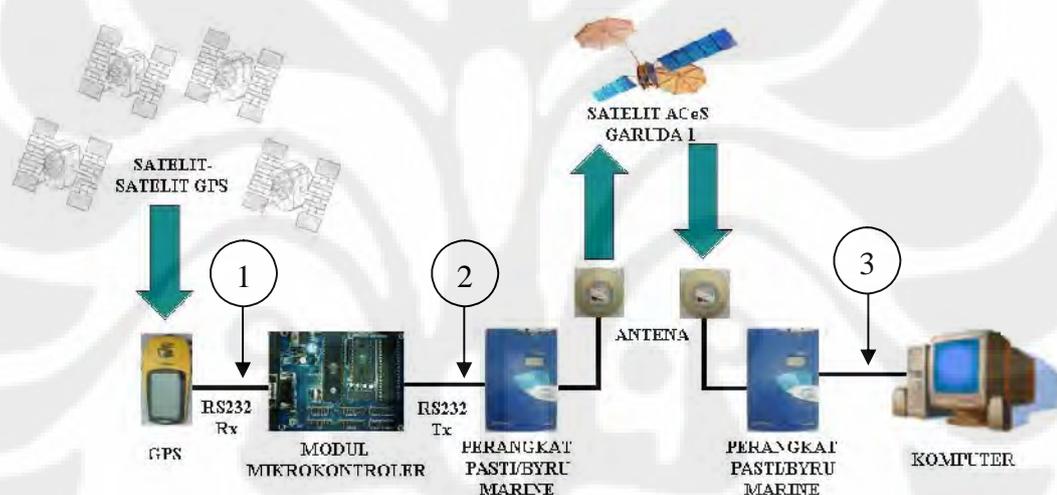
Gambar 4.1(a) Perangkat Keras Sistem Keseluruhan ;(b) Dan (c) Perangkat Keras Sistem Pada Sisi Pengirim

## 4.2 PENGUJIAN SISTEM

Pada sistem yang telah dibuat, dilakukan beberapa skenario pengujian, antara lain:

1. Menampilkan data keluaran GPS dengan format NMEA-0183.
2. Menampilkan data keluaran dari kontroler.
3. Menampilkan data yang telah dikirimkan pada sisi penerima.

Untuk mendapatkan hasil dari skenario pengujian yang dipaparkan pada poin-poin di atas, kami menentukan tempat melakukan uji coba. Tempat yang dimaksudkan adalah poin-poin yang ditunjukkan (poin 1, poin 2, dan poin 3) pada Gambar 4.2.



Gambar 4.2 Blok Diagram Tempat Pengujian Sistem Pengiriman Dan Penerimaan Data GPS Melalui Layanan Komunikasi Satelit Aces Garuda 1

### 4.2.1 Menampilkan Data Keluaran GPS

Menampilkan data keluaran GPS pada layar komputer bertujuan untuk mengetahui apakah GPS telah mengeluarkan data dari terminalnya atau belum. Ini dilakukan dengan cara menghubungkan konektor dari GPS dengan komputer melalui *port* serial PC. Pada komputer dijalankan program *hyper terminal* dan diset sesuai dengan konfigurasi pada GPS. Gambar 4.3 merupakan tampilan dari hasil pengujian keluaran data GPS pada *hyper terminal* dengan format NMEA-0183.

```

$GPRMB,A,0.00,L,,005.0609.789,S,10650.336,E,1.243,274.1,,V,A*54
$GPGGA,105536,0609.8789,S,10651.5812,E,1.07,1.3,26.4,M,2.6,M,,*56
$GPGSA,A,3,02,05,09,12,,,,,24,,29,30,3.0,1.3,2.7*35
$GPGSV,3,1,12,02,28,120,36,05,30,184,45,09,68,122,45,12,28,170,47*77
$GPGSV,3,2,12,14,13,232,00,15,25,009,00,18,08,315,00,21,13,337,00*75
$GPGSV,3,3,12,24,46,012,41,26,08,021,00,29,55,268,44,30,23,211,48*7C
$GPGLL,0609.8789,S,10651.5812,E,105536,A,A*52
$GPBOD,274.2,T,273.6,M,005,*71
$PGRME,7.2,M,14.2,M,16.0,M*2B
$PGRM2,87,f,3*24
$GPRTE,1,1,c,*37
$GPRMC,105538,A,0609.8789,S,10651.5812,E,0.0,235.3,070109,0.5,E,A*0D
$GPRMB,A,0.00,L,,005.0609.789,S,10650.336,E,1.243,274.1,,V,A*54
$GPGGA,105538,0609.8789,S,10651.5812,E,1.07,1.3,26.4,M,2.6,M,,*58
$GPGSA,A,3,02,05,09,12,,,,,24,,29,30,3.0,1.3,2.7*35
$GPGSV,3,1,12,02,28,120,36,05,30,184,45,09,68,119,45,12,28,170,47*7F
$GPGSV,3,2,12,14,13,232,00,15,25,009,00,18,08,315,00,21,13,337,00*75
$GPGSV,3,3,12,24,46,012,41,26,08,021,00,29,55,268,44,30,23,211,48*7C
$GPGLL,0609.8789,S,10651.5812,E,105538,A,A*5C
$GPBOD,274.2,T,273.6,M,005,*71
$PGRME,7.2,M,14.2,M,16.0,M*2B
$PGRM2,87,f,3*24
$GPRTE,1,1,c,*37

```

Gambar 4.3 Data Keluaran GPS Pada Kondisi Aktif

Format data GPS NMEA 0183 memiliki beberapa data berbeda. Terdapat dua belas pengiriman data sebelum kembali ke format data awal, yaitu \$GPRMC.....<CR><LF>. Siklus pengiriman data NMEA 0183 adalah 2 detik dihitung dari pengiriman data \$GPRMC awal sampai \$GPRMC berikutnya. Data posisi berada pada deretan data jika GPS bebas dari halangan (GPS dapat menerima sinyal dari satelit-satelit GPS). Gambar 4.4 memperlihatkan kondisi GPS tidak mendapatkan sinyal dari satelit-satelit GPS (*void*).

```

$GPRMB,V,,,,,V,N*04
$GPGGA,,,,,0,12,,,M,M,*65
$GPGSA,A,1,01,05,06,12,14,16,18,21,22,24,29,30,,,*19
$GPGSV,3,1,12,01,08,322,00,05,29,051,00,06,08,239,00,12,14,049,00*75
$GPGSV,3,2,12,14,12,345,00,16,16,209,00,18,54,061,00,21,50,175,31*7C
$GPGSV,3,3,12,22,39,001,00,24,15,150,00,29,19,133,00,30,49,068,00*75
$GPGLL,,,,,V,N*64
$GPBOD,T,M,*47
$PGRME,M,M,M*00
$PGRMZ,f,1*29
$GPRTE,1,1,c,*37
$GPRMC,V,,,,,150608,0.6,E,N*34
$GPRMB,V,,,,,V,N*04
$GPGGA,,,,,0,12,,,M,M,*65
$GPGSA,A,1,01,05,06,12,14,16,18,21,22,24,29,30,,,*19
$GPGSV,3,1,12,01,08,322,00,05,29,051,00,06,08,239,00,12,14,049,00*75
$GPGSV,3,2,12,14,12,345,00,16,16,209,00,18,54,061,00,21,50,175,32*7F
$GPGSV,3,3,12,22,39,001,00,24,15,150,00,29,19,133,00,30,49,068,00*75
$GPGLL,,,,,V,N*64
$GPBOD,T,M,*47
$PGRME,M,M,M*00
$PGRMZ,f,1*29
$GPRTE,1,1,c,*37

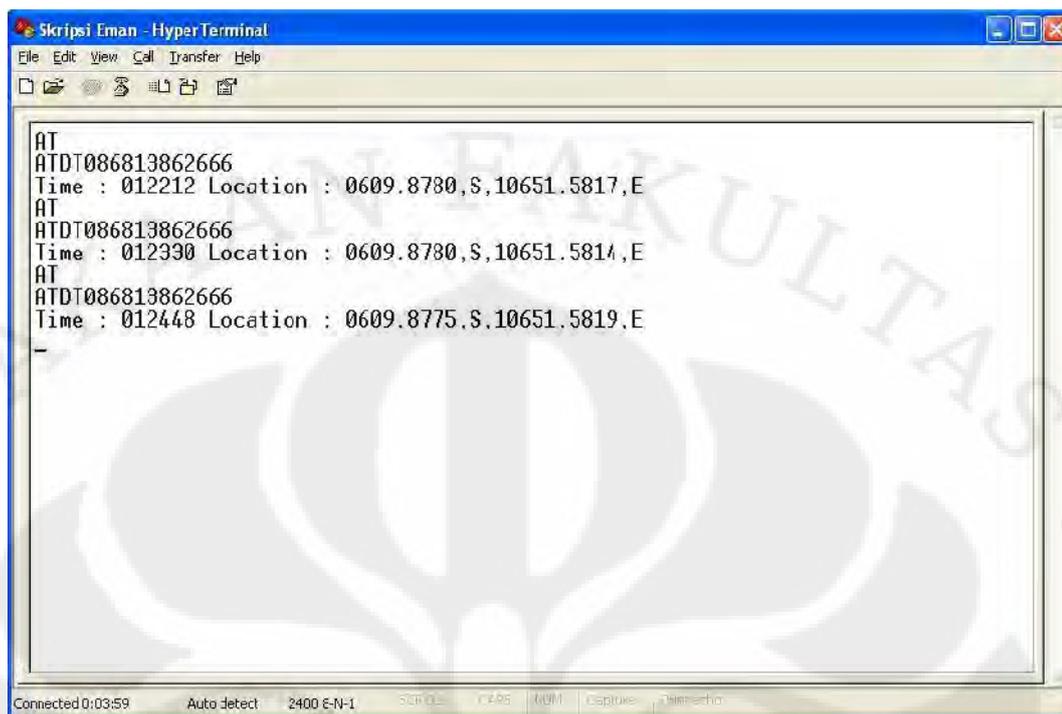
```

Gambar 4.4 Data Keluaran GPS Pada Kondisi Tanpa Sinyal (Void)

#### 4.2.2 Menampilkan Data Keluaran Kontroler

Pada pengujian ini dilakukan *masking* terhadap data mentah GPS seperti Gambar 4.3 pada bagian sebelumnya. Data mentah GPS yang *dimasking* adalah data GPRMC. Data mentah GPS yang digunakan adalah data GPS dengan format NMEA-0183. Setelah data GPRMC *dimasking*, selanjutnya data hasil *masking* tersebut akan disimpan di memori internal mikrokontroler. Setelah melalui beberapa tahapan proses yang terjadi di mikrokontroler, mikrokontroler akan mengirimkan data waktu dan data lokasi saja yang terdapat pada data GPRMC yang sebelumnya telah *dimasking* dan disimpan di memori internal mikrokontroler.

Langkah pengujian dilakukan dengan menghubungkan konektor data GPS dengan modul mikrokontroler yang telah diprogram melalui *port* serial dan menghubungkan mikrokontroler dengan PC melalui *port* serial. PC akan menampilkan data olahan dari mikrokontroler pada program *hyper terminal*. Gambar 4.5 memperlihatkan hasil olahan mikrokontroler yang hanya difungsikan untuk mengirimkan data waktu dan data lokasi saja yang ditampilkan pada *hyper terminal*.



Gambar 4.5 Tampilan Data Dari Mikrokontroler Pada *Hyper Terminal*

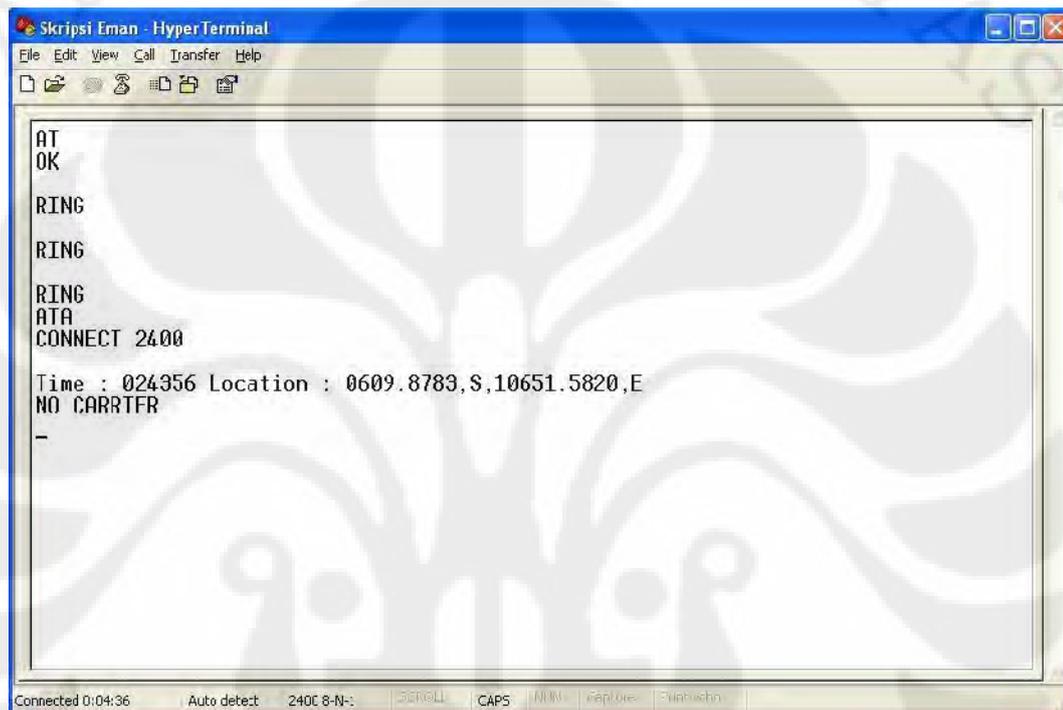
Gambar di atas memperlihatkan hanya data waktu dan data lokasi saja yang dikirimkan oleh mikrokontroler. Data waktu dan data lokasi ini didapatkan dari data GPRMC yang telah dimasking. Dapat dilihat pada baris ketiga bahwa data waktu yang dimaksud ditunjukkan oleh susunan angka 012212 yang berarti jam 01 menit 22 detik 12 dalam UTC. Sedangkan data lokasi yang dimaksud ditunjukkan oleh kombinasi antara angka dan huruf, yaitu 0609.8780,S,10651.5817,E yang berarti  $06^{\circ} 09,8780'$  lintang selatan dan  $106^{\circ} 51,5817$  bujur timur.

Pada baris keenam bahwa data waktu yang dimaksud ditunjukkan oleh susunan angka 012330 yang berarti jam 01 menit 23 detik 30 dalam UTC. Sedangkan data lokasi yang dimaksud ditunjukkan oleh kombinasi antara angka dan huruf, yaitu 0609.8780,S,10651.5814,E yang berarti  $06^{\circ} 09,8780'$  lintang selatan dan  $106^{\circ} 51,5814$  bujur timur.

Pada baris kesembilan bahwa data waktu yang dimaksud ditunjukkan oleh susunan angka 012448 yang berarti jam 01 menit 24 detik 48 dalam UTC. Sedangkan data lokasi yang dimaksud ditunjukkan oleh kombinasi antara angka dan huruf, yaitu 0609.8775,S,10651.5819,E yang berarti  $06^{\circ} 09,8775'$  lintang selatan dan  $106^{\circ} 51,5819$  bujur timur.

### 4.2.3 Menampilkan Data Yang Telah Diterima Pada Sisi Penerima

Pengujian ini bertujuan melihat data GPS yang telah dikirimkan ke satelit Garuda 1 dan diterima di sisi penerima melalui perangkat PASTI/*Byru Marine*. Gambar 4.6 memperlihatkan data GPS yang telah berhasil diterima oleh perangkat PASTI/*Byru Marine* pada sisi penerima. Dapat dilihat pada gambar tersebut bahwa data GPS yang diterima sama dengan data GPS yang dikirimkan.



Gambar 4.6 Tampilan Data Yang Diterima Di Sisi Penerima

Pada baris kedelapan bahwa data waktu yang dimaksud ditunjukkan oleh susunan angka 024356 yang berarti jam 02 menit 43 detik 56 dalam UTC. Sedangkan data lokasi yang dimaksud ditunjukkan oleh kombinasi antara angka dan huruf, yaitu 0609.8783,S,10651.5820,E yang berarti  $06^{\circ} 09,8783'$  lintang selatan dan  $106^{\circ} 51,5820'$  bujur timur.

### 4.3 ANALISIS SISTEM

Dari skenario pengujian yang dilakukan pada bagian 4.2, kita dapat menganalisis pengujian yang dilakukan di atas. Analisis yang dihasilkan adalah sebagai berikut.

#### 4.3.1 Analisis Pengujian Pertama

Dari hasil pengujian pertama ada dua kondisi keluaran dari GPS, yaitu pada kondisi aktif (GPS mendapat sinyal dari satelit-satelit GPS) dan kondisi *void* (GPS tidak mendapatkan sinyal yang cukup dari satelit-satelit GPS). Pada Gambar 4.4 beberapa data tidak dapat menunjukkan status posisi, waktu, dan kecepatannya dikarenakan kondisi *void* pada GPS. Untuk mendapatkan data-data seperti yang disebutkan di atas, GPS sekurang-kurangnya mendapatkan sinyal dari empat satelit GPS.

#### 4.3.2 Analisis Pengujian Kedua

Pada pengujian kedua, mikrokontroler dikondisikan untuk melakukan *masking* terhadap data GPRMC. Data GPRMC yang telah *dimasking* tadi selanjutnya akan disimpan di memori internal mikrokontroler. Setelah melalui beberapa tahapan proses yang terjadi di mikrokontroler, mikrokontroler akan mengirimkan data waktu dan data lokasi saja yang terdapat pada data GPRMC yang sebelumnya telah *dimasking* dan disimpan di memori internal mikrokontroler.

Besarnya waktu yang dibutuhkan mikrokontroler mulai dari pengambilan data GPS sampai dengan pengiriman data GPS apabila GPS diposisikan dalam keadaan tidak bergerak adalah sebagai berikut:

Perhitungan besarnya nilai 1 mesin *cycle* (T) adalah[7]:

$$T = \frac{12}{\text{frekuensi osilator}} \dots\dots\dots (4.1)$$

$$T = \frac{12}{11,0592\text{MHz}}$$

$$T = 1,0850\mu\text{s}$$

Apabila nilai ini dimasukkan sesuai dengan besarnya mesin *cycle* tiap instruksi, maka akan didapatkan waktu untuk pengambilan data GPRMC sebagai berikut:

$$\begin{aligned}
 &= 32,550 + (63 \times (\text{inst.1}+\text{inst.2}+\text{inst.5}+\text{inst.6}+\text{inst.7})) + \\
 &\quad (\text{inst.1}+\text{inst.2}+\text{inst.3}+\text{inst.4}) + 3,2550 \\
 &= 32,550 + (63 \times 16,275) + 13,020 + 3,255 \\
 &= 32,550 + 1025,325 + 13,020 + 3,255 \\
 &= 1074,15 \mu\text{s}
 \end{aligned}$$

Waktu yang dibutuhkan oleh mikrokontroler untuk melakukan proses komunikasi dengan perangkat PASTI/*Byru Marine* sebelum mengirimkan data GPS adalah sebagai berikut:

$$\begin{aligned}
 &= 50,9950 + 24,9550 + (3 \times (\text{inst.8}+\text{inst.9})) + 2,1700 + 162,750 + 24,9550 + \\
 &\quad (13 \times (\text{inst.11}+\text{inst.12})) + 2,1700 \\
 &= 50,9950 + 24,9550 + (3 \times 9,765) + 2,1700 + 162,750 + 24,9550 + (13 \times \\
 &\quad 9,765) + 2,1700 \\
 &= 50,9950 + 24,9550 + 29,2950 + 2,1700 + 162,750 + 24,9550 + 126,9450 \\
 &\quad + 2,1700 \\
 &= 424,235 \mu\text{s}
 \end{aligned}$$

Waktu yang dibutuhkan oleh mikrokontroler untuk mengirimkan data GPS yang telah diolah sebelumnya ke perangkat PASTI/*Byru Marine* adalah sebagai berikut:

$$\begin{aligned}
 &= 64,015 + (6 \times (\text{inst.14}+\text{inst.15}+\text{inst.16}+\text{inst.17})) + 107,4150 + (24 \times \\
 &\quad (\text{inst.18}+\text{inst.19}+\text{inst.20}+\text{inst.21})) + 22,785 + 36,4032 \times 10^6 + 2,1700 \\
 &= 64,015 + (6 \times 14,105) + 107,4150 + (24 \times 14,105) + 22,785 + 36,4032 \times \\
 &\quad 10^6 + 2,1700 \\
 &= 64,015 + 84,630 + 107,4150 + 338,520 + 22,785 + 36,4032 \times 10^6 + 2,1700 \\
 &= 36,40381954 \times 10^6 \mu\text{s}
 \end{aligned}$$

Dari perhitungan di atas, maka total waktu yang dibutuhkan oleh mikrokontroler mulai dari pengambilan data GPS sampai dengan pengiriman data GPS yang telah diolah ke perangkat PASTI/*Byru Marine* adalah sebagai berikut:

$$\begin{aligned}
 &= 1074,15 \mu\text{s} + 424,235 \mu\text{s} + 36,40381954 \times 10^6 \mu\text{s} \\
 &= 36,40531793 \times 10^6 \mu\text{s}
 \end{aligned}$$

= 36,4053 s

Dari perhitungan waktu di atas dapat diketahui bahwa mikrokontroler akan mengulangi prosesnya setelah 36,4053 detik ditambah dengan waktu yang dibutuhkan untuk proses *handshaking* yang berkisar 40 detik. Jadi, total waktu yang dibutuhkan oleh mikrokontroler untuk satu siklus pengiriman data GPS adalah sekitar 76 detik. Setelah 76 detik, maka mikrokontroler akan mengambil data GPS yang baru lagi.

Perhitungan waktu yang telah dijelaskan di atas berlaku ketika perangkat PASTI/*Byru Marine* pada sisi pengirim dalam keadaan siap untuk melakukan komunikasi dan perangkat PASTI/*Byru Marine* pada sisi penerima dengan segera melakukan konfirmasi terhadap panggilan dari perangkat PASTI/*Byru Marine* pada sisi pengirim. Jika tidak demikian, maka program akan melakukan pengulangan beberapa tahapan proses sampai tercapai kondisi yang diinginkan. Dengan demikian, waktu yang dibutuhkan oleh mikrokontroler untuk satu kali siklus pengiriman data GPS akan menjadi bertambah.

Rencana program yang dibuat ada sedikit perbedaan dengan program sebenarnya. Pada rencana program yang dibuat, mikrokontroler diperintahkan untuk mengirimkan data pada kondisi kecepatan yang terekam pada GPS tidak sama dengan nol. Namun, kami mengalami kesulitan untuk mencobanya dalam kondisi bergerak. Oleh karena itu, kami membuat program yang dapat mengirimkan data waktu dan data lokasi dengan kondisi kecepatan yang terekam pada GPS sama dengan nol. Itu artinya bahwa kami melakukan percobaan di lokasi yang tetap, bukan berada di atas armada yang sedang bergerak.

#### **4.3.3 Analisis Pengujian Ketiga**

Pada pengujian ketiga, dapat dilihat bahwa data yang telah diterima sama dengan data yang dikirimkan. Dapat dilihat bahwa data yang dikirimkan muncul setelah tulisan "CONNECT 2400". Data yang diterima adalah data waktu dan data lokasi. Data ini akan diterima setelah perangkat PASTI/*Byru Marine* pada sisi penerima dengan segera melakukan konfirmasi terhadap panggilan dari perangkat PASTI/*Byru Marine* pada sisi pengirim. Konfirmasi yang diberikan berupa pesan ATA. Dan setelah tulisan "CONNECT 2400", data yang dikirimkan akan terlihat pada layar monitor komputer.

## BAB 5

### KESIMPULAN

Berdasarkan hasil pengujian dan analisis sistem pada Bab 4, dapat diambil beberapa kesimpulan sebagai berikut:

1. Sesuai dengan tujuan dari tugas akhir, kami berhasil merancang dan membuat sistem pengiriman data GPS menggunakan perangkat mikrokontroler sebagai penerima, pengolah, dan pengirim data GPS, melalui perangkat PASTI/*Byru Marine* dengan memanfaatkan layanan satelit Garuda 1.
2. Mikrokontroler akan mengulangi prosesnya setelah 36,4032 detik ditambah dengan waktu yang dibutuhkan untuk proses *handshaking* yang berkisar 40 detik. Jadi, total waktu yang dibutuhkan oleh mikrokontroler untuk satu siklus pengiriman data GPS adalah sekitar 76 detik.
3. Rencana program yang dibuat ada sedikit perbedaan dengan program sebenarnya. Program sebenarnya dibuat untuk dapat mengirimkan data waktu dan data lokasi dengan kondisi kecepatan yang terekam pada GPS sama dengan nol atau dalam kondisi tidak bergerak. Hal ini disebabkan karena kami mengalami kesulitan untuk dapat mencobanya di atas armada yang sedang bergerak.
4. Sistem pengiriman dan penerimaan data GPS yang kami buat ini sangat bergantung pada kondisi cuaca . Jika kondisi cuaca sedang cerah, maka proses pengiriman dan penerimaan data GPS akan berjalan dengan lancar. Begitu pula jika sebaliknya. Hal ini disebabkan oleh penggunaan sistem telekomunikasi berbasis satelit.
5. Sistem pengiriman dan penerimaan data GPS melalui layanan telekomunikasi berbasis satelit ini sangat cocok untuk daerah *remote* yang tidak terjangkau. Salah satu aplikasi dari sistem ini adalah untuk pemantauan armada, terutama armada laut dan udara.

## DAFTAR REFERENSI

- [1] Jean Marie Zogg (2009). *GPS Basics*. Diakses 8 April 2009 dari microblox. <http://www.u-blox.com>
- [2] Peter, Joe. *NMEA Data*. Diakses 6 Mei 2009 dari National Marine Electronics Association. [www.nmea.org](http://www.nmea.org)
- [3] Taryudi. “Rancang Bangun Tampilan Data GPS pada Peta GIS dan Data Eksternal pada Monitor PC Berbasis Mikrokontroler”. Skripsi, Program Sarjana Fakultas Teknik UI, Depok, 2007, hal. 33.
- [4] Taryudi. “Rancang Bangun Tampilan Data GPS pada Peta GIS dan Data Eksternal pada Monitor PC Berbasis Mikrokontroler”. Skripsi, Program Sarjana Fakultas Teknik UI, Depok, 2007, hal. 26
- [5] Kenneth J Ayala, *The 8051 Microcontroller Architecture, Programming, and Application 2<sup>nd</sup> Edition* (New York: West Publishing Company, 1997), hal. 63
- [6] Taryudi. “Rancang Bangun Tampilan Data GPS pada Peta GIS dan Data Eksternal pada Monitor PC Berbasis Mikrokontroler”. Skripsi, Program Sarjana Fakultas Teknik UI, Depok, 2007, hal. 80
- [7] Kenneth J Ayala, *The 8051 Microcontroller Architecture, Programming, and Application 2<sup>nd</sup> Edition* (New York: West Publishing Company, 1997), hal. 60.

## DAFTAR PUSTAKA

Ayala, Kenneth J, *The 8051 Microcontroller Architecture, Programming, and Application 2<sup>nd</sup> Edition* (New York: West Publishing Company, 1997)

Jean Marie Zogg (2009). *GPS Basics*. Diakses 8 April 2009 dari microblox.  
<http://www.u-blox.com>

Peter, Joe. *NMEA Data*. Diakses 6 Mei 2008 dari National Marine Electronics Association.  
[www.nmea.org](http://www.nmea.org)

Taryudi. “ Rancang Bangun Tampilan Data GPS pada Perta GIS dan Data Eksternal pada Monitor PC Berbasis Mikrokontroler”. Skripsi , Program Sarjana Fakultas Teknik UI, Depok, 2007.

### LAMPIRAN 1 Program Pengolahan Data GPS

		Mesin	Cycle	Waktu( $\mu$ s)	Total Waktu( $\mu$ s)
terima:	mov	A,sbuf	2	2,1700	<b>7,5950</b>
	jnb	ri,\$	2	2,1700	
	clr	ri	1	1,0850	
	ret		2	2,1700	
kirim:	mov	sbuf,A	2	2,1700	<b>7,5950</b>
	jnb	ti,\$	2	2,1700	
	clr	ti	1	1,0850	
	ret		2	2,1700	
simpan:	mov	@r0,A	1	1,0850	<b>3,2550</b>
	ret		2	2,1700	
ambil:	mov	A,@r0	1	1,0850	<b>3,2550</b>
	ret		2	2,1700	
header1:	mov	A,#'A'	1	1,0850	<b>19,530</b>
	lcall	kirim		7,5950	
	mov	A,#'T'	1	1,0850	
	lcall	kirim		7,5950	
	ret		2	2,1700	
header2:	mov	A,#'A'	1	1,0850	<b>36,890</b>
	lcall	kirim		7,5950	
	mov	A,#'T'	1	1,0850	
	lcall	kirim		7,5950	
	mov	A,#'D'	1	1,0850	
	lcall	kirim		7,5950	
	mov	A,#'T'	1	1,0850	
	lcall	kirim		7,5950	
	ret		2	2,1700	
	nomor:	mov	A,#'0'	1	
lcall		kirim		7,5950	
mov		A,#'8'	1	1,0850	
lcall		kirim		7,5950	
mov		A,#'6'	1	1,0850	
lcall		kirim		7,5950	
mov		A,#'8'	1	1,0850	
lcall		kirim		7,5950	
mov		A,#'1'	1	1,0850	
lcall		kirim		7,5950	
mov		A,#'3'	1	1,0850	
lcall		kirim		7,5950	
mov		A,#'8'	1	1,0850	
lcall		kirim		7,5950	
mov		A,#'6'	1	1,0850	
lcall		kirim		7,5950	
mov		A,#'2'	1	1,0850	
lcall		kirim		7,5950	
mov	A,#'6'	1	1,0850		
lcall	kirim		7,5950		

	mov	A,#'6'	1	1,0850	
	lcall	kirim		7,5950	
	mov	A,#'6'	1	1,0850	
	lcall	kirim		7,5950	
	ret		2	2,1700	
time:	mov	A,#'T'	1	1,0850	
	lcall	kirim		7,5950	
	mov	A,#'i'	1	1,0850	
	lcall	kirim		7,5950	
	mov	A,#'m'	1	1,0850	
	lcall	kirim		7,5950	
	mov	A,#'e'	1	1,0850	
	lcall	kirim		7,5950	<b>62,930</b>
	mov	A,#32	1	1,0850	
	lcall	kirim		7,5950	
	mov	A,#':'	1	1,0850	
	lcall	kirim		7,5950	
	mov	A,#32	1	1,0850	
	lcall	kirim		7,5950	
	ret		2	2,1700	
location:	mov	A,#32	1	1,0850	
	lcall	kirim		7,5950	
	mov	A,#'L'	1	1,0850	
	lcall	kirim		7,5950	
	mov	A,#'o'	1	1,0850	
	lcall	kirim		7,5950	
	mov	A,#'c'	1	1,0850	
	lcall	kirim		7,5950	
	mov	A,#'a'	1	1,0850	
	lcall	kirim		7,5950	
	mov	A,#'t'	1	1,0850	
	lcall	kirim		7,5950	
	mov	A,#'i'	1	1,0850	
	lcall	kirim		7,5950	<b>106,330</b>
	mov	A,#'o'	1	1,0850	
	lcall	kirim		7,5950	
	mov	A,#'n'	1	1,0850	
	lcall	kirim		7,5950	
	mov	A,#32	1	1,0850	
	lcall	kirim		7,5950	
	mov	A,#':'	1	1,0850	
	lcall	kirim		7,5950	
	mov	A,#32	1	1,0850	
	lcall	kirim		7,5950	
	ret		2	2,1700	
baud24:	mov	TMOD,#20h	1	1,0850	
	mov	TH1,#0F4h	1	1,0850	<b>7,5950</b>
	mov	SCON,#52h	1	1,0850	

	mov	PCON,#00h	1	1,0850	
	mov	TCON,#40h	1	1,0850	
	ret		2	2,1700	
baud48:	mov	TMOD,#20h	1	1,0850	
	mov	TH1,#0FAh	1	1,0850	
	mov	SCON,#52h	1	1,0850	<b>7,5950</b>
	mov	PCON,#00h	1	1,0850	
	mov	TCON,#40h	1	1,0850	
	ret		2	2,1700	
delay:	djnz	r2,\$			
	djnz	r3,delay			
	djnz	r4,delay			
	ret		2	2,1700	
set_DTR:	mov	p1,#02h	1	1,0850	
	ret		2	2,1700	<b>3,2550</b>
clr_DTR:	mov	p1,#00h	1	1,0850	
	ret		2	2,1700	<b>3,2550</b>
zzz:	mov	A,#13	1	1,0850	
	lcall	kirim		7,5950	
	mov	A,#10	1	1,0850	<b>19,530</b>
	lcall	kirim		7,5950	
	ret		2	2,1700	
start:	mov	sp,#20h	1	1,0850	
aaa:	mov	r0,#35h	1	1,0850	
	lcall	baud48		9,7650	
	mov	p2,#00h	1	1,0850	<b>32,550</b>
aaa1:	lcall	terima		7,5950	
	cjne	A,#'M',aaa1	2	2,1700	
	lcall	terima		7,5950	
	cjne	A,#'C',aaa1	2	2,1700	
aaa2:	lcall	terima		7,5950	<b>1</b>
	cjne	A,#10,aaa3	2	2,1700	<b>2</b>
	mov	@r0,A	1	1,0850	<b>3</b>
	ljmp	aaa4	2	2,1700	<b>4</b>
aaa3:	lcall	simpan		3,2550	<b>5</b>
	inc	r0	1	1,0850	<b>6</b>
	ljmp	aaa2	2	2,1700	<b>7</b>
aaa4:	mov	r0,#37h	1	1,0850	<b>3,2550</b>
	cjne	@r0,#'V',bbb	2	2,1700	
	ljmp	aaa	2	2,1700	<b>2,1700</b>
bbb:	lcall	baud24		7,5950	
	lcall	set_DTR		3,2550	
	mov	p2,#02h	1	1,0850	<b>50,9950</b>
	lcall	header1		19,530	
	lcall	zzz		19,530	
bbb1:	lcall	terima		7,5950	
	lcall	terima		7,5950	<b>24,9550</b>
	lcall	terima		7,5950	

	cjne	A,#'O',bbb12	2	2,1700	
bbb11:	lcall	terima		7,5950	<b>8</b>
	cjne	A,#10,bbb11	2	2,1700	<b>9</b>
	ljmp	bbb2	2	2,1700	<b>10</b>
bbb12:	lcall	terima		7,5950	
	cjne	A,#10,bbb12	2	2,1700	
	ljmp	bbb	2	2,1700	
bbb2:	lcall	header2		36,890	
	lcall	nomor		106,33	<b>162,75</b>
	lcall	zzz		19,530	
bbb21:	lcall	terima		7,5950	
	lcall	terima		7,5950	<b>24,9550</b>
	lcall	terima		7,5950	
	cjne	A,#'C',bbb23	2	2,1700	
bbb22:	lcall	terima		7,5950	<b>11</b>
	cjne	A,#10,bbb22	2	2,1700	<b>12</b>
	ljmp	ccc	2	2,1700	<b>13</b>
bbb23:	lcall	terima		7,5950	
	cjne	A,#10,bbb23	2	2,1700	
	ljmp	bbb	2	2,1700	
ccc:	mov	r0,#36h	1	1,0850	<b>64,015</b>
	lcall	time		62,930	
ccc1:	lcall	ambil		3,2550	<b>14</b>
	lcall	kirim		7,5950	<b>15</b>
	inc	r0	1	1,0850	<b>16</b>
	cjne	r0,#3Ch,ccc1	2	2,1700	<b>17</b>
ccc2:	mov	r0,#3Fh	1	1,0850	<b>107,4150</b>
	lcall	location		106,33	
ccc3:	lcall	ambil		3,2550	<b>18</b>
	lcall	kirim		7,5950	<b>19</b>
	inc	r0	1	1,0850	<b>20</b>
	cjne	r0,#57h,ccc3	2	2,1700	<b>21</b>
	lcall	zzz		19,530	
	lcall	clr_DTR		3,2550	<b>22,785</b>
	lcall	delay			<b>36,4032 x 10<sup>6</sup></b>
	ljmp	aaa	2	2,1700	<b>2,1700</b>
	end				

## LAMPIRAN 2 Tabel Karakter ASCII

Char	Dec	Oct	Hex
(nul) Null	0	0000	0x00
(soh) Start of heading	1	0001	0x01
(stx) Start of text	2	0002	0x02
(etx) End of text	3	0003	0x03
(eot) End of transmission	4	0004	0x04
(enq) Enquiry	5	0005	0x05
(ack) Acknowledge	6	0006	0x06
(bel) Bell	7	0007	0x07
(bs) Backspace	8	0010	0x08
(ht) Horizontal tab	9	0011	0x09
(nl) New line, line feed	10	0012	0x0a
(vt) Vertical tab	11	0013	0x0b
(np) New page, form feed	12	0014	0x0c
(cr) Carriage return	13	0015	0x0d
(so) Shift out	14	0016	0x0e
(si) Shift in	15	0017	0x0f
(dle) Data link escape	16	0020	0x10
(dc1) Device control 1	17	0021	0x11
(dc2) Device control 2	18	0022	0x12
(dc3) Device control 3	19	0023	0x13
(dc4) Device control 4	20	0024	0x14
(nak) Negative Acknowledge	21	0025	0x15
(syn) Synchronous idle	22	0026	0x16
(etb) End of transmission block	23	0027	0x17
(can) Cancel	24	0030	0x18
(em) End of medium	25	0031	0x19
(sub) Substitute	26	0032	0x1a
(esc) Escape	27	0033	0x1b
(fs) File separator	28	0034	0x1c
(gs) Group separator	29	0035	0x1d
(rs) Record separator	30	0036	0x1e
(us) Unit separator	31	0037	0x1f
(sp) Space	32	0040	0x20
!	33	0041	0x21
"	34	0042	0x22
#	35	0043	0x23
\$	36	0044	0x24
%	37	0045	0x25
&	38	0046	0x26
'	39	0047	0x27
(	40	0050	0x28
)	41	0051	0x29
*	42	0052	0x2a

Char	Dec	Oct	Hex
+	43	0053	0x2b
,	44	0054	0x2c
-	45	0055	0x2d
.	46	0056	0x2e
/	47	0057	0x2f
0	48	0060	0x30
1	49	0061	0x31
2	50	0062	0x32
3	51	0063	0x33
4	52	0064	0x34
5	53	0065	0x35
6	54	0066	0x36
7	55	0067	0x37
8	56	0070	0x38
9	57	0071	0x39
:	58	0072	0x3a
;	59	0073	0x3b
<	60	0074	0x3c
=	61	0075	0x3d
>	62	0076	0x3e
?	63	0077	0x3f
@	64	0100	0x40
A	65	0101	0x41
B	66	0102	0x42
C	67	0103	0x43
D	68	0104	0x44
E	69	0105	0x45
F	70	0106	0x46
G	71	0107	0x47
H	72	0110	0x48
I	73	0111	0x49
J	74	0112	0x4a
K	75	0113	0x4b
L	76	0114	0x4c
M	77	0115	0x4d
N	78	0116	0x4e
O	79	0117	0x4f
P	80	0120	0x50
Q	81	0121	0x51
R	82	0122	0x52
S	83	0123	0x53
T	84	0124	0x54
U	85	0125	0x55
V	86	0126	0x56
W	87	0127	0x57

Char	Dec	Oct	Hex
X	88	0130	0x58
Y	89	0131	0x59
Z	90	0132	0x5a
[	91	0133	0x5b
\	92	0134	0x5c
]	93	0135	0x5d
^	94	0136	0x5e
_	95	0137	0x5f
`	96	0140	0x60
a	97	0141	0x61
b	98	0142	0x62
c	99	0143	0x63
d	100	0144	0x64
e	101	0145	0x65
f	102	0146	0x66
g	103	0147	0x67
h	104	0150	0x68
i	105	0151	0x69
j	106	0152	0x6a
k	107	0153	0x6b
l	108	0154	0x6c
m	109	0155	0x6d
n	110	0156	0x6e
o	111	0157	0x6f
p	112	0160	0x70
q	113	0161	0x71
r	114	0162	0x72
s	115	0163	0x73
t	116	0164	0x74
u	117	0165	0x75
v	118	0166	0x76
w	119	0167	0x77
x	120	0170	0x78
y	121	0171	0x79
z	122	0172	0x7a
{	123	0173	0x7b
	124	0174	0x7c
}	125	0175	0x7d
~	126	0176	0x7e
(del) Delete	127	0177	0x7f