



TUGAS AKHIR - TE090362

**PROTOTYPE SISTEM MONITORING BERAT DAN POSISI KENDARAAN
BERMUATAN DENGAN MENGGUNAKAN PC**

Muhammad Reza Adzani
NRP 2211 030 042
Uinsa Pradana
NRP 2210 030 079

Dosen Pembimbing
Slamet Budiprayitno, ST., MT

PROGRAM D3 TEKNIK ELEKTRO
JURUSAN TEKNIK ELEKTRO
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember
Surabaya 2014



FINAL PROJECT - TE090362

***PROTOTYPE OF WEIGHT AND POSITION MONITORING SYSTEM OF
WEIGHTED VEHICLE BY USING PC***

Muhammad Reza Adzani
NRP 2211 030 042
Uinsa Pradana
NRP 2211 030 079

Counsellor
Slamet Budiprayitno, ST., MT

*Electrical Engineering D3 Program
Industrial Technology Faculty
Institut Teknologi Sepuluh Nopember
Surabaya 2014*

PROTOTYPE SISTEM *MONITORING* BERAT DAN POSISI KENDARAAN BERMUATAN DENGAN MENGGUNAKAN PC

Nama Mahasiswa 1 : Muhammad Reza Adzani
NRP : 2211030042
Nama Mahasiswa 2 : Uinsa Pradana
NRP : 2211030079
Dosen Pembimbing : Slamet Budiprayitno,ST.,MT
NIP : 19781113 201012 1 002

ABSTRAK

Proses pengiriman suatu barang di industri terkadang membutuhkan waktu yang sangat lama karena jarak yang ditempuh. Banyak rintangan yang akan dihadapi ketika pengiriman suatu barang yang ada di industri. Salah satu rintangan yang dihadapi antara lain terjadinya pencurian terhadap barang yang dikirim. Pencurian barang tersebut terkadang tidak diketahui jumlah barang yang dicuri serta posisi saat barang tersebut dicuri.

Oleh karena itu diciptakan suatu sistem yang bertujuan sebagai *monitoring* berat serta posisi dari kendaraan bermuatan. Sistem ini terdiri dari beberapa komponen yaitu : *power supply*, sensor *loadcell*, *op-amp*, mikrokontroler Atmega16, RS232, serta GPS. *Power supply* +5volt akan terhubung ke sensor *loadcell* dan GPS. *Loadcell* dihubungkan ke rangkaian *op-amp* sebelum terhubung ke mikrokontroler. Sedangkan GPS akan langsung terhubung ke mikrokontroler. Kemudian dari dari *loadcell* dan GPS yang masuk ke mikrokontroler akan diteruskan ke rangkaian RS232 untuk disambungkan ke modem (*transmitter*). Dari modem (*transmitter*) akan dihubungkan ke modem (*receiver*) untuk ditampilkan ke Delphi.

Setelah melalui proses pengujian, data yang didapat yaitu untuk *loadcell* mengalami error sebesar 0,2 %, dalam artian untuk 1volt nilai yang keluar sebesar 1Kg. sedangkan data GPS yang didapat saat pengujian mengalami *error* sebesar 0,002% untuk *latitude* dan untuk *longitude* mengalami *error* sebesar 0,004%.

Kata kunci : GPS, Loadcell, Mikrokontroler Atmega16, SMS Gateway, Op-Amp



Halaman ini sengaja dikosongkan

PROTOTYPE OF WEIGHT AND POSITION MONITORING SYSTEM OF WEIGHTED VEHICLE BY USING PC

Nama Mahasiswa 1 : Muhammad Reza Adzani
NRP : 2211030042
Nama Mahasiswa 2 : Uinsa Pradana
NRP : 2211030079
Dosen Pembimbing : Slamet Budiprayitno,ST.,MT
NIP : 19781113 201012 1 002

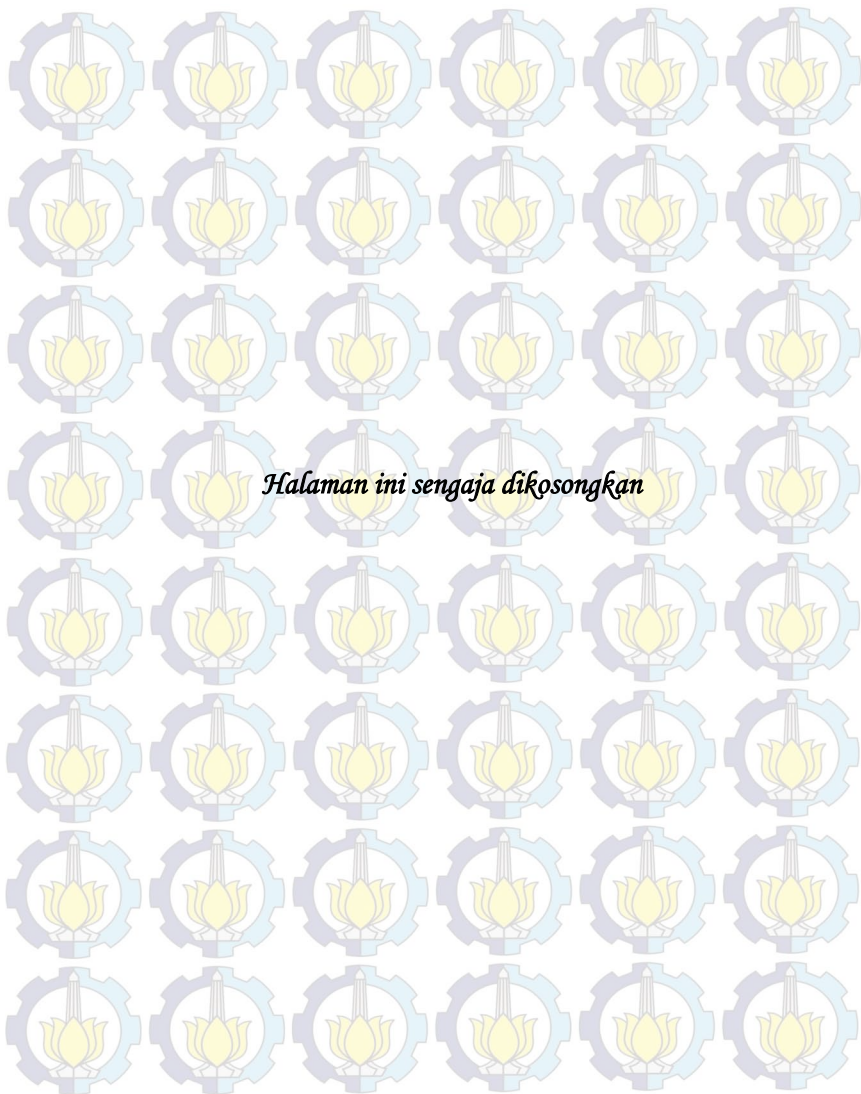
ABSTRACT

The process of sending an item in the industry can sometimes take a very long time because of the distance. Many obstacles to be faced when shipping an item that is in the industry. One of the obstacles faced by, among others, the theft of the goods shipped. Theft of goods in delivering process is sometimes unknown the quantity of goods as well as the current position of the item stolen.

Therefore created a system that is intended as a payload weight monitoring and monitoring the position of the vehicle. The system consists of several components: power supply, loadcell sensors, op-amps, microcontroller, RS232, as well as GPS. +5 Volt power supply will be connected to a load cell sensor and a GPS. Loadcell connected to the op-amp circuit before connecting to the microcontroller. While the GPS will be directly connected to the microcontroller. Then from out of loadcell and GPS are included into the microcontroller will be forwarded to the RS232 circuit to be connected to the modem (transmitter). of the modem (transmitter) will be connected to the modem (receiver) to be displayed to Delphi.

After going through the testing process, the data obtained is for loadcell has an error of 0,2%, meaning that for 1volt value out of 1Kg. while the GPS data obtained while testing has an error of 0.002% for the latitude and longitude for experiencing an error of 0.004%.

Keywords : GPS, Loadcell, Microcontroller, Atmega16, SMS Gateway, Op-Amp



**PROTOTYPE SISTEM MONITORING BERAT DAN POSISI
KENDARAAN BERMUATAN DENGAN MENGGUNAKAN PC**

TUGAS AKHIR

**Diajukan Guna Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar Ahli Madya**

Pada

**Bidang Studi Komputer Kontrol
Program Studi D3 Teknik Elektro
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember**

**Menyetujui :
Dosen Pembimbing,**

Slamet Budiprayitno, ST., MT

NIP. 19781113 201012 1 002

**SURABAYA
JULI, 2014**

KATA PENGANTAR

Puji syukur penulis panjatkan ke hadirat Allah SWT yang telah melimpahkan segala rahmat dan hidayahNya sehingga penulis dapat menyelesaikan tugas akhir dengan judul :

“ PROTOTIPE SISTEM MONITORING BERAT DAN POSISI KENDARAAN BERMUATAN DENGAN MENGGUNAKAN PC ”

Dengan penyusunan tugas akhir ini penyusun tidak terlepas dari bantuan dari berbagai pihak. Oleh karena itu, pada kesempatan ini penulis hendak menyampaikan rasa terima kasih kepada pihak – pihak yang telah membantu. Ucapan terima kasih ini kami sampaikan kepada :

1. Allah SWT yang telah memberi rahmatNya dalam pembuatan tugas akhir ini.
2. Kedua orang tua yang telah memberikan dukungan penuh dan teladan bagi penulis.
3. Bapak Slamet Budiprayitno,ST.,MT selaku Dosen Pembimbing dan Koordinator Program Studi D3 Teknik Elektro Jurusan Teknik Elektro FTI ITS Surabaya, yang telah meluangkan waktunya untuk membimbing penulis dalam pembuatan tugas akhir ini.
4. Teman – teman 2011 atas semangat dan kerjasamanya.

Akhir kata semoga buku ini dapat bermanfaat bagi kita semua.

Surabaya, 10 Juli 2014

Penulis



Halaman ini sengaja dikosongkan

DAFTAR ISI

JUDUL	i
PENGESAHAN	iii
ABSTRAK	v
ABSTRACT	vii
KATA PENGANTAR	ix
DAFTAR ISI	xi
DAFTAR GAMBAR	xiii
DAFTAR TABEL	xv
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Permasalahan	2
1.3 Tujuan	2
1.4 Batasan Masalah	2
1.5 Sistematika Penulisan	3
1.6 Relevansi	3
BAB II TEORI PENUNJANG	5
2.1 Mikrokontroler Atmega16	5
2.2 GPS	7
2.2.1 Parallax GPS Module PMB-688	9
2.3 SMS Gateway	10
2.4 Modem GSM	11
2.4.1 Modem Wavecom <i>Fastrack</i> M1306B	11
2.5 AT-Command	12
2.6 Database	13
2.6.1 Microsoft Access	14
2.7 Loadcell	15
2.8 Op-Amp	16
2.8.1 INA125	16
BAB III PERANCANGAN DAN PEMBUATAN ALAT	19
3.1 Perancangan <i>Hardware</i>	19
3.1.1 Perancangan <i>Power Supply</i>	20
3.1.2 Perancangan Sistem Minimum ATmega16	21
3.1.3 Perancangan Rangkaian <i>Driver</i> RS232	22
3.1.4 Modul GPS <i>Tracker</i> PMB – 688 GPS SiRF	23

3.1.5 Modem GSM Wavecom	26
3.1.6 Perancangan Rangkaian Op-Amp	27
3.1.7 Sensor <i>Loadcell</i>	28
3.2 Perancangan <i>Software</i>	28
3.3 Perancangan Mekanik	32
BAB IV PENGUJIAN DAN ANALISA DATA.....	35
4.1 Pengujian GPS	35
4.2 Pengujian Rangkaian <i>Power Supply</i>	39
4.3 Pengujian Modem GSM.....	39
4.4 Pengujian Mikrokontroler ATmega16	42
4.5 Pengujian <i>Loadcell</i> dengan Op-amp	43
4.6 Pengujian Ketelitian Bit Pada ADC	44
4.7 Pengujian GPS Secara Serial.....	45
4.8 Pengujian Data Penguatan <i>Loadcell</i> secara serial	46
4.9 Pengujian Tingkat Keakuratan Berat Beban dengan Teori ...	47
4.10 Pengujian Berat Beban Via SMS.....	48
4.11 Pengujian Pengiriman SMS ke Delphi Secara Manual.....	49
4.12 Pengujian Sistem Keseluruhan	50
4.13 Deskripsi Tujuan Alat.....	51
BAB V PENUTUP	53
5.1 Kesimpulan.....	53
5.2 Saran.....	53
DAFTAR PUSTAKA	55
LAMPIRAN 1 <i>Listing Program</i>	A-1
LAMPIRAN 2 <i>Datasheet</i>	B-1
RIWAYAT HIDUP PENULIS	C-1

DAFTAR GAMBAR

Gambar 2.1	Mikrokontroler ATmega16	5
Gambar 2.2	Pin-pin ATmega16.....	7
Gambar 2.3	Proses Cara kerja GPS	8
Gambar 2.4	Parallax GPS <i>Module</i> PMB-688	9
Gambar 2.5	Arsitektur SMS <i>content provider</i>	10
Gambar 2.6	Wavecom <i>Fastrack</i> M1306B.....	11
Gambar 2.7	<i>Loadcell</i> 5Kg.....	16
Gambar 2.8	INA125	17
Gambar 3.1	Diagram Alat Secara Keseluruhan	19
Gambar 3.2	Rangkaian <i>Power Supply</i>	20
Gambar 3.3	Rangkaian Mikrokontroler ATmega16.....	22
Gambar 3.4	RS232 <i>Pin Out</i>	23
Gambar 3.5	Rangkaian <i>Driver</i> RS232.....	23
Gambar 3.6	GPS <i>Tracker</i> PMB-688 GPS SiRF	24
Gambar 3.7	Modem GSM <i>Fastrack</i> M1306B.....	26
Gambar 3.8	Rangkaian INA125	27
Gambar 3.9	Sensor <i>Loadcell</i>	28
Gambar 3.10	Tampilan Delphi 7	29
Gambar 3.11	<i>Flowchart</i> Kirim Data.....	30
Gambar 3.12	<i>Flowchart</i> Terima Data.....	31
Gambar 3.13	Tampilan Keseluruhan Alat	32
Gambar 3.14	Tampilan <i>Loadcell</i> Pada Rangka Besi	33
Gambar 4.1	Tampilan <i>Longitude</i> Dan <i>Latitude</i>	35
Gambar 4.2	Tampilan Koordinat di <i>Google Maps</i>	36
Gambar 4.3	Pengujian Modem GSM Secara <i>Serial</i>	40
Gambar 4.4	Pengujian Pengecekan Jumlah Saldo.....	40
Gambar 4.5	Mengirim SMS ke <i>Handphone</i>	40
Gambar 4.6	Jumlah Pesan Yang Ada Pada <i>Inbox</i> Modem	41
Gambar 4.7	Informasi Dasar Modem	41
Gambar 4.8	Tampilan Data GPS pada <i>Hyperterminal</i>	46
Gambar 4.9	Tampilan Data Berat Beban Pada <i>Hyperterminal</i>	46
Gambar 4.10	Pengujian Berat Beban Pada Saat 0 Kg	47
Gambar 4.11	Pengujian Berat Beban Pada Saat 2,5 Kg	48
Gambar 4.12	Tampilan Pesan Pada Delphi	49
Gambar 4.13	Tampilan Delphi Pada Saat Bekerja	50



Halaman ini sengaja dikosongkan

DAFTAR TABEL

Tabel 3.1	Fitur Dan Spesifikasi GPS	24
Tabel 4.1	Perbandingan Data Koordinat	36
Tabel 4.2	Pengukuran Tegangan <i>Power Supply</i>	39
Tabel 4.3	Pengukuran Tegangan Pada <i>Port A, Port C, Port D</i>	42
Tabel 4.4	Data penguatan <i>Loadcell</i> Dengan IC Op-Amp INA125	43
Tabel 4.5	Data Berat Beban Secara Teori Dan Praktek	48
Tabel 4.6	Data Berat Beban Pada LCD Dan Delphi 7	49
Tabel 4.7	<i>Database Monitoring</i>	51





Halaman ini sengaja dikosongkan

BAB I

PENDAHULUAN

1.1 Latar Belakang

Seiring dengan berkembangnya zaman, situs jual-beli *online* bermunculan dengan pesat. Situs jual-beli *online* sangat mudah digunakan karena dapat mencari dan memilih barang yang dibutuhkan tanpa harus bepergian karena bisa diakses dimana saja serta kapan saja. Banyak orang mengakses situs jual-beli *online* untuk mencari barang, membeli barang, maupun untuk menjual barang. Dengan semakin banyaknya transaksi jual-beli *online*, semakin banyak pula proses pengiriman barang yang dilakukan oleh jasa pengiriman untuk mengirim barang pesanan.

Proses pengiriman barang yang dilakukan oleh jasa pengiriman barang terkadang membutuhkan waktu yang lama tergantung jarak yang ditempuh. Selain membutuhkan waktu yang lama, terdapat resiko lain yang dihadapi jasa pengiriman barang. Salah satu resiko yang dihadapi saat pengiriman barang yaitu terjadinya pencurian. Terkadang pencurian tersebut tidak diketahui jumlah barang yang dicuri maupun lokasi saat barang dicuri oleh sopir. Meskipun sering terjadi pencurian, pihak jasa pengiriman kurang melengkapi sistem pendukung untuk mencegah terjadinya pencurian.

Dengan masalah yang ada, untuk itu dibuatlah prototipe sistem yang bertujuan untuk memonitoring kendaraan yang digunakan oleh jasa pengiriman pada saat mengirim barang. Prototipe sistem ini berfungsi untuk mengetahui jumlah barang yang dicuri serta posisi kendaraan saat terjadi pencurian. Komponen yang digunakan pada prototipe ini yaitu : Accu, *Power Supply*, Sensor *Loadcell*, Op-Amp, Mikrokontroler, RS232, Modem (*Receiver* dan *Transmitter*), serta GPS.

Accu yang memiliki tegangan +12Volt berfungsi sebagai sumber tegangan untuk prototipe sistem ini. Accu dihubungkan ke *Power Supply*. *Power Supply* berfungsi untuk membagi tegangan dari Accu menjadi 2 tegangan, yaitu +5Volt dengan +12Volt. Tegangan +5Volt dihubungkan ke Sensor *Loadcell* dan GPS. Sensor *Loadcell* berfungsi untuk mengukur berat beban. Sensor *Loadcell* kemudian dihubungkan ke Op-Amp. Op-Amp pada prototipe sistem ini menggunakan IC INA125 yang berfungsi sebagai penguat tegangan dari *Loadcell*. Dari Op-Amp dihubungkan

ke Mikrokontroler port A0. GPS pada prototipe ini berfungsi untuk memonitoring posisi kendaraan. GPS setelah mendapatkan tegangan +5Volt akan terhubung ke Mikrokontroler port D0 dan D1. Kemudian Mikrokontroler dihubungkan ke RS232. Dari RS232 dihubungkan ke Modem (*transmitter*) untuk mengirim data ke Modem (*receiver*). Data dari modem (*receiver*) akan ditampilkan ke delphi

Tujuan dari pembuatan prototipe ini yaitu sebagai monitoring berat muatan dan posisi pada kendaraan bermuatan yang terhubung ke PC. Untuk monitoring berat muatan, prototipe ini menggunakan Sensor *Loadcell*. Sedangkan untuk monitoring posisi, prototipe ini menggunakan GPS. Jika terjadi perubahan berat muatan dan posisi kendaraan, maka data dari *Loadcell* dan GPS akan mengirim ke Mikrokontroler. Kemudian data dari mikrokontroler akan tampil ke delphi

1.2 Permasalahan

Melihat latar belakang di atas, maka dirumuskan permasalahan antara lain :

1. Belum tersedianya alat yang berfungsi untuk memonitoring berat beban dan posisi pada kendaraan bermuatan secara *real time*.
2. Belum tersedianya alat yang berfungsi untuk memonitoring berat beban dan posisi pada kendaraan bermuatan yang terhubung ke *database*.

1.3 Tujuan

Tujuan dari tugas akhir ini adalah :

1. Dapat memonitoring dan mengetahui posisi koordinat dan berat muatan pada kendaraan bermuatan sehingga kendaraan bermuatan dapat terpantau secara *realtime*.

1.4 Batasan Masalah

Batasan masalah dari Prototipe Sistem Monitoring Berat dan Posisi Kendaraan Bermuatan dengan Menggunakan PC adalah :

1. Mikrokontroler yang digunakan dalam tugas akhir ini adalah mikrokontroler ATmega16 dengan bahasa pemrograman delphi 7 sebagai media penampil.
2. Modul GPS yang digunakan adalah Parallax PMB-688.

3. Modem yang digunakan adalah Modem GSM Wavecom Fastrack M1306B.
4. Operator yang digunakan adalah IM3.
5. Sensor pengukur berat yang digunakan yaitu *loadcell* dengan beban maksimal 5kg.
6. *Database* yang digunakan adalah Microsoft *Access Database*.
7. Data yang diterima oleh modem (*transmitter*) maksimal 10 data
8. Pengujian data sekitar wilayah Institut Teknologi Sepuluh Nopember Surabaya

1.5 Sistematika Penulisan

Dalam penyusunan buku tugas akhir ini, pembahasan mengenai sistem alat yang dibuat disusun dengan sistematika sebagai berikut :

1. BAB I PENDAHULUAN membahas tentang latar belakang, permasalahan, tujuan, batasan masalah, sistematika penulisan serta relevansi yang digunakan dalam tugas akhir yang dibuat.
2. BAB II TEORI PENUNJANG menjelaskan dasar teori yang berisi tentang konsep yang dijadikan landasan dan mendukung dalam perencanaan serta pembuatan alat yang dibuat.
3. BAB III PERANCANGAN SISTEM dalam bab ini membahas tentang perencanaan dan pembuatan perangkat keras (*hardware*) yang terdiri atas rangkaian elektronika, desain mekanik serta perangkat lunak (*software*) yang terdiri atas program yang akan digunakan untuk menjalankan alat tersebut.
4. BAB IV PENGUJIAN DAN ANALISA SISTEM membahas tentang pengujian alat dan analisa data yang didapat dalam pengujian alat.
5. BAB V PENUTUP berisi tentang kesimpulan alat dari tugas akhir ini dan saran-saran untuk pengembangan alat ini selanjutnya.

1.6 Relevansi

Hasil yang diperoleh dari pembuatan alat tugas akhir ini diharapkan dapat memberikan manfaat yaitu membantu perusahaan yang bergerak dalam bidang jasa pengiriman barang. Dengan sistem ini perusahaan dapat memonitoring posisi dan berat muatan pada kendaraan bermuatan dari jarak jauh dalam satu *interface*. Selain itu untuk mencegah terjadinya pencurian saat pengiriman barang.



BAB II TEORI PENUNJANG

Pada bab ini dibahas tentang teori penunjang serta komponen-komponen yang digunakan dalam Monitoring Posisi dan Berat pada kendaraan dengan PC. Komponen yang digunakan dalam Monitoring Posisi dan Berat pada kendaraan dengan PC adalah seperti GPS, modem gsm, *loadcell*, RTC, OP AMP, serta mikrokontroler.

2.1 Mikrokontroler Atmega 16 [1]

Mikrokontroler adalah sebuah sistem komputer yang dibangun pada sebuah keping (*chip*) tunggal. Jadi, hanya sebuah keping IC saja dapat dibuat sebuah sistem komputer yang dapat berfungsi untuk mengontrol alat. Mikrokontroler berisikan ROM (*Read-Only Memory*), RAM (*Read-Write Memory*), beberapa *Port* masukan maupun keluaran, dan beberapa peripheral seperti pencacah/pewaktu, ADC (*Analog to Digital converter*), DAC (*Digital to Analog converter*) dan serial komunikasi. Pada gambar 2.1 adalah gambar dari ATMEGA16, dan berikut adalah tampilan fisik mikrokontroler ATMEGA16.



Gambar 2.1 Mikrokontroler ATMEGA16

Seperti mikroprosesor pada umumnya, secara internal mikrokontroler ATMEGA16 terdiri atas unit-unit fungsionalnya *Arithmetic and Logical Unit* (ALU), himpunan register kerja, register dan dekoder instruksi, dan pewaktu beserta komponen kendali lainnya.

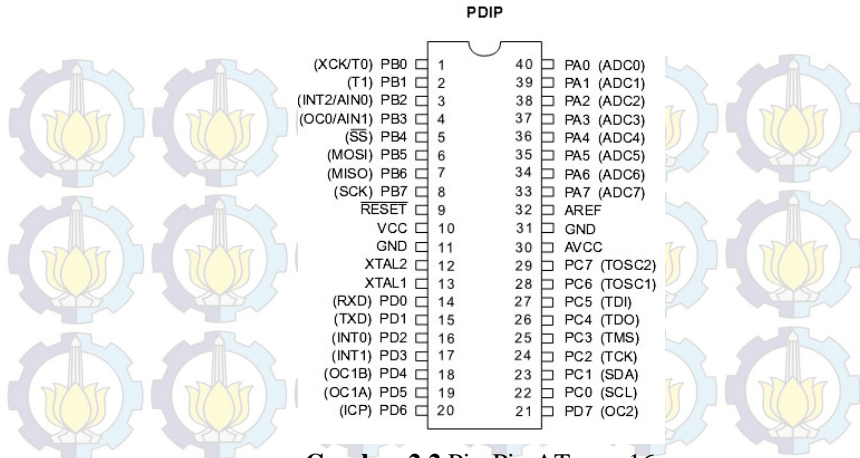
Berbeda dengan mikroprosesor, mikrokontroler menyediakan memori dalam serpih yang sama dengan prosesornya (*in chip*).

Mikrokontroler ini menggunakan arsitektur Harvard yang memisahkan memori program dari memori data, baik bus alamat maupun bus data, sehingga pengaksesan program dan data dapat dilakukan secara bersamaan.

Secara garis besar mikrokontroler ATmega16 terdiri dari :

1. Arsitektur RISC dengan *throughput* mencapai 16 MIPS pada frekuensi 16Mhz.
2. Memiliki kapasitas Flash memori 16Kbyte, EEPROM 512 Byte, dan SRAM 1Kbyte
3. Saluran I/O 32 buah, yaitu *Port A*, *Port B*, *Port C*, dan *Port D*.
4. CPU yang terdiri dari 32 buah register.
5. User interupsi internal dan eksternal
6. *Port* antarmuka SPI dan *Port* USART sebagai komunikasi serial
7. Fitur Peripheral
 - Dua buah 8-bit *timer/counter* dengan prescaler terpisah dan mode compare
 - Satu buah 16-bit *timer/counter* dengan prescaler terpisah, mode *compare*, dan mode *capture*
 - *Real time counter* dengan osilator tersendiri
 - Empat kanal PWM dan antarmuka komparator analog
 - 8 kanal, 10 bit ADC
 - *Byte-oriented Two-wire Serial Interface*
 - *Watchdog timer* dengan osilator internal

Kelebihan utama dari mikrokontroler ialah tersedianya RAM dan peralatan I/O pendukung sehingga ukuran board mikrokontroler menjadi sangat ringkas. Arsitektur ATmega16 mempunyai dua memori utama, yaitu memori data dan memori program. Selain itu, ATmega16 memiliki memori EEPROM untuk menyimpan data. ATmega16 memiliki 16K byte On-chip In-System Reprogrammable Flash Memory untuk menyimpan program. Instruksi ATmega16 semuanya memiliki format 16 atau 32 bit, maka memori flash diatur dalam 8K x 16 bit. Konfigurasi pin mikrokontroler Atmega16 dengan kemasan 40-pi dapat dilihat pada Gambar 2.2. Dari gambar tersebut dapat terlihat ATmega16 memiliki 8 pin untuk masing-masing *Port A*, *Port B*, *Port C*, dan *Port D*.

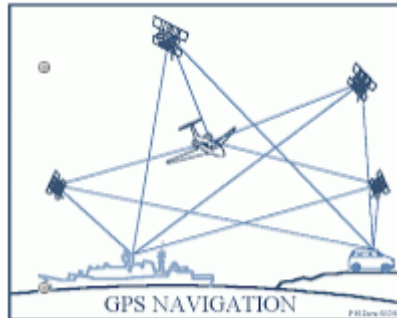


Gambar 2.2 Pin-Pin ATmega16

2.2 GPS [2]

GPS adalah singkatan dari *Global Positioning System*, sistem satelit yang dapat memberikan posisi di mana pun di dunia ini. Satelit GPS tidak mentransmisikan informasi posisi, yang ditransmisikan satelit adalah posisi satelit dan jarak penerima GPS dari satelit. Informasi ini diolah alat penerima GPS dan hasilnya ditampilkan ke delphi. Penerima GPS memperoleh sinyal dari beberapa satelit yang mengorbit bumi. Satelit yang mengitari bumi pada orbit pendek ini terdiri dari 24 susunan satelit, dengan 21 satelit aktif dan 3 buah satelit sebagai cadangan. Dengan susunan orbit tertentu, maka satelit GPS bisa diterima diseluruh permukaan bumi dengan penampakan antara 4 sampai 8 buah satelit. GPS dapat memberikan informasi posisi dan waktu dengan ketelitian sangat tinggi.

Cara kerja GPS adalah dengan memakai perhitungan *triangulation* dari satelit. Untuk perhitungan *triangulation*, GPS mengukur jarak menggunakan *travel time* sinyal radio. Untuk mengukur *travel time*, GPS memerlukan memerlukan akurasi waktu yang tinggi. Untuk perhitungan jarak, pastikan posisi satelit dan ketinggian pada orbitnya. Menggoreksi delay sinyal waktu perjalanan di atmosfer sampai diterima *reciever*. Gambar 2.3 merupakan gambaran proses cara kerja GPS.



Gambar 2.3 Proses Cara Kerja GPS

Aplikasi dari GPS dalam kehidupan sehari hari, yaitu :

a) Militer

GPS digunakan untuk keperluan perang, seperti menuntun arah bom, atau mengetahui posisi pasukan berada. Dengan cara ini maka kita bisa mengetahui mana teman mana lawan untuk menghindari salah target, ataupun menentukan pergerakan pasukan.

b) Navigasi

GPS banyak juga digunakan sebagai alat navigasi seperti kompas. Beberapa jenis kendaraan telah dilengkapi dengan GPS untuk alat bantu navigasi, dengan menambahkan peta, maka bisa digunakan untuk memandu pengendara, sehingga pengendara bisa mengetahui jalur mana yang sebaiknya dipilih untuk mencapai tujuan yang diinginkan.

c) Sistem Informasi Geografis

Untuk keperluan Sistem Informasi Geografis, GPS sering juga diikutsertakan dalam pembuatan peta, seperti mengukur jarak perbatasan, ataupun sebagai referensi pengukuran.

d) Pelacak Kendaraan

Kegunaan lain GPS adalah sebagai pelacak kendaraan, dengan bantuan GPS pemilik kendaraan/pengelola armada bisa mengetahui ada dimana saja kendaraannya/aset Bergeraknya berada saat ini.

e) Pemantauan Gempa

Bahkan saat ini, GPS dengan ketelitian tinggi bisa digunakan untuk memantau pergerakan tanah, yang ordenya hanya mm dalam setahun. Pemantauan pergerakan tanah berguna untuk memperkirakan terjadinya gempa, baik pergerakan vulkanik ataupun tektonik.



Istilah dalam GPS Navigasi, yaitu :

a. *Waypoint*

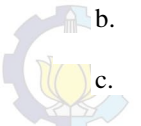
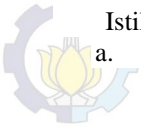
lokasi yang sudah ditandai, biasanya terdapat terdapat antara point awal dan point tujuan.

b. *Route*

Sekumpulan waypoint untuk mencapai suatu tujuan.

c. *Track*

Sekumpulan posisi dari perjalanan yang sedang/sudah ditempuh.



2.2.1 Parallax GPS Module PMB-688

Modul Parallax GPS PMB-688 SiRF *External Antenna* adalah sebuah modul GPS akurasi tinggi dari Parallax yang memiliki performa yang sangat baik. Dengan kemampuan penerimaan data dari 20 satelit yang dapat membantu pelacakan posisi dengan cepat. PMB-648 GPS SiRF *Eksternal Antenna* menyediakan antarmuka serial 4800 bps dengan menggunakan sumber tegangan 3.3 Volt DC – 5 Volt DC @ 65mA. Gambar 2.4 adalah bentuk fisik dari Parallax GPS Module PMB-688 *External Antenna*.

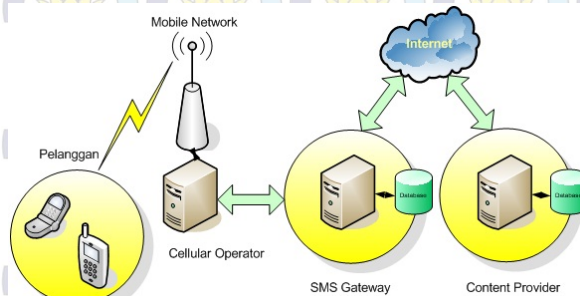


Gambar 2.4 Parallax GPS Module PMB-688
Beberapa fitur yang diberikan oleh modul GPS ini adalah :

1. Sensitivitas tinggi.
2. Berbasis *chipset* SiRFstarIII.
3. Dapat menangkap sinyal dari 20 satelit untuk hasil pelacakan yang lebih cepat dan berakurasi tinggi.
4. Terintegrasi dengan baterai yang *rechargeable* untuk *backup* memori dan RTC.
5. Komunikasi data menggunakan NMEA0183 v2.2.
6. Tegangan *supply* 3.3 Volt DC – 5 Volt DC.
7. Suhu kerja -20 °C - +70 °C.

2.3 SMS Gateway [3]

Short Message Service (SMS) adalah suatu fasilitas untuk mengirim dan menerima suatu pesan singkat berupa teks melalui perangkat nirkabel, yaitu perangkat komunikasi telepon selular, dalam hal ini perangkat nirkabel yang digunakan adalah telepon selular. Salah satu kelebihan dari SMS adalah biaya yang murah. *SMS Gateway* adalah teknologi mengirim, menerima dan bahkan mengolah sms melalui komputer dan sistem komputerisasi. *SMS Gateway* dapat terhubung ke media lain seperti perangkat SMSC dan server milik *Content Provider* melalui link IP untuk memproses suatu layanan SMS. Sebuah sistem *SMS Gateway*, umumnya terdiri komponen *Hardware* (Server/Komputer yang dilengkapi dengan perangkat jaringan) dan *Software* (Aplikasi yang digunakan untuk pengolahan pesan). Dan untuk sebuah sistem yang besar umumnya menggunakan *database* untuk penyimpanan data. Gambar 2.5 menunjukkan arsitektur *SMS content provider*.



Gambar 2.5 Arsitektur *SMS content provider*

2.4 Modem GSM [3]

Modem adalah perangkat yang dapat menghubungkan laptop atau perangkat *Personal Computer* dengan jaringan internet. Modem dapat berupa *handphone* ataupun berupa perangkat lain yang memang didesain khusus sebagai modem, salah satunya adalah *mobile* modem. Sifat dan *mobile* modem, antara lain menggunakan jaringan *wireless* sehingga dapat digunakan di manapun. Selain itu, karena memang didesain khusus untuk modem sehingga koneksi internet menjadi lebih mudah, cepat, dan stabil.

Modem GSM adalah modem yang menggunakan telepon seluler sebagai sistem transfer datanya. Modem GSM ini akan dapat digunakan untuk seluruh kartu koneksi dari *provider* apapun asalkan GSM. Modem GSM ini berbentuk seperti USB yang sangat *mobile* dan dapat digunakan di mana saja dan kapan saja. Tidak harus seperti kebanyakan modem kabel yang hanya bisa digunakan ketika ada kabel jaringan internet yang terhubung antara dan ke komputer dengan ke pusat jaringan internet yang biasanya menggunakan saluran telekomunikasi. Modem GSM ini bahkan tidak memerlukan kabel sedikit pun dalam penggunaannya.

2.4.1 Modem Wavecom Fastrack M1306B

Modem Wavecom M1306B Q2406B adalah sebuah modem yang berisikan E-GSM/GSM-GPRS 900/1800 *dual band* modem yang memiliki GPRS kelas 10. Modem ini dapat digunakan untuk pengiriman transmisi data, fax, *short message*, serta dapat digunakan untuk panggilan suara. Gambar 2.6 merupakan bentuk fisik wavecom *fastrack* M1306B.



Gambar 2.6 Wavecom *Fastrack* M1306B

Fitur yang ada pada modem ini adalah:

1. *Aluminum casing*
2. *Wireless access to internet (GPRS)*
3. *Based on Wavecom M1306B casing (short chassis) with Wismo Quik Q2406B/Q2403A module.*
4. *Dual-band 900/1800MHZ*
5. *3 Volt SIM card slot*
6. *USB interface (prolific driver for USB interface Support Win98 up to Win7)*
7. *Double tone multi-frequency function (DTMF)*
8. *Send and receive voice, data, fax, e-mail, SMS, MMS*
9. *Maximum transmitting speed 115KB/s*
10. *Support AT Command to make remote control (GSM07.07 and 07.05)*
11. *GPRS Class 10*
12. *Antenna with high sensitivity*
13. *Always on-line*
14. *Conform with ETSI GSM Phase-2+ standard*
15. *Output Power : Class 4 2W @ 900MHZ, Class 1 1W @ 1800MHZ*
16. *Working temperature : -20 --+55*
17. *Storage temperature : -25 --+70*
18. *Size : 98*54*25mm*
19. *Weight : 130g*
20. *Accessories: data cable USB (prolific USB-to-serial driver), antenna.*

2.5 AT-Command [3]

AT-Command adalah perintah yang dapat diberikan kepada *handphone* atau GSM/CDMA modem untuk melakukan sesuatu hal, termasuk untuk mengirim dan menerima SMS. Dengan memprogram pemberian perintah ini di dalam komputer/mikrokontroler maka perangkat kita dapat melakukan pengiriman atau penerimaan SMS secara otomatis untuk mencapai tujuan tertentu. Komputer ataupun mikrokontroler dapat memberikan perintah *AT-Command* melalui hubungan kabel data serial ataupun *bluetooth*. *AT-Command* ini sebenarnya adalah pengembangan dari perintah yang dapat diberikan kepada modem Hayes yang sudah ada sejak dulu. Dinamakan *AT-Command* karena semua perintah diawali dengan karakter A dan T.

Antar perangkat *handphone* dan GSM/CDMA modem bisa memiliki *AT-Command* yang berbeda-beda, namun biasanya mirip antara satu perangkat dengan perangkat lain. Untuk dapat mengetahui secara persis maka kita harus mendapatkan dokumentasi teknis dari produsen pembuat *handphone* atau GSM/CDMA modem tersebut.

2.6 *Database* [4]

Database atau basis data adalah kumpulan data yang disimpan secara sistematis di dalam komputer dan dapat diolah atau dimanipulasi menggunakan perangkat lunak (program aplikasi) untuk menghasilkan informasi. Dilihat dari jenisnya, basis data dibagi menjadi dua yaitu, Basis data *flat-file*. Basis data *flat-file* ideal untuk data berukuran kecil dan dapat dirubah dengan mudah. Pada dasarnya, mereka tersusun dari sekumpulan string dalam satu atau lebih file yang dapat diurai untuk mendapatkan informasi yang disimpan. Basis data *flat-file* baik digunakan untuk menyimpan daftar atau data yang sederhana dan dalam jumlah kecil. Basis data *flat-file* akan menjadi sangat rumit apabila digunakan untuk menyimpan data dengan struktur kompleks walaupun dimungkinkan pula untuk menyimpan data semacam itu. Salah satu masalah menggunakan basis data jenis ini adalah rentan pada korupsi data karena tidak adanya penguncian yang melekat ketika data digunakan atau dimodifikasi.

Basis data relasional. Basis data ini mempunyai struktur yang lebih logis terkait cara penyimpanan. Kata "relasional" berasal dari kenyataan bahwa tabel-tabel yang berada di basis data dapat dihubungkan satu dengan lainnya. Basis data relasional menggunakan sekumpulan tabel dua dimensi yang masing-masing tabel tersusun atas baris (tupel) dan kolom (atribut). Untuk membuat hubungan antara dua atau lebih tabel, digunakan key (atribut kunci) yaitu primary key di salah satu tabel dan foreign key di tabel yang lain. Saat ini, basis data relasional menjadi pilihan karena keunggulannya. Beberapa kelemahan yang mungkin dirasakan untuk basis data jenis ini adalah implementasi yang lebih sulit untuk data dalam jumlah besar dengan tingkat kompleksitasnya yang tinggi dan proses pencarian informasi yang lebih lambat karena perlu menghubungkan tabel-tabel terlebih dahulu apabila datanya tersebar di beberapa tabel.

Data secara umum dapat dikatakan sebagai segala sesuatu yang dapat dikumpulkan. Tentu saja hal ini akan membuat segala sesuatu di dunia ini menjadi data, dan masing masing dapat dikumpulkan menurut

jenisnya. Segala bentuk catatan mengenai data-data tersebut sebenarnya dapat dianggap sebagai *database* (tempat kumpulan data-data). Biasanya catatan dari data-data tersebut dilakukan dengan relatif sederhana dan dilakukan dengan cara manual (dicatat di atas lembaran-lembaran kertas, atau paling tidak diketik menggunakan program aplikasi tertentu). Setelah data-data tersebut dikumpulkan, biasanya diperlukan untuk pembuatan laporan, pengambilan keputusan atau segala sesuatu bentuk pengolahan yang berhubungan dengan data tersebut. *Database* dapat dibuat dan diolah dengan menggunakan suatu program komputer, yaitu yang biasa disebut dengan *software* (perangkat lunak). *Software* yang digunakan untuk mengelola dan memanggil kueri (*query*) *database* disebut *Database Management System* (DBMS) atau jika diterjemahkan kedalam bahasa Indonesia berarti “Sistem Manajemen Basis Data”.

DBMS terdiri dari dua komponen, yaitu *Relational Database Management System* (RDBMS) dan *Overview of Database Management System* (ODBMS). RDBMS meliputi *Interface Drivers*, *SQL Engine*, *Transaction Engine*, *Relational Engine*, dan *Storage Engine*. Sedangkan ODBMS meliputi *Language Drivers*, *Query Engine*, *Transaction Engine*, dan *Storage Engine*. Sedangkan level dari *software*nya sendiri, terdapat dua level *software* yang memungkinkan untuk membuat sebuah *database* antara lain *high level software* dan *low level software*. Yang termasuk di dalam *high level software*, antara lain Microsoft SQL Server, Oracle, Sybase, Interbase, XBase, Firebird, MySQL, PostgreSQL, Microsoft Access, dBase III, Paradox, FoxPro, Visual FoxPro, Arago, Force, Recital, dbFast, dbXL, Quicksilver, Clipper, *FlagShip*, *Harbour*, Visual dBase, dan *Lotus Smart Suite Approach*. Sedangkan yang termasuk di dalam *low level software* antara lain Btrieve dan *Tsunami Record Manager*.

2.6.1 Microsoft Access [4]

Microsoft Access adalah suatu program aplikasi basis data komputer relasional yang digunakan untuk merancang, membuat dan mengolah berbagai jenis data dengan kapasitas yang besar. Aplikasi ini menggunakan mesin basis data Microsoft Jet Database *Engine*, dan juga menggunakan tampilan grafis yang intuitif sehingga memudahkan pengguna. Versi terakhir adalah Microsoft Office Access 2007 yang termasuk ke dalam Microsoft Office System 2007.

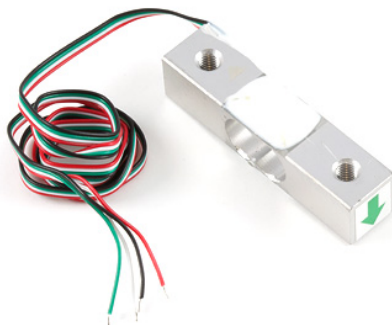
Microsoft Access dapat menggunakan data yang disimpan di dalam format Microsoft Access, Microsoft Jet Database *Engine*,

Microsoft SQL Server, Oracle Database, atau semua kontainer basis data yang mendukung standar ODBC. Para pengguna / *programmer* yang mahir dapat menggunakannya untuk mengembangkan perangkat lunak aplikasi yang kompleks, sementara para programmer yang kurang mahir dapat menggunakannya untuk mengembangkan perangkat lunak aplikasi yang sederhana. Access juga mendukung teknik-teknik pemrograman berorientasi objek, tetapi tidak dapat digolongkan ke dalam perangkat bantu pemrograman berorientasi objek.

2.7 *Loadcell* [5]

Loadcell adalah sebuah alat uji perangkat listrik yang dapat mengubah suatu energi menjadi energi lainnya yang biasa digunakan untuk mengubah suatu gaya menjadi sinyal listrik. Perubahan dari satu sistem ke sistem lainnya ini tidak langsung terjadi dalam dua tahap saja tetapi harus melalui tahap-tahap pengaturan mekanikal, kekuatan dan energi dapat merasakan perubahan kondisi dari baik menjadi kurang baik. Pada *strain gauge (loadcell)* atau biasa disebut dengan deformasi *strain gauge*. *Strain gauge* mengukur perubahan yang berpengaruh pada strain sebagai sinyal listrik, karena perubahan efektif terjadi pada beban hambatan kawat listrik.

Sel/slot beban dari satu strain gauge atau dua pengukur regangan. Output sinyal listrik biasanya disediakan serta di urutkan beberapa milivolt dan membutuhkan amplifikasi oleh penguat instrumentasi sebelum dapat digunakan. Gambar 2.7 adalah tampilan fisik dari *Loadcell*.



Gambar 2.7 *Loadcell* 5Kg

Output dari pemantauan perubahan kondisi dapat ditingkatkan untuk menghitung gaya yang diterapkan untuk perbaikan dan

pemantauan kondisinya. Berbagai jenis sel/slot beban yang ada termasuk sel/slot beban hidrolik, *strain gauge* merupakan bagian terpenting dari sebuah *loadcell*, dengan fungsi untuk mendeteksi besarnya perubahan dimensi jarak yang disebabkan oleh suatu elemen gaya. Strain gages secara umum digunakan dalam pengukuran presisi gaya, berat, tekanan, torsi, perpindahan dan kuantitas mekanis lainnya.

Setelahnya dikonversi menjadi energi tegangan kedalam anggota mekanis. *Strain gauge* menghasilkan perubahan pada nilai tahanan yang proporsional dengan perubahan jangka panjang atau perubahan melalui lamanya proses.

2.8 Op-Amp [5]

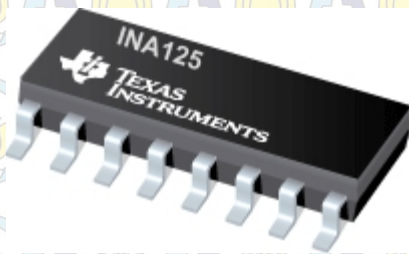
Operational Amplifier atau di singkat op-amp merupakan salah satu komponen analog yang sering digunakan dalam berbagai aplikasi rangkaian elektronika. Aplikasi op-amp yang paling sering dipakai antara lain adalah rangkaian inverter, non-inverter, integrator dan differensiator. Pada op-Amp memiliki 2 rangkaian *feedback* (umpan balik) yaitu *feedback* negatif dan *feedback* positif dimana *feedback* negatif pada op-amp memegang peranan penting. Secara umum, umpan balik positif akan menghasilkan osilasi sedangkan umpan balik negatif menghasilkan penguatan yang dapat terukur. Op-amp pada dasarnya adalah sebuah differential amplifier (penguat diferensial) yang memiliki dua masukan. Masukan op-amp ada yang dinamakan input inverting dan non-inverting. Op-amp ideal memiliki *open loop gain* (penguatan loop terbuka) yang tak terhingga besarnya. Karakteristik dari op-amp ideal adalah :

1. Penguatan tegangan lingkaran terbuka (*open-loop voltage gain*) $A_{VOL} = -\infty$
2. Tegangan offset keluaran (*output offset voltage*) $V_{OO} = 0$
3. Hambatan masukan (*input resistance*) $R_I = \infty$
4. Hambatan keluaran (*output resistance*) $R_O = 0$
5. Lebar pita (*band width*) $BW = \infty$
6. Waktu tanggapan (*respon time*) = 0 detik
7. Karakteristik tidak berubah dengan suhu

2.8.1 INA125

INA125 adalah instrumentasi penguat akurasi tinggi berdayadaya rendah dengan tegangan referensi presisi. Hal ini

memberikan eksitasi jembatan yang lengkap dan diferensial masukan penguatan yang presisi pada sirkuit yang terintegrasi. Sebuah resistor eksternal tunggal menetapkan penguatan apapun dari 4 sampai 10.000. INA125 telah melalui proses *laser-trimmed* sehingga dapat menghasilkan *offset* yang rendah. Untuk tegangan *offset* yang rendah yaitu 250mV, *low offset drift* 2mV/°C, serta *high common-mode rejection* (100dB at G = 100). INA125 bekerja dengan tegangan *supply* dari +2.7V sampai +36V atau tegangan *supply* ganda mulai dari ±1.35V sampai ±18V. Referensi tegangan eksternal disesuaikan dengan menggunakan pin yang dapat dirubah rubah nilai tegangannya dari 2.5V, 5V, atau 10V, memungkinkan penggunaan menggunakan berbagai macam transduser. Tegangan referensi akurat sampai ±0.5% (*max*) dengan ±35ppm/°C *drift* (*max*). Mode tidur memungkinkan operasi untuk di *shutdown* yang berguna untuk menghemat daya. INA125 tersedia dalam 16-pin DIP plastik dan SO-16 permukaan paket dan ditetapkan untuk -40 ° C hingga +85 ° C kisaran suhu di industri. Gambar 2.8 adalah bentuk fisik dari INA125



Gambar 2.8 INA125

Fitur dari INA125 adalah :

1. Tegangan *offset* yang rendah 250mV max
2. Arus *input* bias yang rendah 20nA max
3. Noise nya rendah
4. Mempunyai *sleep mode*
5. Memiliki CMR yang tinggi 100dB min
6. 16-PIN DIP dan kemasan SO-16 SOIC



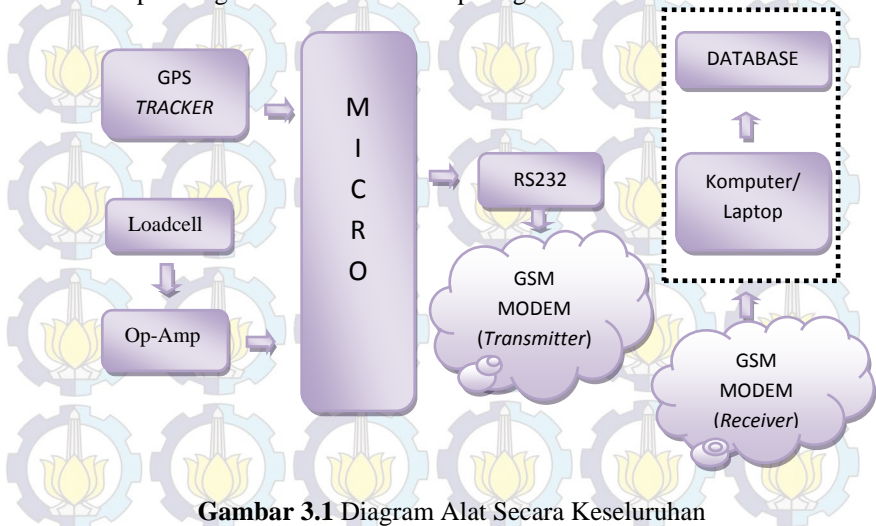
Halaman ini sengaja dikosongkan

BAB III PERANCANGAN DAN PEMBUATAN ALAT

Bab ini membahas perancangan sistem yang meliputi perancangan *hardware*, *software* dan perancangan mekanik. Detail mengenai pembahasan perancangan akan dibahas pada beberapa sub bab berikut.

3.1 Perancangan *Hardware*

Blok diagram dari keseluruhan Prototipe Sistem *Monitoring* Berat dan Posisi Kendaraan Bermuatan dengan Menggunakan PC yang di desain pada tugas akhir ini terlihat seperti gambar 3.1.



Gambar 3.1 Diagram Alat Secara Keseluruhan

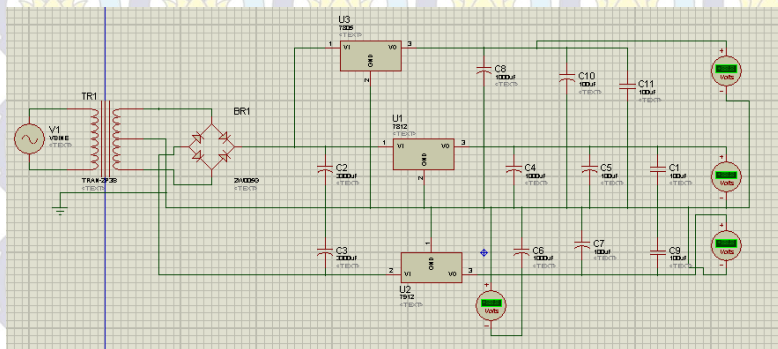
Cara kerja diagram alir sistem diatas adalah pertama data yang ada sistem yaitu posisi dari modul GPS serta berat dari *loadcell* diolah oleh mikrokontroler Atmega16. Untuk data dari berat yang berasal dari *loadcell* dikuatkan terlebih dahulu oleh op-amp INA125 dengan penguatan sebesar 1000 kali. Masing masing data dari *loadcell* dan modul GPS masuk ke pin *A port A0* yang berfungsi sebagai ADC, sedangkan data GPS masuk ke *port D* pin D0. Kabel pada GPS yang masuk pada pin D0 adalah kabel berwarna kuning yang berfungsi sebagai *transceiver* dari GPS. Kedua data tersebut diolah dan dikirimkan ke modem wavecom yang bertindak sebagai *transmitter* dengan

menggunakan RS232. Pada *receiver* pada PC menggunakan modem wavecom, pesan yang dikirim ke modem *receiver* berbasis SMS yang kemudian apabila pesan yang diterima sesuai dengan format yang ditentukan maka posisi dan berat dari kendaraan serta muatan nya dapat ditampilkan pada PC dengan menggunakan delphi 7. Pada tampilan ini kita dapat mengetahui secara *realtime* tentang posisi dan berat muatan kendaraan dari google *maps*. Pada saat menjalankan delphi 7 diperlukan koneksi internet yang berguna untuk menampilkan google *maps* sehingga kita dapat mengetahui posisi dari kendaraan tersebut.

Gambaran kerja dari sistem secara keseluruhan yaitu GPS *receiver* menerima data dari satelit. Data dari GPS ini dikirimkan ke *Microcontroller* AT Mega16 untuk diolah dan diambil data lintang, bujur, dan berat muatan. kemudian data tersebut dikirimkan melalui modem gsm ke PC atau laptop dan disimpan dalam database kemudian ditampilkan menggunakan Google *Maps*. Dari data yang ditampilkan, operator dapat memonitoring posisi kendaraan melalui peta. Demikian seterusnya saat kendaraan masih menyala atau sedang digunakan.

3.1.1 Perancangan Power Supply

Rangkaian *power supply* bertujuan menyediakan tegangan DC untuk keseluruhan kebutuhan alat yang digunakan. Rangkaian *power supply* ini mengubah tegangan AC 220 Volt menjadi tegangan DC yang dibutuhkan. *Power supply* terdiri dari *transformator step down* yang berfungsi untuk menurunkan tegangan 220 Volt. Rangkaian Power supply terlihat pada gambar 3.2 dengan output 5 Volt, 12 Volt, dan -5 Volt

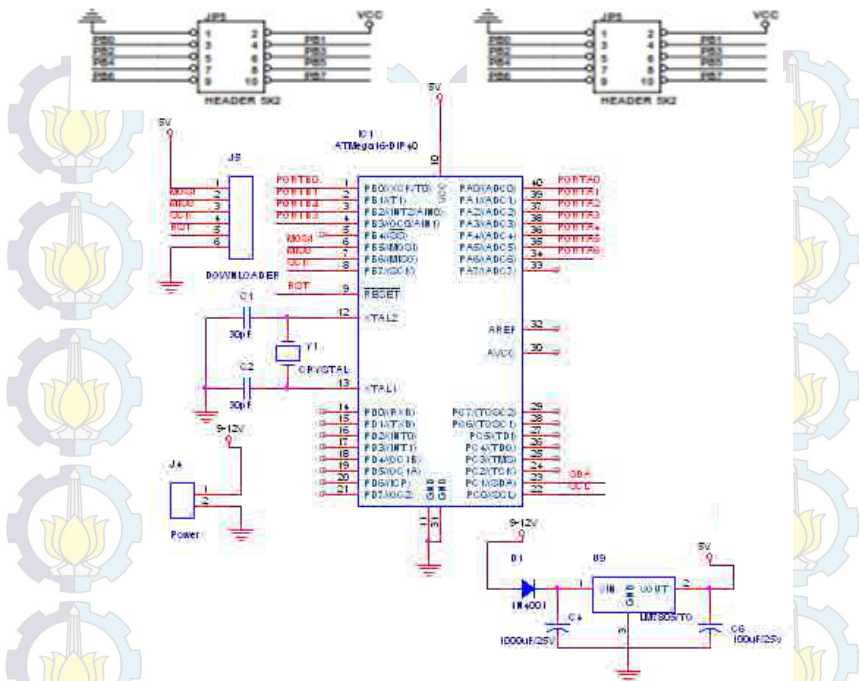


Gambar 3.2 Rangkaian Power Supply

Dioda dalam *power supply* berfungsi untuk menyearahkan tegangan dari AC menjadi DC (*rectifier*). Kapasitor berfungsi untuk menghilangkan *noise* pada sinyal DC. Regulator berfungsi untuk *output* tegangan referensi dan transistor sebagai penguat arus. *Power supply* dibuat dengan memberikan keluaran tegangan DC sebesar 5 Volt, -5 Volt dan 12 Volt. Tegangan 5 Volt untuk *supply* GPS dan *supply* loadcell, 12 Volt untuk modem GSM wavecom.

3.1.2 Perancangan Sistem Minimum ATmega16

Minimum sistem dari *microcontroller* AVR ATmega16 ini digunakan sebagai pengatur dari semua proses yang ada. Fungsi utama dari *microcontroller* ini adalah mengolah data gps serta data loadcell yang diterima dan mengirimkan ke modem gsm untuk ditampilkan ke *software* Delphi 7. Pada *microcontroller* ATmega16 terdapat pin untuk komunikasi serial yaitu Rx dan Tx. Tx digunakan untuk mengirimkan data secara serial sedangkan Rx digunakan untuk menerima data secara serial. Komunikasi serial pada mikrokontroler ini masih menggunakan sinyal TTL atau transistor transistor *logic* yaitu sinyal yang ayunan gelombang datanya antara 0 dan 5 Volt. Rx berfungsi dalam menerima data dari modul GPS *Tracker* dan data *loadcell* yang dikirimkan ke *microcontroller* untuk diolah. Sedangkan Tx berfungsi mengirim data yang telah diolah *microcontroller* ke modem GSM. Gambar 3.3 rangkaian pada modul ST-16 AVR ATmega16 yang kami gunakan.



Gambar 3.3 Rangkaian Mikrokontroler ATmega16

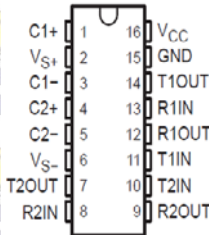
Penggunaan masing-masing port I/O mikrokontroler ATmega16 dalam sistem ini adalah sebagai berikut:

1. Port A0
Dihubungkan dengan rangkaian ADC
2. Port B
Tidak dihubungkan
3. Port C
Tidak dihubungkan
4. Port D0 dan D1
Dihubungkan ke RX dan TX GPS

3.1.3 Perancangan Rangkaian *Driver* RS232

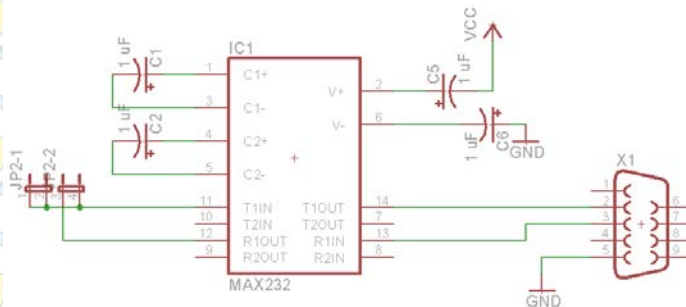
RS232 merupakan salah satu jenis IC rangkaian antar muka *dual RS-232 transmitter/receiver* yang memenuhi semua spesifikasi standar EIA-232-E. IC RS232 hanya membutuhkan *power supply* 5 Volt

(single power supply) sebagai catu. IC RS232 di sini berfungsi untuk merubah level tegangan pada COM menjadi level tegangan TTL/CMOS. Gambar 3.4 adalah Pin out RS232



Gambar 3.4 RS232 Pin Out

Sedangkan untuk skematik rangkaian Rs-232 yang akan dibuat dapat dilihat pada Gambar 3.5.



Gambar 3.5 Rangkaian Driver RS232

3.1.4 Modul GPS Tracker PMB-688 GPS SiRF

GPS yang digunakan pada sistem monitoring gambar 3.6 adalah modul GPS tracker PMB-688 GPS SIRF produksi parallax.



Gambar 3.6 GPS Tracker PMB-688 GPS SiRF

PMB-688 GPS memiliki 20 saluran satelit pelacakan paralel untuk akuisisi data cepat NMEA0183 v2.2 sering digunakan untuk berbagai aplikasi seperti sistem navigasi robotika, telemetri, atau eksperimen. terdapat *rechargeable* baterai untuk memori dan RTC cadangan, kabel untuk listrik, dan koneksi TTL. GPS PMB-688 memiliki 6 pin, fungsi tiap pin antara lain: Pin1 terhubung ke TX, Pin2 terhubung ke RX, Pin3 terhubung ke VCC supply +5volt, Pin4 terhubung ke ground, Pin5 dan Pin6 tidak dihubungkan (*Not connected*). Tabel 3.1 adalah fitur dan spesifikasi GPS

Tabel 3.1 Fitur Dan Spesifikasi GPS

GPS IC	SiRFstar III
Receiver:	Tracking up to 20 satellites L1, 1575.42 MHz, C/A code
Accuracy:	Position: 2DRMS approximately 5m, WAAS support Velocity: 0.1 m/s without SA imposed. Time: $\pm 1\mu\text{sec}$
Acquisition Time:	Cold Start: 42sec (Average) Warm Start: 38sec (Average) Hot Start: 1sec (Min)
Sensitivity:	Acquisition: -148dBm, Tracking: -159 dBm
Dynamics:	Altitude : 18000m (Max.) Velocity : 515m/s (Max.) Acceleration : $\pm 4g$ (Max.)
Navigation update rate	Once per Second
Serial Port:	TTL
Baud Rate:	4800 bps (Optional 9600,19300,38400 bps)
Output	NMEA0183 V2.2 GGA, GSV, GSA, RMC (optimal

<i>Message:</i>	VTG, GLL)
<i>Datum:</i>	WGS 84
<i>Power supply:</i>	DC 3.3 Volt ~ 5 Volt
<i>Power Consumption</i>	<i>Typical 65mA @12 Volt</i>
<i>LED Function</i>	<i>Power on/off and Navigation</i>
<i>Operating Temp.:</i>	-20°C~+70°C
<i>Storage Temp.:</i>	-30°C~+85°C
<i>Humidity</i>	5%~95%
<i>Antenna Type</i>	<i>Built-in patch antenna</i>
<i>RF Connector</i>	<i>MMCX type (Optional : Active Antenna)</i>

Cara kerja GPS secara logika ada 5 langkah:

1. Memakai perhitungan *triangulation* dari satelit.
2. Untuk perhitungan *triangulation*, GPS mengukur jarak menggunakan *travel time* sinyal radio.
3. Untuk mengukur *travel time*, GPS memerlukan akurasi waktu yang tinggi.
4. Untuk perhitungan jarak, kita harus tahu dengan pasti posisi satelit dan ketinggian pada orbitnya.
5. Terakhir harus mengoreksi *delay* sinyal waktu perjalanan di atmosfer sampai diterima *receiver*.

Satelit GPS berputar mengelilingi bumi selama 12 jam di dalam orbit yang akurat dan mengirimkan sinyal informasi ke bumi. GPS *receiver* mengambil informasi itu dan dengan menggunakan perhitungan *triangulation* menghitung lokasi *user* dengan tepat. GPS *receiver* membandingkan waktu sinyal dikirim dengan waktu sinyal di terima. Dari informasi itu didapat diketahui berapa jarak satelit. Dengan perhitungan jarak GPS *receiver* dapat melakukan perhitungan dan menentukan posisi *user* dan menampilkan dalam peta elektronik.

Sebuah GPS *receiver* harus mengunci sinyal minimal tiga satelit untuk menghitung posisi 2D (lintang, bujur, dan berat muatan) dan *track* pergerakan. Jika sudah dapat menentukan posisi *user*, selanjutnya GPS dapat menghitung informasi lain, seperti kecepatan, arah yang dituju, jalur, tujuan perjalanan, jarak tujuan, matahari terbit dan matahari terbenam dan masih banyak lagi.

Satelit GPS dalam mengirim informasi waktu sangat presisi karena satelit tersebut memakai jam atom. Jam atom yang ada pada satelit dengan partikel atom yang diisolasi, sehingga dapat menghasilkan jam yang akurat dibandingkan dengan jam biasa.

Perhitungan waktu yang akurat sangat menentukan akurasi perhitungan untuk menentukan informasi lokasi kita. Selain itu semakin banyak sinyal satelit yang dapat diterima maka akan semakin presisi data yang diterima karena ketiga satelit mengirim *pseudo-random code* dan waktu yang sama.

Ketinggian itu menimbulkan keuntungan dalam mendukung proses kerja GPS, bagi kita karena semakin tinggi maka semakin bersih atmosfer, sehingga gangguan semakin sedikit dan orbit yang cocok dan perhitungan matematika yang cocok. Satelit harus tetap pada posisi yang tepat sehingga stasiun di bumi harus terus memonitor setiap pergerakan satelit, dengan bantuan radar yang presisi selalu dicek tentang *altitude*, posisi dan kecepatannya.

3.1.5 Modem GSM Wavecom

Pada sistem *monitoring* ini sebagai SMS *device*-nya kami menggunakan Modem GSM Wavecom *Fastrack* M1306B. Gambar 3.7 bentuk fisik Modem GSM *Fastrack* M1306B.



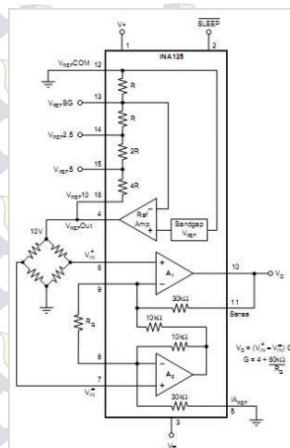
Gambar 3.7 Modem GSM *Fastrack* M1306B

Beberapa keunggulan Modem Wavecom *Fastrack* dibandingkan dengan Modem GSM/HP:

1. Wavecom jauh lebih stabil dibanding Modem GSM/HP
2. Wavecom tidak gampang panas dibanding Modem GSM/HP
3. Pengiriman SMS yang lebih cepat dibanding Modem GSM/HP (1000 s/d 1200 SMS per jam)
4. Support AT Command, bisa digunakan untuk cek sisa pulsa, cek point, cek pemakaian terakhir dll
5. Tidak semua Modem GSM/HP support AT Command
6. Tidak memakai baterai sehingga lebih praktis digunakan
7. Dapat langsung dikoneksikan dengan mikrokontroler melalui RS232

3.1.6 Perancangan Rangkaian Op-Amp

Rangkaian Op-Amp digunakan untuk memperkuat sinyal DC yang keluar dari loadcell agar dapat terbaca oleh mikrokontroler. Rangkaian Op-amp menggunakan IC INA125 yang telah diatur tegangannya. Gambar 3.8 rangkaian INA125



Gambar 3.8 Rangkaian INA125

Nilai penguatan di rangkaian INA125 ini dapat ditentukan dengan rumus sebagai berikut.

$$G = \frac{4 + 60K\Omega}{R_g}$$

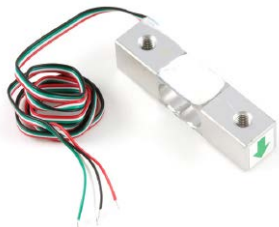
Dimana :

- G = *gain* (penguatan)
- 4 = nilai ketetapan IC INA125
- 60K Ω = nilai ketetapan resistansi INA125
- R_g = nilai resistansi yang dapat diubah

3.1.7 Sensor *Loadcell*

Untuk *monitoring* berat pada sistem ini digunakan sensor *Loadcell* yang berfungsi untuk mengukur berat muatan. *Loadcell* yang digunakan dapat menghitung berat 5Kg dan memiliki tegangan 0,85 milivolt hingga 1 milivolt. *Loadcell* memiliki 4 macam warna kabel yang dihubungkan ke *supply* +5 volt dan rangkaian *Op-amp*.

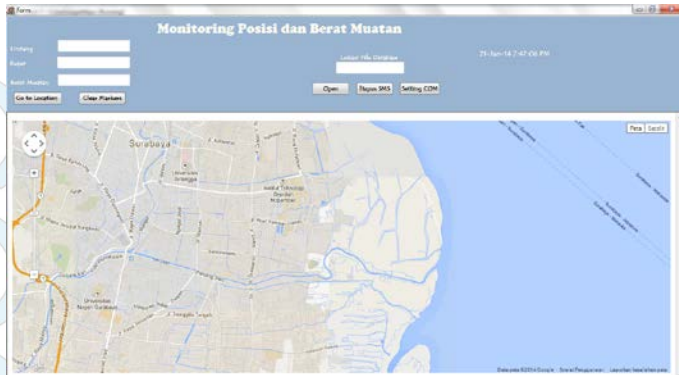
Kabel warna merah dihubungkan ke *supply* +5 volt. Kabel warna putih dihubungkan ke input(-) *Op-amp*. Kabel warna hijau dihubungkan ke input (+) *op-amp*. Kabel warna hitam dihubungkan ke ground. Setelah dihubungkan ke rangkaian *op-amp*, sensor *loadcell* memiliki tegangan maksimal 4,23 Volt. Gambar 3.9 merupakan bentuk dari sensor *loadcell*.



Gambar 3.9 Sensor *Loadcell*

3.2 Perancangan *Software*

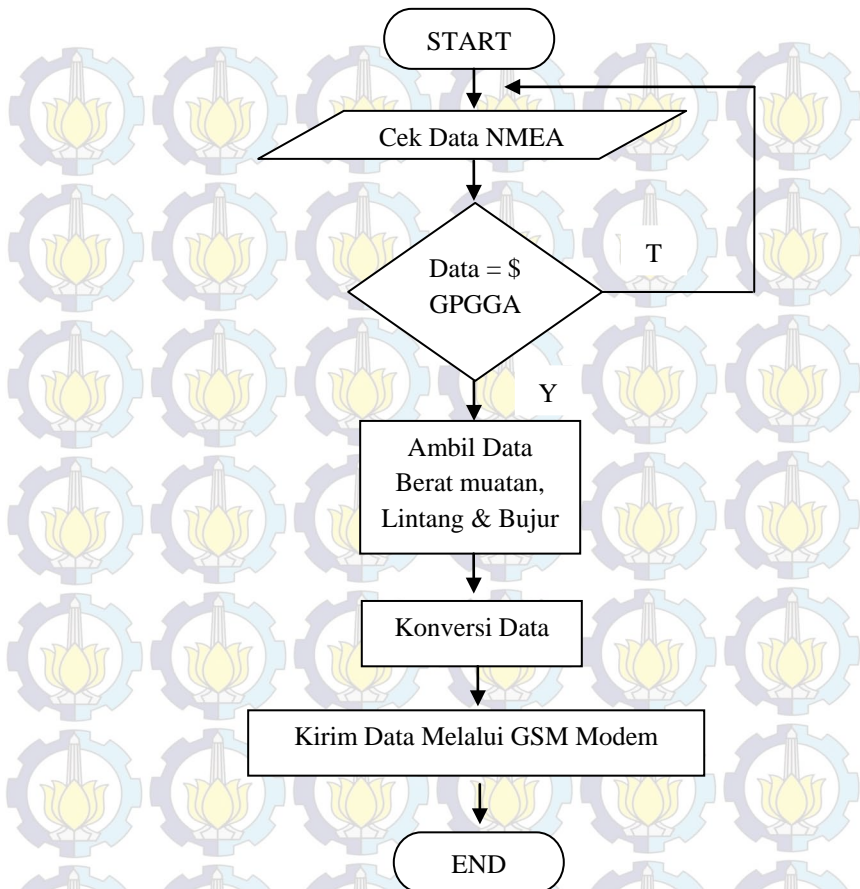
Perancangan perangkat lunak (*software*) pada Prototipe Sistem *Monitoring* Berat dan Posisi Kendaraan Bermuatan dengan Menggunakan PC ini meliputi pembuatan sebuah algoritma pengiriman data GPS pada ATmega16 menggunakan *compiler CodeVision AVR*. Rancangan ini juga menggunakan *software* Delphi 7 untuk tampilannya. Gambar 3.10 tampilan Delphi 7.



Gambar 3.10 Tampilan Delphi 7

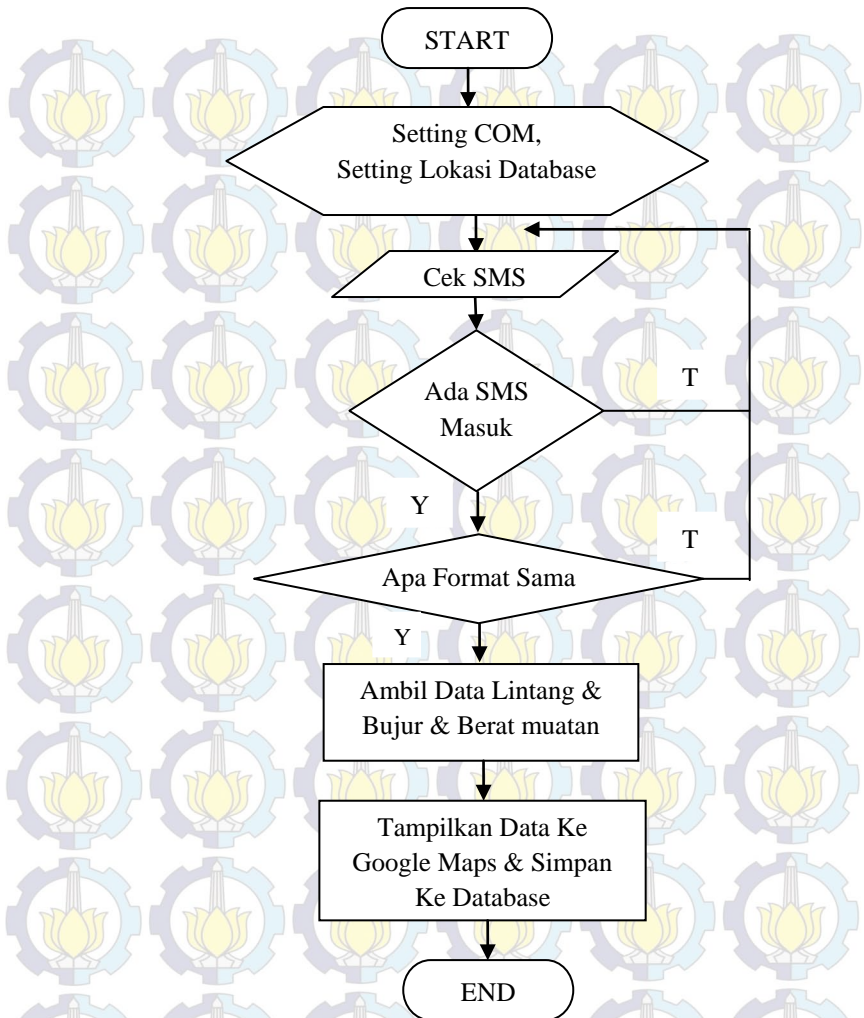
Sebelum merancang *software* maka akan dijelaskan terlebih dahulu alur dari sistem yang dibuat. Pada awalnya GPS akan mengambil data dari satelit, data tersebut dikirimkan ke mikrokontroler untuk diolah dan *parsing* data. Data yang diambil berupa lintang, bujur, dan berat muatan. Setelah *parsing* data, mikrokontroler mengolah data tersebut sehingga sesuai dengan format koordinat pada *Google Maps* dan mengirimkannya melalui SMS. Sehingga dapat terpantau posisi dan berat muatan yang ada pada sebuah kendaraan bermuatan. Tampilan dari posisi dan berat muatan dari kendaraan dibuat dengan menggunakan delphi 7 dengan *google maps* sebagai penunjuk posisi kendaraan. Dibutuhkan koneksi internet pada PC karena berfungsi untuk menampilkan *google maps*, serta dibutuhkan komunikasi antar *provider* yang lancar guna terpantau nya keadaan dari kendaraan tersebut. Alat ini nantinya akan mengatasi solusi semakin maraknya pencurian dan perampokan yang ada pada truk truk bermuatan yang sering terjadi pada jalan raya antar kota. Alat ini dapat dipasang pada bagian bawah dari kendaraan bermuatan sehingga dapat mendeteksi perubahan berat yang ada pada kendaraan yang nantinya perubahan berat ini akan dikirim ke PC pusat.

Flowchart dari perangkat lunak merupakan alur dari jalannya sistem yang akan dibuat. *Flowchart* dari sistem kirim data yang akan dibuat dapat dilihat pada gambar 3.11.



Gambar 3.11 *Flowchart* Kirim Data

Dapat dijelaskan bahwa pengiriman data dilakukan dengan memeriksa data NMEA terlebih dahulu, apabila data adalah \$GPGGA maka data lintang dan bujur akan diambil, apabila tidak maka cek data kembali. Kemudian setelah data diambil data dikonversi agar sesuai dengan format lintang, bujur, dan berat muatan. setelah dikonversi lalu data dikirim melalui modem. Selanjutnya penjelasan *Flowchart* dari sistem terima data yang akan dibuat dapat dilihat pada gambar 3.12



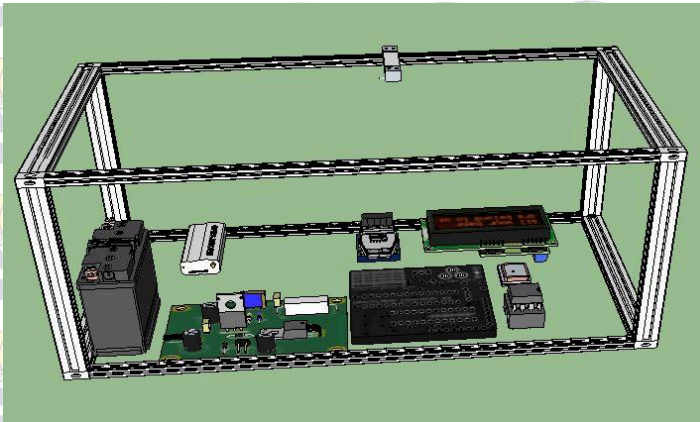
Gambar 3.12 Flowchart Terima Data

Dapat dijelaskan bahwa untuk menerima data yang pertama kali dilakukan adalah *setting* COM dan *setting* lokasi *database*. Setelah itu mengecek SMS yang masuk, jika ada SMS masuk maka format SMS

dicek apakah sama atau tidak. Jika tidak maka mengecek SMS lagi. Jika sama maka mengambil data *latitude*, *longitude* dan Berat muatan. Kemudian data ditampilkan ke *google maps* dan simpan ke *database*.

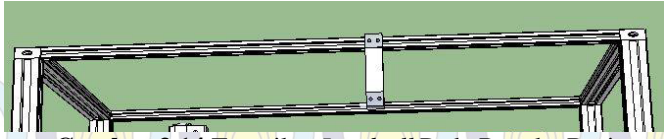
3.3 Perancangan Mekanik

Perancangan mekanik berupa pembuatan *box* yang terbuat dari karton sebagai tempat meletakkan alat pada jok sepeda motor. Gambar 3.13 merupakan tampilan keseluruhan alat yang di dalamnya terdapat rangkaian *power supply*, *minimum system* ATmega16, GPS receiver, dan modem GSM.

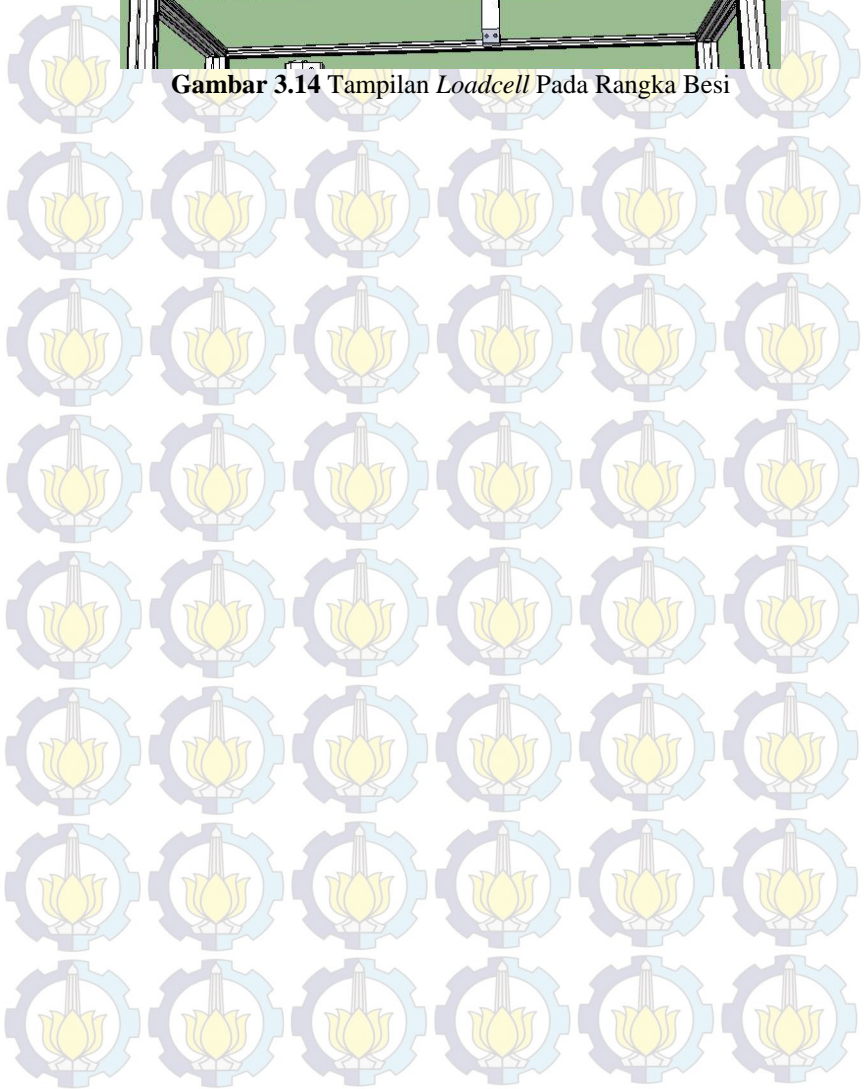


Gambar 3.13 Tampilan Keseluruhan Alat

Tata letak dari komponen komponen yang dibutuhkan telah diatur sedemikian rupa sehingga data yang didapatkan baik dari GPS dan dari *loadcell* dapat terolah dengan baik pada mikrokontroler. Perancangan timbangan sendiri dilakukan dengan cara menaruh *loadcell* pada bagian atas dari kerangka *box*. Kerangka *box* ini terbuat dari besi siku setebal 3 mm yang memungkinkan untuk menahan beban sebesar 5 kg. Perancangan timbangan sendiri dilakukan dengan menempatkan *loadcell* pada bagian atas rangka secara melintang sehingga *loadcell* dapat terpasang dengan kuat pada rangka besi yang nantinya dapat dipasang penampang pada *loadcell* sehingga dapat menampung beban dengan lebih mudah. Gambar 3.14 adalah salah satu sisi dari *loadcell* terpasang pada rangka besi.



Gambar 3.14 Tampilan *Loadcell* Pada Rangka Besi





BAB IV PENGUJIAN DAN ANALISA DATA

Pengujian dan analisa data pembuatan alat bertujuan agar mengetahui apakah alat telah bekerja sesuai dengan yang kita inginkan atau belum. Dan sebagai acuan yang tidak terpisahkan adalah adanya proses evaluasi sehingga akan dapat dilakukan langkah-langkah positif guna membawa alat ini kearah yang lebih baik.

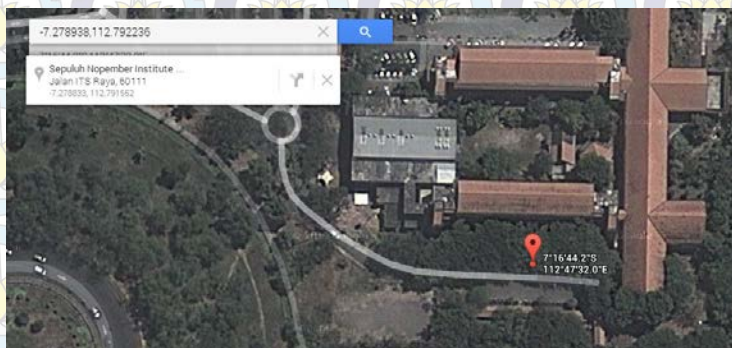
4.1 Pengujian GPS

Pengujian GPS ini bertujuan untuk mencari *error* serta membandingkan koordinat antara hasil data yang ditampilkan oleh GPS kita dengan perangkat GPS lain. Pengujian dilakukan di beberapa lokasi yang berbeda. Selanjutnya dilakukan pengambilan data yang berupa *latitude* dan *longitude* GPS kita lalu dibandingkan dengan data dari GPS. Gambar 4.1 tampilan *longitude* dan *latitude* dengan *display* LCD.



Gambar 4.1 Tampilan *Longitude* dan *Latitude*.

Setelah mendapatkan data yang ada dari LCD lalu langkah berikutnya adalah membandingkan dengan posisi yang ada pada *Google Maps*. Gambar 4.2 adalah tampilan koordinat di *Google Maps*.



Gambar 4.2 Tampilan Koordinat di *Google Maps*

Tabel 4.1 Perbandingan Data Koordinat

No.	Lokasi	Koordinat Alat (<i>Display LCD</i>)	Koordinat IWARE 8801 BLAZE
1.	Bundaran ITS	-7.279216, 112.790870	-7.279365, 112.790835
2.	Jalan Kertajaya Indah	-7.279687, 112.788376	-7.279708, 112.788180
3.	Kerta Indah Timur VII	-7.280930, 112.785720	-7.281073, 112.785696
4.	Manyar Kerta Adi II	-7.281874, 112.784036	-7.281877, 112.783803
5.	Manyar Kerta Adi I	-7.282122, 112.780978	-7.282124, 112.780764
6.	Jalan Arief Rahman Hakim	-7.289467, 112.781367	-7.289451, 112.781582
7.	Jalan Teknik Elektro	-7.289880, 112.792195	-7.289651, 112.792276
8.	Jalan Teknik Mesin	-7.285908, 112.795916	-7.285878, 112.796152
9.	Jalan Teknik Kimia	-7.279913, 112.797780	-7.279613, 112.797790

10.	Jalan Raya ITS	-7.275622, 112.793748	-7.275688, 112.793373
-----	----------------	--------------------------	--------------------------

Berdasarkan Tabel 4.1 di atas maka dapat kita cari kesalahan relatif antara GPS kami dengan GPS pembandingan dengan rumus sebagai berikut :

$$e = \frac{(x-k)}{x} * 100\%$$

dimana e = kesalahan relatif

x = koordinat GPS

k = koordinat GPS IWARE 8801 BLAZE

Data ketidakakuratan untuk masing masing titik adalah sebagai berikut :

1. a. $Lattitude = \frac{7.279216-7.279365}{7.279365} * 100\% = 0.00204\%$

b. $Longitude = \frac{112.790870-112.790835}{112.790835} * 100\% = 0.0035\%$

2. a. $Lattitude = \frac{7.279687-7.279708}{7.279708} * 100\% = 0,00236\%$

b. $Longitude = \frac{112.788376-112.788180}{112.788180} * 100\% = 0,0041\%$

3. a. $Lattitude = \frac{7.280930-7.281073}{7.281073} * 100\% = 0.00264\%$

b. $Longitude = \frac{112.785720-112.785696}{112.785696} * 100\% = 0.0036\%$

4. a. $Lattitude = \frac{7.281874-7.281877}{7.281877} * 100\% = 0.00275\%$

b. $Longitude = \frac{112.784036-112.783803}{112.783803} * 100\% = 0.0049\%$

$$5. \quad a. \textit{Latitude} = \frac{7.282122-7.282124}{7.282124} * 100\% = 0.00212\%$$

$$b. \textit{Longitude} = \frac{112.780978-112.780764}{112.780764} * 100\% = 0.0043\%$$

$$6. \quad a. \textit{Latitude} = \frac{7.289467-7.289451}{7.289451} * 100\% = 0.00211\%$$

$$b. \textit{Longitude} = \frac{112.781367-112.781582}{112.781582} * 100\% = 0.0037\%$$

$$7. \quad a. \textit{Latitude} = \frac{7.289880-7.289651}{7.289651} * 100\% = 0.00285\%$$

$$b. \textit{Longitude} = \frac{112.792195-112.792276}{112.792276} * 100\% = 0.0047\%$$

$$8. \quad a. \textit{Latitude} = \frac{7.285908-7.285878}{7.285878} * 100\% = 0.00201\%$$

$$b. \textit{Longitude} = \frac{112.795916-112.796152}{112.796152} * 100\% = 0.0039\%$$

$$9. \quad a. \textit{Latitude} = \frac{7.279913-7.279613}{7.279613} * 100\% = 0.00292\%$$

$$b. \textit{Longitude} = \frac{112.797780-112.797790}{112.797790} * 100\% = 0.0031\%$$

$$10. \quad a. \textit{Latitude} = \frac{7.275622-7.275688}{7.275688} * 100\% = 0.00191\%$$

$$b. \textit{Longitude} = \frac{112.793748-112.793373}{112.793373} * 100\% = 0.0042\%$$

Untuk rata-rata kesalahan relatif *latitude* didapatkan nilai sebesar 0.00255% dan untuk rata-rata kesalahan relatif *longitude* didapatkan nilai sebesar 0.00401%. Kemudian akan dicari *error* dari selisih jarak

posisi antara koordinat alat dan koordinat pembanding dengan cara menggunakan *goole maps* dan didapatkan.

1. Selisih jarak di Bundaran ITS = 12 m
2. Selisih jarak di Jalan Kertajaya Indah = 11 m
3. Selisih jarak di Kerta Indah Timur VII = 9 m
4. Selisih jarak di Manyar Kerta Adi II = 10 m
5. Selisih jarak di Manyar Kerta Adi I = 11 m
6. Selisih jarak di Jalan Arief Rahman Hakim = 9 m
7. Selisih jarak di Jalan Teknik Elektro = 10 m
8. Selisih jarak di Jalan Teknik Mesin = 10 m
9. Selisih jarak di Jalan Teknik Kimia = 9 m
10. Selisih jarak di Jalan Raya ITS = 10 m

4.2 Pengujian Rangkaian *Power Supply*

Pengujian berikutnya adalah pengujian rangkaian *power supply* dari *accu*. Pengujian dilakukan dengan cara mengukur tegangan *output* sebesar +5 *Volt* serta +12 *Volt* pada saat keadaan tidak ada beban dan saat berbeban. Beban yang disini merupakan beban pada saat keadaan keseluruhan sistem pada keadaan menyala. Hasil pengukuran tegangan *power supply* dapat dilihat pada Tabel 4.2. Dari data pengukuran didapatkan hasil yang mendekati nilai yang diharapkan. Dengan data ini dapat diketahui *error* tegangan *output* dari rangkaian *power supply accu* saat tanpa beban dan berbeban.

Tabel 4.2 Pengukuran Tegangan *Power Supply*

<i>Output</i>	Pembacaan		<i>Error</i>	
	Tanpa Beban	Berbeban	Tanpa Beban	Berbeban
+5 <i>Volt</i>	4,99 <i>Volt</i>	4.97 <i>Volt</i>	0,2 %	0,6%
+12 <i>Volt</i>	11,98 <i>Volt</i>	11.95 <i>Volt</i>	0,16 %	0.4%
+7,5 <i>Volt</i>	7,5 <i>Volt</i>	7,47 <i>Volt</i>	0 %	0,4%

4.3 Pengujian Modem GSM Secara *Serial*

Pengujian modem dilakukan dengan menghubungkan modem ke *hyperterminal*. Gambar 4.3 menunjukkan *hyperterminal* berhasil menerima sms yang dikirimkan oleh *handphone*.

```
+CMGL: 6, "REC UNREAD", "+6289677332171", "14/05/31, 22:04:51+28"  
Oke  
OK
```

Gambar 4.3 Pengujian Modem Pada *Hyperterminal*

Terlihat pada Gambar 4.3, SMS berhasil terkirim pada tanggal 31 Mei 2014, jam 22:04 dengan isi SMS yaitu oke. Lalu untuk pengecekan jumlah saldo pulsa dengan menggunakan modem wavecom pada *hyperterminal* ditunjukkan oleh gambar 4.4.

```
AT+CUSD=1,-111-1#  
OK  
+CUSD: 1, "Account detail  
Nomor Tri: 628993377988 -Main balance: Rp 15669 -Other bal  
ance: Rp 0 -Internet Balance: Rp 0 Total Balance: Rp 15669 valid until 05-JUN-14.  
0. Lanjut",15  
+CUSD: 4  
-
```

Gambar 4.4 Pengujian Pengecekan Jumlah Saldo

Pada gambar 4.4 diatas menunjukkan bahwa saldo yang ada berjumlah Rp 15.669 yang berlaku sampai tanggal 5 juni 2014, perintah at+cusd=1,(perintah pengecekan pulsa) berfungsi untuk mengetahui jumlah saldo yang ada pada suatu kartu *provider* melalui *hyperterminal*. Lalu untuk mengirim sms melalui *hyperterminal* ke perangkat elektronik seperti *handphone* dengan menggunakan modem wavecom ditunjukkan oleh gambar 4.5.

```
AT+CMGS="085656486117"  
> TESTING+  
+CMGS: 48  
OK  
-
```

Gambar 4.5 Mengirim SMS Ke *Handphone* Melalui *Hyperterminal*

Pada gambar 4.5 diatas menunjukkan bahwa pesan berhasil terkirim ke *handphone* dengan nomor tujuan 085656486117 dengan pesan TESTING. Perintah yang digunakan adalah at+cmgs="nomor tujuan" yang berfungsi untuk mengirim pesan dari *hyperterminal* ke

handphone. Lalu untuk mengetahui ada berapa pesan yang masuk ke *inbox* kita dapat mengeceknya melalui *hyperterminal*. Gambar 4.6 menunjukkan jumlah pesan yang ada di dalam *inbox*.

```
AT
OK
AT+CMGF=1
OK
AT+CMGL="ALL"
+CMGL: 1,"REC READ","+6285648951819",,"14/05/31,19:39:54+28"
Opo
+CMGL: 2,"STO UNSENT","082232544666",,
TESTING
+CMGL: 3,"STO UNSENT","082232544666",,
TES
+CMGL: 4,"STO UNSENT","+82232544666",,
TES
+CMGL: 5,"REC READ","+6289677332171",,"14/05/31,21:14:53+28"
y
+CMGL: 6,"REC UNREAD","+6289677332171",,"14/05/31,22:04:51+28"
Oke
OK
-
```

Gambar 4.6 Jumlah Pesan Yang Ada Pada *Inbox* Modem

Pada gambar 4.6 diatas menunjukkan bahwa terdapat 6 pesan yang ada pada *inbox* modem, dengan menggunakan perintah `at+cmgl="ALL"` kita dapat mengetahui seluruh daftar pesan yang ada pada *inbox* modem. Lalu untuk mengetahui informasi mendasar tentang modem ini ditunjukkan oleh gambar 4.7.

```
AT
OK
AT+CGMI
WAVECOM MODEM

OK
AT+CGMR
651b09gg.Q2406B 244 041410 13:45

OK
AT+CGSN
012345678901234

OK
-
```

Gambar 4.7 Informasi Dasar Modem

Pada gambar 4.7 diatas kita dapat mengetahui merek modem, jenis modem, dan untuk mengetahui IMEI modem kita. Dengan merek

modem wavecom dan nomor IMEI atau *International Mobile Equipment Identity* yaitu 01234567901234 dengan menggunakan perintah `at+cgmi`, `at+cgsn`, `at+cgsn`.

4.4 Pengujian Mikrokontroler ATmega16

Pada pengujian mikrokontroler dilakukan dengan menggunakan rangkaian *push button* dan memberikan program sederhana yaitu *input output*. Untuk mengetahui mikrokontroler bekerja dengan baik, kita tekan tombol *push button* kemudian mengukur *logic* setiap *input* dan *output* dari mikrokontroler. Pada tabel 4.3 menunjukkan data tegangan yang ada pada pin pin mikrokontroler.

Tabel 4.3 Pengukuran Tegangan Pada *Port A* Dan *Port D*

Alamat Port	High (Volt)	Low (Volt)
A.0	4,70	0
A.1	4,68	0
A.2	4,68	0
A.3	4,69	0
A.4	4,68	0
A.5	4,68	0
A.6	4,67	0
A.7	4,69	0
C.0	4,71	0
C.1	4,72	0
C.2	4,72	0
C.3	4,74	0
C.4	4,75	0
C.5	4,76	0
C.6	4,73	0
C.7	4,73	0
D.0	4,75	0
D.1	4,74	0
D.2	4,76	0
D.3	4,73	0
D.4	4,71	0
D.5	4,75	0
D.6	4,74	0
D.7	4,72	0

Dari Tabel 4.3, data hasil pengujian dari setiap pin pada *Port A*, *Port B* dan *Port D* mikrokontroler ATMega16, didapatkan tegangan rata – rata 4,7 Volt apabila dalam kondisi *active High* dan tegangan 0 Volt apabila dalam kondisi *active low*. Sehingga dapat disimpulkan bahwa IC ATMega16 yang digunakan berfungsi dengan baik.

4.5 Pengujian Loadcell Dengan Op-Amp

Pada *loadcell* dilakukan pengukuran penguatan tegangan dengan IC Op-Amp dengan penguatan antara 1000 sampai *range* maksimal IC Op-Amp yaitu 10.000 kali. Dengan adanya data hasil penguatan diharapkan kita dapat mendapatkan hasil penguatan yang linier sehingga mudah ketika dihubungkan dengan ADC. Pada tabel 4.4 menunjukkan data tegangan awal *loadcell* yang telah dikuatkan oleh INA125..

Tabel 4.4 Data Penguatan *Loadcell* Dengan IC Op-Amp INA125

Penguatan / Berat	0 Kg	0,5 Kg	1 Kg	1,5 Kg	2 Kg	2,5 Kg
1000x (65 Ω)	137,3 mV	0,57 V	1,03 V	1,48 V	1,9 V	2,4 V
2000x (30 Ω)	242,5 mV	1,202 V	2,1 V	3,1 V	4,08 V	4,23 V
3000x (20 Ω)	378 mV	1,79 V	3,21 V	4,23 V	4,23 V	4,23 V
4000x (15 Ω)	0,48 V	2,45 V	4,23 V	4,23 V	4,23 V	4,23 V
5000x (12 Ω)	0,56 V	2,89 V	4,23 V	4,23 V	4,23 V	4,23 V
6000x (10 Ω)	0,76 V	3,8 V	4,23 V	4,23 V	4,23 V	4,23 V
7000x (8,5 Ω)	0,87 V	4,15 V	4,23 V	4,23 V	4,23 V	4,23 V

8000x (7,5 Ω)	0,92 V	4,23 V	4,23 V	4,23 V	4,23 V	4,23 V
9000x (6,6 Ω)	1,05 V	4,23 V	4,23 V	4,23 V	4,23 V	4,23 V
10000x (6 Ω)	1,28 V	4,23 V	4,23 V	4,23 V	4,23 V	4,23 V

IC INA125 memiliki penguatan mulai dari range 2 – 10000 kali penguatan. Tetapi pada saat proses pengambilan data terjadi masalah yaitu data tegangan yang diambil tidaklah linier setiap kenaikan 1 Kg. Oleh karena itu kita pilih resistor dengan resistansi 60K Ω dengan penguatan 1000 kali yang setiap 1 Kg berat yang ditempatkan diatas *loadcell* maka tegangan yang dihasilkan linier dengan kenaikan 1 V setiap 1 Kg. Hal ini mempermudah kita dalam melakukan proses selanjutnya yaitu membuat program ADC agar berat yang ada dapat ditampilkan dengan nilai akurasi yang tinggi.

4.6 Pengujian Ketelitian Bit Pada ADC

Pengujian ini bertujuan untuk mengetahui nilai ketelitian setiap bit yang ada pada ADC terhadap berat dan tegangan penguatan Hal ini dapat kita peroleh dengan rumus perhitungan. Rumus untuk mengetahui besarnya tegangan perubahan dari setiap bit yang masuk pada ADC dapat kita ketahui dari rumus berikut.

$$\text{Tegangan} = \frac{5}{255} = 0,02\text{V}$$

Dimana: 5 = nilai tegangan yang ada pada mikrokontroler
255 = nilai dari 8-bit

Lalu untuk menguji mengetahui nilai kenaikan berat yang ada pada setiap 1 nilai dari 255 adalah dapat kita ketahui dengan rumus berikut.

$$\text{Berat} = \frac{B}{N}$$

Dimana: B = nilai berat beban

N = jumlah pengurangan dari nilai bit antara 2,5 kg dengan 0 kg.

Pengurangan antara beban 2,5 kg dan 0 kg ini dilakukan dengan cara mengurangi nilai bit antara beban 2,5 kg dengan 0 kg yaitu.

$$Y = \frac{V \ 0 \text{ kg}}{\text{Nilai V per bit}} = \frac{137,3}{20} = 6 \text{ bit}$$

$$X = \frac{V \ 2,5 \text{ kg}}{\text{Nilai V per bit}} = \frac{2500}{20} = 125 \text{ bit}$$

Lalu yang selanjutnya adalah mengurangi nilai bit dari 2 keadaan ini.

$$\begin{aligned} N &= X - Y \\ &= 125 - 6 \\ &= 119 \text{ bit} \end{aligned}$$

$$\begin{aligned} \text{Berat} &= \frac{B}{N} \\ &= \frac{2,5 \text{ Kg}}{119} = 0,02 \text{ Kg} \end{aligned}$$

Sehingga dapat kita ketahui setiap kenaikan 1 bit pada ADC maka tegangan yang naik adalah 20 mV atau 0,02 V dan setiap kenaikan 1 bit ADC maka berat bebannya adalah 0,02 Kg.

4.7 Pengujian GPS Secara Serial

Pengujian GPS dilakukan dengan menghubungkan GPS ke *hyperterminal* dengan media *driver* USB to TTL. Gambar 4.8 menunjukkan *hyperterminal* berhasil menerima data yang dikirimkan oleh GPS.

```

$GPGSA,A,1,,,,,,,,,,,,,*1E
$GPRMC,163407.523,V,0716.7363,S,11247.5343,E,,270514,,N*6C
$GPGGA,163408.512,0716.7363,S,11247.5343,E,0,00,48.6,M,12.3,M,,0000*5B
$GPRMC,163408.512,V,0716.7363,S,11247.5343,E,,270514,,N*61
$GPGGA,163409.512,0716.7363,S,11247.5343,E,0,00,48.6,M,12.3,M,,0000*5A

$GPGSA,A,1,,,,,,,,,,,,,*1E
$GPRMC,163409.512,V,0716.7363,S,11247.5343,E,,270514,,N*60
$GPGGA,163410.523,0716.7363,S,11247.5343,E,0,00,48.6,M,12.3,M,,0000*50

$GPGSA,A,1,,,,,,,,,,,,,*1E
$GPGSV,1,1,00,*79
$GPRMC,163410.523,V,0716.7363,S,11247.5343,E,,270514,,N*6A
$GPRMC,163411.512,V,0716.7363,S,11247.5343,E,,270514,,N*69
$GPGGA,163412.512,0716.7363,S,11247.5343,E,0,00,48.6,M,12.3,M,,0000*50

$GPGSA,A,1,,,,,,,,,,,,,*1E
$GPRMC,163412.512,V,0716.7363,S,11247.5343,E,,270514,,N*6A
$GPGGA,163413.525,0716.7363,S,11247.5343,E,0,00,48.6,M,12.3,M,,0000*55

$GPGSA,A,1,,,,,,,,,,,,,*1E
$GPRMC,163413.525,V,0716.7363,S,11247.5343,E,,270514,,N*6F

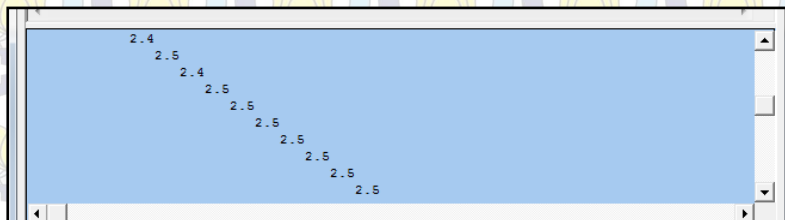
```

Gambar 4.8 Tampilan Data GPS Pada *Hyperterminal*

Pada data diatas dapat kita ketahui posisi dari GPS serta informasi lain seperti waktu data tersebut diambil. *GPGGA adalah Global Positioning System Fix Data*. Pada *GPGGA* terdapat informasi dasar tentang GPS yaitu posisi lintang, bujur, waktu, jumlah satelit, serta ketinggian. *GPRMC* berisikan data yang hampir sama yaitu posisi dan waktu, tetapi di dalam *GPRMC* terdapat informasi tentang kecepatan gerak kendaraan. *GPGSA* berisikan jumlah satelit yang tertangkap oleh modul GPS.

4.8 Pengujian Data Penguatan *Loadcell* Secara Serial

Pengujian penguatan *loadcell* ini dilakukan dengan menghubungkan tegangan *loadcell* yang telah dikuatkan oleh *INA125* ke *hyperterminal* dengan media mikrokontroler *Atmega16*. Gambar 4.9 menunjukkan *hyperterminal* berhasil menerima data yang dikirimkan oleh mikrokontroler *Atmega16* dengan berat beban 2,5 kilogram.



Gambar 4.9 Tampilan Data Berat Beban Pada *Hyperterminal*

Pada gambar diatas dapat kita ketahui bahwa berat yang ada pada *loadcell* adalah 2,5 kilogram. Tujuan dari pengambilan data ini adalah untuk mengetahui apakah ada perbedaan besar data yang signifikan antara perhitungan teori, berat beban yang ada pada tampilan lcd, serta dengan data berat beban yang dikirim ke PC melalui *hyperterminal*.

4.9 Pengujian Tingkat Keakuratan Berat Beban Dengan Teori

Pengujian ini bertujuan untuk mengetahui tingkat keakuratan dari op-amp atau penguatan yang dihasilkan dari tegangan *loadcell* yang kemudian dikuatkan oleh INA125. Lalu dengan menggunakan program ADC dari AVR kita dapat menampilkan berat bebabn pada media LCD. Pada gambar 4.10 menunjukkan bahwa berat beban 0 kg dan pada gambar 4.11 menunjukkan bahwa berat beban 2,5 kg.



Gambar 4.10 Pengujian Berat Beban Pada Saat 0 Kg



Gambar 4.11 Pengujian Berat Beban Pada Saat 2,5 Kg

Pada tabel 4.5 dapat kita lihat antara data perhitungan secara teori dengan data yang kita dapatkan dari melakukan percobaan. Sehingga kita dapat mengetahui besarnya tingkat keakuratan dari penguatan op-amp.

Tabel 4.5 Data Berat Beban Secara Teori dan Praktek

Berat Beban	Teori	Praktek	% Error
0 Kg	0 Kg	0 Kg	0 %
1 Kg	1 Kg	1 Kg	0 %
1,5 Kg	1,5 Kg	1,5 Kg	0 %
2 Kg	2 Kg	2 Kg	0 %
2,5 Kg	2,5 Kg	2,5 Kg	0 %

Dari tabel 4.5 dapat kita lihat bahwa op-amp INA125 memiliki keakuratan beban sebesar 100% apabila ditampilkan pada LCD.

4.10 Pengujian Berat Beban Via SMS

Pengujian ini berfungsi untuk membandingkan data berat beban yang didapatkan dari percobaan pengukuran berat beban yang pada LCD dengan pengiriman secara langsung ke tampilan di PC pusat yang

menggunakan delphi 7. Pada tabel 4.6 dapat dilihat perbedaan antara kedua data tersebut.

Tabel 4.6 Data Berat Beban Pada LCD dan Delphi 7

Berat Beban	LCD	Delphi 7	% Error
0 Kg	0 Kg	0,00045 Kg	1 %
1 Kg	1 Kg	1,00032 Kg	0,03 %
1,5 Kg	1,5 Kg	1,50019 Kg	0,01%
2 Kg	2 Kg	2,00052 Kg	0,01 %
2,5 Kg	2,5 Kg	2,50031 Kg	0,01 %

Dari data diatas dapat dilihat adanya perbedaan antara data berat yang belum terkirim dengan data berat yang telah terkirim dengan rata-rata *error* sebesar 0,2 %. Hal ini disebabkan karena nilai ADC yang masuk pada Atmega16 tidak sesuai dengan perhitungan yang ada dan data yang dikirimkan memiliki lebar nilai yang lebih besar yaitu 5 angka dibelakang koma.

4.11 Pengujian Pengiriman SMS Ke Delphi Secara Manual

Pada pengujian sistem pengiriman ini, perangkat yang dirangkai adalah modem wavecom yang dihubungkan dengan menggunakan kabel *serial* dan menggunakan juga kabel *serial* to USB ke laptop. Pesan yang dikirimkan adalah pesan dengan format a(pesan)b(pesan)c(pesan)d. Gambar 4.12 adalah tampilan pesan yang muncul pada Delphi.

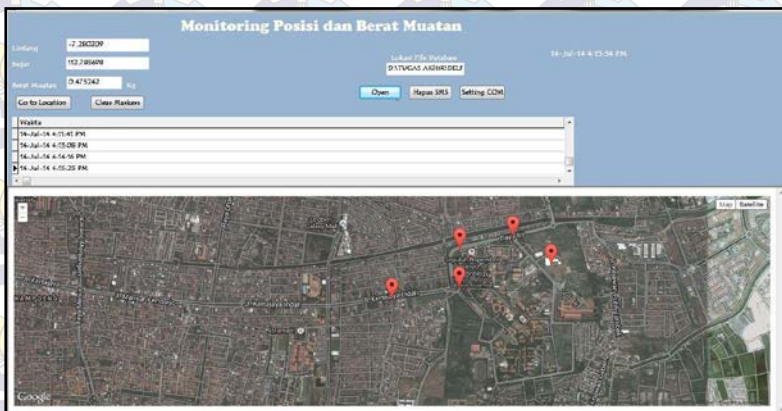


Gambar 4.12 Tampilan Pesan Pada Delphi

Pada gambar diatas dapat kita ketahui bahwa pesan terkirim secara manual pada Delphi dengan pesan yang berisikan 123,456,2kg yang terkirim pada tanggal 9 juni 2014 pada pukul 11.02 WIB.

4.12 Pengujian Sistem Keseluruhan

Pada pengujian sistem, semua perangkat dirangkai menjadi satu. Perangkat ini terdiri dari GPS *receiver*, mikrokontroler, modem GSM, dan *power supply*, dan *loadcell* yang ditempatkan di dalam kotak yang nantinya akan dipasang pada kendaraan. Pengujian ini dilakukan dengan menghubungkan laptop atau PC dengan modem GSM wavecom. Data posisi kendaraan seperti *latitude* dan *longitude* serta berat beban dari *loadcell* kemudian diolah oleh mikrokontroler lalu dikirim dengan media modem GSM yang nantinya ditampilkan pada akan ditampilkan pada *user interface* Delphi yang berfungsi untuk monitoring. Gambar 4.13 adalah tampilan Delphi pada saat aktif.



Gambar 4.13 Tampilan Delphi Pada Saat Bekerja

Data yang telah dikirimkan lalu disimpan dalam sebuah *database*. Data yang disimpan dalam *database* antara lain, waktu, *latitude*, *longitude*, dan berat muatan sehingga memiliki *record* selama pengawasan yang nantinya berfungsi untuk mengetahui sejarah pengiriman dan mengetahui apa saja yang terjadi pada tanggal tanggal tertentu sesuai dengan keinginan kita. *Database* dibuat menggunakan *Microsoft Access*. Pengiriman data antar modem *transmitter* dan *receiver* dikirim berdasarkan kelancaran dari sebuah *provider* seluler

yang digunakan. Apabila terjadi pengiriman data yang tidak sesuai dengan waktu yang ditentukan, hal ini dapat disebabkan data berupa SMS terkendala oleh jaringan telekomunikasi *provider* yang sedang mengalami gangguan. Pada Tabel 4.7 adalah tampilan *database* monitoring yang berisikan data waktu, posisi, serta berat muatan.

Tabel 4.7 Database Monitoring

Waktu	Berat	Latitude	Longitude
24-Jun-14 4:45:23 pm	0.04535	-7.279216	112.790870
24-Jun-14 4:45:53 pm	0.5321	-7.279216	112.790870
24-Jun-14 4:46:23 pm	1.0342	-7.279216	112.790870
24-Jun-14 4:46:53 pm	1.9743	-7.279216	112.790870
24-Jun-14 4:47:23 pm	0.04535	-7.279216	112.790870
24-Jun-14 4:47:53 pm	0.04535	-7.279216	112.790870

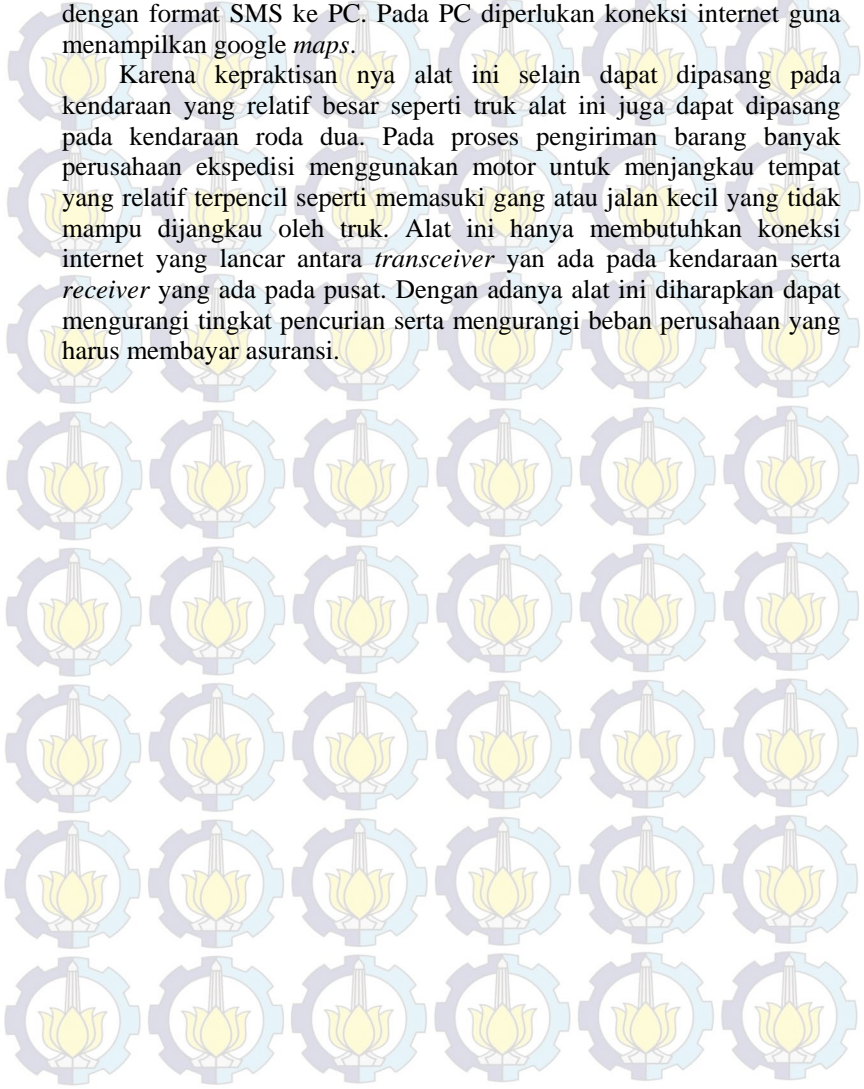
4.13 Deskripsi Tujuan Alat

Pembuatan alat ini ditujukan untuk perusahaan ekspedisi yang dewasa ini semakin diperlukan. Dengan meningkatnya jumlah manusia yang menggunakan internet hal ini meningkatkan juga transaksi jual beli. Untuk pengiriman barang antar penjual dan pembeli masih menggunakan jasa kurir yang disediakan oleh perusahaan ekspedisi yang ada di negara ini. Semakin banyak barang yang dikirim hal ini menyebabkan semakin banyak kendaraan pengantar yang dijalankan oleh perusahaan ekspedisi. Dengan banyaknya kendaraan yang kurang terawasi pada saat pengiriman menambah resiko pencurian yang dilakukan komplotan perampok pada saat kendaraan pengirim sedang pada proses pengiriman.

Alat ini bertujuan untuk mengurangi tingkat pencurian yang disebabkan oleh kurangnya pengawasan perusahaan pusat terhadap kendaraan pengirimannya. Alat ini berfungsi dengan cara memonitoring posisi dan berat muatan dari kendaraan secara *realtime*. Posisi dan muatan dari kendaraan akan terpantau oleh pusat yang memantunya dari tampilan delphi 7 yang berisikan posisi dengan menggunakan media

google *maps* serta berat yang tampil pada kolom di delphi 7. Alat ini bekerja dengan cara mengirimkan pesan yang berupa posisi dan berat dengan format SMS ke PC. Pada PC diperlukan koneksi internet guna menampilkan google *maps*.

Karena kepraktisan nya alat ini selain dapat dipasang pada kendaraan yang relatif besar seperti truk alat ini juga dapat dipasang pada kendaraan roda dua. Pada proses pengiriman barang banyak perusahaan ekspedisi menggunakan motor untuk menjangkau tempat yang relatif terpencil seperti memasuki gang atau jalan kecil yang tidak mampu dijangkau oleh truk. Alat ini hanya membutuhkan koneksi internet yang lancar antara *transceiver* yan ada pada kendaraan serta *receiver* yang ada pada pusat. Dengan adanya alat ini diharapkan dapat mengurangi tingkat pencurian serta mengurangi beban perusahaan yang harus membayar asuransi.



BAB V PENUTUP

Setelah perencanaan dan pembuatan alat serta pengujian dan analisis selesai dilakukan dengan mendapat hasil yang terbaik, maka dapat ditarik kesimpulan dan saran dari yang telah dilakukan sebagai berikut.

5.1 Kesimpulan

Hasil dari pengujian serta analisa data dari Prototipe Sistem Monitoring Berat Dan Posisi Kendaraan Bermuatan Dengan Menggunakan Pc, dapat diambil kesimpulan sebagai berikut :

1. Sistem Monitoring posisi menggunakan GPS ini memiliki nilai *error* yang kecil yaitu untuk *latitude* memiliki % *error* sebesar 0,002 % dan untuk % *error* dari *longitude* GPS adalah 0,004 %. Yang pada hitungan jarak masing masing dari % *error* ini bernilai 10–15 m.
2. Untuk % *error* berat, memiliki % *error* sebesar 0,2 %. Data ini didapatkan mengecek tampilan berat yang ada pada LCD dan membandingkan dengan perhitungan secara teori. Sedangkan untuk data berat beban yang tampil pada delphi 7 memiliki % *error* sebesar 2,6 %.
3. Gangguan jaringan pada *provider* dapat menyebabkan tertundanya pesan yang dikirim antara modem *transmitter* ke modem *receiver* yang terpasang pada PC.

5.2 Saran

Saran yang dapat diberikan penulis yaitu dalam pengujian GPS dilakukan sebaiknya di tempat terbuka sehingga GPS dapat menangkap sinyal satelit dengan lebih cepat dan lebih akurat apabila dibandingkan dengan melakukan pengujian di dalam ruangan. Untuk mekanik timbangan dari *loadcell* sebaiknya diukur dulu luas penampang nya, sehingga berat yang ada memiliki *error* yang kecil.



DAFTAR PUSTAKA

- [1] Andrianto, H., ***Pemrograman Mikrokontroler AVR ATMEGA16 Menggunakan Bahasa C (CodeVision AVR)***, Informatika: Bandung, 2008.
- [2] Tanoë, Andre, ***Berkenalan dengan GPS dan Penerapannya Pada Masyarakat***, Pohon Cahaya: Jakarta, 2011.
- [3] Lingga, W. Dan Makodian, N, ***Teknologi Wireless Communication dan Wireless Broadband***, Penerbit Andi: Jogjakarta, 2010.
- [4] Yuhfizard, ***Database Management Menggunakan Microsoft Access 2003***, Elex Media Komputindo: Jakarta, 2008.
- [5] Zuhail, ***Prinsip Dasar Elektroteknik***, Gramedia Pustaka Utama: Jakarta, 2004.
- [6] Budiharto, Widodo, ***Panduan Praktikum Mikrokontroler AVR Atmega16***, Elex Media Komputindo: Jakarta, 2008.
- [7] Ciang, David, ***Cara Mudah Pemrograman Database Delphi 7 Menggunakan Class Generator***, Elex Media Komputindo: Jakarta, 2004.



LAMPIRAN A

LISTING PROGRAM

a. Program GPS dan Loadcell

/*

*/

This program was produced by the
CodeWizardAVR V2.04.4a Advanced
Automatic Program Generator
© Copyright 1998-2009 Pavel Haiduc, HP InfoTech s.r.l.
<http://www.hpinfotech.com>

Project :
Version :
Date : 17/12/2013
Author : NeVaDa
Company : ITS Surabaya
Comments:

Chip type : ATmega16
Program type : Application
AVR Core Clock frequency: 12.000000 MHz
Memory model : Small
External RAM size : 0
Data Stack size : 256

*/

```
#include <mega16.h>
#include <delay.h>
#include <math.h>
#include <stdlib.h>
int t=0,kirim;
// Alphanumeric LCD Module functions
#asm
.equ __lcd_port=0x15 ;PORTC
#endasm
#include <lcd.h>
```

```
#ifndef RXB8
```



```
#define RXB8 1
#endif
```

```
#ifndef TXB8
#define TXB8 0
#endif
```

```
#ifndef UPE
#define UPE 2
#endif
```

```
#ifndef DOR
#define DOR 3
#endif
```

```
#ifndef FE
#define FE 4
#endif
```

```
#ifndef UDRE
#define UDRE 5
#endif
```

```
#ifndef RXC
#define RXC 7
#endif
```

```
#define FRAMING_ERROR (1<<FE)
#define PARITY_ERROR (1<<UPE)
#define DATA_OVERRUN (1<<DOR)
#define DATA_REGISTER_EMPTY (1<<UDRE)
#define RX_COMPLETE (1<<RXC)
```

```
//unsigned char lcd_buff[20];
unsigned char lcd_buff1[20];
unsigned char lcd_buff2[20];
//unsigned char lcd_buff3[20];
//unsigned char lcd_buff4[20];
```

```

//deklarasi variabel gps
//unsigned char ind[10];
unsigned char longitude[15],latitude[15];
unsigned char cLintang[10],cBujur[10],linmen[10],bujmen[10];
//unsigned char jam[15],satelite[10];
unsigned int iLintang,iBujur;
unsigned int Lintang=0,Bujur=0;
float fLintang, fBujur, lintGMAPS, bujurGMAPS;

// USART Receiver buffer
#define RX_BUFFER_SIZE 8
char rx_buffer[RX_BUFFER_SIZE];

#if RX_BUFFER_SIZE<256
unsigned char rx_wr_index,rx_rd_index,rx_counter;
#else
unsigned int rx_wr_index,rx_rd_index,rx_counter;
#endif

// This flag is set on USART Receiver buffer overflow
bit rx_buffer_overflow;
unsigned char buffgps[100];
unsigned int countgps=0;
unsigned char i;

// USART Receiver interrupt service routine
interrupt [USART_RXC] void usart_rx_isr(void)
{
char status,data;
status=UCSRA;
data=UDR;
if(data=='$')
{
countgps=0;
}
buffgps[countgps]=data;
countgps++;

```

```
if(buffgps[countgps-1]=='*')
```

```
{  
    if(buffgps[0]=='$' && buffgps[1]=='G' && buffgps[2]=='P' &&  
buffgps[3]=='G' && buffgps[4]=='G' && buffgps[5]=='A')  
    {  
//        for(i=7;i<13;i++)  
//        {  
//            jam[0]=buffgps[7];  
//            jam[1]=buffgps[8];  
//            jam[2]=':';  
//            jam[3]=buffgps[9];  
//            jam[4]=buffgps[10];  
//            jam[5]=':';  
//            jam[6]=buffgps[11];  
//            jam[7]=buffgps[12];  
//        }  
        for(i=18;i<=26;i++)  
        {  
            latitude[i-18]=buffgps[i];  
            if(i<20){  
                cLintang[i-18]=buffgps[i];  
                if(i==19)  
                    iLintang=atoi(cLintang);  
            }  
            else{  
                linmen[i-20]=buffgps[i];  
            }  
            if(i==26){  
                fLintang=atof(linmen)/60;  
                lintGMAPS=iLintang+fLintang;  
                Lintang=lintGMAPS*1000000;  
            }  
        }  
        i++;  
        for(i=30;i<=39;i++)
```

```

{
longitude[i-30]=buffgps[i];
if(i<33){
    cBujur[i-30]=buffgps[i];
    if(i==32)iBujur=atoi(cBujur);
}
else{
    bujmen[i-33]=buffgps[i];
}
if(i==39){
    fBujur=atof(bujmen)/60;
    bujurGMAPS=iBujur+fBujur;
    Bujur=bujurGMAPS*1000000;
}
}
i++;
// for(i=43;i<44;i++){
// ind[0]=buffgps[43];// detekssi fix satelit
// }
//
// for(i=45;i<47;i++){
// satelite[0]=buffgps[45];
// satelite[1]=buffgps[46];
// }
// countgps=0;
}

if ((status & (FRAMING_ERROR | PARITY_ERROR |
DATA_OVERRUN))==0)
{
rx_buffer[rx_wr_index]=data;
if (++rx_wr_index == RX_BUFFER_SIZE) rx_wr_index=0;
if (++rx_counter == RX_BUFFER_SIZE)
{
rx_counter=0;
rx_buffer_overflow=1;
};
};

```

```

    };
}

#ifndef _DEBUG_TERMINAL_IO_
// Get a character from the USART Receiver buffer
#define _ALTERNATE_GETCHAR_
#pragma used+
char getchar(void)
{
    char data;
    while (rx_counter==0);
    data=rx_buffer[rx_rd_index];
    if (++rx_rd_index == RX_BUFFER_SIZE) rx_rd_index=0;
    #asm("cli")
    --rx_counter;
    #asm("sei")
    return data;
}
#pragma used-
#endif

// USART Transmitter buffer
#define TX_BUFFER_SIZE 8
char tx_buffer[TX_BUFFER_SIZE];

#if TX_BUFFER_SIZE<256
unsigned char tx_wr_index,tx_rd_index,tx_counter;
#else
unsigned int tx_wr_index,tx_rd_index,tx_counter;
#endif

// USART Transmitter interrupt service routine
interrupt [USART_TXC] void usart_tx_isr(void)
{
    if (tx_counter)
    {
        --tx_counter;
        UDR=tx_buffer[tx_rd_index];
        if (++tx_rd_index == TX_BUFFER_SIZE) tx_rd_index=0;
    }
}

```

```

    };
}

#ifndef _DEBUG_TERMINAL_IO_
// Write a character to the USART Transmitter buffer
#define _ALTERNATE_PUTCHAR_
#pragma used+
void putchar(char c)
{
    while (tx_counter == TX_BUFFER_SIZE);
    #asm("cli")
    if (tx_counter || ((UCSRA & DATA_REGISTER_EMPTY)==0))
    {
        tx_buffer[tx_wr_index]=c;
        if (++tx_wr_index == TX_BUFFER_SIZE) tx_wr_index=0;
        ++tx_counter;
    }
    else
        UDR=c;
    #asm("sei")
}
#pragma used-
#endif

// Standard Input/Output functions
#include <stdio.h>
interrupt [TIM0_OVF] void timer0_ovf_isr(void)
{
    TCNT0=0x21;
    t++;
    if (t==3)
    {
        kirim=1;
        t=0;
    };
}

void main(void)

```

```

{
// Declare your local variables here

// Input/Output Ports initialization
// Port A initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In
Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T
State0=T
PORTA=0x00;
DDRA=0xFF;

// Port B initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In
Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T
State0=T
PORTB=0x00;
DDRB=0x00;

// Port C initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In
Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T
State0=T
PORTC=0x00;
DDRC=0x00;

// Port D initialization
// Func7=In Func6=In Func5=In Func4=In Func3=In Func2=In
Func1=In Func0=In
// State7=T State6=T State5=T State4=T State3=T State2=T State1=T
State0=T
PORTD=0x00;
DDRD=0x00;

// Timer/Counter 0 initialization
// Clock source: System Clock
// Clock value: 11.719 kHz

```

```
// Mode: Normal top=FFh
// OC0 output: Disconnected
TCCR0=0x05;
TCNT0=0x21;
OCR0=0x00;

// Timer/Counter 1 initialization
// Clock source: System Clock
// Clock value: Timer1 Stopped
// Mode: Normal top=FFFFh
// OC1A output: Discon.
// OC1B output: Discon.
// Noise Canceler: Off
// Input Capture on Falling Edge
// Timer1 Overflow Interrupt: Off
// Input Capture Interrupt: Off
// Compare A Match Interrupt: Off
// Compare B Match Interrupt: Off
TCCR1A=0x00;
TCCR1B=0x00;
TCNT1H=0x00;
TCNT1L=0x00;
ICR1H=0x00;
ICR1L=0x00;
OCR1AH=0x00;
OCR1AL=0x00;
OCR1BH=0x00;
OCR1BL=0x00;

// Timer/Counter 2 initialization
// Clock source: System Clock
// Clock value: Timer2 Stopped
// Mode: Normal top=FFh
// OC2 output: Disconnected
ASSR=0x00;
TCCR2=0x00;
TCNT2=0x00;
OCR2=0x00;
```



```
// External Interrupt(s) initialization
// INT0: Off
// INT1: Off
// INT2: Off
MCUCR=0x00;
MCUCSR=0x00;

// Timer(s)/Counter(s) Interrupt(s) initialization
TIMSK=0x00;

// USART initialization
// Communication Parameters: 8 Data, 1 Stop, No Parity
// USART Receiver: On
// USART Transmitter: On
// USART Mode: Asynchronous
// USART Baud Rate: 4800
UCSRA=0x00;
UCSRB=0xD8;
UCSRC=0x86;
UBRRH=0x00;
UBRRL=0x9B;

// Analog Comparator initialization
// Analog Comparator: Off
// Analog Comparator Input Capture by Timer/Counter 1: Off
ACSR=0x80;
SFIOR=0x00;

// LCD module initialization
lcd_init(16);

// Global enable interrupts
#pragma asm("sei")
kirim=0;
while (1)
{
while (kirim==0)
{
// Place your code here
```

```

//sprintf(lcd_buff,"%s",jam);
//lcd_gotoxy(0,0);
//lcd_putsf("Jam :");
//lcd_puts(lcd_buff);

sprintf(lcd_buff1,"-%f",lintGMAPS);
lcd_gotoxy(0,0);
lcd_putsf("Lat :");
lcd_puts(lcd_buff1);

sprintf(lcd_buff2,"%f",bujurGMAPS);
lcd_gotoxy(0,1);
lcd_putsf("Lon :");
lcd_puts(lcd_buff2);
delay_ms(100);
 kirim=1;

//printf("#-%f#%f\r",lintGMAPS,bujurGMAPS);
// if (mnt%1==0 && dtk<=2)
// {
//   sms();
// }
// printf("#-%f#%f\r",lintGMAPS,bujurGMAPS);
};

//void sms()
// {
// //Kirim SMS
// printf("AT+CMGF=1"); //Mengeset Modem sebagai Mode text
// putchar(13);
// putchar(10);
// delay_ms(100);
// printf("AT+CMGS="); //Mengirim SMS
// putchar(34); //karakter "
// printf("085733926379"); //No. Tujuan
// putchar(34); // karakter "

```

```
// putchar(13); // enter
// delay_ms(1000);
// printf("a-%fb%fcMOTOR 1d", lintGMAPS, bujurGMAPS); // isi
data yang akan dikirimkan
// putchar(13); //enter
// delay_ms(1000);
// putchar(26); // karakter ctrl+z
// putchar(13);
// putchar(10);
// delay_ms(1000);
```



b. Program Delphi

unit fMain;

interface

uses

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
Forms,
Dialogs, OleCtrls, SHDocVw, StdCtrls, ExtCtrls, XPMAN,
ComCtrls,MSHTML,
DBCtrls, Grids, DBGrids, CPort, DB, ADOdb;

type

TfrmMain = class(TForm)
 WebBrowser1: TWebBrowser;
 PanelHeader: TPanel;
 XPManifest1: TXPManifest;
 LabelLatitude: TLabel;
 LabelLongitude: TLabel;
 Longitude: TEdit;
 Latitude: TEdit;
 ButtonClearMarkers: TButton;
 DataSource1: TDataSource;
 OpenDialog1: TOpenDialog;
 ComPort1: TComPort;
 Button1: TButton;
 Label1: TLabel;
 Label2: TLabel;
 Edit1: TEdit;
 Button2: TButton;
 Label3: TLabel;
 Edit2: TEdit;
 Timer1: TTimer;
 Label4: TLabel;
 ButtonGotoLocation: TButton;
 Button3: TButton;
 ADOTable1: TADOTable;
 ADOConnection1: TADOConnection;
 Label5: TLabel;

```

DBGrid1: TDBGrid;
procedure FormCreate(Sender: TObject);
procedure ButtonGotoLocationClick(Sender: TObject);
procedure CheckBoxTrafficClick(Sender: TObject);
procedure CheckBoxBicyclingClick(Sender: TObject);
procedure CheckBoxStreeViewClick(Sender: TObject);
procedure ButtonClearMarkersClick(Sender: TObject);
procedure ComPort1RxChar(Sender: TObject; Count: Integer);
procedure Button1Click(Sender: TObject);
procedure Button2Click(Sender: TObject);
procedure Timer1Timer(Sender: TObject);
procedure FormClose(Sender: TObject; var Action: TCloseAction);
procedure Button3Click(Sender: TObject);
private
  { Private declarations }
  HTMLWindow2: IHTMLWindow2;
public
  { Public declarations }
end;

var
  frmMain: TfrmMain;
  isi:TStringList;
  data,data1,data2,data3:string;
implementation
uses
  ActiveX;

{$R *.dfm}

const
  HTMLStr: AnsiString = //untuk buka halaman halaman web yang
mengawali terjadinya google maps dalam const String
'<html> '+
'<head> '+
'<meta name="viewport" content="initial-scale=1.0, user-scalable=yes"
/> '+

```

```

<script type="text/javascript"
src="http://maps.google.com/maps/api/js?sensor=true"></script> '+
<script type="text/javascript"> '+
"+
"+ //Deklarasikan variabel yang akan digunakan dalam fungsi
javascript
' var geocoder; '+ //menampilkan data ke peta
' var map; '+ //menampilkan peta
' var trafficLayer;'+ //menampilkan lalu lintas
' var bikeLayer;'+ //menampilkan lintasan bersepeda
' var markersArray = [];'+ //sebagai penanda posisi
"+
"+
' function initialize() { '+
' geocoder = new google.maps.Geocoder();'+
' var latlng = new google.maps.LatLng(-7.28167314,112.7898994); '+
//Mengatur koordinat awal untuk peta
' var myOptions = { '+
' zoom: 13, '+
' center: latlng, '+
' mapTypeId: google.maps.MapTypeId.ROADMAP '+
//Mengatur jenis standar peta
' }; '+
' map = new
google.maps.Map(document.getElementById("map_canvas"),
myOptions); '+
' trafficLayer = new google.maps.TrafficLayer();'+ //Buat contoh
lalu lintas Lapisan
' bikeLayer = new google.maps.BicyclingLayer();'+ //Buat
contoh lintasan sepeda
' map.set("streetViewControl", false);'+ //mengetahui
jalan
' } '+
"+
"+
' function codeAddress(address) { '+ //fungsi untuk menerjemahkan
dan penanda alamat untuk mengkoordinasikan dan menempatkan alamat
' if (geocoder) {'+
' geocoder.geocode( { address: address}, function(results, status) { '+

```

```

'   if (status == google.maps.GeocoderStatus.OK) {'+
'       map.setCenter(results[0].geometry.location);'+
'       PutMarker(results[0].geometry.location.lat(),
results[0].geometry.location.lng(),
results[0].geometry.location.lat()+", "+results[0].geometry.location.lng())
';'+
'       } else {'+
'           alert("Geocode was not successful for the following reason: " +
status);'+
'       }'+
'   });'+
' }'+
' }'+
'+
'+
' function GotoLatLng(Lat, Lang) { '+ //Mengatur peta dalam
koordinat dan menempatkan penanda
'   var latlng = new google.maps.LatLng(Lat,Lang);'+
'   map.setCenter(latlng);'+
'   PutMarker(Lat, Lang, Lat+", "+Lang);'+
' }'+
'+
'+
'function ClearMarkers() { '+ //menghapus penanda
' if (markersArray) { '+
'   for (i in markersArray) { '+
'     markersArray[i].setMap(null); '+
'   }'+
' }'+
' }'+
'+
' function PutMarker(Lat, Lang, Msg) { '+ //menandai posisi garis
lintang dan bujur
'   var latlng = new google.maps.LatLng(Lat,Lang);'+
'   var marker = new google.maps.Marker({'+
'     position: latlng, '+
'     map: map, '+
'     title: Msg+" ("+Lat+", "+Lang+)"'+
'   });'+

```

```

' markersArray.push(marker); '+
' }'+
"+
"+
' function TrafficOn() { trafficLayer.setMap(map); }'+ //Aktifkan
layer Lalu Lintas
"+
' function TrafficOff() { trafficLayer.setMap(null); }'+
//menonaktifkan layer lalu lintas
"+
"+
' function BicyclingOn() { bikeLayer.setMap(map); }'+ //Aktifkan
layer lintasan sepeda
"+
' function BicyclingOff(){ bikeLayer.setMap(null); }'+
//menonaktifkan layer lintasan sepeda
"+
' function StreetViewOn() { map.set("streetViewControl", true); }'+
//Aktifkan tampilan jalan
"+
' function StreetViewOff() { map.set("streetViewControl", false); }'+
//menonaktifkan tampilan jalan
"+
"+'</script> '+
'<head> '+
'<body onload="initialize()"> '+
' <div id="map_canvas" style="width:100%; height:100%"></div> '+
//ukuran panjang dan lebar tampilan google maps
'</body> '+
'</html> ';

procedure TfrmMain.FormCreate(Sender: TObject);
var
  aStream : TMemoryStream;
begin
  WebBrowser1.Navigate('about:blank'); //Mengatur lokasi ke
halaman kosong
  if Assigned(WebBrowser1.Document) then
  begin

```



```
aStream := TMemoryStream.Create; //membuat memory system
untuk memuat halaman dari string
```

```
try
  aStream.WriteBuffer(Pointer(HTMLStr)^, Length(HTMLStr));
//Salin string ke listing
  //aStream.Write(HTMLStr[1], Length(HTMLStr));
  aStream.Seek(0, soFromBeginning);
  (WebBrowser1.Document as
  IPersistStreamInit).Load(TStreamAdapter.Create(aStream));
//Memuat halaman dari listing
finally
  aStream.Free;
end;
HTMLWindow2 := (WebBrowser1.Document as
IHTMLDocument2).parentWindow //Mengatur contoh dari
parentWindow untuk memanggil fungsi javascripts

end;
end;
```

```
procedure TfrmMain.ButtonGotoLocationClick(Sender: TObject);
begin
  //Panggil kode fungsi Alamat untuk pergi ke alamat
yang dituju
```

```
HTMLWindow2.execScript(Format('GotoLatLng(%s,%s)',[Latitude.Text,
Longitude.Text]), 'JavaScript');
end;
```

```
procedure TfrmMain.ButtonClearMarkersClick(Sender: TObject);
begin
  HTMLWindow2.execScript('ClearMarkers()', 'JavaScript')
//menghapus penanda
end;
```

```
procedure TfrmMain.CheckBoxStreeViewClick(Sender: TObject);
begin
  //if CheckBoxStreeView.Checked then
```

```

//HTMLWindow2.execScript('StreetViewOn()', 'JavaScript')
//Aktifkan opsi tampilan jalan
//else
//HTMLWindow2.execScript('StreetViewOff()', 'JavaScript');
//Nonaktifkan opsi tampilan jalan
end;

procedure TfrmMain.CheckBoxBicyclingClick(Sender: TObject);
begin
//if CheckBoxBicycling.Checked then
//HTMLWindow2.execScript('BicyclingOn()', 'JavaScript')
//Aktifkan lintasan sepeda
//else
//HTMLWindow2.execScript('BicyclingOff()', 'JavaScript');
//Nonaktifkan lintasan sepeda
end;

procedure TfrmMain.CheckBoxTrafficClick(Sender: TObject);
begin
//if CheckBoxTraffic.Checked then
//HTMLWindow2.execScript('TrafficOn()', 'JavaScript')
//Aktifkan Lalu Lintas Lihat pilihan
//else
//HTMLWindow2.execScript('TrafficOff()', 'JavaScript');
//Nonaktifkan Lalu Lintas Lihat pilihan
end;

procedure TfrmMain.ComPort1RxChar(Sender: TObject; Count:
Integer);
var a,n,i,a1,a2,a3,a4: Integer;
jml,number,sms,d:string;
begin
ComPort1.ReadStr(data,Count); //membaca semua data serial
//memo1.Text:=data;
d:="";
//baca sms masuk

```

```
if (Pos('+CMTI',data)>0) then
```

```
begin
```

```
  a:=Pos(',',data);
```

```
  n:=a+1;
```

```
  while(n<Length(data)-1) do
```

```
  begin
```

```
    d:=d+data[n];
```

```
    n:=n+1;
```

```
  end;
```

```
  jml:=d;
```

```
  Sleep(200);
```

```
  ComPort1.WriteStr('AT+CMGR='+d+#13);
```

```
  Sleep(250);
```

```
end;
```

```
if (Pos('+CMGR',data)>0) then
```

```
begin
```

```
  //pisah nomer
```

```
  a:=Pos(" ",",",data);
```

```
  n:=a+3;
```

```
  a:=pos(" ",",",data);
```

```
  for i:=n to a-1 do
```

```
    d:=d+data[i];
```

```
  number:=d;
```

```
  sleep(200);
```

```
  //pisah data sms
```

```
  d:=";
```

```
  sms:=";
```

```
  sleep(100);
```

```
  a:=pos(" "+#13,data);
```

```
  n:=a+3;
```

```
  a:=pos(#13#10+'OK',data);
```

```
  for i:=n to a-3 do
```

```
  begin
```

```
    d:=d+data[i];
```

```
  end;
```

```
  sms:=d;
```

```

sleep(500);

// proses program
a1:=Pos('a',sms);
a2:=Pos('b',sms);
a3:=Pos('c',sms);
a4:=Pos('d',sms);

data1:=""; data2:=""; data3:="";
for i:=a1+1 to a2-1 do
begin
    data1:=data1+sms[i];
end;
for i:=a2+1 to a3-1 do
begin
    data2:=data2+sms[i];
end;
for i:=a3+1 to a4-1 do
begin
    data3:=data3+sms[i];
end;
sleep(100);
Latitude.Text:=data1;
Longitude.Text:=data2;
edit2.Text:=data3;

//masukkan data ke access
if (latitude.Text<>"") and (longitude.Text<>"") then
begin
ADOTable1.Open;
//ADOTable1.Locate('ID Kendaraan',data3,[]);
ADOTable1.Append;
    ADOTable1.FieldName('Latitude').AsString:=data1;
    ADOTable1.FieldName('Longitude').AsString:=data2;
    ADOTable1.FieldName('Berat').AsString:=data3;

ADOTable1.FieldName('Waktu').AsString:=DateTimeToStr(Now);
ADOTable1.Post;

```

```
sleep(10);
```

```
HTMLWindow2.execScript(Format('GotoLatLng(%s,%s)',[data1,data2]), 'JavaScript');
```

```
//hapus SMS
```

```
//if(jml='10') then
```

```
//begin
```

```
//ComPort1.WriteStr('AT+CMGD=1,4'+#13);
```

```
//sleep(1000);
```

```
//end;
```

```
end;
```

```
end;
```

```
procedure TfrmMain.Button1Click(Sender: TObject);
```

```
begin
```

```
comport1.ShowSetupDialog;
```

```
comport1.Open;
```

```
end;
```

```
procedure TfrmMain.Button2Click(Sender: TObject);
```

```
begin
```

```
OpenDialog1.Execute;
```

```
Edit1.Text:=OpenDialog1.FileName;
```

```
ADOConnection1.ConnectionString:='Provider=Microsoft.ACE.OLEDB.12.0;User ID=Admin;'+
```

```
'Data Source='+OpenDialog1.FileName+
```

```
';Mode=Share Deny None;Jet OLEDB:System
```

```
database="'+
```

```
'';Jet OLEDB:Registry Path=''';Jet
```

```
OLEDB:Database Password=''';+
```

```
'Jet OLEDB:Engine Type=6;Jet OLEDB:Database
```

```
Locking Mode=1;'+
```

```
'Jet OLEDB:Global Partial Bulk Ops=2;Jet
```

```
OLEDB:Global Bulk Transactions=1;'+
```

```
'Jet OLEDB:New Database Password=''';Jet
```

```
OLEDB:Create System Database=False;'+
```

```

'Jet OLEDB:Encrypt Database=False;Jet
OLEDB:Don't Copy Locale on Compact=False;'+
'Jet OLEDB:Compact Without Replica
Repair=False;Jet OLEDB:SFP=False;'+
'Jet OLEDB:Support Complex Data=False;';
ADOTable1.Open;

end;

procedure TfrmMain.Timer1Timer(Sender: TObject);
begin
Label4.Caption := DateTimeToStr(Now);
end;

procedure TfrmMain.FormClose(Sender: TObject; var Action:
TCloseAction);
begin
adotable1.Close;
comport1.close;
end;

procedure TfrmMain.Button3Click(Sender: TObject);
begin
ComPort1.WriteString('AT+CMGD=1,4'+#13);
sleep(1000);
end;
end.

```



LAMPIRAN B DATASHEET

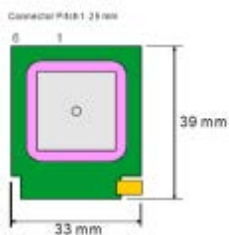
POLSTAR

PMB-688 GPS module



PMB-688 FEATURES

- Built-in SiRFstarIII chipsets receivers give unparalleled GPS performance and precision, 20 parallel satellite-tracking channels for fast acquisition and reacquisition.
- Built-in WAAS/EGNOS Demodulator.
- Low power consumption and ultra mini size only 33x39mm.
- Built-in rechargeable battery for backup memory and RTC backup.
- Support NMEA0183 v2.2 data protocol.
- Enhanced algorithms providing superior navigation performance in urban, canyon and foliage environments.
- For Car Navigation, Marine Navigation, Fleet Management, AVL and Location-Based Services, Auto Pilot, Personal Navigation or touring devices, Tracking devices/systems and Mapping devices application.
- Include RF MMCX connector (Optional: Active Antenna)



PMB-888 Specifications

GPS IC	SIRFstar III
Receiver:	Tracking up to 20 satellites L1, 1575.42 MHz, C/A code
Accuracy:	Position: 2DRMS approximately 6m, WAAS support Velocity: 0.1 m/s without SA imposed. Time: $\pm 1\mu\text{sec}$
Acquisition Time:	Cold Start: 42sec (Average) Warm Start: 38sec (Average) Hot Start: 1sec (Min.)
Sensitivity:	Acquisition: -148dBm Tracking: -159dBm
Dynamics:	Altitude: 18000m (Max.) Velocity: 515m/s (Max.) Acceleration: $\pm 4g$ (Max.)
Navigation update rate :	Once per second
Serial Port :	TTL
Baud Rate :	4800 bps (Optional 9600,19300,38400 bps)
Output Message:	NMEA0183 V2.2 GGA, GSV, GSA, RMC (optional VTG, GLL)
Datum:	WGS 84
Power supply :	DC 3.3V ~ 5V
Power Consumption :	Typical 65mA @12V
LED Function	Power on/off and Navigation
Operating Temp.:	-20°C ~ $+70^{\circ}\text{C}$
Storage Temp.:	-30°C ~ $+85^{\circ}\text{C}$
Humidity:	5%~95%
Antenna Type:	Built-in patch antenna
RF Connector:	MMCX type (Optional : Active Antenna)

3 General Presentation

3.1 Description

The Fastrack M1308B description is given in the Figure 4 below.



Figure 4: Fastrack M1308B general description

Fasttrack M1306B User Guide General Presentation

In addition, two holding bridles are provided to tighten the M1306B on a support.

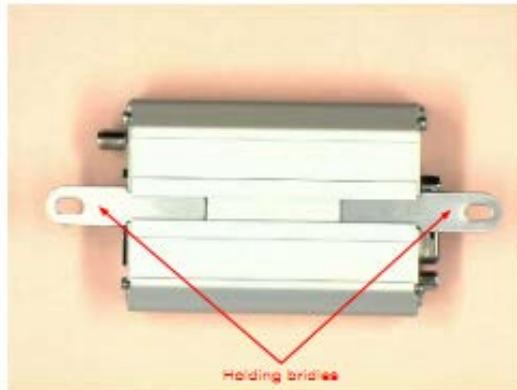


Figure 8: Fasttrack M1306B holding bridles

3.2 External Connections

3.2.1 Connectors

3.2.1.1 Antenna Connector

The antenna connector is a SMA type connector for a 50 Ω RF connection.

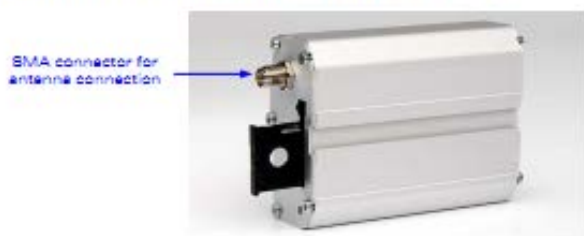


Figure 6: SMA connector for antenna connection

3.2.1.2 Power Supply Connector

The power supply connector is a 4-pin Micro FIT connector for:

- external DC Power Supply connection,
- GPIOs connection (two General Purpose Input/Output signals available).



Figure 7: Power supply connector

Featrack M1308B User Guide
General Presentation

Table 1: Power supply connector pin description

Pin #	Signal	I/O	I/O type	Description	Comment
1	V+BATTERY	I	Power supply	Battery voltage input: • 5.5 V Min. • 13.2 V Typ. • 32 V Max.	High current
2	GND		Power supply	Ground	
3	GPIO4	I/O	CMOS(2X)	General Purpose Input/output	
4	GPIO5	I/O	CMOS(2X)	General Purpose Input/output	

Warning:

Both pin 3 and pin 4 are used by GPIO interface. It is strictly prohibited to connect them to any power supply at the risk of damage to the M1308B.

3.2.1.3 Sub HD 15-pin Connector

The Sub D high density 15-pin connector is used for:

- RS232 serial link connection,
- Audio lines (microphone and speaker) connection,
- BOOT and RESET signal connection.



Figure 8: Sub HD 15-pin connector

Featrack M1308B User Guide
General Presentation

Table 2: Sub HD 15-pin connector description

Pin #	Signal (OCITT / EIA)	I/O	I/O type	Description	Comment
1	CCDC/CT109	O	STANDARD RS232	RS232 Data Carrier Detect	
2	CTXD/CT103	I	STANDARD RS232	RS232 Transmit serial data	
3	BOOT	I	CMOS	Boot	This signal must not be connected. Its use is strictly reserved to Wavocom or competent retailers.
4	CMIC2P	I	Analog	Microphone positive line	
5	CMIC2N	I	Analog	Microphone negative line	
6	CRXD/CT104	O	STANDARD RS232	RS232 Receive serial data	
7	CDBR/CT107	O	STANDARD RS232	RS232 Data Bit Ready	
8	CDTR/CT108-2	I	STANDARD RS232	RS232 Data Terminal Ready	
9	GND	-	GND	Ground	
10	CSPK2P	O	Analog	Speaker positive line	
11	CCTB/CT106	O	STANDARD RS232	RS232 Clear To Send	
12	CRTS/CT105	I	STANDARD RS232	RS232 Request To Send	
13	CR/CT126	O	STANDARD RS232	RS232 Ring Indicator	
14	RESET	I/O	Schmitt	Wireless CPU® reset	Active low
15	CSPK2N	O	Analog	Speaker negative line	

3.2.2 Power supply cable

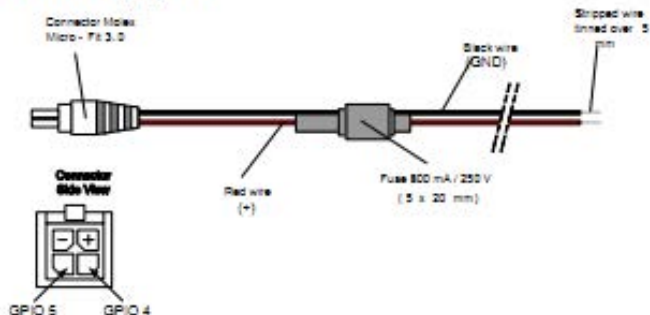


Figure 9: Power supply cable

Component	Characteristic
Micro-Fit 4-pin connector	Part number: MOLEX 45025-0400
Cable	Cable length: ~1.5 m
Wire	Core: tinned copper 24 x 0.2 mm Section: 0.75 mm ²

4 Basic Features and Services

Basic features of the M1306B and available services are summarized in the table below.

Table 3: Basic features of the M1306B

Features	GSM	DCS
Open AT[®]	Open AT [®] programmable: Native execution of embedded standard ANSI C applications, Custom AT command creation, Custom application library creation, Standalone operation.	
Standard	900 MHz. E-GSM compliant. Output power: class 4 (2W). Fully compliant with ETSI GSM phase 2 + small MB.	1800 MHz. Output power: class 1 (1W). Fully compliant with ETSI GSM phase 2 + small MB.
GPRS	Class 10. RBCCH support. Coding schemes: CS1 to CS4. Compliant with SMGS1bis. Embedded TCP/IP stack (optional).	
Interface	RS232 (V.24/V.28) Serial interface supporting: <ul style="list-style-type: none"> • Baud rate (bits/s): 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200, • Autobauding (bits/s): 2400, 4800, 9600, 19200, 38400, 57600. 2 General Purpose Input/Output gates (GPIOs) available. 3 V BIM interface. AT command set based on V.25ter and GSM 07.05 & 07.07. Open AT [®] interface for embedded application.	
SMS	Text & PDU. Point to point (MT/MO). Cell broadcast.	

Featrack M1306B User Guide

Basic Features and Services

Feature	GSM	DCS
Data	Data circuit asynchronous. Transparent and Non Transparent modes. Up to 14,400 bits/s. MNP Class 2 error correction. V42 bis data compression.	
Fax	Automatic fax group 3 (class 1 and Class 2).	
Audio	Echo cancellation Noise reduction Telephony. Emergency calls. Full Rate, Enhanced Full Rate and Half Rate operation (FR/EFR/HR). Dual Tone Multi Frequency function (DTMF).	
GSM supplement services	Call forwarding. Call barring. Multiparty. Call waiting and call hold. Calling line identity. Advice of charge. USSD	
Other	DC power supply Real Time Clock with calendar Complete shielding	

For other detailed technical characteristics, refer to Chapter B.

7 Functional Description

7.1 Architecture

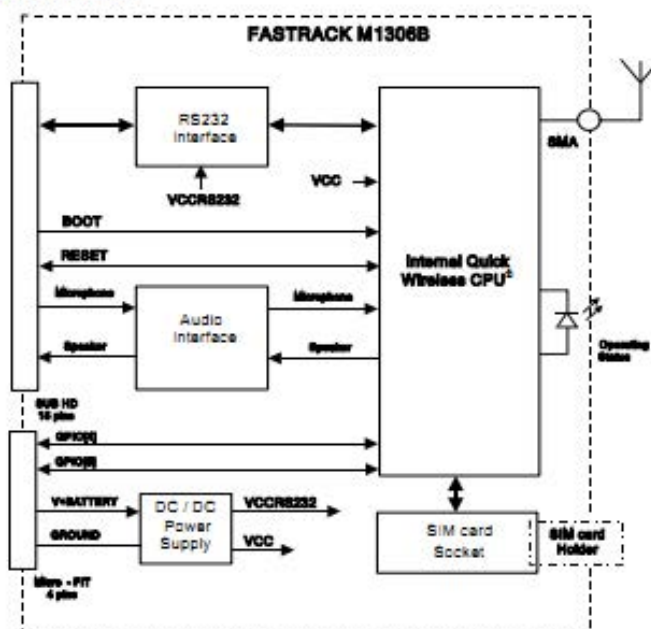


Figure 13: Functional architecture

8.2 Electrical Characteristics

8.2.1 Power Supply

Table 13: Electrical characteristics

Operating Voltage ranges	5.5 V to 32 V DC (GSM or DCB or GPRS).
Maximum current	480 mA Average at 5.5V. 2.1 A Peak at 5.5 V.

Note

The M1306B is permanently powered once the power supply is connected. The following table describes the consequences of over-voltage and under-voltage with the Fastrack M1306B.

Warning:

All the input voltage specification described in this chapter are at the M1306B input. While powering the M1306B, take into account the input drop caused by the power cable. With the delivered cable, this input drop is around 700 mV at 5.5 V and 220 mV at 32V.

Table 14: Effects of power supply defect

If the voltage	then
falls below 5.5 V,	the GSM communication is not guaranteed.
is over 32 V (Transient peaks),	the M1306B guarantees its own protection.
is over 32 V (continuous overvoltage)	the protection of the M1306B is done by the fuse (the supply voltage is disconnected).

The fuse is a 800 mA / 250 V FAST-ACTING 5*20mm. See Chapter 10 for recommended references.

The following table provides information on power consumption of the Fastrack M1306B, assuming an operating temperature of +25 °C and using a 3 V SIM card.

2 NMEA Messages

2.1 General Format of NMEA Messages

NMEA messages use the ASCII character set and have a defined format. Each message begins with a \$ (hex 0x24) and end with a carriage return and line feed (hex 0x0D 0x0A, represented as <CR><LF>). Each message consists of one or more fields of ASCII letters and numbers, separated by commas. After the last field, and before the <CR><LF> is a checksum consisting of an asterisk (*, hex 0x2A) followed by two ASCII characters representing the hexadecimal value of the checksum. The checksum is computed as the exclusive OR of all characters between the \$ and * characters.

Table 2-1 illustrates the NMEA output/input message parameters.

Table 2-1: NMEA output/input message parameters

Parameter	Example	Contents
Start	\$GPGGA \$PSRF<MID>	Message Identifier. Input messages begin at MID 100.
Sequence	Data	Message specific data. Refer to a specific message section for <data>...<data> definition.
Checksum	*CKSUM	CKSUM is a two-hex ASCII character. Checksum is required in all input messages.
Payload	<CR> <LF>	Each message is terminated using Carriage Return (CR) Line Feed (LF) which are \r\n. Because \r\n are not printable ASCII characters, they are omitted from the example strings, but must be sent to terminate the message and cause the receiver to process that input message.

Note:

- (1). All fields in all proprietary NMEA messages are required, none are optional and are comma delimited.
- (2). In some numeric fields representing a single data element, leading zeros before a decimal are suppressed. A single "0" character preceding the decimal point is maintained. In compound numeric structures (such as LAT or LONG), leading zeros are suppressed only on the leftmost element. Trailing zeros are not suppressed.

2.2 NMEA Output Messages

Table 2-2: SIM18 Frequently Used NMEA Output Messages

Message	Description
---------	-------------



GGA	Time, position and fix type data
GLL	Latitude, longitude, LTC time of position fix and status
GSA	GPS receiver operating mode, satellites used in the position solution, and DOP values
GSV	Number of GPS satellites in view satellite ID numbers, elevation, azimuth, & SNR values
RMC	Time, date, position, course and speed data
VTG	Course and speed information relative to the ground
ZDA	PPS timing message (synchronized to PPS)
154	Extended Ephemeris ACK
168	Watchdog Timeout and Exception Condition

A full description of the listed NMEA messages is provided in the following sections.

2.2.1 Message ID GGA: Global Positioning System Fixed Data

Table 2-3 contains the values for the following example:

```
$GPGGA,091925.000,3113.3166,N,12121.2682,E,1,09,0.9,36.9,M,7.9,M,0000*56<CR><LF>
```

Table 2-3: GGA Data Format

Name	Example	Unit	Description
Message ID	\$GPGGA		GGA protocol header
LTC Time	091925.000		hhmmss.ss
Latitude	3113.3166		ddmm.mmmmm
N/S Indicator	N		N=North or S=south
Longitude	12121.2682		dddmm.mmmmm
E/W Indicator	E		E=East or W=West
Position Fix Indicator	1		See Table 2-4
Satellites Used	09		Range 0 to 12
HDOP	0.9		Horizontal Dilution of Precision
MSL Altitude	36.9	meters	
Units	M	meters	
Geoid Separation	7.9	meters	Geoid-to-ellipsoid separation. Ellipsoid altitude = MSL Altitude + Geoid Separation.
Units	M	meters	
Age of Diff. Corr.		sec	Null fields when DGPS is not used
Diff. Ref. Station ID	0000		
Checksum	*56		
<CR><LF>			End of message termination

Table 2-4: Position Fix Indicator

Value	Description
0	Fix not available or invalid
1	GPS SPS Mode, fix valid
2	Differential GPS, SPS Mode, fix valid
3-5	Not supported
6	Dead Reckoning Mode, fix valid

Note: A valid status is derived from all the parameters set in the software. This includes the minimum number of satellites required, any DOP mask setting, presence of DGPS corrections, etc. If the default or current software setting requires that a factor is met, then if that factor is not met, the solution will be marked as invalid.

Datasheet

1313 - Micro Load Cell (0-5kg) - CZL635



Contents

- 1 What do you have to know?
- 1 How does it work - For curious people
- 1 Installation
- 2 Calibration
- 2 Product Specifications
- 3 Glossary

What do you have to know?

A load cell is a force sensing module - a carefully designed metal structure, with small elements called strain gauges mounted in precise locations on the structure. Load cells are designed to measure a specific force, and ignore other forces being applied. The electrical signal output by the load cell is very small and requires specialized amplification. Fortunately, the 1046 PhidgetBridge will perform all the amplification and measurement of the electrical output.

Load cells are designed to measure force in one direction. They will often measure force in other directions, but the sensor sensitivity will be different, since parts of the load cell operating under compression are now in tension, and vice versa.

How does it work - For curious people

Strain-gauge load cells convert the load acting on them into electrical signals. The measuring is done with very small resistor patterns called strain gauges - effectively small, flexible circuit boards. The gauges are bonded onto a beam or structural member that deforms when weight is applied, in turn deforming the strain-gauge. As the strain gauge is deformed, its electrical resistance changes in proportion to the load.

The changes to the circuit caused by force is much smaller than the changes caused by variation in temperature. Higher quality load cells cancel out the effects of temperature using two techniques. By matching the expansion rate of the strain gauge to the expansion rate of the metal it's mounted on, undue strain on the gauges can be avoided as the load cell warms up and cools down. The most important method of temperature compensation involves using multiple strain gauges, which all respond to the change in temperature with the same change in resistance. Some load cell designs use gauges which are never subjected to any force, but only serve to counterbalance the temperature effects on the gauges that measuring force. Most designs use 4 strain gauges, some in compression, some under tension, which maximizes the sensitivity of the load cell, and automatically cancels the effect of temperature.

Installation

This Single Point Load Cell is used in small jewelry scales and kitchen scales. It's mounted by bolting down the end of the load cell where the wires are attached, and applying force on the other end **in the direction of the arrow**. Where the force is applied is not critical, as this load cell measures a shearing effect on the beam, not the bending of the beam. If you mount a small platform on the load cell, as would be done in a small scale, this load cell provides accurate readings regardless of the position of the load on the platform.



Calibration

You can use this simple formula to convert the measured mV/V output from the load cell to the measured force:

$$\text{Expected Force/Weight} = K * (\text{Measured mV/V} - \text{Offset})$$

Where K is gain value that will change depending on what unit of force or weight you want to measure. Since the offset varies between individual load cells, it's necessary to measure it for each sensor. Record the output of the load cell at rest on a flat surface with no force on it. The mV/V output measured by the PhidgetBridge is the offset.

Once you've found the offset, measure something with a known weight and solve the equation for K. You can also calibrate the load cell at multiple known weights and use these points to model a linear function.

Product Specifications	
Mechanical	
Housing Material	Aluminum Alloy
Load Cell Type	Strain Gauge
Capacity	5kg
Dimensions	55.25x12.7x12.7mm
Mounting Holes	M5 (Screw Size)
Cable Length	550mm
Cable Size	30 AWG (0.2mm)
Cable - no. of leads	4
Electrical	
Precision	0.05%
Rated Output	1.0±0.15 mV/V
Non-Linearity	0.05% FS
Hysteresis	0.05% FS
Non-Repeatability	0.05% FS
Creep (per 30 minutes)	0.1% FS
Temperature Effect on Zero (per 10°C)	0.05% FS
Temperature Effect on Span (per 10°C)	0.05% FS
Zero Balance	±1.5% FS
Input Impedance	1130±10 Ohm
Output Impedance	1000±10 Ohm
Insulation Resistance (Under 50VDC)	≥5000 MOhm
Excitation Voltage	5 VDC
Compensated Temperature Range	-10 to ~+40°C
Operating Temperature Range	-20 to ~+55°C
Safe Overload	120% Capacity
Ultimate Overload	150% Capacity

Glossary

Capacity

The maximum load the load cell is designed to measure within its specifications.

Creep

The change in sensor output occurring over 30 minutes, while under load at or near capacity and with all environmental conditions and other variables remaining constant.

FULL SCALE or FS

Used to qualify error - FULL SCALE is the change in output when the sensor is fully loaded. If a particular error (for example, Non-Linearity) is expressed as 0.1% F.S., and the output is 1.0mV/V, the maximum non-linearity that will be seen over the operating range of the sensor will be 0.001 mV/V. An important distinction is that this error doesn't have to only occur at the maximum load. If you are operating the sensor at a maximum of 10% of capacity, for this example, the non-linearity would still be 0.001mV/V, or 1% of the operating range that you are actually using.

Hysteresis

If a force equal to 50% of capacity is applied to a load cell which has been at no load, a given output will be measured. The same load cell is at full capacity, and some of the force is removed, resulting in the load cell operating at 50% capacity. The difference in output between the two test scenarios is called hysteresis.

Excitation Voltage

Specifies the voltage that can be applied to the power/ground terminals on the load cell. In practice, if you are using the load cell with the PhidgetBridge, you don't have to worry about this spec.

Input Impedance

Determines the power that will be consumed by the load cell. The lower this number is, the more current will be required, and the more heating will occur when the load cell is powered. In very noisy environments, a lower input impedance will reduce the effect of Electromagnetic interference on long wires between the load cell and PhidgetBridge.

Insulation Resistance

The electrical resistance measured between the metal structure of the load cell, and the wiring. The practical result of this is the metal structure of the load cells should not be energized with a voltage, particularly higher voltages, as it can arc into the PhidgetBridge. Commonly the load cell and the metal framework it is part of will be grounded to earth or to your system ground.

Maximum Overload

The maximum load which can be applied without producing a structural failure.

Non-Linearity

Ideally, the output of the sensor will be perfectly linear, and a simple 2-point calibration will exactly describe the behaviour of the sensor at other loads. In practice, the sensor is not perfect, and Non-linearity describes the maximum deviation from the linear curve. Theoretically, if a more complex calibration is used, some of the non-linearity can be calibrated out, but this will require a very high accuracy calibration with multiple points.

Non-Repeatability

The maximum difference the sensor will report when exactly the same weight is applied, at the same temperature, over multiple test runs.

Operating Temperature

The extremes of ambient temperature within which the load cell will operate without permanent adverse change to any of its performance characteristics.

Output Impedance

Roughly corresponds to the input impedance. If the Output Impedance is very high, measuring the bridge will distort the results. The PhidgetBridge carefully buffers the signals coming from the load cell, so in practice this is not a concern.

Rated Output

Is the difference in the output of the sensor between when it is fully loaded to its rated capacity, and when it's unloaded. Effectively, it's how sensitive the sensor is, and corresponds to the gain calculated when calibrating the sensor. More expensive sensors have an exact rated output based on an individual calibration done at the factory.



Safe Overload

The maximum axial load which can be applied without producing a permanent shift in performance characteristics beyond those specified.

Compensated Temperature

The range of temperature over which the load cell is compensated to maintain output and zero balance within specified limits.

Temperature Effect on Span

Span is also called rated output. This value is the change in output due to a change in ambient temperature. It is measured over 10 degree C temperature interval.

Temperature Effect on Zero

The change in zero balance due to a change in ambient temperature. This value is measured over 10 degree C temperature interval.

Zero Balance

Zero Balance defines the maximum difference between the +/- output wires when no load is applied. Realistically, each sensor will be individually calibrated, at least for the output when no load is applied. Zero Balance is more of a concern if the load cell is being interfaced to an amplification circuit - the PhidgetBridge can easily handle enormous differences between +/- . If the difference is very large, the PhidgetBridge will not be able to use the higher Gain settings.



INSTRUMENTATION AMPLIFIER With Precision Voltage Reference

FEATURES

- LOW QUIESCENT CURRENT: 460µA
- PRECISION VOLTAGE REFERENCE:
1.24V, 2.5V, 5V or 10V
- SLEEP MODE
- LOW OFFSET VOLTAGE: 250µV max
- LOW OFFSET DRIFT: 2µV/°C max
- LOW INPUT BIAS CURRENT: 20nA max
- HIGH CMR: 100dB min
- LOW NOISE: 38nV/√Hz at f = 1kHz
- INPUT PROTECTION TO ±40V
- WIDE SUPPLY RANGE
Single Supply: 2.7V to 36V
Dual Supply: ±1.35V to ±18V
- 16-PIN DIP AND SO-16 SOIC PACKAGES

DESCRIPTION

The INA125 is a low power, high accuracy instrumentation amplifier with a precision voltage reference. It provides complete bridge excitation and precision differential-input amplification on a single integrated circuit.

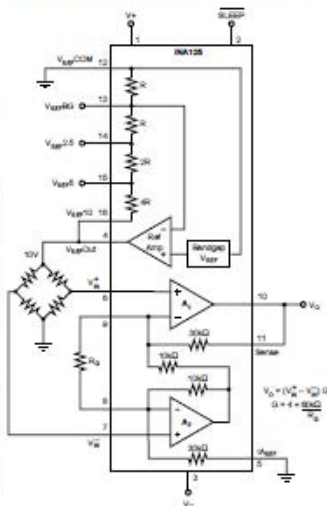
A single external resistor sets any gain from 4 to 10,000. The INA125 is laser-trimmed for low offset voltage (250µV), low offset drift (2µV/°C), and high common-mode rejection (100dB at G = 100). It operates on single (+2.7V to +36V) or dual (±1.35V to ±18V) supplies.

The voltage reference is externally adjustable with pin-selectable voltages of 2.5V, 5V, or 10V, allowing use with a variety of transducers. The reference voltage is accurate to ±0.5% (max) with ±35ppm/°C drift (max). Sleep mode allows shutdown and duty cycle operation to save power.

The INA125 is available in 16-pin plastic DIP and SO-16 surface-mount packages and is specified for the -40°C to +85°C industrial temperature range.

APPLICATIONS

- PRESSURE AND TEMPERATURE BRIDGE AMPLIFIERS
- INDUSTRIAL PROCESS CONTROL
- FACTORY AUTOMATION
- MULTI-CHANNEL DATA ACQUISITION
- BATTERY OPERATED SYSTEMS
- GENERAL PURPOSE INSTRUMENTATION



SPECIFICATIONS: $V_S = \pm 15V$

At $T_A = +25^\circ C$, $V_S = \pm 15V$, I_A common = $0V$, V_{REF} common = $0V$, and $R_L = 10k\Omega$, unless otherwise noted.

PARAMETER	CONDITIONS	INA129P, U			INA129PA, UA			UNITS
		MIN	TYP	MAX	MIN	TYP	MAX	
INPUT								
Offset Voltage, RTI								
Initial	$V_S = \pm 1.25V$ to $\pm 15V$, $G = 4$		± 50	± 250		*	± 500	μV
vs Temperature			± 0.25	± 2		*	± 15	$\mu V/^\circ C$
vs Power Supply			± 2	± 20		*	± 150	$\mu V/V$
Long-Term Stability				± 0.2		*		$\mu V/mo$
Impedance, Differential				$10^{11} \parallel 2$		*		$\Omega \parallel pF$
Common-Mode			$10^{11} \parallel 9$		*		$\Omega \parallel pF$	
Safe Input Voltage				± 40		*		V
Input Voltage Range	$V_{CM} = -10.7V$ to $+10.2V$		See Test			*		V
Common-Mode Rejection	$G = 4$	75	84		72	*		dB
	$G = 10$	86	94		80	*		dB
	$G = 100$	100	114		90	*		dB
	$G = 500$	100	114		90	*		dB
BIAS CURRENT	$V_{CM} = 0V$			10	25	*	50	nA
vs Temperature				± 50		*		$\mu A/^\circ C$
Offset Current				± 0.5	± 2.5	*	± 5	nA
vs Temperature				± 0.5		*		$\mu A/^\circ C$
NOISE, RTI	$R_F = 0\Omega$							
Voltage Noise, $f = 10Hz$			40			*		nV/\sqrt{Hz}
$f = 100Hz$			38			*		nV/\sqrt{Hz}
$f = 1kHz$			36			*		nV/\sqrt{Hz}
$f = 0.1Hz$ to $10Hz$			0.8			*		$\mu V/p-p$
Current Noise, $f = 10Hz$			170			*		fA/\sqrt{Hz}
$f = 1kHz$			56			*		fA/\sqrt{Hz}
$f = 0.1Hz$ to $10Hz$			5			*		$pA/p-p$
GAIN								
Gain Equation			$4 + 60k/R_F$	10,000	*	*	*	V/V
Range of Gain	$V_{IO} = -14V$ to $+13.3V$	4				*	*	V/V
Gain Error	$G = 4$		± 0.01	± 0.075		*	± 0.1	%
	$G = 10$		± 0.03	± 0.3		*	± 0.5	%
	$G = 100$		± 0.05	± 0.5		*	± 1	%
	$G = 500$		± 0.1			*		%
Gain vs Temperature	$G = 4$		11	± 15		*	*	ppm/°C
	$G = 4^{1/2}$		125	± 100		*	*	ppm/°C
Nonlinearity	$V_{IO} = -14V$ to $+13.3V$					*		% of FS
	$G = 4$		± 0.004	± 0.003		*	± 0.004	% of FS
	$G = 10$		± 0.004	± 0.002		*	± 0.004	% of FS
	$G = 100$		± 0.001	± 0.01		*	*	% of FS
	$G = 500$		± 0.002			*	*	% of FS
OUTPUT								
Voltage, Positive			$(V_{IO}) - 1.7$		*	*	*	V
Negative			$(V_{IO}) + 1$		*	*	*	V
Load Capacitance Stability			1000		*	*	*	pF
Short-Circuit Current			-20 ± 12		*	*	*	mA
VOLTAGE REFERENCE	$V_{REF} = +2.5V, +5V, +10V$							
Accuracy	$I_L = 0$		± 0.15	± 0.5		*	± 1	%
vs Temperature	$I_L = 0$		± 15	± 25		*	± 100	ppm/°C
vs Power Supply, V_{IO}	$V_{IO} = (V_{REF} + 1.25V) + 430V$		± 20	± 50		*	± 100	ppm/V
vs Load	$I_L = 0$ to $5mA$		3	75		*	*	ppm/A
Dropout Voltage, $(V_{IO}) - V_{REF(1)}$	Ref Load = $2k\Omega$	1.25	1		*	*		V
Bandgap Voltage Reference			1.24			*		V
Accuracy	$I_L = 0$		± 0.5			*		%
vs Temperature	$I_L = 0$		± 15			*		ppm/°C

The information provided herein is believed to be reliable; however, BURR-BROWN assumes no responsibility for inaccuracies or omissions. BURR-BROWN assumes no responsibility for the use of this information, and all use of such information shall be entirely at the user's own risk. Prices and specifications are subject to change without notice. No patent rights or licenses to any of the circuits described herein are implied or granted to any third party. BURR-BROWN does not authorize or warrant any BURR-BROWN product for use in life support devices and/or systems.

BURR-BROWN



INA125

SPECIFICATIONS: $V_S = \pm 15V$ (CONT)

At $T_A = +25^\circ C$, $V_S = \pm 15V$, IA common = 0V, V_{REF} common = 0V, and $R_L = 10k\Omega$, unless otherwise noted.

PARAMETER CONDITIONS		INA125P, U			INA125PA, UA			UNITS	
		MIN	TYP	MAX	MIN	TYP	MAX		
FREQUENCY RESPONSE	Bandwidth, -3dB	G = 4	150		*			kHz	
		G = 10	45		*			kHz	
		G = 100	4.5		*			kHz	
		G = 500	0.9		*			kHz	
		G = 4, 10V/Step	0.2		*			V/μs	
Slew Rate	G = 4, 10V/Step	80		*			μs		
	G = 10, 10V/Step	83		*			μs		
Settling Time, 0.01%	G = 100, 10V/Step	375		*			μs		
	G = 500, 10V/Step	1700		*			μs		
Overload Recovery	50% Overdrive	5		*			μs		
POWER SUPPLY	Specified Operating Voltage		±1.35	±1.0	*	*	*	V	
		Quiescent Current, Positive	$I_Q = I_{REF} = 0mA$	460	525	*	*	*	μA
			$I_Q = I_{REF} = 0mA$	-330	-325	*	*	*	μA
		Reference Ground Current ⁽²⁾		100		*	*	*	μA
			Sleep Current ($V_{SLEEP} < 100mV$)	$R_L = 10k\Omega$, Ref Load = 2kΩ	±1	±25	*	*	*
SLEEP MODE PIN ⁽⁴⁾	V_{IH} (Logic High Input Voltage)		+2.7	V^+	*	*	*	V	
		V_{IL} (Logic Low Input Voltage)	0	+0.1	*	*	*	V	
		I_{IH} (Logic High Input Current)		15		*	*	μA	
		I_{IL} (Logic Low Input Current)		0		*	*	μA	
		Wake-up Time ⁽⁵⁾		150		*	*	μs	
TEMPERATURE RANGE	Specification Range		-40	+65	*	*	*	°C	
		Operation Range	-55	+125	*	*	*	°C	
		Storage Range	-65	+125	*	*	*	°C	
		Thermal Resistance, θ_{JA}		80		*	*	°C/W	
		16-Pin DIP		100		*	*	°C/W	
SO-16 Surface-Mount									

* Specification same as INA125P, U.

NOTES: (1) Temperature coefficient of the "internal Resistor" in the gain equation. Does not include TCR of gain-setting resistor, R_G . (2) Dropout voltage is the positive supply voltage minus the reference voltage that produces a 1% decrease in reference voltage. (3) $V_{REF} < 0.01V$ pin. (4) Voltage measured with respect to Reference Common. Logic low input selects Sleep mode. (5) IA and Reference, see Typical Performance Curves.

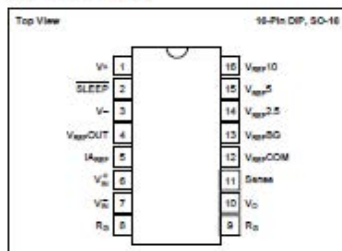
SPECIFICATIONS: $V_S = +5V$

At $T_A = +25^\circ C$, $V_S = +5V$, IA common at $V_{P/2}$, V_{REF} common = $V_{P/2}$, $V_{CM} = V_{P/2}$, and $R_L = 10k\Omega$ to $V_{P/2}$, unless otherwise noted.

PARAMETER	CONDITIONS	INA125P, U			INA125PA, UA			UNITS	
		MIN	TYP	MAX	MIN	TYP	MAX		
INPUT	Offset Voltage, RTI	Initial	±75	±500		*	±750	μV	
		vs Temperature	±2.7V to +36V	10.25		*	*	μV/°C	
		vs Power Supply		3	20	*	50	μV/V	
		Input Voltage Range	Common-Mode Rejection	$V_{CM} = +1.1V$ to +3.6V	See Test		*	*	
				G = 4	78	84	72	*	*
	G = 10	85	94	80	*	*	dB		
	G = 100	100	114	90	*	*	dB		
	G = 500	100	114	90	*	*	dB		
GAIN	Gain Error	$V_{IO} = +0.3V$ to +3.0V				*	*	%	
		G = 4		±0.01		*	*		
OUTPUT	Voltage, Positive		(V ⁺)-1.2	(V ⁺)-0.8	*	*	V		
		Negative	(V ⁻)+0.3	(V ⁻)+0.15	*	*	V		
POWER SUPPLY	Specified Operating Voltage		+2.7	+5	*	*	*	V	
		Operating Voltage Range		460	525	*	*	*	μA
			Quiescent Current	$I_Q = I_{REF} = 0mA$		±1	±25	*	*
		Sleep Current ($V_{SLEEP} < 100mV$)	$R_L = 10k\Omega$, Ref Load = 2kΩ						

* Specification same as INA125P, U.

PIN CONFIGURATION



ELECTROSTATIC DISCHARGE SENSITIVITY

This integrated circuit can be damaged by ESD. Burr-Brown recommends that all integrated circuits be handled with appropriate precautions. Failure to observe proper handling and installation procedures can cause damage.

ESD damage can range from subtle performance degradation to complete device failure. Precision integrated circuits may be more susceptible to damage because very small parametric changes could cause the device not to meet its published specifications.

ABSOLUTE MAXIMUM RATINGS⁽¹⁾

Power Supply Voltage, V ⁺ to V ⁻	30V
Input Signal Voltage	±40V
Output Short Circuit	Continuous
Operating Temperature	-55°C to +125°C
Storage Temperature	-55°C to +125°C
Lead Temperature (soldering, 10s)	+300°C

NOTE: Stresses above these ratings may cause permanent damage.

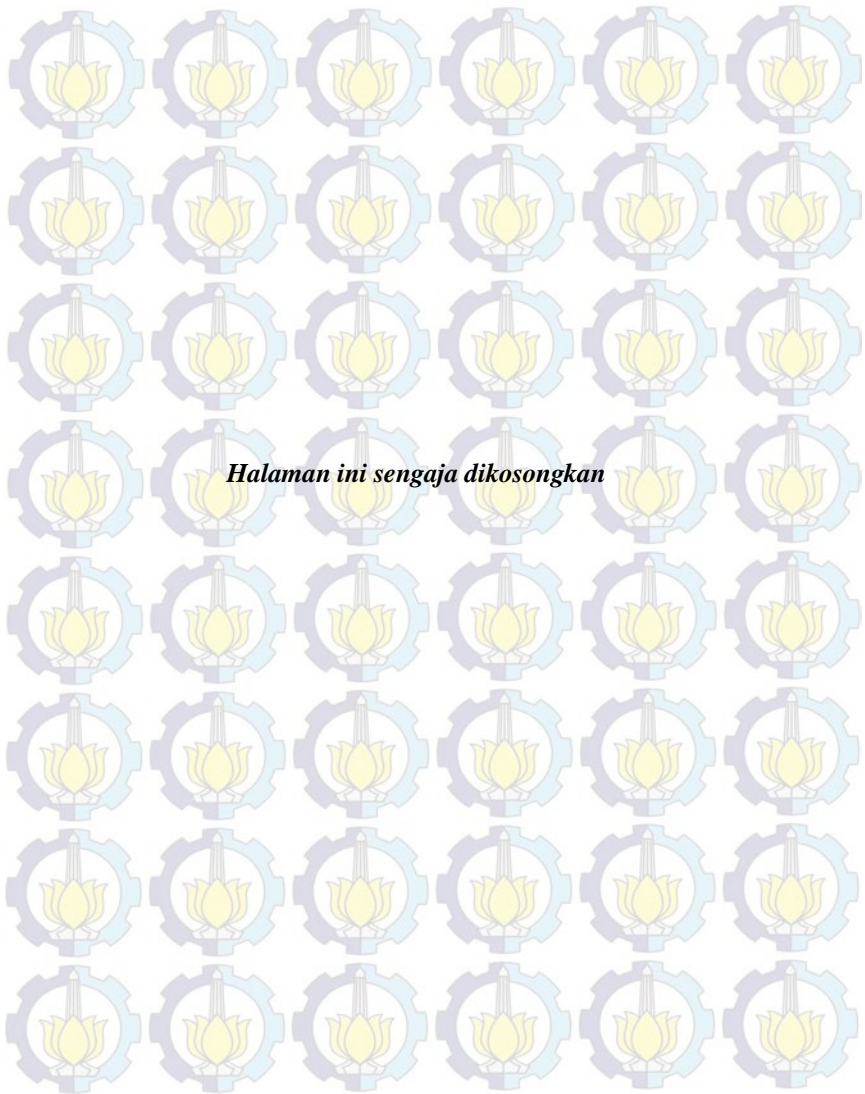
PACKAGE INFORMATION

PRODUCT	PACKAGE	PACKAGE DRAWING NUMBER ⁽¹⁾
INA125PA	16-Pin Plastic DIP	190
INA125P	16-Pin Plastic DIP	190
INA125UA	SO-16 Surface-Mount	205
INA125U	SO-16 Surface-Mount	205

NOTES: (1) For detailed drawing and dimension table, please see end of data sheet, or Appendix C of Burr-Brown IC Data Book.



INA125



RIWAYAT HIDUP PENULIS



Nama : Muhammad Reza
Adzani
TTL : Surabaya, 24 April
1993
Jenis Kelamin : Laki-laki
Agama : Islam
Alamat Rumah : Pulosari 3D/2,
Surabaya
Telp/HP : 085749860272
E-mail : adzani.reza@gmail.com
Hobi : Membaca Buku

RIWAYAT PENDIDIKAN

- 1999 – 2005 : SDN Sawunggaling VII Surabaya
- 2005 – 2008 : SMP Negeri 06 Surabaya
- 2008 – 2011 : SMA Negeri 02 Surabaya
- 2011 – sekarang : Bidang Studi Komputer Kontrol, Program D3
Teknik Elektro, ITS

PENGALAMAN KERJA

Kerja Praktek di PT. Angkasa Pura I (PERSERO) (Januari–Februari
2014).



RIWAYAT HIDUP PENULIS



Nama : Uinsa Pradana
TTL : Surabaya, 02
Desember 1991
Jenis Kelamin : Laki-laki
Agama : Islam
Alamat Rumah : Jl. Wonokitri 7 No.8,
Surabaya
Telp/HP : 08993377988
E-mail : uinsa.8@gmail.com
Hobi : Membaca, olahraga,
jalan-jalan.

RIWAYAT PENDIDIKAN

- 1998 – 2004 : SDN Sawunggaling 8
- 2004 – 2007 : SMPN 16 Surabaya
- 2007 – 2010 : SMAN 13 Surabaya
- 2011 – sekarang : Bidang Studi Komputer Kontrol, Program D3
Teknik Elektro, FTI-ITS

PENGALAMAN KERJA

- Kerja Praktek di PT. PJB UPHT, Gresik (Juli - Agustus 2013)

