

Interfacing the Lassen SQ GPS Receiver to PALOS on MK-2 Node



Zhengming Fu
zhengming.fu@yale.edu


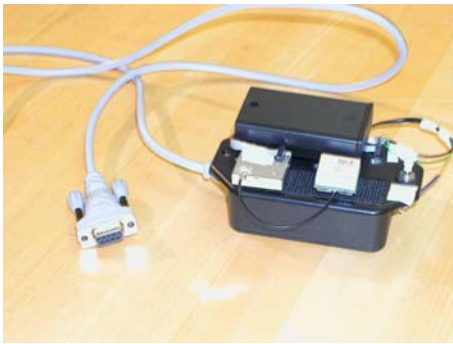


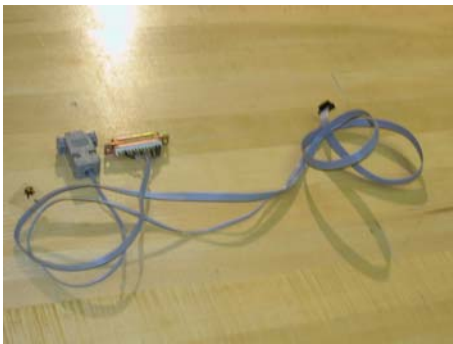
ENALAB
<http://www.eng.yale.edu/enalab>
Department of Electrical Engineering
Yale University

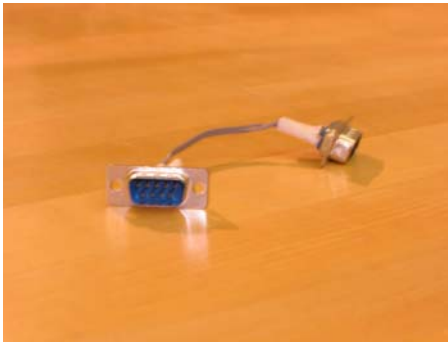
Version 0.1 Released on October 4, 2003

Trimble's Lassen™ SQ models add complete GPS functionality to your mobile embedded system. This document is mainly about the installation and implementation of Lassen SQ GPS module on MK-2 node.

Terminology

MK-2 Node	A wireless sensor node designed by UCLA. It has 2 micro-controllers onboard. AVR ATMEGA128L (AVR ISA) and ATMEGA FR9081 (ARM-Thumb ISA).	
GPS Receiver	Trimble Lassen GPS Signal Receiver. It accepts the GPS satellite signal from antenna and outputs TTL signals in the form of TSIP or NMEA0183 protocol. Development Manual: http://www.trimble.com/lassensq_ts.asp?Nav=Collection-19088	

<p>TTL- RS232 converter</p>	<p>A port converter, which reads TTL signals from GPS receiver and output corresponding RS-232 COM signals.</p>	
<p>GPS Node</p>	<p>A MK-2 node, which connects with GPS receiver. The PALOS programs running on it is compiled from the code under /palos-gps/test/GPS-MK2</p>	
<p>PC Node</p>	<p>Also called gateway node. It is a MK-2 node, which connects with PC. The palos programs running on it is compiled from the code under /palos-gps/test/GPS-MK2</p>	
<p>PC Interpreter:</p>	<p>It is a program, running on your PC, which reads the GPS raw data and translates it into readable GPS location data.</p>	
<p>MK-2 Parallel- Serial Cable</p>	<p>Also called AVR programming. It is a cable, one end of which connects with MK-2 with AVR connector. The other end provides both parallel and serial port connectors.</p>	

<p>Male-2-Male Adaptor</p>	<p>It bridges the TTL-RS232 converter serial cable and the MK-2 Parallel-Serial cable. It is different from the common Male-2-Male adaptor from market since it is cross-wired, i.e. the TXD of one port connects to RXD of the other, vise versa.</p>	
<p>ECEF Coordinate</p>	<p>Earth Centered Earth Fixed Coordinate System. We use meter as unit in our application</p> <ul style="list-style-type: none"> • The origin of the system is at the mass center of the earth. • The Z axis is along the axis of rotation. • The X axis passes through the intersection of the CTP's¹ equatorial plane and a reference meridian.² • The Y axis is defined in equatorial plane to complete a right hand 	

¹ CTP: The mean position of the spin-axis for the years 1900-1905.

² Prime Meridian: Meridian line is an imaginary line, which runs from the North Pole to the South Pole. The Prime Meridian line runs through “the primary transit” instrument (main telescope) at the Royal Observatory in Greenwich, UK.

	coordinate system.	
LLA Coordinate	<p>Latitude, Longitude and Altitude Coordinate System</p> <ul style="list-style-type: none"> • The origin of the system is at the mass center of the earth • The latitude is an angle from the equatorial plane to a line normal to the reference ellipsoid³ • The longitude is angle between a reference plane (Prime Meridian) and a plane passing through the point, both planes being perpendicular to the equatorial plane • The altitude is the height difference from the reference ellipsoid to the point in a direction that is normal to the ellipsoid 	

³ Reference ellipsoid: an imaginary perfect ellipsoid covering the earth. The shape of the geoid(i.e. mean sea level surface) varies around the globe, therefore difference sized ellipsoids have used for different region.

Serial Port Hardware Installation

The Lassen SQ GPS module uses a single 8-pin (2x4) male header connector for both power and data I/O. Since the output of Lassen GPS is TTL level serial port, you need an adaptor to convert it to RS-232 before it is connected with your laptop or MK-2 node.

On the MK-2 node, the GPS receiver connects to the UART driver pins on the MK-2 with AVR connector, if a UART driver is used. (MK-2 has 2 UART pins: one is for AVR microcontroller, the other is for ARM THURM microcontroller. We use the AVR one here.)

The PC or Laptop communicates with GPS model through COM1 or COM2 port.

In order to connect to PC or MK-2, you need the following hardware.

1. FFSD-04-D-05 from SAMTEC.com⁴
2. ASP 69533-01 from SAMTEC.com
3. AVR programming cable from UCLA
4. Male-2-Male Adaptor
5. TTL-RS232 converter (you can use one of the MICA serial port accessory boards)

⁴ You can reach the manufacturers by 1-800-SAMTEC-9 or <http://www.samtec.com/>.

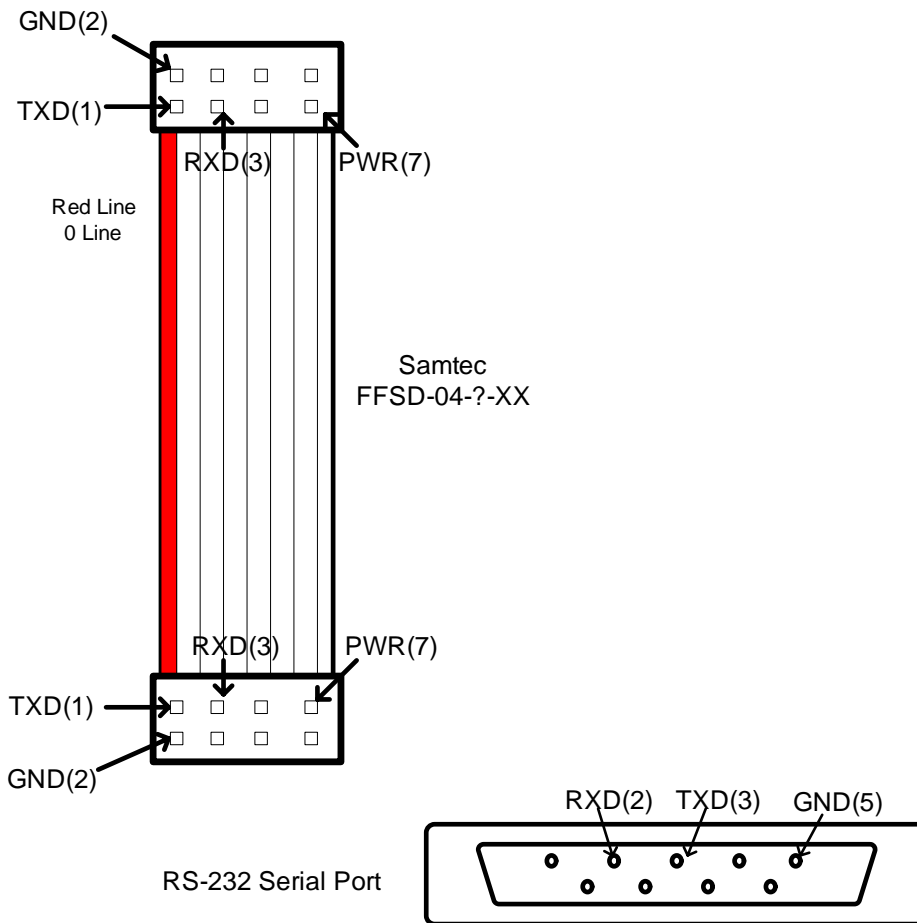
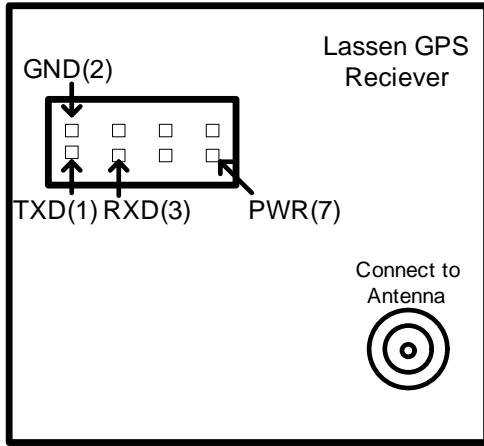


Figure 1 GPS Receiver Digital IO/Power Connector



Figure 2 FFSD connector mount on GPS Receiver

Figure 1 shows the different connectors, pins and their purposes. Besides the GPS receiver and antenna you also need 2 types of connectors from Samtec: FFSD-04-D-05

(http://www.samtec.com/technical_specifications/overview.asp?series=FFSD&menu=Search) and ASP 69533-01.

NOTE: FFSD-04-?-XX: ? could be S for single end and D double end; XX represents the overall length in inches, 1/8 inch, with 2 inch minimum.

Figure 2 show how the FFSD connector mounts onto the SQ GPS receiver. The receiver should face up.

Communication Protocol in Lassen SQ GPS Receiver

In our design, we use TSIP (Trimble Standard Interface Protocol) as GPS receiver and MK-2 or Laptop. Trimble SQ GPS offers 2 options in terms of GPS data output: TSIP and NMEA 0183. TSIP is a protocol defined by Trimble Co. It is also the default protocol configuration in the receiver. It is a binary language. The NMEA 0183 is an industrial standard protocol. It is an ASCII format. The entire NMEA protocol encompass over 50 messages, but Lassen SQ GPS receiver only supports a subset of them. (Lassen Manual lists 9 output messages).

We use TSIP since it is more powerful, scalable and flexible. In terms of output messages, The TSIP protocol offers over 20 commands and their associated response packets for use in configuration to meet user requirements.

In terms of coordinate system, the NMEA 0183 protocol only supports the LLA (Latitude, Longitude and Altitude) coordinate system, while TSIP can suppose both the LLA position, ENU velocity and ECEF position and velocity (Earth Centered

Earth Fixed) coordinate system. We consider TSIP is more powerful and scalable in Lassen SQ GPS Receiver System.

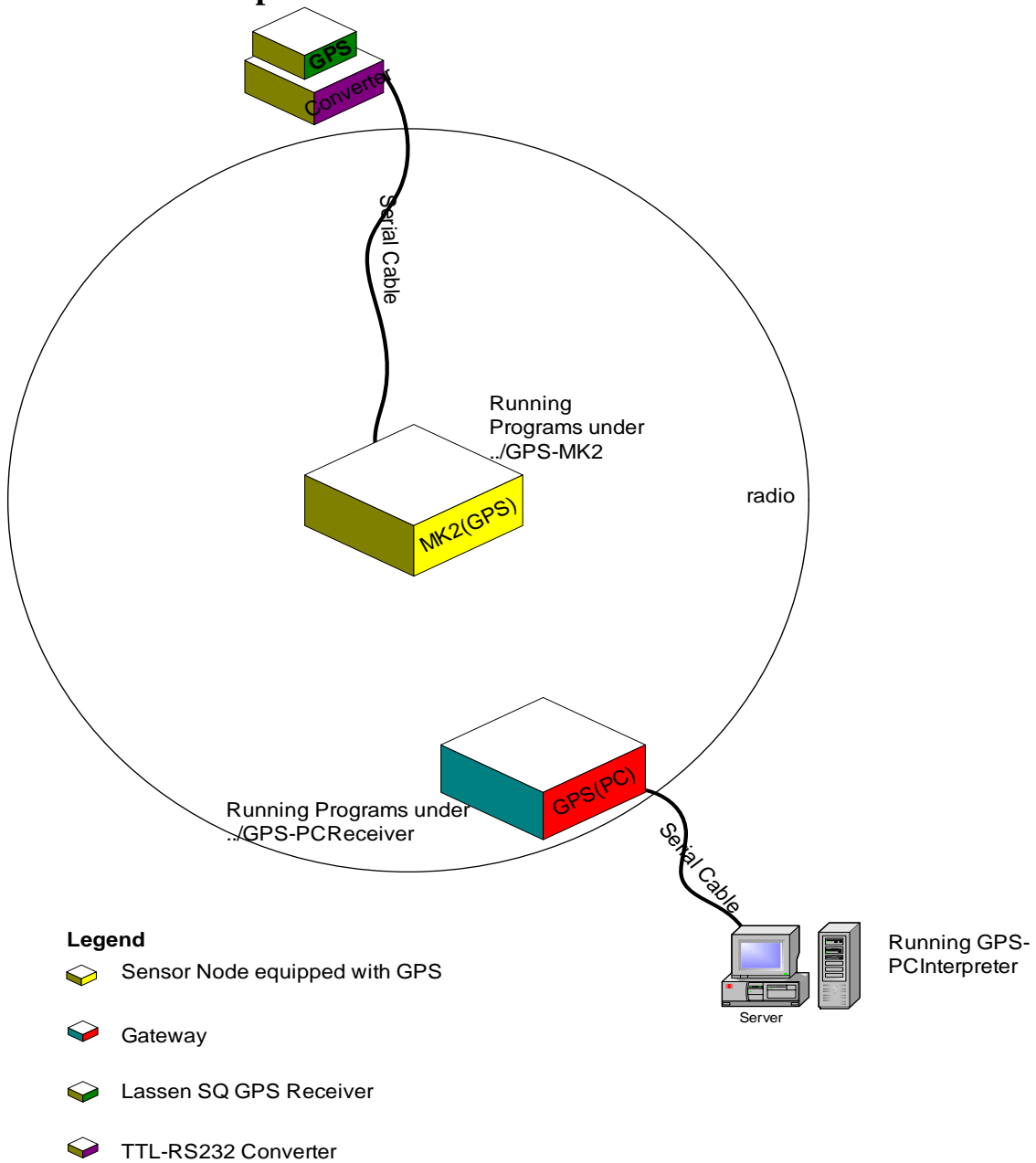
Serial Port Configuration

Lassen SQ GPS module has the single I/O port. The Table provides the default protocol and port configuration for the receiver, as delivered from factory. We continue to use this configuration in PALOS GPS module.

Table 1 Default Protocol and Port Configuration

Input		Output	
TSIP	Baud Rate: 9600 Data Bits: 8 Parity: Odd Stop Bits: 1 No Flow Control	TSIP	Baud Rate: 9600 Data Bits: 8 Parity: Odd Stop Bits: 1 No Flow Control

Hardware Set-up



Car 2

Figure 3 GPS Testbed Implementation

The GPS wireless systems are connected in the following way: Lassen GPS receiver connects to TTL-RS232 converter with FFSD-04 cable and ASP 69533-01 (J2 connector).

Then the output of the converted is a serial cable, which connects with GPS MK-2 nodes. The GPS serial cable and MK-2 Node serial cable must be bridged with a Male-2-Male adaptor. (see figure 7) The Adaptor must be cross-wired, i.e. the RXD of one Male Port connects to TXD of the other port, vice versa.

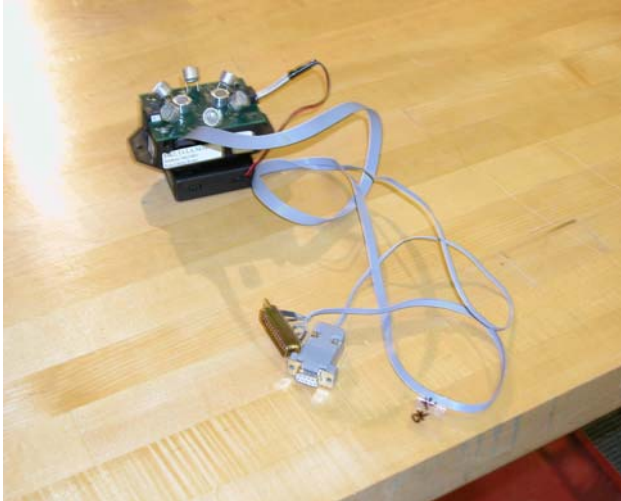


Figure 4 connect AVR connector with GPS node

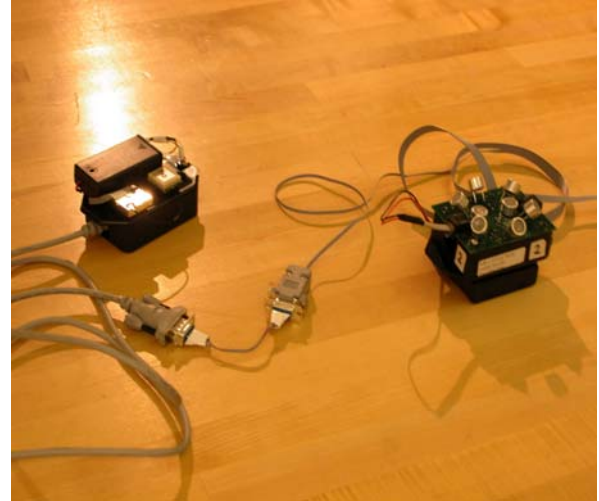


Figure 5 Bridge GPS serial cable and MK-2 serial cable with Male-2-Male adaptor

On the other side, the PC-Receiver node connects PC with a serial COM port. (see Figure 8) It is important to notice that when Serial-Parallel cable is used in application mode (see Figure 4), the parallel port must be disconnected from PC. Otherwise the COM serial port can not function properly. (Basically the PC COM port can write but can not read).

Now you are READY to GO. Figure 9 shows the complete-setup after the completion.



Figure 6 PC node connect with Laptop



Figure 7 The Final Implementation

Software Component

1. PALOS running on sensor node with GPS receiver.
2. PALOS running on gateway sensor node.
3. GPS Interpreter running on the PC, which connects to gateway sensor node.

Software Installation

It is assumed that you are using Windows XP or Windows 2K.

STEP 1: Install Cygwin. Programming Environment Setting-up

1. **Tiny-OS Linux Emulation Environment:** If you are setting a development environment under Windows, you first need to install TinyOS (<http://webs.cs.berkeley.edu/tos/>) TinyOS is a component-based OS from the networked sensor research.

(CAUTION: You must use ADMINISTRATOR account to setup the emulator)

2. **GNU compilers for ATmega processors**

- o **GNU avr-gcc compilers(For AVR Processor),**: the most current precompiled version for Windows can be found on the AVR Freaks website.
- o Uncompress the appropriate archive file in the directory of your choice:

```
gunzip gnutools_windows.tar.gz
```

```
tar -xvf gnutools_windows.tar
```

3. It will create a */cygdrive/c/gnutools* directory.

4. Edit your profile point to this directory.

1. open */cygdrive/c/cygwin/bin/.bashprofile*

2. add export

```
PATH=$PATH:/cygdrive/c/avr-gcc/bin:/cygdrive/c/gnutools/bin
```

3. close the file and restart *cygwin* again.

4. (TEST) type *make*, you should get

```
make:*** No targets specified and no makefile found. Stop. (It means that your make program is installed. )
```

5. (TEST) type *avr-gcc*, you should get

```
avr-gcc.exe:no input files. (It means that your avr-gcc compiler is installed.)
```

6. Download and compile the operating systems: PALOS for the ATmega128L

STEP 2: Install PALOS (including GPS codes) on GPS Node (MK-2)

1. Unzip the palos-GPS.zip into your root directory. There are 2 relevant directories specified for GPS applications. One is /palos-gps/test/GPS-MK2, which is for GPS node; the other is /palos-gps/test/GPS-PCReceiver, which is for PC-connecting gateway node.
2. Connect your GPS node with PC by Parallel port. , i.e. compiling mode. (see Figure3)
3. Enter /palos-gps/test/GPS-MK2, then type *make -f Makefile_MK2clean*
4. After clean up the files, type *make -f Makefile_MK2 install*.



Figure 8 Parallel-Serial Cable at Compiling Mode **Figure 9 Parallel-Serial Cable at Application Mode**

STEP 3: Install PALOS (including GPS codes) on gateway Node (MK-2), which connects to PC

1. Connect your gateway node with PC by Parallel port.
2. Enter /palos-gps/test/GPS-PCReceiver, then type *make -f Makefile_MK2clean*
3. After clean up the files, type *make -f Makefile_MK2 install*.

STEP 4: Install GPS Interpreter on your PC

1. Download GPS-PCInterpret.zip
2. Unzip it, then compile it with Visual C compiler. For your testing convenience we also include compiled a compiled executable under /Debug directory.

GPS Output on PC

The GPS receiver generates the location information and periodically sends to GPS node. Once the GPS node successfully extracts a complete GPS packet, (not restricted to ECEF location information), it broadcasts it while updating the local GPS data structure. The GPS flows among nodes are raw data, which is binary data in TSIP form. (In terms of TSIP format, refer to SQ Manual

http://www.trimble.com/lassensq_ts.asp?Nav=Collection-19088). In order to make it readable, we developed an interpreter on PC to translate them into user-friendly output. There are two simultaneous outputs: file output and real-time screen output. Here is an output example from the file output.

```
*****XYZ ECEF (Earth-Centered, Earth-Fixed Frame) Coordinate
with Single Precision*****
the origin of the coordinate system is at the center of mass of the
earth
```

```
Position XYZ ECEF X = 1408758.125000 meters
```

```
Position XYZ ECEF Y = 4586261.500000 meters
```

```
Position XYZ ECEF Z = 4188586.000000 meters
```

```
Pos ECEF Time of Fix5 = 510736.000000 second
```

```
Present Time To the Latest Sunday in Days = 5
```

```
Present Time To the Latest Sunday in left hours = 21
```

```
Present Time To the Latest Sunday in left minutes = 52
```

```
Present Time To the Latest Sunday in left seconds = 16
```

```
-----END OF XYZ Position-----
```

Example of GPS File Position Output

```
***** Start of GPS Time Message *****
```

```
GPS TIME Process: Time of Week
```

```
packet[0]= 152
```

```
packet[1]= 64
```

```
packet[2]= 92
```

```
packet[3]= 71
```

```
GPS TIME = 56384.593750
```

```
GPS Weeks: 1239
```

```
Year: 2003, Month:10, Day:4
```

Example of GPS File Time Output

⁵ Fix Of Time: the time when the position is fixed. It is in second unit and referenced to the beginning of the week.

We used a Cartesian coordinate system, which is called Earth-Centered, Earth-Fixed (ECEF) Coordinate. The ECEF is defined as follows.

- The origin of the system is at the mass center of the earth.
- The Z axis is along the axis of rotation.
- The X axis passes through the intersection of the CTP's⁶ equatorial plane and a reference meridian.⁷
- The Y axis is defined in equatorial plane to complete a right hand coordinate system.
- The position unit used here is meter.

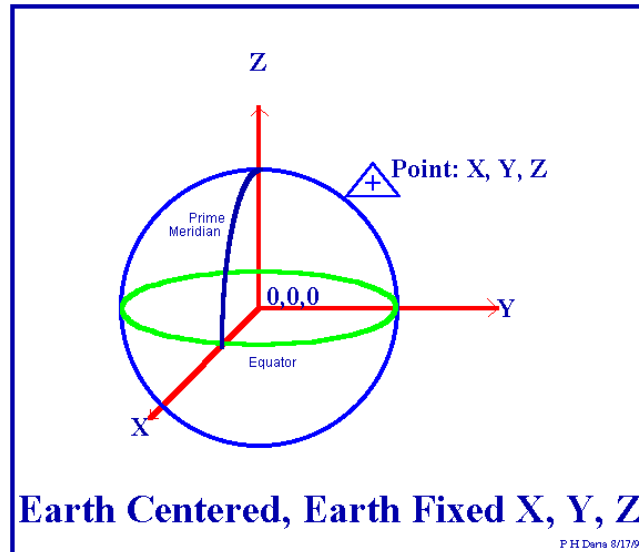


Figure 10 A Earth Centered Earth Fixed Coordinate System

We also provide a Greenwich (GMT) GPS time, which recycle every week. For example, you can see that the information is got on Friday 21:52:16. The clock is reset every Sunday morning 0:00 am. It is fairly enough for the research-oriented application.

***NOTE:** The Lassen SQ does not have the power management. If you are interested in power management, I recommend **Lassen LP Model** (<http://www.trimble.com/lassenlp.html>) In Schedule Track™ Model, the developer can program the unit to power up, quickly acquire satellites and output position to a schedule. After gathering satellite data and computing its location, the receiver may be directed to power down to a minimal mode of operation for a programmed interval or until awakened by a hardware interrupt-start performance at a programmed interval or in response to a hardware event. Schedule Track mode provides an advantage over normal battery-backed fast start modes with automatic wakeup to maintain current satellite data for fastest possible acquisition. Schedule Track offers the lowest power consumption in a deep sleep mode but provides position data as quickly as possible when needed.*

⁶ CTP: The mean position of the spin-axis for the years 1900-1905.

⁷ Prime Meridian: Meridian line is an imaginary line, which runs from the North Pole to the South Pole. The Prime Meridian line runs through “the primary transit” instrument (main telescope) at the Royal Observatory in Greenwich, UK.

GPS Design in PALOS

The GPS driver is implemented as a task in PALOS. THE GPS task communicates directly with the UART by GetNBytes() function and Task Event Queue. The UART is responsible to extract a long data stream from UART port. The data stream should be continuous and at least 50 bytes long. After the stream is received, the UART layer puts the streams into data buffer of GPS task event queue and handles the control over to GPS Task Layer with PalosTask_putEvent() function.

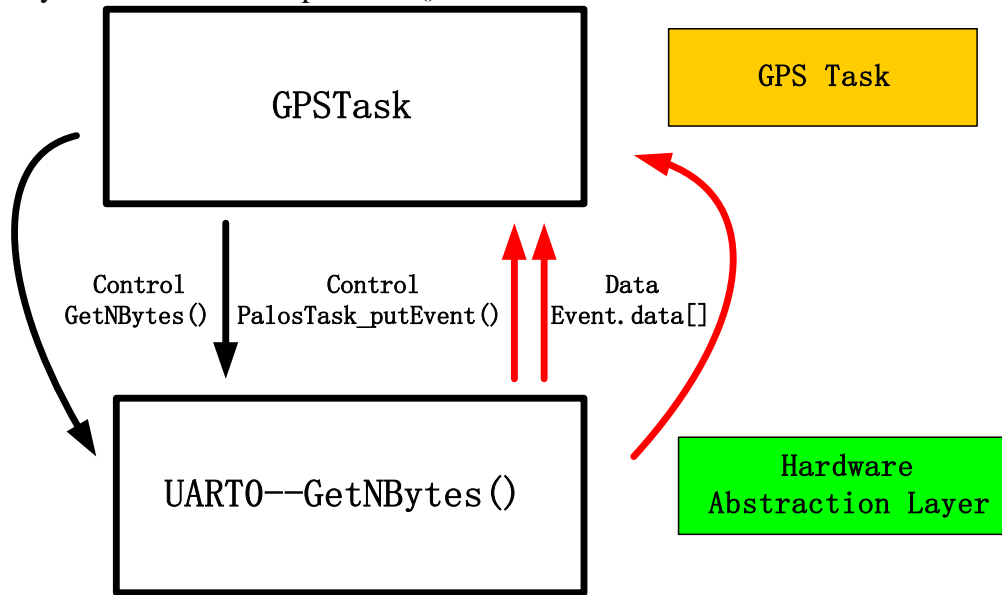


Figure 11 Design of GPS Task

After taking over the control, GPS Task Layer extracts those valid GPS packets from the received data stream. Figure 12 shows an example of an incoming stream as it is received from the GPS module. For those fragments at the stream's head and tail, task simply discards them.

After the valid packets are received, the GPS task processes it with their report ID⁸ and broadcast the necessary data through the radio channel. When the GPS task is done, GPS task returns the control back to UART layer by calling GetNBytes() function and wait for the next data stream from UART.

NOTE: The TSIP packets are in binary format form. It begins with "0x10" and ends with "0x10, 0x03". The heading tag, "0x10", is immediately followed by a report ID, which indicates what purpose this packet is for, such as position, GPS status, time etc.

On the other side, the gateway node receives the broadcasted data and sends it directly to PC through UART port. A GPS interpreter on PC will translate the raw GPS data into readable GPS position and time information on the screen.

⁸ Report ID is a field in GPS packet, which identifies the meaning of this packet output.

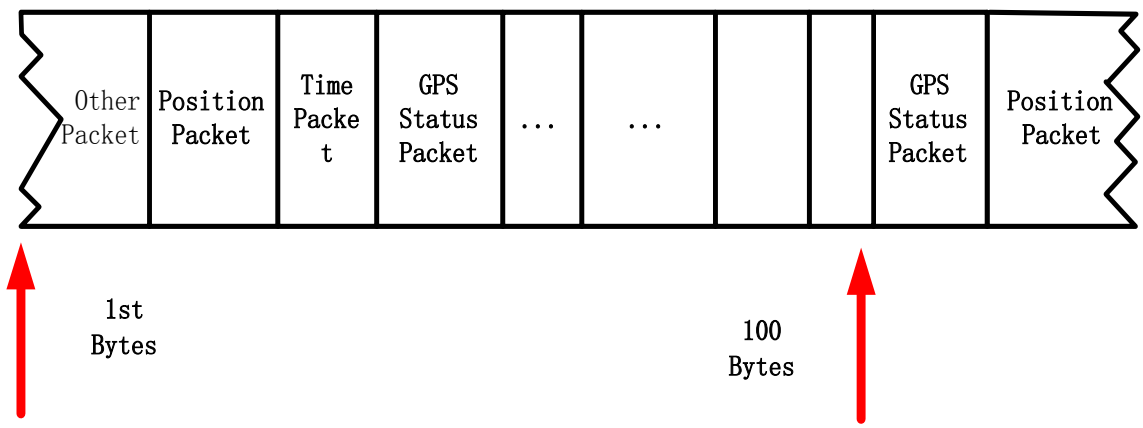


Figure 12 Case when a long data stream extracted