

BAB IV PENGUJIAN DAN ANALISIS SISTEM

4.1 HASIL RANCANG BANGUN

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



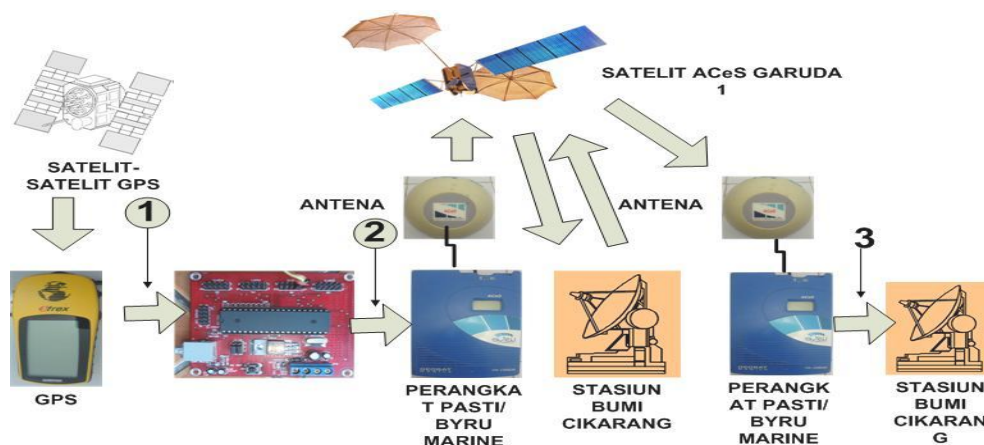
Gambar 4.1. Perangkat keras sistem

4.2 PENGUJIAN SISTEM

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

1. Menampilkan data keluaran GPS (NMEA-0183)
2. Menampilkan data keluaran GPS berdasar kecepatan yang disetting dari *hyperterminal*.
3. Menampilkan data keluaran dari perangkat PASTI disisi receiver:

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



Gambar 4.2. Blok diagram tempat pengujian sistem

4.2.1 Menampilkan Data Keluaran GPS

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

```

$GPGGA,06.21,9896,S,10649.2145,E,043224,A,A=5A
$GPBOD,T,M,*47
$GPRME,3.0,M,8.2,M,8.7,M=28
$GPRMZ,314,F,3=1D
$GPRTE,1.1,c,*37
$GPRMC,043226,A,0621.9896,S,10649.2145,E,0.0,81.9,090608,0.6,E,A=35
$GPRMB,A,V,A=1C
$GPGGA,043226,0621.9896,S,10649.2145,E,1.12,1.1,95.7,M,7.3,M,*51
$GPGSA,A,3,01,05,06,09,12,14,18,21,22,24,30,31,2.4,1.1,2.2*34
$GPGSV,3,1,12,01,11,300,00,05,42,077,35,06,03,261,00,09,05,026,00*77
$GPGSV,3,2,12,12,26,069,35,14,15,324,28,18,40,029,35,21,73,200,39*74
$GPGSV,3,3,12,22,20,350,00,24,36,153,29,30,54,109,41,31,42,243,29*74
$GPGLL,0621.9896,S,10649.2145,E,043226,A,A=58
$GPBOD,T,M,*47
$GPRME,3.0,M,8.1,M,8.7,M=2B
$GPRMZ,314,F,3=1D
$GPRTE,1.1,c,*37
$GPRMC,043228,A,0621.9895,S,10649.2145,E,0.0,81.9,090608,0.6,E,A=38
$GPRMB,A,V,A=1C
$GPGGA,043228,0621.9895,S,10649.2145,E,1.12,1.1,95.7,M,7.3,M,*5C
$GPGSA,A,3,01,05,06,09,12,14,18,21,22,24,30,31,2.4,1.1,2.2*34
$GPGSV,3,1,12,01,11,300,00,05,42,077,35,06,03,261,00,09,05,026,00*77
$GPGSV,3,2,12,12,26,069,36,14,15,324,28,18,40,029,35,21,73,200,39*77
$GPGSV,3,3,12,22,21,350,00,24,36,
  
```

Gambar 4.3. Data keluaran GPS yang ditampilkan pada *hyper terminal*

Format data GPS NMEA 0183 memiliki beberapa data berbeda. Terdapat dua belas pengiriman data sebelum kembali ke format data awal, yaitu \$GPRMC.....<CR><LF>. Siklus pengiriman data NMEA 0183 adalah 2 detik dihitung dari pengiriman data \$GPRMC awal sampai \$GPRMC berikutnya. Data

posisi berada pada deretan data jika GPS bebas dari halangan (GPS tidak dapat menerima sinyal dari satelit-satelit GPS). Gambar 4.4 memperlihatkan kondisi GPS tidak mendapatkan sinyal dari satelit-satelit GPS (*void*).

```

skripsi - HyperTerminal
File Edit View Call Transfer Help
$GPRMB,V,0.12,M,M,*04
$GPGGA,1.01,05.06,12.14,16.18,21,22,24,29,30,0.0,*19
$GPGSA,A,1,01,08,322,00,05,29,051,00,06,08,239,00,12,14,049,00*75
$GPGSV,3,2,12,14,12,345,00,16,16,209,00,18,54,061,00,21,50,175,31*7C
$GPGSV,3,3,12,22,39,001,00,24,15,150,00,29,19,133,00,30,49,068,00*75
$GPGRL,V,N*64
$GPBOD,T,M,*47
$PGRME,M,M,M*00
$PGRMZ,f,1*29
$GPRTE,1,1,c,*37
$GPRMC,V,0.150608,0.6,E,N*34
$GPRMB,V,0.12,M,M,*04
$GPGGA,1.01,05.06,12.14,16.18,21,22,24,29,30,0.0,*19
$GPGSA,A,1,01,08,322,00,05,29,051,00,06,08,239,00,12,14,049,00*75
$GPGSV,3,2,12,14,12,345,00,16,16,209,00,18,54,061,00,21,50,175,31*7C
$GPGSV,3,3,12,22,39,001,00,24,15,150,00,29,19,133,00,30,49,068,00*75
$GPGRL,V,N*64
$GPBOD,T,M,*47
$PGRME,M,M,M*00
$PGRMZ,f,1*29
$GPRTE,1,1,c,*37

```

Gambar 4.4. Data keluaran GPS pada kondisi tanpa sinyal (*void*)

4.2.2 Menampilkan Data Keluaran Kontroler

Pengujian ini bertujuan untuk mengambil data \$GPRMC dari format data GPS NMEA-0183 yang dikirimkan. Mikrokontroler akan melakukan *masking* terhadap data mentah GPS yang diterimanya. Hasilnya hanya data \$GPRMC yang dikirimkan. Gambar 4.5 memperlihatkan data GPS yang telah diolah pada mikrokontroler sehingga hanya data \$GPRMC yang dilewatkan.

```

,072342,A,0619.6069,S,10708.8055,E,0.0,191.1,241208,0.6,E,A*04
,072344,A,0619.6069,S,10708.8055,E,0.0,191.1,241208,0.6,E,A*02
,072346,A,0619.6069,S,10708.8055,E,0.0,191.1,241208,0.6,E,A*00
,072348,A,0619.6071,S,10708.8055,E,0.0,191.1,241208,0.6,E,A*07
,072350,A,0619.6071,S,10708.8055,E,0.0,191.1,241208,0.6,E,A*0E
,072352,A,0619.6071,S,10708.8055,E,0.0,191.1,241208,0.6,E,A*0C
,072354,A,0619.6073,S,10708.8056,E,0.0,191.1,241208,0.6,E,A*0B
,072356,A,0619.6073,S,10708.8055,E,0.0,191.1,241208,0.6,E,A*0A
,072358,A,0619.6074,S,10708.8055,E,0.0,191.1,241208,0.6,E,A*03

,072400,A,0619.6075,S,10708.8055,E,0.0,191.1,241208,0.6,E,A*08

,072402,A,0619.6075,S,10708.8055,E,0.0,191.1,241208,0.6,E,A*0A

,072404,A,0619.6075,S,10708.8055,E,0.0,191.1,241208,0.6,E,A*0C

,072406,A,0619.6074,S,10708.8055,E,0.0,191.1,241208,0.6,E,A*0F
,072408,A,0619.6074,S,10708.8055,E,0.0,191.1,241208,0.6,E,A*01
,072410,A,0619.6073,S,10708.8055,E,0.0,191.1,241208,0.6,E,A*0F
,072412,A,0619.6071,S,10708.8055,E,0.0,191.1,241208,0.6,E,A*0F
,072414,A,0619.6071,S,10708.8056,E,0.0,191.1,241208,0.6,E,A*0A

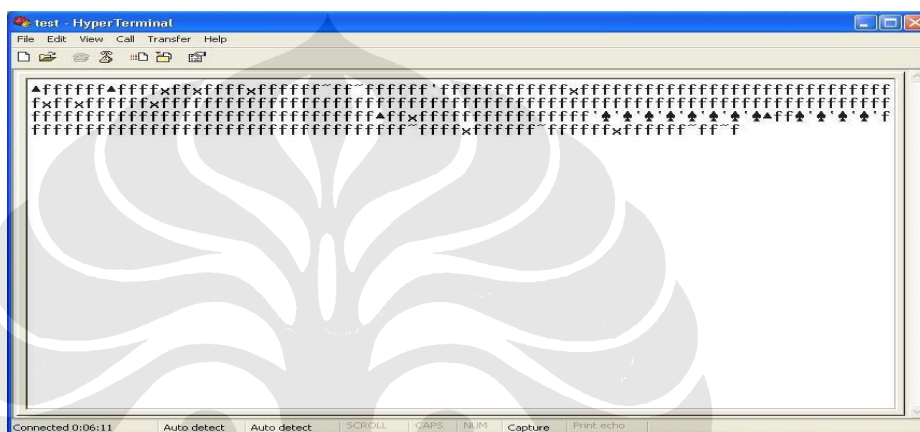
```

Gambar 4.5. Tampilan data dari kontroler pada *hyperterminal* dengan *masking* data \$GPRMC

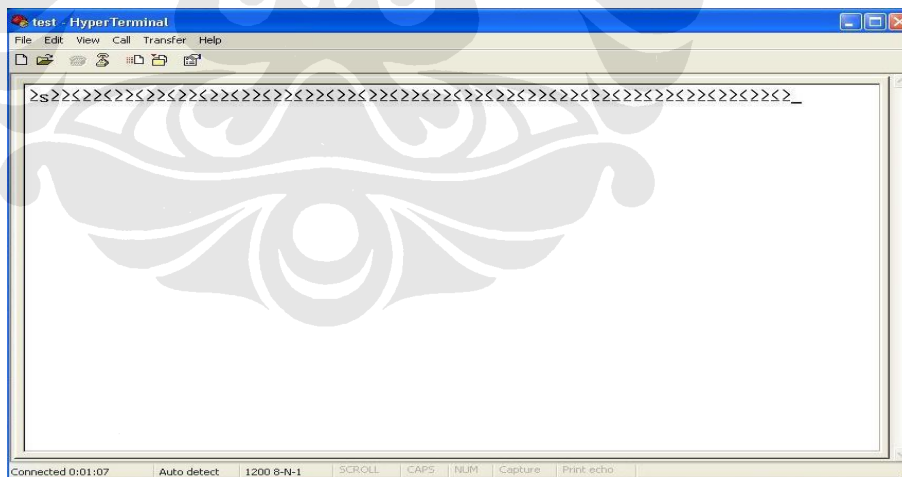
Pada baris pertama waktunya menunjukkan 072342 yang berarti jam 07, 23 menit 42 detik. Sedangkan pada baris kedua *hyper terminal* data waktunya menunjukkan 072344 yang berarti jam 07,23 menit 44 detik.

4.2.2.1 Data Keluaran dengan memberikan variasi kecepatan hyperterminal

Pengujian ini bertujuan untuk melihat data yang akan dikirimkan ke satelit Garuda 1 dengan menggunakan kecepatan pada hyperterminal dengan kecepatan yang digunakan adalah 4800 baud dan 1200 baud.



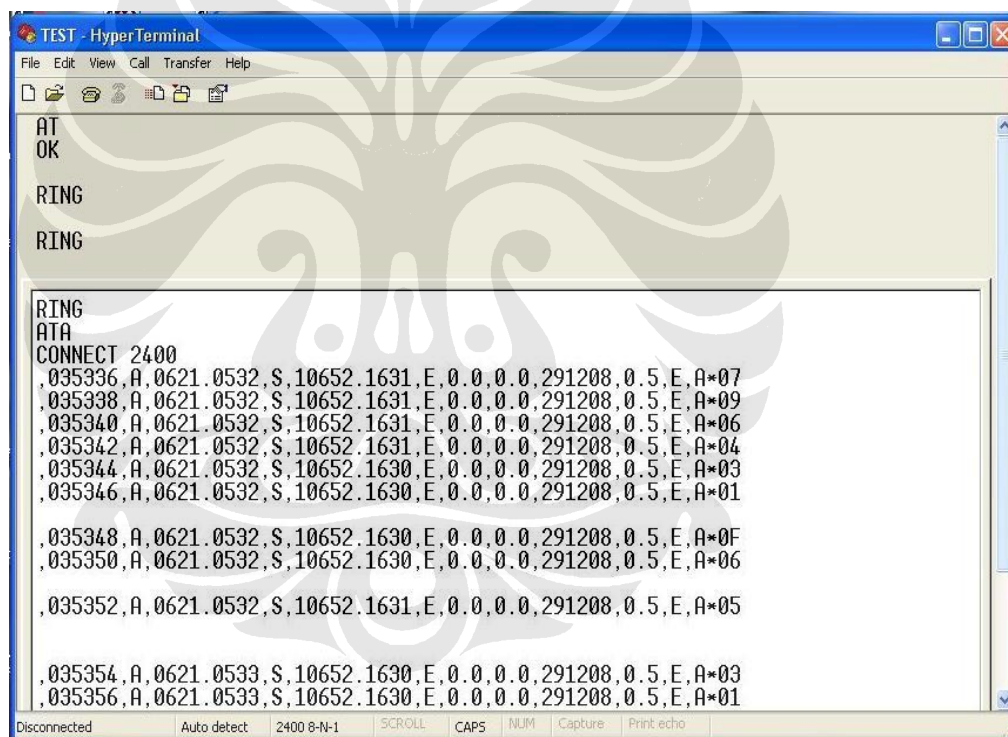
Gambar 4.6. Data keluaran dengan kecepatan 4800 baud



Gambar 4.6. Data keluaran dengan kecepatan 1200 baud

4.2.3 Menampilkan Data Final yang diterima PASTI/*User Terminal* sebagai sisi receiver

Pengujian ini dilakukan untuk melihat apakah data yang dikirim akan sampai tujuan sesuai dengan yang diinginkan dengan data GPS dalam bentuk \$GPRMC tidak berubah datanya. Dalam pengujian ini dilakukan beberapa langkah seperti gambar 4.7 yaitu dengan mengetik kata AT pada *hyperterminal*, kata OK adalah balasan bahwa perangkat PASTI/*user terminal* siap melakukan komunikasi yang dilanjutkan dengan kata RING tiga kali ini menyatakan sistem komunikasi yang dilakukan adalah antara perangkat PASTI/*user terminal*. Agar didapatkan datanya maka perlu mengetik kata ATA berarti perangkat siap menerima data. Terakhir akan didapatkan data \$GPRMC secara terus menerus.



```

TEST - HyperTerminal
File Edit View Call Transfer Help
[Icons]
AT
OK
RING
RING
RING
ATA
CONNECT 2400
,035336,A,0621.0532,S,10652.1631,E,0.0,0.0,291208,0.5,E,A*07
,035338,A,0621.0532,S,10652.1631,E,0.0,0.0,291208,0.5,E,A*09
,035340,A,0621.0532,S,10652.1631,E,0.0,0.0,291208,0.5,E,A*06
,035342,A,0621.0532,S,10652.1631,E,0.0,0.0,291208,0.5,E,A*04
,035344,A,0621.0532,S,10652.1630,E,0.0,0.0,291208,0.5,E,A*03
,035346,A,0621.0532,S,10652.1630,E,0.0,0.0,291208,0.5,E,A*01
,035348,A,0621.0532,S,10652.1630,E,0.0,0.0,291208,0.5,E,A*0F
,035350,A,0621.0532,S,10652.1630,E,0.0,0.0,291208,0.5,E,A*06
,035352,A,0621.0532,S,10652.1631,E,0.0,0.0,291208,0.5,E,A*05
,035354,A,0621.0533,S,10652.1630,E,0.0,0.0,291208,0.5,E,A*03
,035356,A,0621.0533,S,10652.1630,E,0.0,0.0,291208,0.5,E,A*01
Disconnected Auto detect 2400 8-N-1 SCROLL CAPS NUM Capture Print echo

```

Gambar 4.6. Tampilan data \$GPRMC pada PASTI/*user terminal* sisi receiver

4.3 ANALISIS SISTEM

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

4.3.1 Analisis Pengujian Pertama

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

4.3.2 Analisis Pengujian Kedua

Pada pengujian kedua yang terbagi dua pengujian dengan mencoba beberapa skenario.

1. Mikrokontroler dikondisikan untuk melakukan *masking* terhadap data mentah dari GPS. Data yang akan diambil dan dilewati hanya format \$GPRMC, yaitu data minimum yang direkomendasikan pada sistem navigasi GPS. Tujuannya adalah agar mikrokontroler dapat mengambil data \$GPRMC persiklus pengiriman data GPS. Untuk mengetahui besarnya waktu yang diperlukan untuk melakukan pengambilan dan pengiriman data pada mikrokontroler, dapat dengan menghitung total mesin *cycle* pada programnya. Besarnya mesin *cycle* dan waktu yang diperlukan dapat dilihat pada *lampiran 1*. Untuk memudahkan *masking*, program dibatasi dengan jangkauan antara karakter khusus data \$GPRMC (karakter C) dengan data akhir <LF>.

Lama waktu pengambilan data dari GPS oleh mikrokontroler dengan *masking* \$GPRMC adalah:

$$a. \text{ \$GPRMC terima} = (11,9350 + A + B)\mu\text{s}$$

A adalah banyaknya pengulangan sampai mikrokontroler mengenali karakter C pada data GPS.

$$A = (5 \times (\text{Inst.1} + \text{Inst.2})) + (\text{Inst.1} + \text{Inst.2} + \text{Inst.3})$$

$$A = (5 \times (9,7650)) + (11,9350)$$

$$A = 60,7600\mu\text{s}$$

B adalah banyaknya pengulangan sampai mikrokontroler memperoleh semua data \$GPRMC yaitu dengan ditandai dengan <LF>.

$$B = (63 \times (\text{Inst.1} + \text{Inst.2} + \text{Inst.5} + \text{Inst.6} + \text{Inst.7}) +$$

$$(Inst.1 + Inst.2 + Inst.3 + Inst.4)$$

$$B = 1038,3450\mu s$$

$$\$GPRMC \text{ terima} = (11,9350 + 60,7600 + 1038,3450)\mu s$$

$$\$GPRMC \text{ terima} = 1111,0400\mu s$$

b. $\$GPRMC \text{ kirim} = (26,0400 + C + 19,5300)\mu s$

C adalah banyaknya pengulangan sampai akhirnya mikrokontroler mengirimkan data terakhir \$GPRMC.

$$C = (62 \times (Inst.1 + Inst.2 + Inst.5 + Inst.6 + Inst.7) + (Inst.1 + Inst.2 + Inst.3 + Inst.4))$$

$$C = 1024,2400\mu s$$

$$\$GPRMC \text{ kirim} = (26,0600 + 1024,2400 + 19,5300)\mu s$$

$$\$GPRMC \text{ kirim} = 1069,8300\mu s$$

c. $\$GPRMC \text{ total} = \$GPRMC \text{ terima} + \$GPRMC \text{ kirim}$

$$\$GPRMC \text{ total} = (1111,0400 + 1069,8300)\mu s$$

$$\$GPRMC \text{ total} = 2180,8700\mu s$$

d. $Delay = \$GPRMC \text{ kirim} - \$GPRMC \text{ terima}$

$$Delay = (1069,8300 - 1111,0400)\mu s$$

$$Delay = -41,2100\mu s$$

Nilai *delay* yang menunjukkan hasil minus (-) berarti, waktu yang diperlukan untuk mengirimkan data yang telah disimpan lebih cepat 41,2100μs. Sedangkan untuk mengambil data \$GPRMC diperlukan waktu 2180,8700μs pada pengambilan pertama. Pada pengujian ini, mikrokontroler tetap mengambil data-data lain dari format NMEA-0183. Oleh karena itu waktu pengiriman data lainnya tetap diperhitungkan. Pada pengujian pertama, dapat diketahui besarnya waktu pengiriman data dari GPS, yaitu 0,1670 detik per data. Berarti ada sekitar sebelas data lagi yang dikirimkan oleh GPS ke mikrokontroler. Besarnya waktu untuk mengirimkan ke sebelas data lainnya adalah:

$$\text{Waktu total} = 11 \times 0,1670 \text{ detik} = 1,8370 \text{ detik}$$

Pada *lampiran I* dapat dilihat bahwa program melakukan pengulangan untuk sampai menemukan karakter khusus \$GPRMC. Berarti total besarnya waktu yang digunakan untuk mengambil data \$GPRMC berikutnya adalah:

\$GPRMC berikutnya = \$GPRMC total + 11,9350 μ s + X

Nilai X pada rumus di atas adalah waktu total dari 11 data yang dikirimkan GPS.

\$GPRMC berikutnya = 2180,8700 μ s + 11,9350 μ s + 1,8370s

\$GPRMC berikutnya = 2180,8700 μ s + 11,9350 μ s + 1837000 μ s

\$GPRMC berikutnya = 1839192,8050 μ s = 1,8392s

Ini memperlihatkan bahwa kecepatan mikrokontroler dalam mengambil dan mengirimkan data \$GPRMC tiap siklusnya mendekati besar siklus data GPS sebesar 2 detik, yaitu berbeda sebesar 0,1608s. Perbedaan ini bahkan lebih kecil dari nilai waktu pengiriman data GPS yang sebesar 0,1677 detik per datanya. Rentang waktu yang sedikit ini masih mempengaruhi pengiriman data dari mikrokontroler. Seperti yang terlihat pada gambar 4.6, walau dapat mengambil data persiklus GPS, rentang waktu yang masih ada menyebabkan mikrokontroler mengirimkan *header* ATDT tanpa data \$GPRMC.

2. Pengaturan kecepatan pada *hyperteminal* yaitu dengan kecepatan 4800 baud dengan maksud apakah data yang dikirim akan semakin cepat diterima dan 1200 baud apakah data yang diterima semakin lambat. Berdasarkan sistem yang dipakai yaitu perangkat PASTI dengan standar kecepatan 2400 baud sehingga jika diatur kecepatannya yang tidak sesuai maka akan didapatkan data yang berbeda.

4.3.3 Analisis Pengujian Ketiga

Pengujian ini dilakukan dengan maksud apakah data \$GPRMC yang dikirim dari perangkat PASTI/byru marine (*sisi transmitter*) sama dengan data \$GPRMC yang diterima perangkat PASTI/byru marine (*sisi receiver*). Berdasarkan Gambar 4.8 maka bisa diartikan untuk data posisi *latitude* seperti pada baris kedua tampilan *hyper terminal* adalah 06210532,S yang berarti *latitude* 06⁰ 21.0532'S atau lebih tepatnya data posisi GPS dapat dikonversi menjadi data dalam bentuk *Geographic Information Systems* (GIS) dalam dua langkah[22].

Langkah 1

Membagi koordinat dengan konstanta 100

$$\frac{0621.0532}{100} = 6.210532 = 6.210$$

Langkah 2

Mengalikan angka di belakang koma (,) dengan konstanta 10 dan membaginya dengan konstanta 6. Hasilnya ditambahkan dengan angka di depan koma pada hasil langkah 1.

$$\text{Data posisi peta GIS} = 6 + \frac{0.210 \times 10}{6} = 6.35LS$$

Karena kota Jakarta berada pada bagian selatan dari garis khatulistiwa maka di depan angka lintang selatan ditambahkan tanda minus (-) sehingga didapat -6.3665 LS.

Sedangkan data A di depan data *latitude* menunjukkan GPS dalam keadaan aktif atau menerima sinyal dari satelit-satelit GPS.

Untuk contoh data posisi *longitude* seperti pada baris pertama tampilan *hyper terminal* adalah 165021631,E yang berarti *longitude* 106° 02.1631'E. Apabila dikonversi ke posisi peta GIS dengan menggunakan langkah yang sama, didapatkan:

$$\frac{105021631}{100} = 165.021631 = 165.0216$$

$$\text{Data posisi peta GIS} = 165 + \frac{0.021631 \times 10}{6} = 165.21631BT$$

Jika dilihat pada gambar 4.8 terlihat terdapat jeda kosong walaupun selang waktu antara masing-masing data \$GPRMC tetap 2 sekon. Hal ini dikarenakan perbedaan waktu pengambilan dan pengiriman data \$GPRMC yang berbeda sedikit oleh mikrokontroler.