Ashtech

Geodetic Base Station Software™

User's Manual

V 3.2.0

January 2002

For information on translations and distribution outside the U.S.A. please contact Magellan.

Printed in the United States of America.

Part Number: 630248, Revision B

January 2002

SOFTWARE LICENSE AGREEMENT

IMPORTANT: BY OPENING THE SEALED DISK PACKAGE CONTAINING THE SOFTWARE MEDIA, YOU ARE AGREEING TO BE BOUND BY THE TERMS AND CONDITIONS OF THE LICENSE AGREEMENT ("AGREEMENT"). THIS AGREEMENT CONSTITUTES THE COMPLETE AGREEMENT BETWEEN YOU ("LICENSEE") AND MAGELLAN CORPORATION. ("LICENSOR"). CAREFULLY READ THE AGREEMENT AND IF YOU DO NOT AGREE WITH THE TERMS, RETURN THIS UNOPENED DISK PACKAGE AND THE ACCOMPANYING ITEMS TO THE PLACE WHERE YOU OBTAINED THEM FOR A FULL REFUND.

LICENSE. LICENSOR grants to you a limited, non-exclusive, non-transferable, personal license ("License") to (i) install and operate the copy of the computer program contained in this package ("Program") in machine acceptable form only on a single computer (one central processing unit and associated monitor and keyboard) and (ii) make one archival copy of the Program for use with the same computer. LICENSOR and its third-party suppliers retain all rights to the Program not expressly granted in this Agreement.

OWNERSHIP OF PROGRAMS AND COPIES. This License is not a sale of the original Program or any copies. LICENSOR and its third-party suppliers retain the ownership of the Program and all copyrights and other proprietary rights therein, and all subsequent copies of the Program made by you, regardless of the form in which the copies may exist. The Program and the accompanying manuals ("Documentation") are copyrighted works of authorship and contain valuable trade secret and confidential information proprietary to LICENSOR and its third-party suppliers. You agree to exercise reasonable efforts to protect the proprietary interests of LICENSOR and its third-party suppliers in the Program and Documentation and maintain them in strict confidence.

USER RESTRICTIONS. The Program is provided for use in your internal commercial business operations and must remain at all times upon a single computer owned or leased by you. You may physically transfer the Program from one computer to another provided that the Program is operated only on one computer at a time. You may not operate the Program in a time-sharing or service bureau operation or rent, lease, sublease, sell, assign, pledge, transfer, transmit electronically or otherwise dispose of the Program or Documentation, on a temporary or permanent basis, without the prior written consent of LICENSOR. You agree not to translate, modify, adapt, disassemble, decompile, or reverse engineer the Program, or create derivative works of the Program or Documentation or any portion thereof.

TERMINATION. The License is effective until terminated. The License will terminate without notice from LICENSOR if you fail to comply with any provisions of this Agreement. Upon termination, you must cease all use of the Program and Documentation and return them, and any copies thereof, to LICENSOR.

GENERAL. This Agreement shall be governed by and construed in accordance with the Laws of the State of California and the United States without regard to conflict of laws provisions thereof and without regard to the United Nations Convention on Contracts for the International Sale of Goods.

DISCLAIMER OF WARRANTIES AND LIMITATION OF LIABILITY

LICENSOR AND ITS THIRD-PARTY SUPPLIERS MAKE NO WARRANTIES OR REPRESENTATIONS, EXPRESS OR IMPLIED, REGARDING THE PROGRAM, MEDIA, DOCUMENTATION, RESULTS OR ACCURACY OF DATA AND HEREBY EXPRESSLY DISCLAIM ANY WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE AND NONFRINGEMENT. LICENSOR AND ITS THIRD-PARTY SUPPLIERS DO NOT WARRANT THE PROGRAM WILL MEET YOUR REQUIREMENTS OR THAT ITS OPERATION WILL BE UNINTERRUPTED OR ERROR-FREE.

LICENSOR, its third-party suppliers, or anyone involved in the creation or delivery of the Program or Documentation to you shall have no liability to you or any third-party for special, incidental, indirect or consequential damages (including, but not limited to, loss of profits or savings, downtime, damage to or replacement of equipment or property, or recovery or replacement of programs or data) arising from claims based in warranty, contract, tort (including negligence), strict liability, or otherwise even if LICENSOR or its third-party suppliers have been advised of the possibility of such claim or damages. The liability of LICENSOR and its third-party suppliers for direct damages shall not exceed the actual amount paid for this Program License.

Some states do not allow the exclusion of limitation of implied warranties or liability for incidental or consequential damages, so some of the above limitations or exclusions may not apply to you.

U.S. GOVERNMENT RESTRICTED RIGHTS

The Program and Documentation are provided with RESTRICTIVE RIGHTS. Use, duplication, or disclosure by the Government is subject to restrictions as set forth in subdivision (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 or subdivision 9(C)(1) and (2) of the Commercial Computer Software - Restricted Rights 48 CFR 52.227.19, as applicable.

Should you have any questions concerning the License Agreement or the Limited Warranties and Limitation of Liability, please contact in writing: Magellan Corporation, 471 El Camino Real, Santa Clara, CA 95050

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1

INTRODUCTION TO GEODETIC BASE STATION SOFTWARE (GBSS)

1.0 Overview

Thales Navigation Geodetic Base Station Software (GBSS) is a PC-based program that has been specifically designed for continuous logging of high-quality GPS data. The GBSS software provides the user with sophisticated file creation, file management and file distribution. In addition, all copies of GBSS software come with a Post-Session Command feature. The Post-Session Command feature allows powerful system integration tools to be deployed and extends tremendous flexibility to the base station operator. The result is an advanced and automated continuous reference station system capable of creating multiple files simultaneously (even compressed files) and automatically distributing them anywhere in the world.

GBSS will operate on a Windows 95/98/NT or Windows 2000 platform. However, the Windows NT Workstation and Windows NT Server or Windows 2000 are strongly recommended over Windows 95/98. Windows NT Server (Version 4.0 or higher) is recommended for users desiring to make their data available via FTP or via Web pages. GBSS is 32-bit in nature and takes full advantage of NT's preemptive multi-tasking and multi-threading capabilities. These features provide a stable and secure base station platform that requires minimal maintenance.

GBSS currently supports the following Thales Navigation GPS receivers:

- All Z-12 receivers
- All Z Surveyor receivers
- All Z-FX receivers
- All µZ receivers
- All Super C/A receivers
- The Z-Xtreme receiver
- The G12 receiver
- The GG-24 single-frequency GPS/GLONASS receiver
- The Z-18 dual-frequency GPS/GLONASS receiver.

GBSS allows you to simultaneously create a wide variety of different file types. Each of the file types are easily activated and deactivated through the GBSS point-and-click interface. Data files can be automatically stored in any one of four user-selectable directory structures. GBSS supports creation of the following different file types:

- Dual-frequency Ashtech format (GPS)
- Single-frequency Ashtech format (GPS)
- Dual-frequency RINEX format (GPS)
- Single-frequency RINEX format (GPS)
- Dual-Frequency Ashtech format (GPS/GLONASS)
- Single-frequency Ashtech format (GPS/GLONASS)
- Ionospheric model file
- Trap File (described later)
- NMEA file
- Diagnostic log file
- Compressed files

Please note that all of the files supported by a particular receiver can be created *simultaneously* by GBSS. For instance, the GBSS software allows you to simultaneously log dual-frequency and single-frequency Ashtech GPS data and dual-frequency and single-frequency RINEX files while connected to only one dual-frequency receiver. GBSS accomplishes this by automatically splitting the dual-frequency data stream into dual-frequency and single-frequency components, and then storing each component in separate files (which could then be stored in different directories).

All of the above files can be automatically compressed by GBSS. This feature facilitates archiving of data and automated FTP transfers where file size is important.

In addition, GBSS can be configured to automatically create different epoch intervals for the same time period. For example, a 1-hour dual-frequency RINEX file could be created at a 1-second interval, a 20-second interval and a 30-second interval with no interpolation of data points. This feature allows the base station operator to post data from the same time period at different epoch intervals.

The Post-Session Command feature allows you to create even more file types than those listed above. Any third-party command-line driven program can be called by GBSS. This feature allows you to call such a program to automatically do work on one of the above files. This results in entirely new data formats not directly supported by GBSS.

Many file management tools have been built into GBSS, and these tools provide sophisticated control over the collected data. GBSS comes with four user-selectable directory structures. For example, dual-frequency RINEX data can be stored in the Primary directory structure and single-frequency RINEX data can be stored in the Secondary structure. These file management tools thus allow the GBSS operator to provide different users with different file types.

GBSS allows you to set the File Duration (file length) to a value between 3 minutes and 84 hours. Each copy of GBSS also comes with a user-selectable automatic file deletion feature. This feature automatically deletes any file older then the user-specified age. For example, if the File Deletion Age is set to 30 days, any file created by GBSS older then 30 days will automatically be deleted.

GBSS also allows you to effectively manage any incoming NMEA messages. These incoming NMEA messages are automatically culled into their own NMEA files. This feature is especially useful when interfacing GBSS with tiltmeters, meteorological stations, digital seismometers or any other digital instrument outputting industry-standard NMEA messages.

GBSS provides an extensive *Session Logging*/Programming capability. Through this feature, you can configure GBSS to record data only during specified time periods. These "*logging sessions*" can be both recurring and nonrecurring. The recurring *logging sessions* repeat on a daily or weekly basis. For example, GBSS can be configured to record data only on Mondays, Wednesdays, and Fridays between the hours of 9:00 AM and 5:00 PM. The nonrecurring *logging sessions*, which occur once, are defined by a start time (that is, year, month, day, hour, minute) and a duration. For example, GBSS can be configured to start recording data on November 12, 2002 at 03:00 and continue to log for 10 days.

An automatic sub-directory creation feature can be enabled for each of these directory structures. Data are automatically stored in daily sub-directories eliminating the confusion of storing all data in a single directory. Furthermore, the Post-Session Command feature provides virtually unlimited file management tools by allowing the user to tailor the software for individual applications.

The Post-Session Command feature allows command-line driven programs to be launched in accordance with the File Duration setting. One example of the Post-Session Command feature is automated file distribution. Thales Navigation has worked with Ipswitch, Inc. to develop an automated data distribution system using Ipswitch's WS_FTP Pro program. GBSS can be programmed by the operator to open up an FTP connection at the end of each *session* and "push" the data to any remote FTP site in the world. Consider the example where GBSS is configured to create 1-hour files and has FTP Post-Session Commands enabled. At the end of the 1-hour file *session* GBSS will launch the FTP Post-Session Command and automatically distribute the data to remote FTP servers. Any of the files created by GBSS can thus be automatically pushed around the world to remote users. This feature provides the ultimate in data management and distribution over the Internet.

In addition, any of the four user-selected directory structures on the local PC running GBSS can be replicated on remote FTP sites. GBSS can be programmed by the operator to automatically open an FTP connection to a remote FTP site and then automatically create the same directory structures that are currently present in GBSS. This process occurs in accordance with the File Duration parameter. For example, if the file interval is set to 1 hour, GBSS will open the FTP connection every hour and create the GBSS directory structures on the remote FTP site. Once these directory structures are created, GBSS can then push any of its files to these directories on the remote computer. The end result is that it appears that the remote FTP site is directly connected to a GPS station, even though it is not.

Data files created by GBSS can also be made passively available to other users via Windows NT Server. GBSS has been specifically designed to work in concert with the FTP and Web page utilities that come standard with NT Server. Windows NT allows the user to provide the right subset of GPS data to the right group of users. For example, the system administrator may wish to grant access to the single-frequency data, but restrict access to the dual-frequency data. This is accomplished by setting up different access levels for different users. For example, dual-frequency RINEX data can be placed in the Primary directory structure and single-frequency RINEX data can be placed in the Secondary directory structure. Windows NT Server then allows you to define different permissions to these different directory structures. For example, the single-frequency data could be made available via anonymous FTP, whereas the dual-frequency could be made available via restricted logon. Users can then access this data through an FTP connection, or through a Web page interface. Because of the multi-tasking and multi-threading nature of Windows NT Server (Version 4.0), multiple users may access the data simultaneously. Windows NT Server can be easily setup with the NTFS file structure for added security.

GBSS can be configured to work with a BBS system. There are many off-the-shelf packages that provide a means to call in via telephone line and download data.

It must be emphasized that GBSS does not, by itself, contain *all* of the tools necessary for automated Internet and BBS operation. Rather, it has been designed to work in concert with such packages. For example, in order to employ the automated FTP distribution, you must first purchase the program WS-FTP Pro from Ipswitch, Inc. Ipswitch, Inc. can be reached at the following Internet address:

http://www.ipswitch.com/

GBSS also has a real-time external program interfacing capability. This feature provides applications external to GBSS with the ability to obtain the real-time data collected by GBSS. In fact, currently available are two specialized program modules available from Thales Navigation that exploit this interface: 1) the GBSS Meteorological module and 2) the GBSS Tilt module. The GBSS Meteorological module is used when connecting a Paroscientific MET3 or MET3A station. The Meteorological Option permits the automatic collection, conversion and archival (even RINEX Met. Files) of meteorological data. The Tilt Module integrates a tilt meter supplied by Applied Geomechanics, Inc. With this module, you can monitor and record the tilt information output by the tilt meter attached to the GPS receiver.

1.1 Minimum System Requirements

GBSS requires the target platform to be a Windows 95 or Windows NT-based computer. While GBSS requires less than 5 megabytes of memory to run, Windows 95 and NT impose higher minimums. It is recommended, for performance reasons, that your computer has no less than 16 megabytes of RAM memory for Windows 95 and no less than 24 megabytes of RAM memory for Windows NT. Thales Navigation strongly recommends a Windows NT platform.

Your disk space requirements will vary depending upon your unique configuration of GBSS. GBSS actually requires less than 10 megabytes of disk space to store the program and its ancillary files but it is recommended that your available disk space be much larger to accommodate your data storage needs. Because GBSS can create so many files simultaneously, Thales Navigation recommends installing a large hard drive for users who need to create multiple files.

To optimize the performance of GBSS, please give special consideration to the PC you choose to run the software. It is important to choose a leading brand system to avoid cheap components such as serial cards often found in off-name brands. Although GBSS will run on most all Intel processors (486 and up), it's performance will be maximized on a well-built Pentium system specifically designed for NT.

1.2 Special Requirements

If you desire to use the file compression capabilities of GBSS, you will need to have a valid copy of PKZIP 2.04g or PKZIPC.EXE (from the PKZIP 4.5 Suite for Windows) installed on your computer. Additionally, your PATH must be configured to provide programs access to the PKZIP or PKZIPC programs. These programs are off-the-shelf packages and are not provided as part of GBSS. If you plan on using the long file name options of GBSS (see Section 3.1.4.2, page page 25), you will need to use PKZIPC.exe (from the PKZIP 4.5 Suite for Windows).

If you will be using the Auto-Startup feature of GBSS (see Section 3.2, page 60) and are installing GBSS on a Windows NT platform, you will need to bypass the Windows NT logon screen. For this you will need to edit the registry to allow automatic logon. This is described in Section 3.2.

If you desire to use the automated FTP push feature of GBSS, it is necessary to purchase WS_FTP Professional (Version 4.53) from Ipswitch, Inc. WS_FTP Pro is not distributed with GBSS because Thales Navigation does not own it. WS_FTP Pro can be purchased from Ipswitch, Inc. at the following Internet address:

http://www.ipswitch.com

During installation it is highly recommended that you not install WS_FTP Pro into a directory that has spaces in directory names (such as the default directory "\Program Files\WS_FTP\"). This is because Windows NT and Windows 95 have difficulty interpreting some command line calls containing spaces in path names. Note: *the location of FTP95PRO must be in the "PATH" of your system.*

1.3 Demo Versions of GBSS

There are two basic configurations of GBSS: fully operational and demonstration versions. This document applies to both configurations. Demonstration versions, which are freely distributed over the Internet or provided on diskette, have a greatly reduced capability when compared with a fully operational version. Demonstration versions will only allow a 30-second epoch interval, log a maximum of 100 epochs, and will not perform file compression. Most importantly, demonstration versions do not allow Post- Session Commands to be executed nor is the passive mode permitted (see Section 3.1.2.1). To learn more about the Post-Session Command feature, please contact Thales Navigation for more details. Please note that the installation instructions documented herein apply to both configurations.

You can obtain a demonstration version from the Thales Navigation web page at the following address:

http://www.ashtech.com/

2

INSTALLATION INSTRUCTIONS

2.0 Overview

For most users of GBSS the installation is very straightforward. The installation diskettes use an industry-recognized installer program. If for any reason you decide to remove GBSS (such as to install an upgrade to GBSS), it can be removed (and all of its support components) using normal Windows 95 or Windows NT software uninstall mechanisms. Please see the end of this section for details on uninstalling GBSS.

Please note that when installing GBSS on a Windows NT machine, it is necessary to install GBSS under an account that has full administrative privileges (such as the Administrator account). If you attempt to install GBSS under an account that does not have full administrative rights, GBSS will not install and run properly. This is because the GBSS installer needs to add device drivers for its sentinel key.

During the installation of GBSS the following major components will be installed:

- GBSS Program files
- Sentinel Drivers
- GBSS Sound Files
- Stand-alone RINEX converter
- Stand-alone FTP program (creates directories on remote computers)
- Stand-alone GPS/GLONASS → GPS conversion program

The program files include the executable program, its configuration file, and a simulation test file. The sentinel drivers are required to allow GBSS to communicate with its sentinel key (without these drivers, GBSS may not run). This sentinel key comes standard with your copy of Geodetic Base Station. The sentinel key allows GBSS and its support utilities to run on a single workstation. Please note that multiple copies of GBSS can be run on a single workstation with a single sentinel keys.

The GBSS sound files are a set of WAV files that GBSS can be configured to play when certain events occur (see Section 3.1.6.3). You select which sound files to play during the configuration of GBSS.

The stand-alone support utilities are all 32-bit in nature and are command line-driven. These support programs provide powerful system integration tools to the base-station operator and facilitate many specialized applications. The stand-alone RINEX converter can be used as a command line-driven program or as a standard Windows menu program. For example, the RINEX program can be called from the GBSS Post-Session Command feature for special RINEX applications not supported by the built-in RINEX converter in GBSS. Alternately, the RINEX converter can be used to manually 'RINEX' data through a simple Windows interface.

The stand-alone FTP program is used when it is necessary to automatically create any of the four GBSS directory structures on a remote FTP server, or really any directory structure on a remote FTP server. This FTP program is called ASHFTPMD.EXE and was developed to work in tandem with WS_FTP Pro (not supplied with GBSS). ASHFTPMD.EXE is called from the Post-Session Command feature and can automatically create the four GBSS directory structures on any remote FTP computers. Furthermore, ASHFTPMD.EXE can be used to automatically create many different directory structures. For example, ASHFTPMD.EXE could be used to automatically create a directory structure on the remote computer based upon site name, year and month, rather than year and month alone. ASHFTPMD.EXE supports these specialized applications. For additional information, please see Appendix C.

The stand-alone GPS/GLONASS → GPS program allows you to take GPS/GLONASS data files and convert them to GPS-only data files. This program is called GNSS2GPS.EXE and is command-line driven. There is no graphical Windows user interface for GNSS2GPS.EXE. Appendix D provides detailed documentation on this program.

2.1 The Installation Process

To install GBSS onto your computer, insert the GBSS installation diskette labeled "GBSS Install Disk 1" into the A (or B) drive of your computer. Press the "Start" button and select "Run". Use the "Browse" command to locate and run the "Setup" program on the diskette located in the A (or B) drive of the computer.

The install program guides you through the installation of the GBSS software. At each step you will be given an opportunity to accept default options or tailor these to your individual needs. You will be required to enter your 8-character serial number. This serial number is located on each of the GBSS installation diskettes. The GBSS serial number is the first eight characters located on your installation diskettes. For example, your diskette may be labeled as follows:

KF004561-GBSS00-041698

For this example the GBSS serial number is KF004561. Please note that without the proper serial number you will not be allowed to continue the GBSS installation.

Upon completing the installation of the GBSS program and data files, you will be asked three questions:

1. Whether or not you want a GBSS entry in the Windows Start Program menu.
2. Whether or not you want a shortcut to GBSS added to your Windows desktop.
3. Whether or not you want GBSS to be automatically started with each start of Windows.

Answering no to any of these questions does not prohibit you from later manually activating or deactivating the features. Likewise, answering yes to any of these questions will not prohibit you from manually deactivating the features. Manually activating and deactivating these features can be accomplished through standard Windows configuration parameters (such as creating shortcuts).

If you decide to add GBSS to the Windows Start Program menu, then you will be able to quickly launch GBSS using the Windows "Start" button.

If you decide to add a shortcut to GBSS to your Windows desktop, then you will be able to quickly launch GBSS by double-clicking the its icon on you Windows desktop.

If you choose for GBSS to be automatically started with Windows, then each and every time you start Windows GBSS will be launched and will attempt to automatically connect to a GPS receiver. Please note that this feature should only be enabled for users who wish GBSS to connect without human intervention (such as permanent reference station sites). Unless you have some alternative power supply, your computer will shut down whenever there is a power failure. When the Auto-Connect feature is enabled, your computer will automatically start Windows, which in turn will then automatically start GBSS, which will then re-connect to the GPS receiver. This feature provides the base station operator with the assurance that the continuous reference station will weather power failures and will automatically re-start after a power failure.

Please note that there are some special installation instructions described in Sections 2.5 and 2.6 for various configurations of GBSS. Although you may not need to use these special instructions, it is strongly advised that you familiarize yourself with them before completing the GBSS installation process.

Finally, after installing GBSS and before collecting data operationally, it is suggested that you collect some sample GPS data for about 5 minutes and then terminate GBSS through its normal termination methods described later in this manual. The reason is simply that GBSS collects information about your computer and receiver to which GBSS is connected and then stores that information in its configuration files. This information is primarily used in the naming of output files. If you do not follow the procedure you could have some incorrectly named files.

2.2 GBSS Support Utilities

GBSS comes standard with three stand-alone utility programs. These programs are installed into the "utils" sub-directory of the GBSS installation directory. In order for these programs to function properly, the location of the "utils" directory must be made known to the Windows operating system. This can be accomplished by adding a statement to the "Path" indicating the location of the "utils" directory. Please contact your system administrator or MIS department for complete instructions on how to make these utilities available for your use. These support utilities are documented in the appendices of this manual. Further installation instructions on these utilities can be found in those sections.

2.3 External Modules

GBSS also has a real-time external program interfacing capability. This feature provides applications external to GBSS with the ability to obtain the real-time data collected by GBSS. There are two such programs currently available from Thales Navigation: 1) the GBSS Meteorological module and 2) the GBSS Tilt module. The GBSS Meteorological module is used when connecting a Paroscientific MET3 or MET3A station. The Meteorological module permits the automatic collection, conversion and archival of meteorological data, including the creation of RINEX meteorological files. The Tilt Module integrates a tilt meter supplied by

Applied Geomechanics, Inc. With this module, one can monitor and record the tilt information output by a tilt meter attached to the GPS receiver.

To install any of these modules, first complete the installation of GBSS and then follow the installation instructions supplied with that external module. Section 3.1.9 provides some information about configuring GBSS for external modules. The information in that section is generic for all external programs using the real-time interface of GBSS. For specifics on each module supplied by Thales Navigation, please consult the documentation for that module.

2.4 Installing the Sentinel Key

Before actually running GBSS, you will need to install the software sentinel key. Please note that GBSS will not run without this sentinel key. Also note that you cannot start GBSS with the key and then later remove the key while GBSS is running. The software sentinel key is installed by attaching the end of the sentinel key labeled ↑COMPUTER↑ to a parallel printer port of your computer. Please tighten the screws of the sentinel key to securely connect the key to your computer. If a printer is connected to your computer, attach that cable to the sentinel. If the sentinel cannot be installed because of an obstruction behind the computer, you can place the sentinel key later in the parallel sequence (for example, you could attach the sentinel key to a DB-25 male to DB-25 female cable which is connected to your computer's parallel port). To ensure a good connection between the computer, the sentinel key and other parallel devices, use only IEEE standard parallel printer cables.

The sentinel key allows GBSS and its support utilities to run on a single workstation. Multiple copies of GBSS can be run on a single workstation without need of additional keys.

2.5 Windows 95-Specific Installation Instructions

The only special installation that you must perform under Windows 95 depends upon whether or not you will be using the compression capability (that is, PKZIP or PKZIPC). This section assumes that PKZIP or PKZIPC has already been installed onto your computer and that your "PATH" is set appropriately (Again, neither PKZIP nor PKZIPC are installed as part of, and do not come with, GBSS).

When GBSS compresses files, it uses PKZIP 2.04g or PKZIPC.EXE (from the PKZIP 4.5 Suite for Windows). Because PKZIP 2.04g is an MS-DOS program, when GBSS calls PKZIP or PKZIPC, an MS-DOS prompt window is created (it is opened as a minimized window: that is, the only sign of the MS-DOS window is an icon on the task bar). By default, Windows 95 does not close this window when PKZIP or PKZIPC terminates. To solve this problem, you will need to run GBSS and wait until it compresses files for the first time. Once the task bar icon for that window is created, simply follow these steps:

1. Click on the icon and the window will become maximized.
2. Put your cursor into the title bar of the window (that is, the top of the window, which is normally blue) and press the Right (not left) button on your mouse. This will cause a pop-up menu to appear.
3. Select the "Properties…" option on the pop-up menu. This will cause a new window to appear.
4. In the newly created window, select the "Program" Tab.
5. At the bottom of the program tab is a checkbox labeled "Close on Exit", set it to checked and press the OK button.

GBSS compresses files based upon your selected compression options (see Sections 3.1.4.5 to 3.1.4.8) and the "File Duration" (see Sections 3.1.4.10). You will need to wait for the "File Duration" to expire before GBSS actually compresses files. For configuration testing purposes, you can initially speed this process by setting the "File Duration" parameter to 0.1 (thereby setting the "File Duration" to 6 minutes).

2.6 Windows NT-Specific Installation Instructions

These special NT installation instructions should only be followed if you want GBSS to automatically start with each start of Windows. Section 2.1 explains that a fundamental decision needs to be made regarding whether GBSS is configured to automatically start with Windows or not. If the decision to automatically start GBSS is made, then a special installation procedure must be accomplished because of the following:

1. When Windows NT is started, you are normally required to follow a logon process where you must type your account name and password.
2. Windows NT has a built-in plug-and-play feature that may incorrectly detect your GPS receiver as a serial mouse.
3. Windows NT may not completely initialize hardware and device drivers before GBSS is started.

If it is not clear why item 1 is important, remember that you may not be present when your computer is re-started (for example, because of a momentary power failure). To overcome the Windows NT login requirement you can edit the registry to allow automatic logon for the computer. The procedure is as follows. Start the Registry Editor by clicking on the Start button, click on Run, and type regedit in the window that appears

1. In Registry Editor, select the following subkey:

   HKEY_LOCAL_MACHINE\Software

   \Microsoft

   \Windows NT

   \CurrentVersion

   \Winlogon
2. Add a new string value entry named AutoAdminLogon of type REG_SZ, and specify a value of 1. Do this using Edit/New/String Value to create the new value. Then highlight the new value and use modify to specify the value.
3. Add a new string value entry named DefaultPassword of type REG_SZ, and enter the password of the user who is listed under the value of **DefaultUserName**

Item number 2 is important because the RS-232 lines are set 'high' by a powered Ashtech receiver and can be misinterpreted by Windows NT when Windows is started. That is, each time Windows NT is started, its plug-and-play feature looks for any new peripheral devices that may have been recently connected. This feature may inadvertently determine that the GPS receiver connected to a communication port is a serial mouse device. To prevent this, you must modify your BOOT.INI file as described in the following steps.

1. Make a backup copy of the BOOT.INI file.
2. 2Determine the existing hidden, system, and read-only attributes from the BOOT.INI file (that is, write them down somewhere).
3. 3)Remove the hidden, system, and read-only attributes from the BOOT.INI file.
4. 4)Using a text editor (such as Notepad.exe) open the BOOT.INI file.
5. 5)Add the "/NoSerialMice" option to the end of each entry in the "[operating systems]" section of BOOT.INI. See the example below for more information.
6. 6)Save BOOT.INI and exit Notepad.
7. 7)Restore the hidden, system, and read-only attributes to the BOOT.INI file determined under step 2.

The following example shows modifications made to a BOOT.INI to prevent the detection of serial mouse devices:

```
[boot loader]
timeout=5
default=multi(0)disk(0)rdisk(0)partition(1)\WINDOWS
(operating systems)
multi(0) disk (0) partition (1)\WINDOWS="Windows NT Workstation
      Version 4.00" /NoSerialMice
multi(0)rdisk (0) partition (1)\WINDOWS="Windows NT Workstation
      Version 4.00 [VGA model]" /basevideo /sos /NoSerialMice
```

Complete documentation on preventing the detection of the serial mouse can be found on Microsoft's home page at *http://www.microsoft.com* under Article ID Q131976. As per that article, the syntax of the "NoSerialMice" is as follows:

/NoSerialMice	Disables the detection of serial mice on all COMM ports.
/NoSerialMice:COMn	Disables the detection of serial mice on COMM port n, where n is the number of the port.
/NoSerialMice:COMx,y,z	Disables the detection of serial mice on COMM ports x, y and z.

The final item in the list of special Windows NT installation instructions has to do with Windows NT's multi-tasking environment, Windows NT drivers and the Auto-Connect feature of GBSS. Specifically, the Rainbow sentinel drivers, which are installed as part of the automatic installation, are started and initialized by Windows each time NT is started. When Windows NT automatically starts GBSS, it first attempts to connect with its software sentinel and assumes that the sentinel driver is fully initialized. However, because of the multi-tasking nature of Windows NT, those drivers may not initialize before GBSS actually tries to interface with them. In other words, in some instances GBSS will attempt to communicate with the sentinel drivers before Windows has a chance to fully initialize those drivers. To prevent this race condition, the command-line call to GBSS in the Start-Up folder can specify a delay that GBSS will exercise before attempting to communicate with the sentinel device. Most Windows NT users will not need to change the automatic Start-Up. However, if you experience sentinel related errors when GBSS automatically starts, you may need to increase the automatic startup delay. Section 3.2 on page 60 provides complete details on setting this delay.

2.7 Uninstalling GBSS

GBSS and all of its components can be uninstalled via the "Add/Remove Programs" feature of the "Control Panel" in Windows. Please note that GBSS must be removed prior to installing a new version. The Install Shield program that installs GBSS does not detect and remove old versions.

3

CONFIGURING THE GEODETIC BASE STATION SOFTWARE

3.0 Configuration Overview

Before actually running GBSS, you will need to install the software sentinel key. Please note that GBSS will not run without this sentinel key. Also note that you cannot start GBSS with the key and then later remove the key while GBSS is running. The software sentinel key is installed by attaching the end of the sentinel key labeled ↑COMPUTER↑ to a parallel printer port of your computer. Please tighten the screws of the sentinel key to connect the key securely to your computer. If a printer is connected to your computer, attach that cable to the sentinel. If the sentinel cannot be installed because of an obstruction behind the computer, you can place the sentinel key later in the parallel sequence (for example, you could attach the sentinel key to a DB-25 male to DB-25 female cable which is connected to your computer's parallel port). To ensure a good connection between the computer, the sentinel key and other parallel devices, use only IEEE standard parallel printer cables.

The sentinel key allows GBSS and its support utilities to run on a single workstation. Multiple copies of GBSS can be run on a single workstation without need of additional keys.

Prior to connecting to the GPS receiver, GBSS needs to be configured to suit your data collections needs. Please note that this configuration process is extremely important, as the GBSS factory defaults will more than likely *not* meet your needs.

The GBSS configuration information is stored in two files, GBSS.INI and GBSS.SES, which are located in the directory in which the GBSS EXE program resides. GBSS.INI contains most of the GBSS configuration information. GBSS.SES contains configuration data that is specific to the *Logging Sessions* feature (described in Section 3.1.5, page page 34). GBSS automatically updates the contents of these files as you make changes using the "configuration" menus of GBSS. Changes in the configuration of GBSS are written to its configuration files (`GBSS.INI` and `GBSS.SES`) so that the configuration may be recalled at the start of the next run of the program. In this way, once the desired configuration is set, the operator no longer needs to change it -- unless, or course, it needs to be altered to support a new configuration. The majority of these parameters are set using the "Configuration" sub-menus. Details of the contents of the Configuration sub-menus follow.

Since `GBSS.INI` and `GBSS.SES` are ASCII text files, the configuration can also be modified with any text file editor. However, you are strongly discouraged from making configuration changes using this is approach. In fact, after you get GBSS configured to meet your needs, a prudent computer procedure would be to make a backup copy of the configuration files.

After configuring GBSS and before collecting data operationally, it is suggested that you collect some sample GPS data for about 5 minutes and then terminate GBSS through its normal termination methods described later in this manual. This will save the configuration files so that they will be available the next time GBSS starts.

3.1 Configuration Menus

Most GBSS parameters are set through the main menu "Configuration" option. The "Configuration" drop-down menu is divided into 9 different selections. These 9 selections become available when "Configuration" is selected, calling the drop-down menu shown in Figure 3.1.

Figure 3.1 Configuration Drop-Down Menu

The following sub-sections describe each of the nine configuration windows.

GBSS is designed such that the software is configured before connecting to a receiver. It is important to note that GBSS also needs to be configured before using the receiver simulation capability (see Sections 4.6 and 4.7 page 75). Once GBSS is connected to a GPS receiver, the GBSS “Configuration” screens are not editable. *GBSS must be disconnected from the GPS receiver before the “Configuration” screens become accessible again.*

You can, however, change the configuration of the *receiver* while GBSS is connected to that receiver. Special features have been built into GBSS to change the GPS receiver’s parameters while GBSS is still connected to the GPS receiver. The receiver’s parameters can be changed by sending commands through the terminal window, or by uploading a script file. These features are covered in Sections 4.4, page page 73 and 4.5, page page 74, respectively.

3.1.1 Communications (Communication Settings)

The Communications Configuration window, Figure 3.2, allows you to set the following communication parameters:

1. The PC communication port (labeled "Port”),
2. The PC communication speed (labeled "Speed”), and

3. Expected RS-232 status lines.

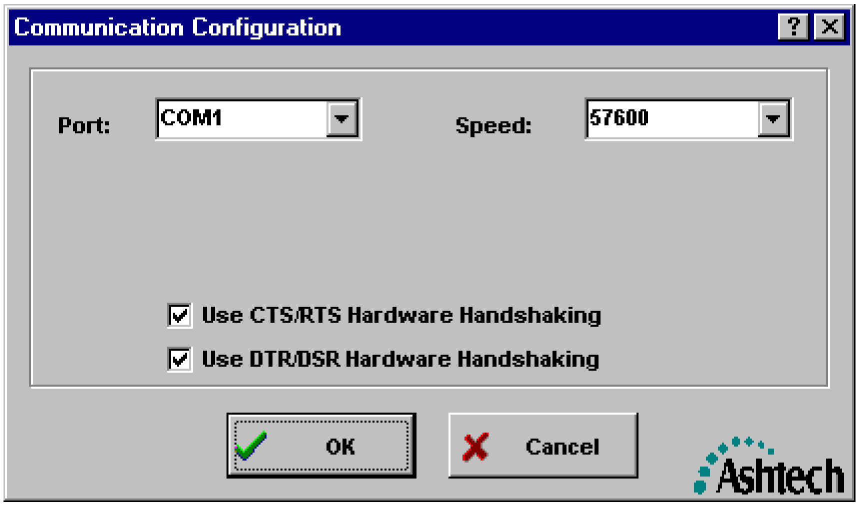


Figure 3.2 Commumnication Configuration Window

3.1.1.1 Configuration | Comms / Port

The port selection allows the operator to specify the communications port of the computer used to communicate with the Ashtech receiver. The permissible values are "COM1", "COM2", "COM3"… through "COM16".

3.1.1.2 Configuration | Comms / Speed

The speed selection allows the specification of the communications speed of the communications port of the computer used to communicate with the Ashtech receiver. The permissible values are 300, 600, 1200, 2400, 4800, 9600, 19200, 38400, 57600 and 115200. When GBSS is in Active Mode (see Section 3.1.2.1), it will command the receiver to the baud rate you select here.

3.1.1.3 Configuration | Comms / Use CTS/RTS Hardware Handshaking

The CTS/RTS Hardware Handshaking checkbox allows you to specify whether or not the normal CTS/RTS hardware flow control handshaking is enabled. In most configurations, this checkbox should be checked. Those who uncheck this checkbox should have clear rationale as to why they should eliminate the CTS/RTS hardware handshaking. For example, there are certain Ashtech receivers that do not employ the CTS/RTS hardware handshaking. In these cases, GBSS needs to be made aware of the difference.

3.1.1.4 Configuration | Comms / Use DTR/DSR Hardware Handshaking

The DTR/DSR Hardware Handshaking checkbox allows you to specify whether or not the normal DTR/DSR hardware flow control handshaking is enabled. In most configurations, this checkbox should be unchecked. Those who check this checkbox should have a clear rationale as to why they should enable the DTR/DSR hardware handshaking. For example, there are certain radio modems that implement the DTR/DSR hardware handshaking. In these cases, GBSS will not communicate with the connected receiver unless the DTR/DSR handshaking is enabled.

3.1.2 GPS Receiver (Receiver Settings)

The Receiver Configuration window, Figure 3.3 allows you to do the following:

1. Place GBSS is in Active or Passive mode;
2. Command the receiver's recording interval and elevation mask;
3. Command the receiver to use its compressed message format;

4. Command the receiver to disable storage of data to the receivers internal memory;
5. Upload the site data to the receiver;
6. Upload a script file to the receiver at connection time;
7. Enter information about the GPS receiver if GBSS is placed in Passive mode.

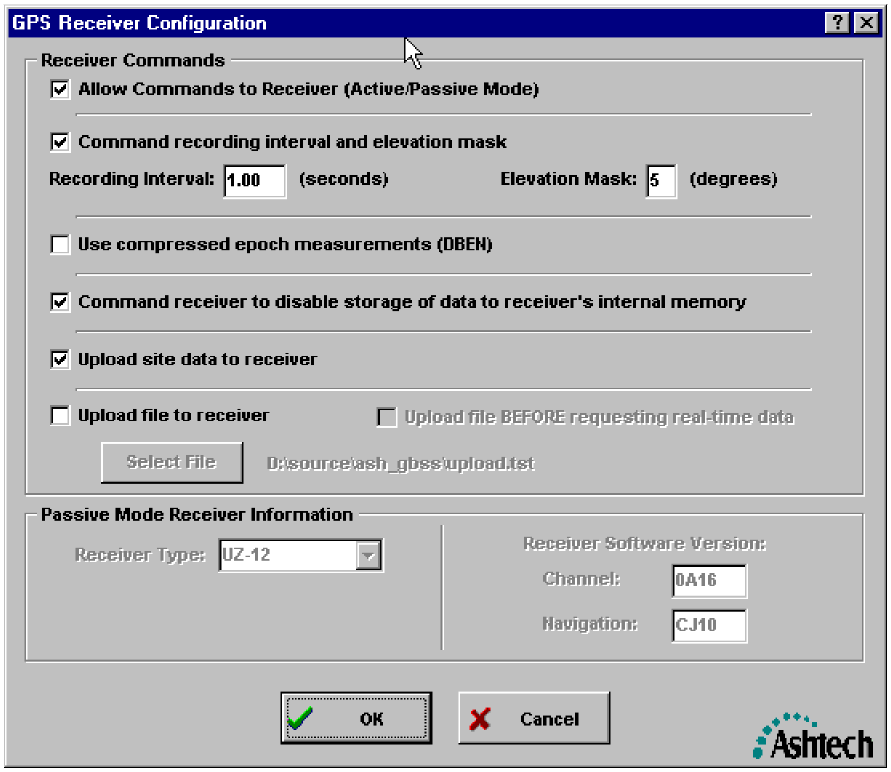


Figure 3.3 GPS Receiver Configuration Window 1

3.1.2.1 Configuration | Receiver / Active or Passive Mode

Checking the "Allow Commands to Receiver" checkbox places GBSS in the ACTIVE mode and allows the software to send commands to the receiver. These commands include the following:

- Query of receiver type
- Setting receiver's communication speed
- Enabling and disabling real-time outputs
- Setting receiver's elevation mask angle
- Setting receiver's recording interval
- Enabling requests for latest satellite navigation messages
- Enabling requests for ION messages
- Disabling receiver's internal (RAM) storage of epoch data
- Uploading site data to the GPS receiver
- Uploading commands placed in the upload file

The factory default for this setting is the ACTIVE mode, which Magellan recommends for most users.

When the check is removed from the check box (thereby placing the program into the PASSIVE mode), no commands can be issued to the receiver. If GBSS is placed in PASSIVE mode, the receiver must be configured independently of GBSS to the desired communication speed and configured to transmit the binary MBEN, PBEN, and SNAV messages. In brief, one must have *a priori* knowledge of the receiver settings (for example, baud rate) and then enter or select the corresponding values within GBSS. For instructions on configuring the receiver manually, please consult the appropriate receiver manual.

When in Passive mode, you must set the communication speed of GBSS under Configuration | Comms / Speed. Additionally, GBSS must be configured to identify the receiver type and receiver software version numbers (see Section 3.1.2.7). **Failure to properly identify receiver type and software version numbers could result in unusable output data files**.

Section 3.1.2 shows GBSS configured for Active Mode. Figure 3.4 shows GBSS in Passive Mode and connected to a Z-FX GPS receiver.

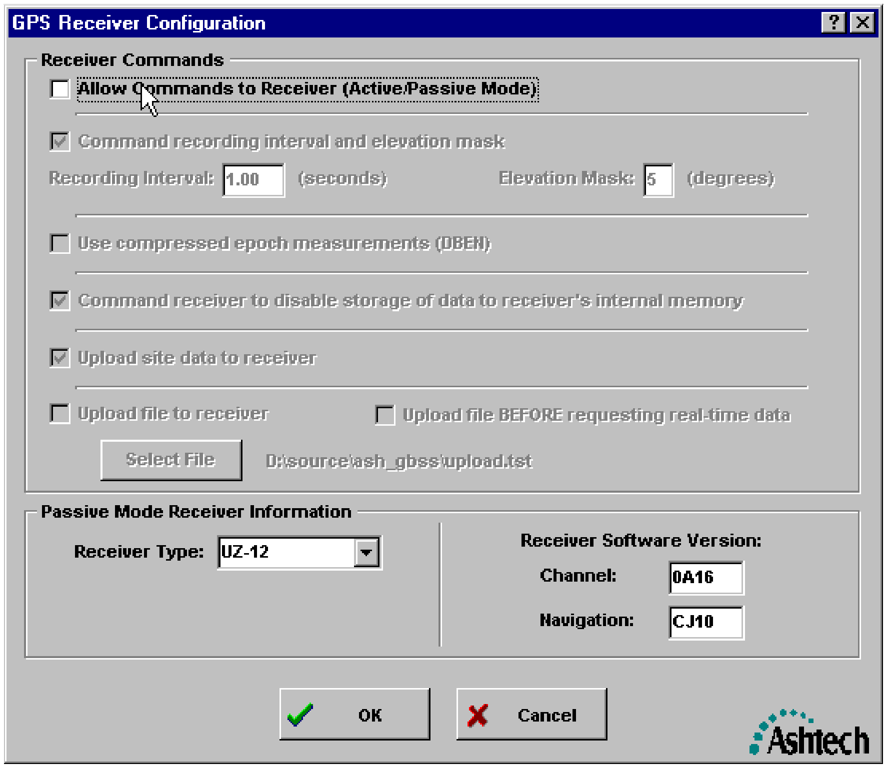


Figure 3.4 GPS Receiver Configuration Window 2

3.1.2.2 Configuration | Receiver / Epoch Interval and Elevation Mask

The "Command recording interval and elevation mask" checkbox is available and has meaning when GBSS is in its Active Mode (see Section 3.1.2.1). Placing a check into this box instructs GBSS to set the receiver's recording interval (or epoch interval) and elevation mask to the values displayed in the "Recording Interval" and "Elevation Mask" edit fields. These commands will be sent to the receiver each time GBSS is connected to the receiver (see Sections 3.2, 4.1, 4.6 and 4.7).

The "Recording Interval" and "Elevation Mask" edit fields are editable and have meaning when the "Command recording interval and elevation mask" checkbox is checked and active. It is important to note that the permissible values in the Recording Interval edit field may be different than that available for your receiver type. For example, some Z-XII receivers are not capable of a 0.1-second epoch interval but the edit field will still permit the entry of a 0.1-second epoch interval. This is because GBSS supports a wide array of Ashtech receivers and must support the maximum range of values found across the Ashtech product line. Sending an illegal epoch interval to the receiver will result in the receiver rejecting the command. In this case the receiver will continue with its current recording interval (and NO error message will be generated by the receiver or GBSS). As such, it is up to the operator of GBSS to ensure that only legal values for the recording interval are properly entered into the Recording Interval edit field.

3.1.2.3 Configuration | Receiver / Disable Receiver Epoch Storage

The "Command receiver to disable storage of data to receiver's internal memory" checkbox is available and has meaning when GBSS is in its Active Mode (see Section 3.1.2.1). It does not apply when GBSS is in its Passive Mode. Placing a check into this box instructs GBSS to command the receiver to stop recording epoch and navigation data to the receiver's internal memory (but the receiver will still output epoch-related data to GBSS). This command will be sent to the receiver each time GBSS is instructed to "Connect" to the receiver (see Sections 4.1, 3.2, 4.6 and 4.7).

3.1.2.4 Configuration | Receiver / Receiver Outputs Its Compressed Data Records

The "Use compressed epoch measurements" checkbox is available and has meaning when GBSS is in its Active Mode (see Section 3.1.2.1). It does not apply when GBSS is in its Passive Mode. Placing a check into this box instructs GBSS to command the receiver to use its compressed observation message format (for example, DBEN and MACM). At times it can be advantageous to have the GPS receiver output in its compressed message format. A limited bandwidth radio link is such an example. Please note that some receivers do not support these formats, and you should consult your receiver operations manuals for details.

3.1.2.5 Configuration | Receiver / Upload Site Data to Receiver

The "Upload site data to receiver" checkbox is available and has meaning when GBSS is in its Active Mode (see Section 3.1.2.1). It does not apply when GBSS is in its Passive Mode. Placing a check into this box instructs GBSS to upload the GBSS stored site-specific data to the receiver. The site-specific data is entered through the Configuration | Site GBSS menu (see Section 3.1.3). This site data will be sent to the receiver each time GBSS instructed to "Connect" to the receiver (see Sections 3.2, 4.1, 4.6 and 4.7). The site data consists of the site name, the antenna height and the GPS receiver reference position.

3.1.2.6 Configuration | Receiver / Upload File to Receiver

The "Upload file to receiver" checkbox is available and has meaning when GBSS is in Active Mode (see Section 3.1.2.1). It does not apply when GBSS is in its Passive Mode. Placing a check into this box instructs GBSS to upload the data stored in the specified upload script file each time GBSS is instructed to "Connect" to the receiver (see Sections 3.2, 4.1, 4.6 and 4.7).

This feature enables you to maintain a directory on your PC with a variety of script files for different applications. For example, you might define a script file named `RTCM.TXT` that contains the commands needed to configure the GPS receiver to output RTCM Type 1 corrections on a specified port and baud rate. Alternately, you might have a script file named `RTK.TXT` that configures the GPS receiver to output RTK corrections on a specified port at a specified baud rate. This feature enables you to predefine an unlimited number of configurations that are then always available for rapid use at a later date.

The "Select File" Button and the text to the right of the button are only enabled when the "Upload file to Receiver" checkbox is checked and GBSS is in Active Mode. It does not apply when GBSS is in its Passive Mode. When enabled, the text to the right of the "Select File" button describes the file currently selected for upload to the receiver on the next "connection" to a receiver. To change the currently selected upload file, press the "Select File" button and select the desired file from your hard drive. Figure 3.5 provides an example of the file selection window that is displayed when an active "Select File" button is pressed.

Figure 3.5 Select File to Upload

The “Upload file BEFORE requesting real-time data” checkbox allows you to specify whether or not the script file is uploaded before or after GBSS issues the real-time data request commands.

The format of the upload file is described in Appendix B.

3.1.2.7 Configuration | Receiver / Passive Mode Receiver Information

The information within the area entitled “Passive Mode Receiver Information” is only editable and will only have meaning when GBSS is in Passive Mode (see Section 3.1.2.1, page page 15). When the receiver is in Passive Mode, it is VERY IMPORTANT that GBSS know the exact receiver type to which it is connected. Because the software cannot determine the receiver type without sending commands to the receiver, and because the software needs to know the receiver type to know how to interpret data from the receiver, the “Receiver Type” edit field must be set correctly for passive base station operations. Failure to provide the correct receiver type may result in incorrect outputs.

The Receiver Channel Software Version (Labeled “Channel:”) edit field will take any 4-character sequence. This field allows the specification of the Ashtech receiver's channel software version and is stored within the output B-File. The Receiver Navigation Software Version (Labeled “Navigation:”) edit field will take any 4-character sequence. This field allows the specification of the Ashtech receiver's navigation software version and is stored within the output B-File. While the channel and navigation version information is not needed for the operation of the base station software, it is useful in troubleshooting receiver problems. That is, if data from this program is ever provided to Thales Navigation to analyze a problem, providing incorrect receiver software version information could mislead the engineers and delay the isolation of the problem.

When the base station software is in Active Mode (that is, GBSS can send commands to the receiver), this menu parameter is ignored. That is, the program can query the receiver and determine its channel and navigation software versions. When this program cannot send commands to an Ashtech receiver (or when the program is in simulation mode and the simulation file does not contain the RID packet from the receiver) the values of the “Channel:” and “Navigation:” edit fields will be used.

For Z-XII receivers, the navigation and channel software versions can be obtained by turning on your Ashtech receiver and selecting menu 0 (zero). The right side of the second line from the bottom contains a number of the form:

NNNN-CCCC

where, *NNNN* is the navigation software version and *CCCC* is the channel software version.

3.1.3 Site (Site Settings)

The Site Configuration window, Figure 3.6, allows you to enter the following site-specific data:

1. 4-character site name;
2. Antenna height; and
3. WGS-84 position of the site.

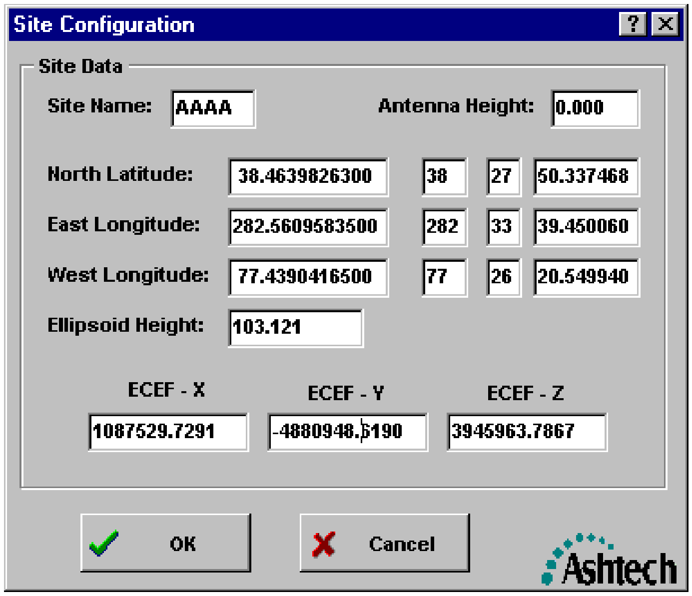


Figure 3.6 Site Configuration Window

Please note that if you wish to send the information on this menu to the receiver (for example, to set the 4-character site name output with each epoch of data), then the "Upload site data to receiver" checkbox of the "Configuration | GPS Receiver" menu must be checked (see Section 3.1.2.5 for further details).

When using the built in RINEX converter, all of the information in this menu will be stored as part the RINEX file header data (see Section 3.1.4.5).

3.1.3.1 Configuration | Site / Site Name

The "Site Name" edit field allows the entry of any 4 non-blank characters for the site name. With the exception of the '?' character, each of the four non-blank characters must be legal characters in MS-DOS file names. Any '?' character in the file name will be translated to the '_' (underscore) character.

This parameter may be used for three purposes. The first purpose depends upon whether or not the "Upload site data to receiver" checkbox of the "Configuration | GPS Receiver" menu (see Section 3.1.2.5) has been checked. If it has (and GBSS is in Active Mode) GBSS commands the receiver to place this site name in each epoch of the B-File data. This command will be sent to the receiver each time GBSS is "connected" to the receiver (see Sections 3.2, 4.1, , 4.6, and 4.7). The second purpose of this site name is in the creation of the names of the output B-, E-, S-, and Trap Files. Ashtech file names are of the form:

tnnnnsyy.ddd

where	t	is the file type (B, E, S, or T),
	nnnn	is the 4-character station name,
	s	is the *session* code,
	yy	is the last two digits of the year, and
	ddd	is the day of the year.

See the Appendices for complete details on file naming.

The third and final purpose of the entered site name is that it will always be output as part of the RINEX header.

3.1.3.2 Configuration | Site / Antenna Height

The "Antenna Height" edit field allows the entry of the vertical antenna height offset from the mark (that is, the HI). This parameter may be used for three purposes. The first purpose depends on whether or not GBSS is configured to output Site files (see Section 3.1.4.5). That is, if GBSS is configured to create Site files, the antenna height value entered on the site menu will be placed into the vertical antenna height field of the Ashtech Site file.

Secondly, the antenna height parameter will be sent to the receiver if the "Upload site data to receiver" checkbox of the "Configuration | GPS Receiver" menu (see Section 3.1.2.5) is checked. This antenna height information will be sent to the receiver each time GBSS is "connected" to the receiver (see Sections 3.2, 4.1, 4.6 and 4.7).

The third and final purpose of the entered antenna height is that it will always be output as part of the RINEX header (see Section 3.1.4.5).

3.1.3.3 Configuration | Site / Site Position

The WGS-84 position of the reference station can be entered through the position-related entry fields on the Site Configuration menu. When entering positional data, the other components will automatically be updated as data is entered. For example, entering a value for decimal degrees West Longitude will cause the Degrees, Minutes, and Seconds fields of West longitude to be updated as well as all components of East longitude. Additionally, the Earth-Centered Earth-Fixed positional components will also be updated.

The entered positional data will be sent to the receiver each time GBSS is "connected" to the GPS receiver (see Sections 3.2, 4.1, 4.6 and 4.7). The entered positions are ultimately converted to North latitude, East longitude and Ellipsoidal height which are the values sent to the receiver.

Additionally, all positional data entered will be output as part of the RINEX header data (see Section 3.1.4.5).

3.1.4 File Outputs (File Output Configuration)

The "File Outputs" Configuration window provides a simple point-and-click interface for file creation settings and file management settings. GBSS allows you to store a wide variety of GPS data in up to four user-selectable directory structures. Storing a particular file type in any one of the four directory structures is as easy as enabling that files checkbox.

GBSS allows you to create the following different file types:

- Dual-frequency Ashtech format (GPS)
- Single-frequency Ashtech format (GPS)
- Dual-frequency RINEX format (GPS)
- Single-frequency RINEX format (GPS)
- Dual-Frequency Ashtech format (GPS/GLONASS)
- Single-frequency Ashtech format (GPS/GLONASS)
- Ionospheric model file
- Trap File
- NMEA file
- Diagnostic log file
- Compressed files

The Post-Session Command feature (see Section 3.1.8) allows you to create even more file types than those listed above. Any third-party command line driven program can be called by GBSS. This feature allows you to call such a program to automatically do work on one of the above file formats. The result can be an entirely new data format not directly supported by GBSS.

The File Output Configuration window contains five different sections that are accessible via the five tabs located at the top of the window. These tabs are as follows:

- File Modes tab
- Data Files tab
- Compression Files tab
- NMEA Capture File tab
- RSIM Files tabs

Figure 3.7 through Figure 3.11 show five examples of the "File Outputs" Configuration window, one for each of these tabs:

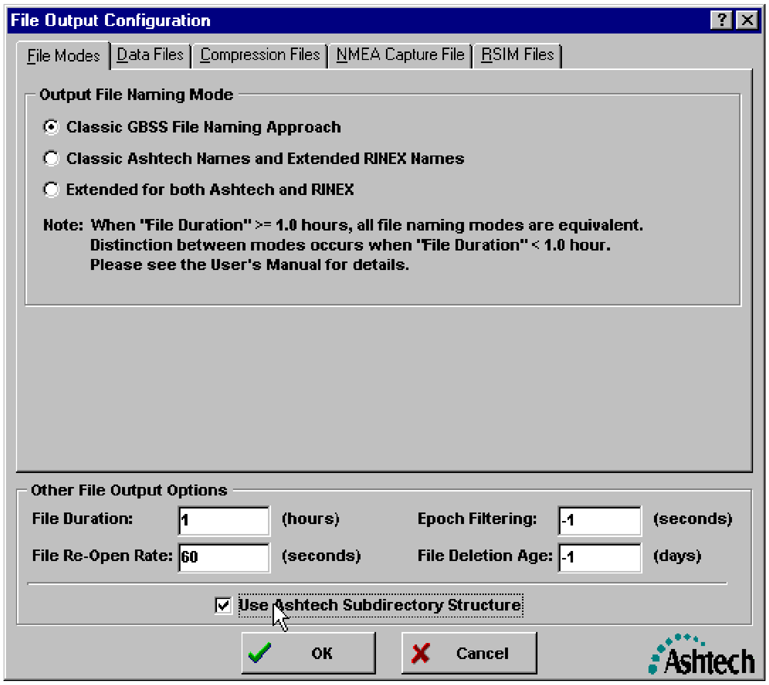


Figure 3.7 File Output Configuration Window 1

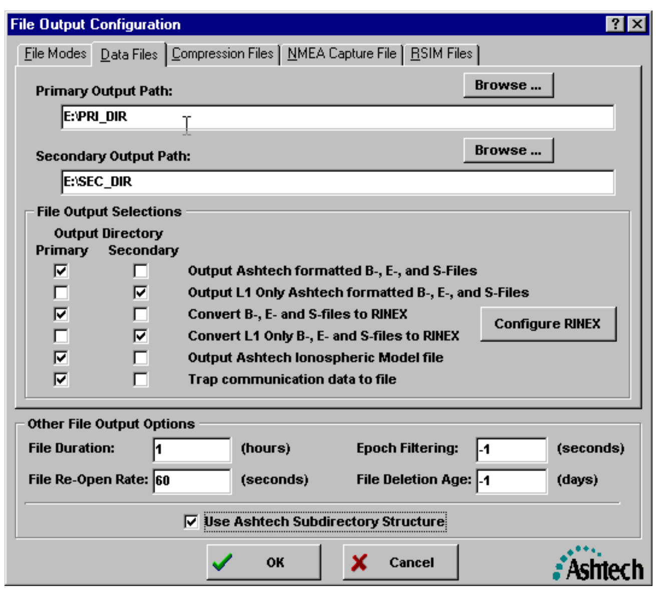


Figure 3.8 File Output Configuration Window 2

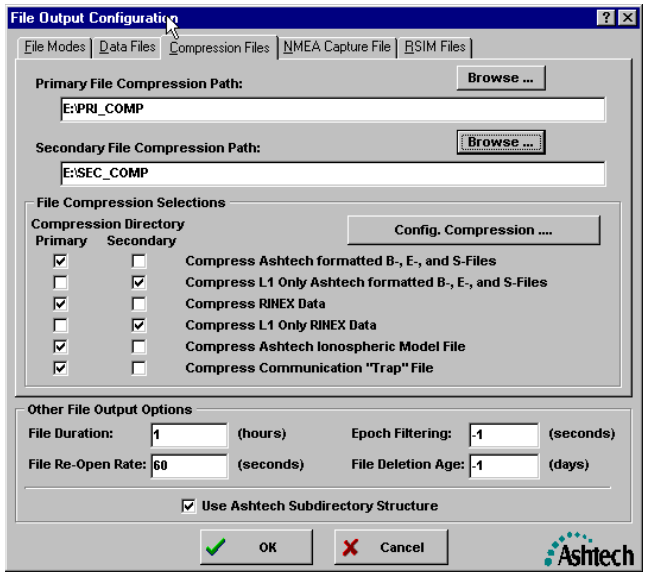


Figure 3.9 File Output Configuraiton Window 3

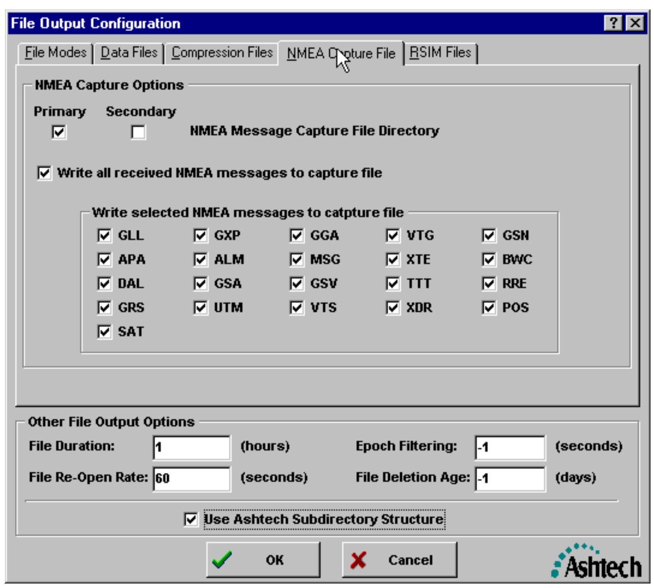


Figure 3.10 File Output Configuration Window 4

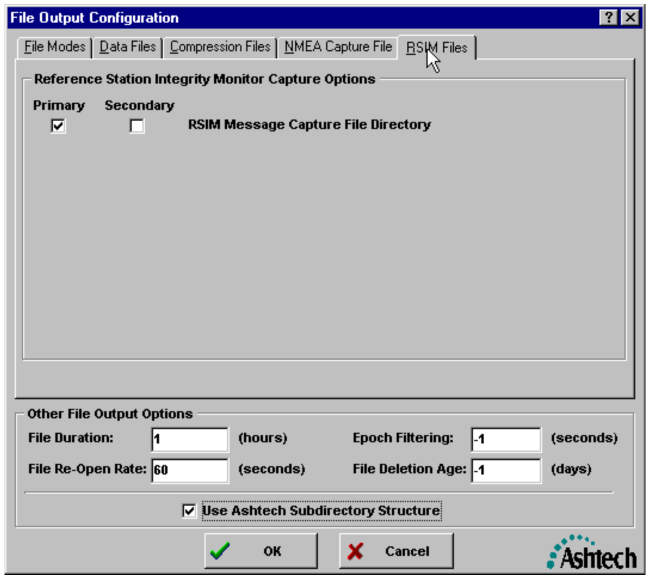


Figure 3.11 File Output Configuration Window 5

To move between each of these three screens simply click on the tab of the screen you wish to go to. It is important to note that there is a close relationship between the Data Files screen and the Compression Files screen. Files cannot be compressed unless they have been first enabled in the Data Files tab.

The following lists the types of parameters (not an all-inclusive list) that can be set through the Output Files Configuration window:

- File Output/Naming Modes
- Primary data file output directory;
- Secondary data file output directory;
- Primary compression file output directory;
- Secondary compression file output directory;
- RINEX file outputs
- RINEX header information;
- Which output data files to store and in which directories;
- Which compression file to create and in which directories;
- Whether or not to use the Ashtech automatic sub-directory creation feature;
- The duration of data collection *sessions*;
- The rate at which the FAT (that is, the File Allocation Table) is updated;
- The epoch filtering rate; and
- The age of files to be automatically deleted from the system.

3.1.4.1 Configuration | Output Files / Ashtech Subdirectories

The "Use Ashtech Subdirectory Structure" checkbox is located at the bottom of the "File Output Configuration" window and controls the automatic subdirectory creation feature. When enabled, this feature will automatically create new subdirectories for file data management purposes and provide you with completely automated data management. When the Automatic Sub-directory feature is enabled, new subdirectories are created at the beginning of each day and at the beginning of each month. For example, if your primary directory is `C:\PRI_DIR`, and you are logging data on Nov 16, 1997, then the your data will automatically be stored in the following directory structure:

C:\PRI_DIR\Nov97\Day16\

If you're logging data on March 21, 1998 and your secondary directory is `D:\SEC_DIR`, then your data will automatically be stored in the following directory structure:

D:\SEC_DIR\Mar98\Day21\

The automatic subdirectory feature conforms to the following structure:

mmmYY\DAY*dd*

where, mmm is the 3-character month of the year

YY is the last two digits of the year; and

dd is the day of the current month

Please note that when this subdirectory feature is not enabled (box unchecked), no automatic subdirectory creation will occur. That is, all data files will be stored directly in the user-specified Primary and Secondary directories. For example, if you specify `C:\PRI_DIR` as the Primary directory, then data will be stored in `C:\PRI_DIR`.

See the Appendices for complete details on subdirectory naming.

3.1.4.2 Configuration | File Outputs / File Modes

When the "File Modes" tab is selected, the File Output Configuration window will have an appearance similar to Figure 3.12.

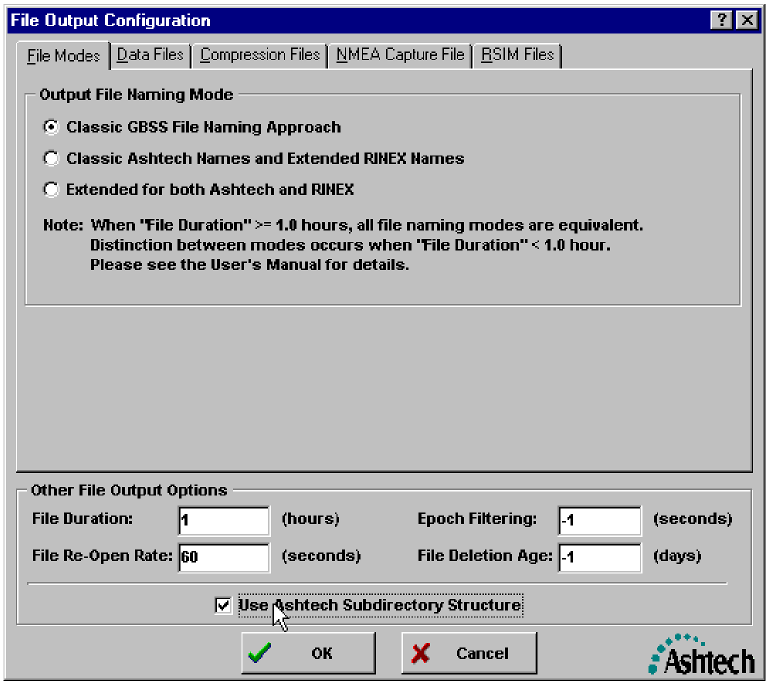


Figure 3.12 File Output Configuration Window 6

Notice in the sample window provided, there are three file-naming modes. Most GBSS users will not see any difference between the file naming modes. This is because most GBSS users desire files of 1.0, or more, duration. That is, most GBSS users set the File Duration parameter (see Section 3.1.4.10) to 1.0 or larger.

If you need to have files of lesser duration (for example, it is desired to close out a data logging period and fire the post-download commands every 30 minutes), then you must decide upon the file naming approach that will be used. Basically, the problem boils down to the idea of independent file names that are human readable. For example, classic Ashtech file names follow an 8.3 name format:

tnnnnsyy.ddd

where,	t	is the file type (B, E, S, or T),
	nnnn	is the 4-character station name,
	s	is the session code,
	yy	is the last two digits of the year, and
	ddd	is the day of the year.

In this classic form, the session code is usually a letter. Under this approach, and with a File Duration of 1.0 hour, the session code can start at 'A' (that is, for the 0 hour of the day), increment to 'B' on the next session (that is, for hour 1 of the day), and continue with each session until 'X' is reached (that is, for hour 23 of the day). However, when the File Duration is less than 1.0 hour, this form of human-readable/discernible file names can no longer be used. Therefore, we have provided three options:

1. Classic GBSS names
2. Classic Ashtech names and Extended RINEX names

3. Extended names for both Ashtech and RINEX

As can be seen, there are two choices containing “classic” naming and two choices where “extended” naming is available.

Under the “classic” naming form, and with the File Duration less than 1 hour, the normal file naming convention does not apply in that the ‘s’ (that is, session), and ‘ddd’ (that is, day of year) codes become incrementors. At the start of logging, the session code ‘s’ becomes ‘A’ and ‘ddd’ becomes ‘000’. As the File Duration is reached, the files are closed and new ones are opened with the ‘s’ set to ‘A’ and the ‘ddd’ becomes ‘001’. With each File Duration, this process continues until ‘s’ is set to ‘A’ and ‘ddd’ is set to ‘999’. At this point, the increment causes a rollover, where ‘s’ is set to ‘B’ and ‘ddd’ is reset to ‘000’. This rollover process continues until ‘s’ reaches ‘Z’ and ‘ddd’ reaches ‘999’. At this point, each time ‘ddd’ reaches ‘999’ it is simply reset to ‘000’.

Under the “extended” naming form, and with the File Duration less than 1 hour, the session code ‘s’ becomes a three-character field. It has the form *hmm*, where *h* is the hour code (that is, ‘A’ to ‘X’) and *mm* represents the start minute of the hour. Because the File Duration parameter has a lower limit of 0.05 (that is, 3 minutes), independent file names will be generated. See Appendix F for a complete description of the extended RINEX naming convention.

Please note that there are both advantages and disadvantages of each mode. For example, many processing programs expect an 8.3 file name format (and do not support the 10.3 format created in the “extended” naming modes). However, it is difficult, using a directory listing only, to determine the approximate start times of “classic” file names generated when the File Duration is less than 1.0.

In summary, most users will not need to worry about the file naming approach (that is, those users that set the File Duration parameter to 1.0 or greater). Those users that need smaller File Durations must weight the naming options. For details on file naming conventions, please see Appendix A.

3.1.4.3 Configuration | File Outputs / Primary Output Path

The “Primary Output Path” Edit field is used to specify the top-level primary directory where data files will be stored. This directory specifies where files will be stored when the file’s “Primary” checkbox of the “File Output Selection” is checked (see Section 3.1.4.5).

To change the directory, you can manually edit the output path or use the browse feature. To use the browse feature, click on the **BROWSE** button to the right of the text “Primary Output Path“. On selecting this button, you will be provided with a window that is capable of navigating over the entire set of directories of your computer. Use this window to select the desired primary output directory. If the desired directory does not exist, simply use the New Folder button in the upper right corner of the directory selection window to create it.

Once you have identified a Primary Output Path, then any data type that has the Primary field checked alongside it will be stored in that directory. Consider the example shown in Figure 3.13.

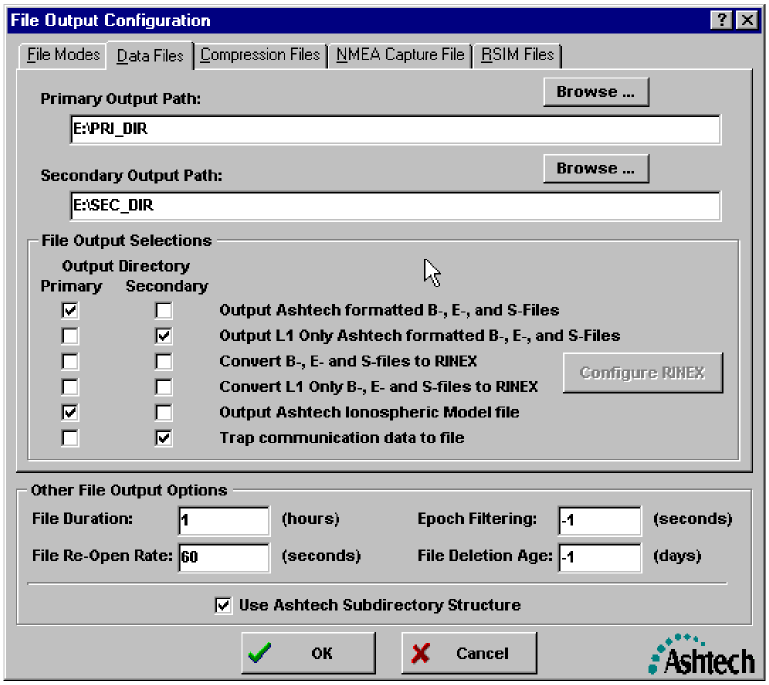


Figure 3.13 File Output Configuration Window 7

The above example displays a configuration where the primary directory is `E:\PRI_DIR` and the following file types have been selected for output into the primary directory:

- Ashtech Formatted B, E and S Files
- Ashtech Ionospheric Model File

Please note that it is permissible to use the same directory for both the primary and secondary output directories.

3.1.4.4 Configuration | File Output / Secondary Output Path

The "Secondary Output Path" edit field is used to specify the top-level secondary directory where data files will be stored. This directory indicates where files will be stored when the file's "Secondary" checkbox of the "File Output Selection" is checked (see Section 3.1.4.5).

To change the directory, you can manually edit the output path or use the browse feature. To use the browse feature, click on the **BROWSE** button to the right of the text "Secondary Output Path". On selecting this button, you will be provided with a window that is capable of navigating over the entire set of directories of your computer. Use that window to select the desired secondary output directory. If the desired directory does not exist, simply use the New Folder button in the upper right corner of the directory selection window to create it.

Once you have identified a Secondary Output Path, then any data type that has the secondary field checked alongside it will be stored in that directory. In the example of Section 3.1.4.3 the Secondary Output Path is `E:\SEC_DIR` and the following files types have been selected for output to the secondary directory:

- L1 only Ashtech Formatted B, E and S Files
- Trap communication data to file

Please note that it is permissible to use the same directory for both the primary and secondary output directories.

3.1.4.5 Configuration | File Output / File Output Selections

The File Output Selections area of the Data Files screen is used to specify which data files will be created and whether they will be stored in the Primary or Secondary Output paths. The File Output Selections is composed of two columns of checkboxes. The left-most column of checkboxes is for the Primary Output Path, and the right-most column of checkboxes is for the Secondary Output Path. These items are labeled as follows:

- Output Ashtech Formatted B-, E-, and S-Files
- Output L1 Only Ashtech Formatted B-, E-, and S-Files
- Convert B-, E- and S-files to RINEX
- Convert L1 Only B-, E- and S-Files to RINEX
- Output Ashtech Ionospheric Model File
- Trap communication data to file

The left-most checkbox of each is labeled with a heading of "Primary" and the right-most checkbox is labeled with "Secondary". When a check is placed into the Primary column, GBSS will output that file to the Primary Output Path. When a check is placed in the Secondary column, GBSS will output that file to the Secondary Output Path. Please note that no file can be simultaneously output to the primary and secondary directory (GBSS protects the user from setting this illegal combination).

The format of the B-, E- and S-Files will depend entirely upon the connected Ashtech receiver. Please see the Ashtech GPS receiver manual for the structure of your B, E and S files. For example, see the Ashtech Continuous Geodetic Reference Station (CGRS) manual for information on the dual-frequency data files or see the Ashtech Super CA manual for information on the single-frequency data files. Please note that if your GPS receiver is of dual-frequency type, then files output when one of the "Output Ashtech Formatted B-, E-, and S-Files" checkboxes is checked, then the output B, E and S files will be dual-frequency. Likewise, when connected to a single frequency receiver, then a check in the same checkboxes will cause GBSS to output L1 only B-, E-, and S-files.

GBSS can create the L1 Only versions of the B-, E-, and S-Files when connected to a dual-frequency receiver or a single-frequency receiver. This data is created from the L1 C/A observations reported by the receiver. These L1-only files can be created regardless of the receiver type. Please note that GBSS allows you to simultaneously create dual-frequency and single-frequency GPS data files while connected to a single dual-frequency GPS receiver.

It is also important to note that if GBSS is connected to a single-frequency receiver, only single-frequency files will be created. In this case, if both the Ashtech formatted B-, E- and S-files and the L1-Only Ashtech Formatted B-, E- and S-files checkboxes are checked, then you will be creating two identical single-frequency data sets. Conversely, if you are connected to a dual-frequency receiver and both the Ashtech formatted B-, E- and S-files and the L1-Only Ashtech Formatted B-, E-, and S-files checkboxes are checked, then you will be creating a dual-frequency file set and a single-frequency file set.

The format of standard RINEX files is defined in the paper entitled "*RINEX: The Receiver Independent Exchange Format Version 2"*; Dr. Werner Gurtner, University of Berne, revised in July of 1998. There is also now an extended RINEX format, version 2.2, which handles files of less than one-hour duration. These formats are described in Appendix F. As noted above, GBSS gives you the option of using either of these formats.

Please note that the RINEX files can only be created if the corresponding B-, E- and S-files are being created. For instance, if you are connected to a dual-frequency receiver and you wish to create dual and single-frequency RINEX files, then you must enable the following checkboxes

- Output Ashtech Formatted B-, E-, and S-Files
- Output L1 Only Ashtech Formatted B-, E-, and S-Files
- Convert B-, E- and S-files to RINEX
- Convert L1 Only B-, E- and S-Files to RINEX

The corresponding Ashtech formats must be created first in order for their RINEX counterparts to be created. If you do not wish to keep the Ashtech formatted files you can have GBSS automatically delete them through the Post-Session Command feature (see Section 3.1.8)

The Ionospheric Model file will only be created when GBSS is in active mode. This is because the ION data must be requested from the receiver through a command and the passive mode does not permit commands to the receiver. The Ionospheric Model file is a binary disk file which exactly replicates that which is described in your receiver manual for the $PASHR,ION, message, less the "$PASHR,ION," header and checksum information: that is, the file is exactly 74 bytes long.

The “Trap” file records all of the bytes that enter into GBSS via the communication port (that is, a capture of ALL raw communication activity between the GPS receiver and GBSS). In other words, the Trap File is simply a byte-for-byte copy of all of the data received over the communication port (before it is interpreted by GBSS).

The Trap File is an extremely powerful feature that enables you to create multiple files for the same time period by “recycling” them through GBSS. Using the Trap File feature, each of these files from the same period can have *different recording intervals.* For example, you can set the fundamental data rate of the receiver to 1 second and enable the creation of a Trap File (see Section 3.1.4.5). This Trap File will open and close in accordance with the File Duration parameter as is done with all other data files, but it will store the raw data stream output by the receiver. You can then use the Epoch Thinning feature within GBSS (see Section 3.1.4.12) to thin this raw data stream to 30 seconds for the creation of your Ashtech formatted files and your RINEX formatted files. You will then have formatted Ashtech and formatted RINEX files at a 30-second epoch interval, but you will have a Trap File at a 1-second interval. These 1-second Trap Files can then be used in three different ways:

- Automated Playback mode
- Manual Playback mode
- Manual Simulation mode

In Automated Playback Mode, GBSS can be configured to automatically create files for the same time period with different epoch intervals. This is because GBSS.EXE can also be run in command-line mode. One can configure GBSS to play back the Trap File through a second copy of GBSS, having a different epoch filtering, to create output data files with different epoch intervals (see Section 3.1.8.4.1). With this feature, for example, one could effectively create data files of 30, 15, 10, 5, 3, and 1-second epochs, each placed in their own directory structures.

Please note that the Trap File bytes are written before GBSS has a chance to interpret them. However, the trap feature has been designed to ensure that the epoch data of a Trap File coincides with the epoch in the associated B-, E- and S-Files.

Trap Files can be concatenated through a simple MS-DOS command. For example, suppose that it is desired to playback three Trap Files from a receiver on a given day. Further, suppose that those three Trap Files were named TXYZ_A97.035, TXYZ_B97.035, and TXYZ_C97.035. Issuing the following DOS command can create a single Trap File:

COPY TXYZ_A97.035 /B + TXYZ_B97.035 /B + TXYZ_C97.035 /B TRAPDATA

The file TRAPDATA will contain the concatenation of the three files. This TRAPDATA file can now be played back through GBSS to create the desired GPS data files.

3.1.4.6 Configuration | Output Files / Primary Compression Directory

The “Primary File Compression Path” edit field is used to specify the top-level primary directory where compressed versions of data files will be stored. This directory indicates where files will be stored when the file’s “Primary” checkbox of the “File Compression Selections” is checked (see Section 3.1.4.8). Please note that the primary compression directory should not be confused with the primary output directory. GBSS allows you to specify separate directory paths for the primary output directory and for the primary compression directory. For example, you might specify the following:

Primary output directory of `C:\PRI_DIR\`
Primary compression directory of `D:\PRI_COMP\`

The primary output directory and the primary compression directory are related only in that the creation of a certain file needs to be enabled in the primary (or secondary) output directory before it can be compressed in the primary (or secondary) compression directory.

To change the primary compression directory, you can manually edit the output path or use the browse feature. To use the browse feature, click on the **BROWSE** button to the right of the text “Primary File Compression Path “. Upon selecting this button, you will be provided with a window that is capable of navigating over the entire set of directories of your computer. Use that window to select the desired primary compression directory. If the desired directory does not exist, simply use the New Folder button in the upper right corner of the directory selection window to create it.

Please note that it is permissible to use the same directory for both the primary and secondary compression directories.

3.1.4.7 Configuration | Output Files / Secondary Compression Directory

The "Secondary File Compression Path" edit field is used to specify the top-level secondary directory where compressed versions of the data files will be stored. This directory indicates where files will be stored when the file's "Secondary" checkbox of the "File Compression Selections" is checked (see Section 3.1.4.8). Please note that the secondary compression directory should not be confused with the secondary output directory. GBSS allows you to specify separate directory paths for the secondary output directory and for the secondary compression directory. For example, you might specify the following:

Secondary output directory of `C:\SEC_DIR\`

Secondary compression directory of `D:\SEC_COMP\`

The secondary output directory and the secondary compression directory are related only in that the creation of a file needs to be enabled in the secondary (or primary) output directory before it can be compressed in the secondary (or primary) compression directory.

To change the secondary compression directory, you can manually edit the output path or use the browse feature. To use the browse feature, click on the **BROWSE** button to the right of the text "Secondary File Compression Path ". Upon selecting this button, you will be provided with a window that is capable of navigating over the entire set of directories of your computer. Use that window to select the desired secondary compression directory. If the desired directory does not exist, simply use the New Folder button in the upper right corner of the directory selection window to create it.

Please note that it is permissible to use the same directory for both the primary and secondary compression directories.

3.1.4.8 Configuration | Output Files / File Compression Selections

The Output Files Configuration menu is used to specify if or where selected compression files will be stored. This part of the menu consists of two checkboxes per item. These items are labeled as follows:

- Compress Ashtech Formatted B-, E-, and S-Files
- Compress L1 Only Ashtech Formatted B-, E-, and S-Files
- Compress RINEX Data
- Compress L1 Only RINEX Data
- Compress Ashtech Ionospheric Model File
- Compress communication "Trap" file

The left-most checkbox of each is labeled with a heading of "Primary" and the right-most checkbox is labeled with "Secondary". When a check is placed into the Primary column, GBSS will add that file into the compression file placed into the primary compression directory. When a check is placed in the Secondary column, GBSS will add that file into the compression file placed into the secondary compression directory. Please note that a file cannot be compressed and output to the primary and secondary compression directories simultaneously (GBSS protects you from setting this illegal combination).

The checkboxes of the File Compression Selections section are only available if their corresponding files have been enabled in either the primary or secondary output directories. In other words, if you have not enabled a particular file to be created into the primary or secondary data directory structures, then that particular file will NOT be available for compression into the primary or secondary compression directories. For example, the "Compress Ashtech formatted B-, E-, and S-File" checkboxes will be disabled when neither of the "Output Ashtech formatted B-, E-, and S-File" checkboxes is selected.

All of the files selected for compression into the Primary Compression Directory will be stored into a single "primary" file for the current *session*. Likewise, all of the files selected for compression into the Secondary Compression Directory will be stored into a single "secondary" file for the current *session*.

Compression file names are of the following form when file duration is 1 hour or greater:

syyddd.ZIP

where	s	is the *session* code,
	yy	is the last two digits of the year, and
	ddd	is the day of the year.

Under the "extended" naming form, and with the File Duration less than 1 hour, the session code 's' becomes a three-character field. It has the form *hmm*, where *h* is the hour code (that is, 'A' to 'X') and *mm* represents the start minute of the hour. Because the File Duration parameter has a lower limit of 0.05 (that is, 3 minutes), independent file names will be generated.

Like the data file names, the compression file names depend upon the "File Duration" parameter (see Section 3.1.4.10) and the current corrected CPU GPS time (see Section 3.1.7). Unlike the data files, however, the compressed versions of these files are created AFTER the associated data files are CLOSED. No attempt is made to compress a file until after its File Duration period has passed. As such, depending upon the speed of your computer and its processing load, compression of recently closed *session* files may not complete for quite some time after the data files for that *session* were closed.

The appendices provide a complete description of the file naming conventions used by GBSS.

3.1.4.8.1 Configuration | Output Files / Selecting PKZIP or PKZIPC Compression

GBSS is designed to call PKZIP or PKZIPC (from PKWARE Inc.) to perform the compression of data files. GBSS does no checking to ensure that PKZIP 2.04G or PKZIPC.exe (from the PKZIP 4.5 Suite for Windows) is installed onto your computer. Nor does GBSS ensure that PKZIP or PKZIPC are callable (that is, it does not check to ensure that PKZIP or PKZIPC can actually run). As such, GBSS will still permit the selection of the compression options even though PKZIP and PKZIPC cannot be called. IT IS UP TO YOU TO INSTALL PKZIP OR PKZIPC ONTO YOUR COMPUTER AND ADD PKZIP OR PKZIPC TO YOUR PATH.

To configure which form of PKZIP (that is, PKZIP 2.04g or PKZIPC.exe) that GBSS will use, press the button labeled "Config. Compression …". Upon pressing this button, a window similar to Figure 3.14 is displayed.

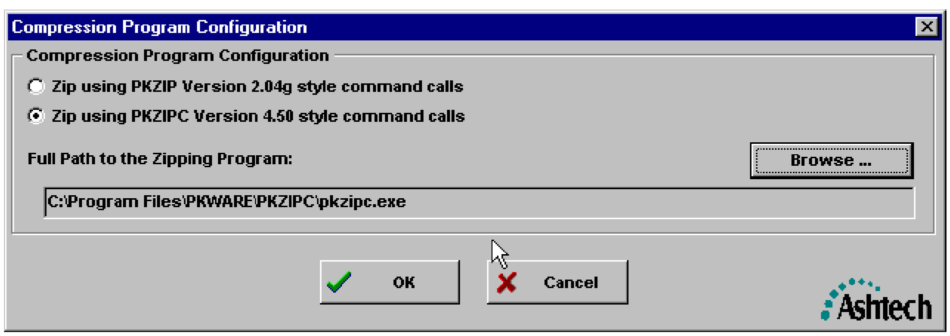


Figure 3.14 Compression Program Configuration Window

Through this window, you can select which compression program will be called and the path where that program is located. Please note that it is essential that the correct form be selected.

Furthermore, PKZIPC supports long file names, where PKZIP 2.04g does not. If you select a GBSS file-naming mode that supports long file names (see Section 3.1.4.2), you will need to use PKZIPC instead of PKZIP 2.04g.

3.1.4.9 Configuration | Output Files / NMEA Capture File

The NMEA Capture screen is accessed through the File Output Configuration window. Once in the File Output screen, simply click the mouse pointer on the NMEA Capture File tab to open the NMEA Capture parameters. The NMEA capture parameters allow you to capture up to 21 different NMEA messages into log files. NMEA messages are industry-standard messages that contain a wide variety of information. The NMEA capture feature is extremely useful for capturing data from auxiliary sensors such as meteorological stations and tilt meters. The output from these sensors is stored in the NMEA XDR message.

NMEA log files will be opened and closed on the same interval that you have specified for the primary output directory files. It is important to note that enabling the "capture" of any NMEA messages to a file does not cause GBSS to request these messages from the GPS receiver. That is, it is up to you to command the receiver (either through GBSS Terminal Window commands, File Upload commands, or through the front panel of the GPS receiver) to send the NMEA messages to GBSS.

3.1.4.9.1 NMEA Capture File Directory

The NMEA Message Capture File Directory allows you to select whether or not you want to activate the NMEA log files feature, and if so, where the resultant files will be stored. NMEA files can be stored in either the primary or the secondary output directory. To activate NMEA messages, use the mouse to check either the primary or secondary directory checkbox. Please note that if neither one of these fields are checked, no NMEA files will be created. Also note that only one of these fields can be checked at any given time.

It is important to stress that enabling the "capture" of any NMEA messages to a file does not cause GBSS to request these messages from the GPS receiver. That is, it is up to the operator to command the receiver (either through GBSS Terminal Window commands, File Upload commands, or through the front panel of the GPS receiver) to send the NMEA messages to GBSS.

3.1.4.9.2 Write All Received NMEA Messages to Capture File

This field allows you to enable the capture of all NMEA messages with the click of a mouse button. When this field is checked, checks will automatically appear next to all of the individual NMEA messages in the "Write selected NMEA messages to capture file" area of the menu.

It is important to stress that enabling the "capture" of any NMEA messages to a file does not cause GBSS to request these messages from the GPS receiver. That is, it is up to the operator to command the receiver (either through GBSS Terminal Window commands, File Upload commands, or through the front panel of the GPS receiver) to send the NMEA messages to GBSS.

3.1.4.9.3 Write Selected NMEA Messages to Capture File

This "Write selected NMEA messages to log file" field allows you to enable or disable the capture of 21 individual NMEA messages. These messages are: GLL, APA, DAL, GRS, SAT, GXP, ALM, GSA, UTM, GGA, MSV, GSV, VTS, VTG, XTE, TTT, XDR, GSN, BWC, RRE and POS. The capture of an individual NMEA message will be enabled if the box to the immediate left of it has been checked. If the box to the left of it has not been checked, then the message will not be captured by GBSS. The "Write all received NMEA messages to capture file" field immediately above this field allows you to select all 21 NMEA messages with the click of a mouse button.

It is important to stress that enabling the "capture" of any NMEA messages to a file does not cause GBSS to request these messages from the GPS receiver. That is, it is up to you to command the receiver (either through GBSS Terminal Window commands, File Upload commands, or through the front panel of the GPS receiver) to send the NMEA messages to GBSS.

3.1.4.10 Configuration | Output Files / File Duration

The "Other File Output Options" area at the bottom of the File Output Configuration window contains five user-configurable parameters controlling various aspects of file creation. These fields are as follows:

- File Duration (hours)
- File Re-Open Rate (seconds)
- Epoch Filtering (seconds)
- File Deletion Age (days)
- Use Ashtech Subdirectory Structure (On/OFF)

Please note that the "Other File Output Options" fields will appear at the bottom of the Compression Files tab, the NMEA Capture File tab as well as the Data Files tab. Changes to any of the fields in these five fields affect all three tabbed areas equally. That is, these "Other File Output Options" are global file-creation parameters

The File Duration parameter specifies the duration, in hours, of the files logged. A value of -1 disables the File Duration option. A disabled File Duration implies that the files are closed only after GBSS is disconnected from the receiver. When the File Duration is enabled (that is, not –1) and the File Duration is reached, GBSS automatically closes all files currently being recorded and then automatically opens new files (such as the B-, E-, S-, RINEX, Trap Files etc.). Each time the output files reach their "duration" (based upon the GPS time embedded in the data received from the receiver) they are closed and the files for the next *session* are opened.

The computation for file closure is based upon the GPS time obtained from the received data. That is, if the File Duration is one hour, file closures will occur on the GPS hour boundary (that is, the first set of files may not contain one whole hour of data).

The range of acceptable values for this field are -1 and 0.05 … 84 (hours). Please note that GBSS will accept values for the File

Duration that are greater than zero but less than 1, but these values impose the special file naming described in Section 3.1.4.2. During initial installation, setup, and/or experimentation, values in the range of 0.05 to 1.0 can be used to facilitate testing (especially the Post-Session Commands described in Section 3.1.8) in that they will cause completion of *sessions* more frequently.

In GBSS, there are two uses of the term "*session*". In this manual, we refer to "*sessions*" and "*logging sessions*" as independent concepts. Both are related to a period over which data are collected. The term "*session*" (by itself) is related to the File Duration described in this section. In this use, we recognize that files are closed at the specified duration and the session codes used to name the files (see Appendix A) are based upon the "Corrected CPU GPS Time" (see Section 3.1.7). The other use (that is, "*logging sessions*") is used to describe the periods in which data are actually recorded into files (see Section 3.1.5) – independent of the file naming and closure mechanisms.

3.1.4.11 Configuration | File Output / File Re-Open Rate

The File Re-Open Rate parameter specifies how often the File Allocation Table (FAT) is updated. The FAT, which is disk resident, contains information describing the content of the file to the operating system. Among this information is the length of the file. When a fatal system error occurs, such as a system or power failure, the operating system is not given a chance to update the FAT. Unfortunately, the FAT is not normally updated until the file is closed. The file re-open rate specifies how often the data files will be closed and then re-opened (in append mode), thereby updating the FAT with each re-open. In this way, the possible data loss is limited to the amount of data collected since the most recent FAT update.

One must be cautioned that the re-open rate does affect the performance of your computer. That is, the faster the re-open rate, the more computer resources (such as CPU time and disk access time) required by GBSS. Specifying too fast a re-open rate can degrade system performance and possibly cause loss of data over the RS-232 ports (which the base station software will report). It is recommended that the File Re-Open Rate be set to 60 seconds. If a faster rate is desired, it is recommended that the desired rate be tested ON THE TARGET COMPUTER, in its operational configuration, before it is selected as a permanent setting.

The range of acceptable values for this field are -1 and 1 … 3600 (seconds). A value of -1 disables the File Re-Open Rate option (that is, the FAT is updated when the file is closed at the end of the *session*). The choice of 0 (zero) is expressly disallowed.

3.1.4.12 Configuration | Output Files / Epoch Filtering

The Epoch Filtering parameter specifies the rate at which epochs will be stored in the following output files:

- Dual-frequency Ashtech format (GPS)
- Single-frequency Ashtech format (GPS)
- Dual-frequency RINEX format (GPS)
- Single-frequency RINEX format (GPS)
- Dual-Frequency Ashtech format (GPS/GLONASS)
- Single-frequency Ashtech format (GPS/GLONASS)

Please note that the Epoch Filtering feature has no impact on data stored to the Trap File.

Epoch filtering allows GBSS to accept the raw data stream coming in over the RS-232 port, write that raw data to the Trap File, and then thin that data when writing it to the output B-Files and RINEX files. For instance if the raw data stream has been set to 1 second (see Section 3.1.2.2) and the Epoch Filtering feature is disabled (set to –1) then the above data files will contain 1-second epoch data. But, if the Epoch filtering feature has been set to 30 seconds, then the output files will contain 30-second epoch data.

It is very important to note, however, that the Trap Files will always contain the entire raw data stream coming in over the RS-232 port. *The Trap File will never be thinned through this Epoch Filtering feature.* This feature of GBSS very useful when you wish to post data with a low epoch rate, say 30 seconds, yet retain the possibility of turning the same *session* data into a higher epoch rate, say 1 second., at a later date. This can easily be done with GBSS and the epoch filtering feature combined with the Trap File feature. For example, if the epoch filtering value is set to 30.0 (seconds) and the receiver is set to output real-time data at 1 second, then GBSS will store the raw trap data unfiltered and then create 30-second epoch B-File and RINEX files. Later, you can use the playback feature of GBSS and the Trap Files to obtain Ashtech Format files and RINEX format files with true epoch intervals at up to a 1 second rate.

The range of acceptable values for this field are -1 and 0.01 ... 3600 (seconds). A value of -1 disables epoch filtering (that is, all received epochs are output to the B-File).

3.1.4.13 Configuration | Output Files / File Deletion Age

The “File Deletion Age” parameter instructs GBSS to delete files older than a given number of days. In determining if a candidate file can be deleted, GBSS uses the name of the file and the current corrected CPU GPS time (see Section 3.1.7). GBSS will only select candidate files that meet the Ashtech naming convention. Additionally, the candidate files come only from those directories specified in the primary and secondary data and compression directories, and any subdirectories automatically created by GBSS. After deleting files from any Ashtech-named subdirectory, GBSS will attempt to remove these subdirectories if they are empty.

GBSS can take up to (and not exactly) 2 days beyond the value specified for the deletion age. This is because GBSS does not delete files until at least one day after the end of the day in which the file is named. For example, suppose that we set up 1-hour *sessions*, we have a deletion age of 5 days specified, and file BMEGFA97.233 is created. GBSS does not consider the deletion of this file until day 240 (that is, day 233 does not complete until day 234 begins and GBSS adds 1 day to the 5 as described above, thus 234 + 6).

The range of acceptable values for this field are -1 and 1 ... 180 (days). A value of -1 disables the deletion of aged files.

3.1.5 Logging Sessions (Recording Periods)

GBSS can be configured to record data only during specified periods. For example, if you need only record data from 9:00 AM to 5:00 PM, Monday through Friday each week, then you can use the *Logging Sessions* feature to achieve this goal.

Before continuing, however, it is important to make a distinction between the terms “*session*” and “*logging sessions*”. In this manual, we refer to “*sessions*” and “*logging sessions*” as independent concepts. Both are related to a period over which data are collected. The term “*session*” (by itself) is related to the File Duration described in this Section 3.1.4.10. In this use, we recognize that files are closed at the specified duration and the session codes used to name the files (see Appendix A) are based upon the “Corrected CPU GPS Time” (see Section 3.1.7). The other use (that is, “*logging sessions*”), described in this section, is used to describe the periods during which data are actually recorded into files – independent of the file naming and closure mechanisms.

Upon selecting the “Configuration” option from the main menu and then selecting the “Logging Session” sub-menu option, you will be presented with a window similar to Figure 3.15.

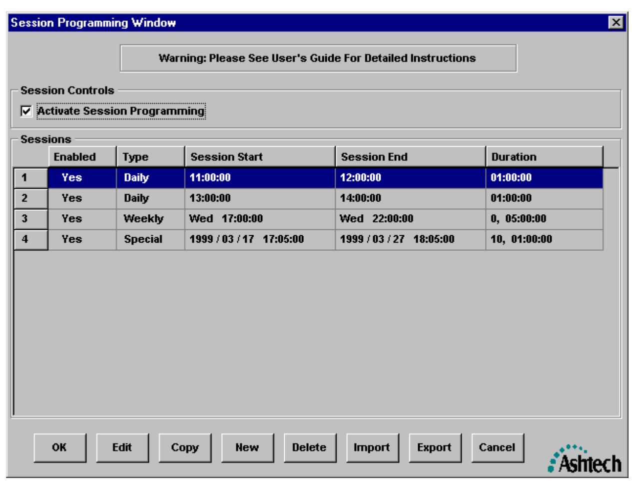


Figure 3.15 Session Programming Window

Through this window, you can create, edit, delete, copy, import, and export *logging sessions*. These *logging sessions* describe the periods during which GBSS will actually record data. When the “Activate Session Programming” checkbox is unchecked and/or there are no programmed *logging sessions*, then the *Logging Sessions* feature (also referred to as the Session Programming feature) of GBSS is said to be disabled. When **disabled**, GBSS will collect and record all data received. When the Session Programming feature is **enabled**, the periods describe by each *logging session* indicate when data will actually be written to files. It is important to note that **enabled** *logging sessions* do affect the contents of the Trap Files (see Section 3.1.4.5). That is, when **enabled** *logging sessions* are encountered, only data that falls within the *logging session* periods will be written to the Trap Files (that is, the data in the Trap Files will correspond with the associated data stored in the other output files).

There are three categories of *logging sessions*:

1. Daily;
2. Weekly; and
3. Special.

Each of these categories is defined by a start time and duration. Upon reaching the start time of a *logging session*, GBSS will begin to write the data and will stop writing data upon reaching the duration (or end time). Daily *logging sessions* are defined by a start time of day and duration. Weekly *logging sessions* are defined by a start day of week, time of day, and duration. With Daily (Weekly) *logging sessions*, you are permitted to have the start lie within one day (week) and end in the next day (week). Daily (Weekly) *logging sessions* are recurring in that they will repeat each day (week). The special *logging session* is defined by a start year, month, day, time of day and duration. These special *logging sessions* are nonrecurring, that is, they will occur only once.

The following are some important notes related to the *logging sessions*.

1. All start and end times defining the *logging sessions* are in the GPS time system. Even though we express them in a Gregorian (that is, UTC-like) format, the times governing the *logging sessions* are in the GPS system time frame. Obviously then, if your needs are expressed in a local time system, you must convert them to GPS time before entering them into GBSS.
2. GBSS will not write, to its output files, the epoch with a time equivalent to the end time of a *logging session* (unless it is overlapped by another *logging session*). That is, GBSS will record data up to, but not including, the *logging session* end time.
3. It is permissible to overlap *logging sessions*. That is, GBSS does not prohibit you from creating *logging sessions* that overlap.

4. If GBSS connects to a receiver and begins receiving data such that the current epoch time falls within an existing *logging session*, GBSS will operate as expected; that is, GBSS will record the received data until the end of the *logging session*.

In the above example window one can see that there are basically five components of each *logging session*. These are described in Table 3.1.

Table 3.1 Components of Logging Session

Column Label	Description
"Enabled"	Indicates whether or not the *logging session* is enabled. Note: you can individually enable and disable each *logging session* as well as globally enable/disable all *logging sessions* (using the "Activate Session Programming" checkbox).
"Type"	Describes the type of *logging session*: 1) Daily; 2) Weekly; or 3) Special.
"Session Start"	Describes the start of the *logging session* for the given type. The format of the column differs depending upon the *logging session* type. For Daily types, you see the hours, minutes and seconds (that is, in the HH:MM:SS format). For Weekly types, the time is expressed in day of week, hours, minutes and seconds (that is, in the ddd HH:MM:SS format). For the Special *logging sessions*, the time is expressed by year, month, day, hour, minutes and seconds (that is, in the YYYY / TT / DD HH:MM:SS format).
"Session End"	Describes the end time of the *logging session*. The format of the column differs depending upon the *logging session* type. For Daily types, you see the hours, minutes and seconds (that is, in the HH:MM:SS format). For Weekly types, the time is expressed in day of week, hours, minutes and seconds (that is, in the ddd HH:MM:SS format). For the Special *logging sessions*, the time is expressed by year, month, day, hour, minutes and seconds (that is, in the YYYY / TT / DD HH:MM:SS format). GBSS will record data up to, but not including, the *logging session* end time.
"Duration"	Describes the duration of the *logging session*. The format of the column differs depending upon the *logging session* type. For Daily types, you see the duration expressed as hours, minutes and seconds (that is, in the HH:MM:SS format). For Weekly and Special types, the duration is expressed in days, hours, minutes and seconds (that is, in the DD HH:MM:SS format). The duration for Weekly *logging sessions* are limited to 24 hours, Weekly *logging sessions* are limited to seven days, and Special *logging sessions* are limited to 35 days.

The following describes the functionality of each button of this window:

Accepts changes made to the *logging sessions* (including any imported *logging sessions*).

Begins the editing of a highlighted *logging session*. Simply use the mouse cursor to select the *logging session* to be edited and then press this button. You can also begin editing a *logging session* by double-clicking on the desired entry. The actual editing takes place using the "L*ogging Session* Editor" window described in Section 3.1.5.1.

Copies the highlighted *logging session* to the end of the set of *logging sessions* and begins editing that *logging session* by launching the "*Logging Session* Editor" window described in Section 3.1.5.1. Up to 50 *logging sessions* are supported by GBSS.

Creates a new *logging session* at the end of the list of *logging sessions*. After creating the new *logging session* you immediately begin editing the new command using the "*Logging Session* Editor" window described in Section 3.1.5.1. Up to 50 *logging sessions* are supported by GBSS.

Deletes the highlighted *logging session*.

Import Upon pressing this button, you will be presented with a file selection window in which you will enter and/or browse for the name of the file containing *logging sessions* to be imported.

Export Upon pressing this button, you will be presented with a file selection window in which you will enter and/or browse for the name of the file in which to write the exported *logging sessions*.

Cancel Cancels all changes to the *logging sessions*.

After editing each *logging session* GBSS will automatically sort them. The primary sort key used is the file type, where the order is Daily, Weekly, and then Special. The secondary sort key is the start time of the *logging session*.

3.1.5.1 Editing a Single Logging Session

Section 3.1.5 describes which actions cause the editing of a particular *logging session*. Here we seek to discuss only the particulars regarding the editing of a given *logging session*. After taking the required action to edit a *logging session*, you will be presented with a screen similar to one of the three that follow. The format displayed will depend upon the type of *logging session* being edited.

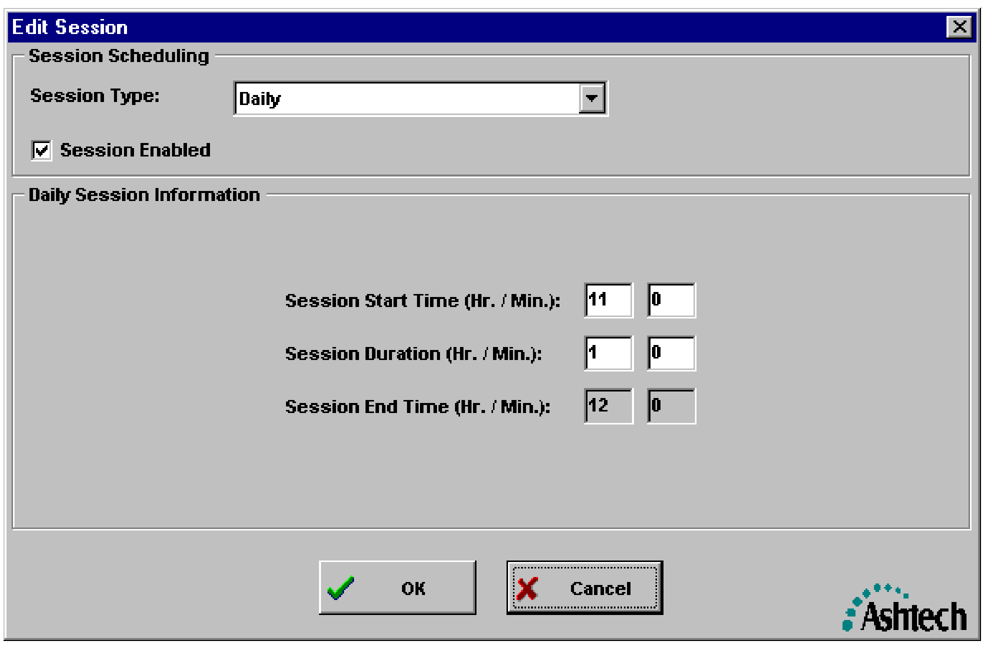


Figure 3.16 Edit Session Window 1

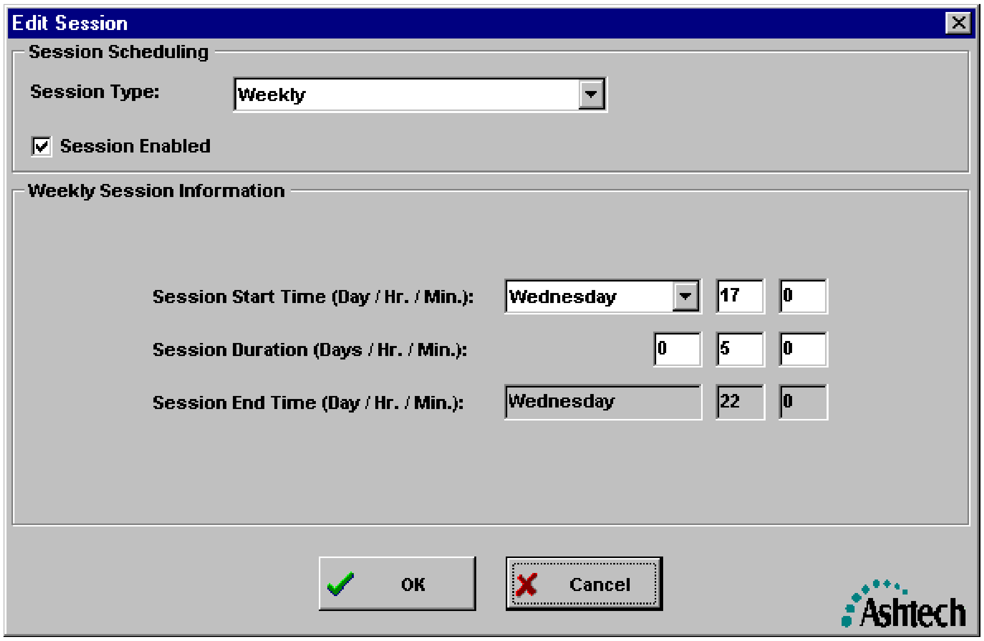


Figure 3.17 Edit Session Window 2

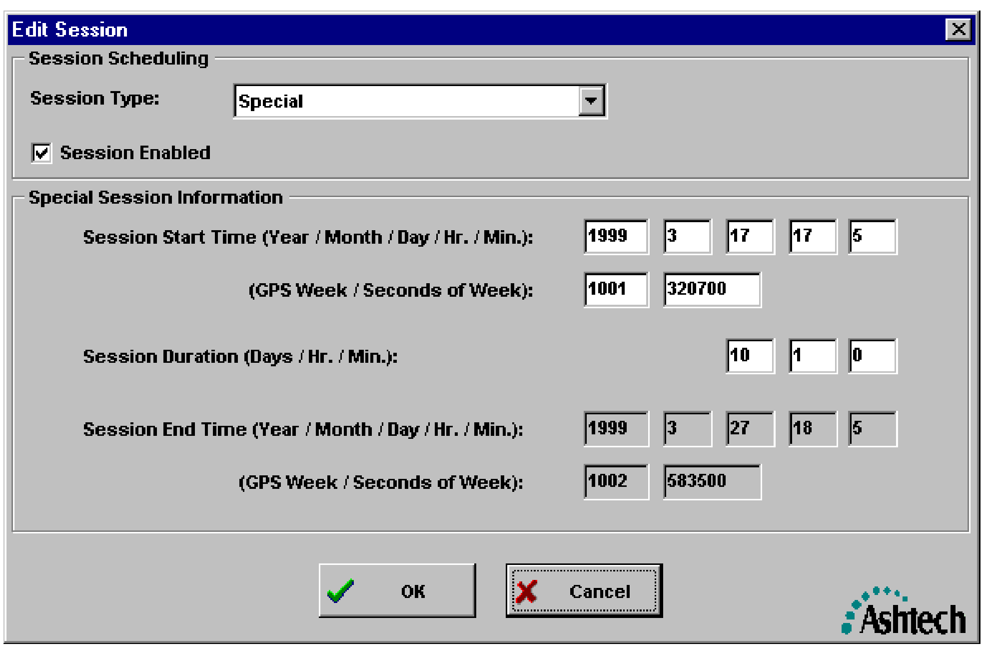


Figure 3.18 Edit Session Window 3

If the desired *logging session* type is not shown when you begin editing, simply change its type using the drop-down list box labeled "Session Type". For each type, you enable/disable the *logging session* using the checkbox labeled "Session Enabled". It should be clear that the *logging session* is enabled when the "Session Enabled" checkbox is checked and disabled otherwise.

Also for each type, you defined the start time and the duration (these components of the *logging session* are described in Section 3.1.5).

Upon completing the editing, simply press the "OK" button. If you want to abort any changes, simply press the "Cancel" button.

3.1.6 Other Setup Options

The Other Configuration Options window, Figure 3.19, allows the operator to set the following options:

Whether or not, and how verbose, diagnostic messages will be written to a log file.

Whether or not the verbose the diagnostic messages will be displayed on the screen.

Whether or not sound files, if any, will be played when alert or warning conditions arise.

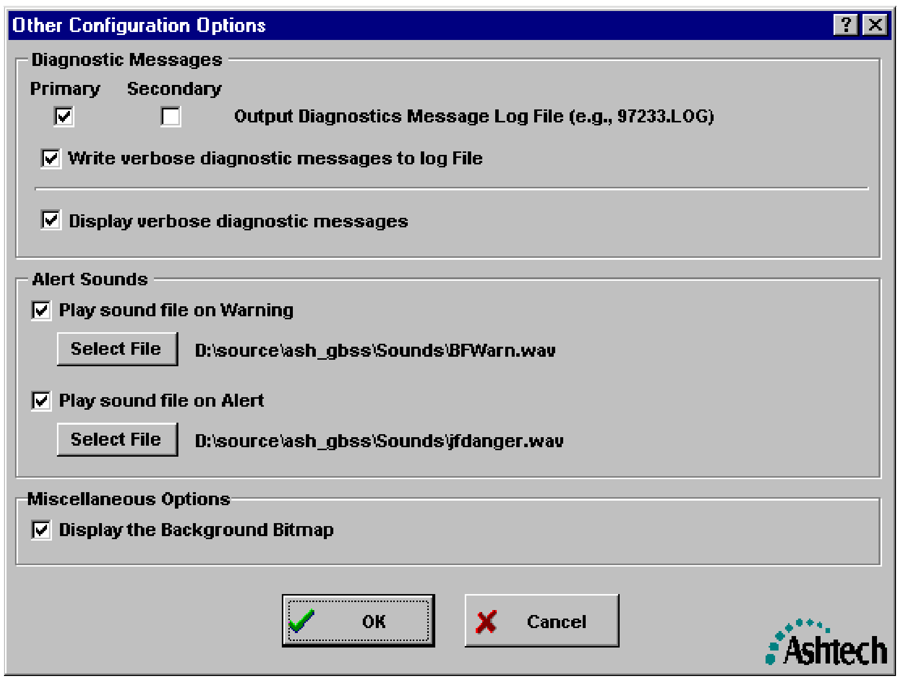


Figure 3.19 Other Configurations Options Window

3.1.6.1 Configuration | Other Options / Logging Diagnostic Messages

Within the Other Configuration Options Menu is a checkbox labeled Primary and one labeled Secondary. Jointly, these two checkboxes specify if and where a Log file will be written. When "Primary" is checked, the Log file will be written to the Primary Data Directory (see Section 3.1.4.3). When "Secondary" is checked, the Log file will be written to the Secondary Data Directory (see Section 3.1.4.4). When neither is checked, no Log file will be written. When either is checked, diagnostic messages will be written to the Log file. Some of these diagnostic messages may not be of interest to some users. For this reason you are provided a means of stripping the diagnostic messages down to only those that are critical. The checkbox labeled "Write verbose diagnostic messages to log file", when unchecked, keeps the diagnostic messages to their minimum.

Log files contain the diagnostic messages generated by GBSS for an entire GPS day. The format of the Log file name is described in the appendices.

3.1.6.2 Configuration | Other Options / Diagnostic Message Display

The "Display verbose diagnostic messages" controls the level of diagnostic messages written to the Diagnostic Messages window (see Section 4.3.4). Some users may not be interested in seeing all of the diagnostic messages. Leaving this checkbox unchecked keeps the diagnostic messages to a minimum (that is, displaying only critical messages).

The system resources (for example, CPU time) of some older computers may be severely taxed when updating the Diagnostic Messages Display window. Leaving this checkbox unchecked will help to free up some of these resources.

3.1.6.3 Configuration | Other Options / Warning and Alert Sounds

GBSS is capable of playing WAV files when an alert or warning condition arises (see Section 4.2.6 for an explanation of these conditions). The Other Configurations Menu allows you to set the sound files to be played when the alert or alarm condition arises. IT IS IMPORTANT TO NOTE THAT GBSS DOES NOT CHECK TO ENSURE THAT A) YOU HAVE A SOUND CARD AND B) THAT YOUR SOUND CARD IS CAPABLE OF PLAYING WAV FILES. If your computer does not have a sound card, it is suggested that you not attempt to play any sounds: that is, that you leave the “Play sound file on Warning” and “Play sound file on Alert” checkboxes unchecked.

To play a sound file on the Warning condition, ensure that the “Play sound file on Warning” checkbox is checked and you use the associated “Select File” to select the desired WAV file. Upon making your file selection, GBSS will test-play that selected sound file.

To play a sound file on the Alert condition, ensure that the “Play sound file on Alert” checkbox is checked and you use the associated “Select File” to select the desired WAV file. Upon making your file selection, GBSS will test-play that selected sound file.

When either a warning condition or an alert condition exists, GBSS will attempt to play the selected sound files repeatedly with about a 1.5 second timing. While GBSS will play sound files over 1.5 seconds, you are advised to keep your warning and alert sound files to less than 1.5 seconds.

3.1.7 GPS Time

Many sections of this document refer to a “corrected CPU GPS time”. This section describes how GBSS can be configured to determine this time. It can be a difficult subject to explain, but it all boils down to the following simple facts:

1. GBSS needs GPS time to properly name files.
2. Your CPU has a clock that is not normally synchronized with GPS clocks.

GBSS must open files before it connects to a receiver. To open those files, it needs to know how to name them. Those names are supposed to be based upon the current GPS time. As such, GBSS needs the value of GPS time. GBSS receives data from the receiver that contains the current GPS time. GBSS is capable of determining the delta between the Raw CPU clock and the received data. Then, when GBSS needs the current GPS time, it can apply this delta to the raw CPU time, thereby generating this “corrected CPU GPS time”.

There are three ways of determining this delta (and ultimately the corrected CPU GPS time).

1. No delta at all (GBSS is to treat the raw CPU time as GPS time).
2. A manually entered delta.

3. Let GBSS determine the delta between the time stamp of the data received and the raw CPU time.

The GPS time configuration menu allows one to specify one of these three methods. This manual refers to this corrected CPU GPS time and is applicable under all three options (that is, under option 1 the delta is simply zero). It is important to note that GBSS uses the corrected CPU GPS time for file naming purposes only and that files are closed based upon the time stamps within the actual data received from the GPS receiver.

Figure 3.20 provides a sample of the GPS Time Configuration window.

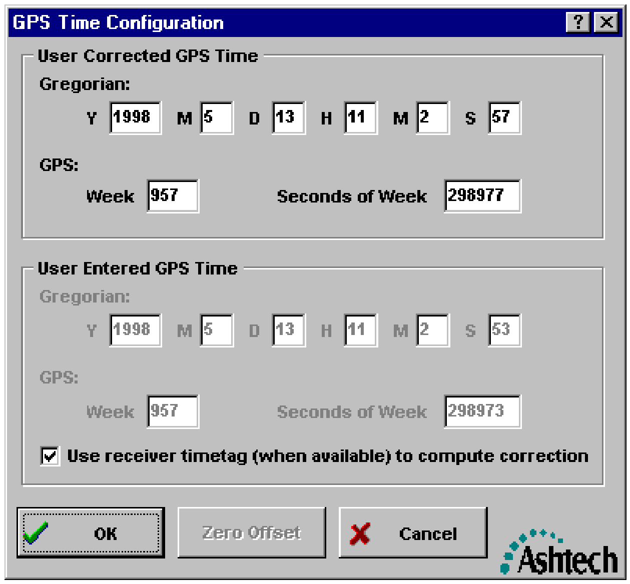


Figure 3.20 GPS Time Configuration Window

The upper portion of the screen, labeled "User Corrected GPS time" shows the current corrected CPU GPS time in both the Gregorian and GPS systems. The section labeled "User Entered GPS Time" allows a choice between Options 2 and 3. That is, when the "Use receiver timetag …" checkbox is checked, GBSS will use the time tags within the data received to compute the correction. Note that when this check box is checked, the edit fields for the User Entered GPS Time become disabled. Also disabled is the "Zero Offset" button (which will be explained shortly). When the checkbox is unchecked, the edit fields for the User Entered GPS Time become enabled. To set a desired time, enter a time that is in the very near future. When that time arrives, press the OK button and GBSS will compute the delta. The "Zero Offset" button is used to implement Option 1 described above. That is, pressing this button forces GBSS to use the RAW CPU time as the corrected CPU GPS time.

When using option 3 (that is, the "Use receiver timetag …" checkbox is checked) GBSS must collect some data to compute the offset. Upon initially selecting this checkbox or immediately after installing GBSS, the offset is not known. GBSS needs to collect some data to compute the offset. Thus, the first time you run GBSS with this feature enabled, the files created may have incorrect names. Therefore, it is suggested that after selecting the "Use receiver timetag…" option (or after initially installing GBSS), you collect data for several minutes (to allow GBSS to calculate the offset), terminate GBSS, and then restart it. This will ensure that when you start to log data operationally the files are named correctly. Failure to follow this procedure could lead to some incorrectly named files.

It is important to note that there is a caveat in using the "Use receiver time tag (when available) to compute correction" checkbox. That is, GBSS must receive both epoch data and broadcast message data to determine the GPS time. When GBSS is in its Active mode (see Section 3.1.2.1), there should not normally be a problem. However, when in the Passive mode, GBSS may not receive broadcast messages (that is, SNAV messages) and, because it is in the passive mode, GBSS cannot request the receiver to send the needed messages. When this occurs, GBSS will not be able to correctly determine the delta. This is why the manual entry of GPS time is provided.

It is also important to note that GBSS will "store" whatever options are selected from this menu. If GBSS, through this menu, is instructed to compute the delta from the time tags within the data, GBSS will continually compute and "store" this delta. In this context, “store” is meant to imply that the information will be stored as part of the GBSS configuration data. Thus, if GBSS is exited and then started at a later time, these parameters will be recalled (including the computed delta).

One final point. Some purists view GPS time as a quantity that can only be expressed as a GPS-week, seconds-of-GPS week couple. This is not our view. For easy interpretation we express GPS time in the format of UTC. Hopefully it is clear that the time is not UTC. GPS and UTC differ by leap seconds and in other small ways.

3.1.8 Post-Session Commands

Before describing the post-session command feature, it is important to make a distinction between the terms “*session*” and “*logging sessions*”. In this manual, we refer to “*sessions*” and “*logging sessions*” as independent concepts. Both are related to a period over which data are collected. The term “*session*” (by itself) is related to the File Duration described in Section 3.1.4.10. In this use, we recognize that files are closed at the specified duration and the session codes used to name the files (see Appendix A) are based upon the “Corrected CPU GPS Time” (see Section 3.1.7). The other use (that is, “*logging sessions*”) is used to describe the periods in which data are actually recorded into files (see Section 3.1.5) – independent of the file naming and closure mechanisms.

GBSS allows you to specify programs to be called at the completion of a *session* (that is, when the “File Duration” expires). Through this feature you can have GBSS pass information created from within GBSS to other programs. For example, you can have GBSS call an FTP program to distribute all of the files just collected to several Internet FTP sites. This simple feature provides you with a very powerful system integration capability that exploits programs supporting command-line parameters or scripting.

Before continuing, however, it is important to state that the Post-Session Command feature provides great flexibility and power. With this flexibility and power comes the potential to incorrectly call programs external to GBSS. This is because GBSS has no knowledge of correct vs. incorrect calls to external programs and cannot, therefore, provide any checks of correctness before the calls to these external programs are actually made. Only advanced/knowledgeable users should exploit the Post-Session Command feature. Presented in Section 3.1.8.3 of this document is an additional set of warnings regarding the Post-Session Command feature that should be reviewed.

3.1.8.1 Post-Session Commands Window

The Post-Session Command Window, Figure 3.21, is the window through which you enter, modify, order, and delete post-session commands.

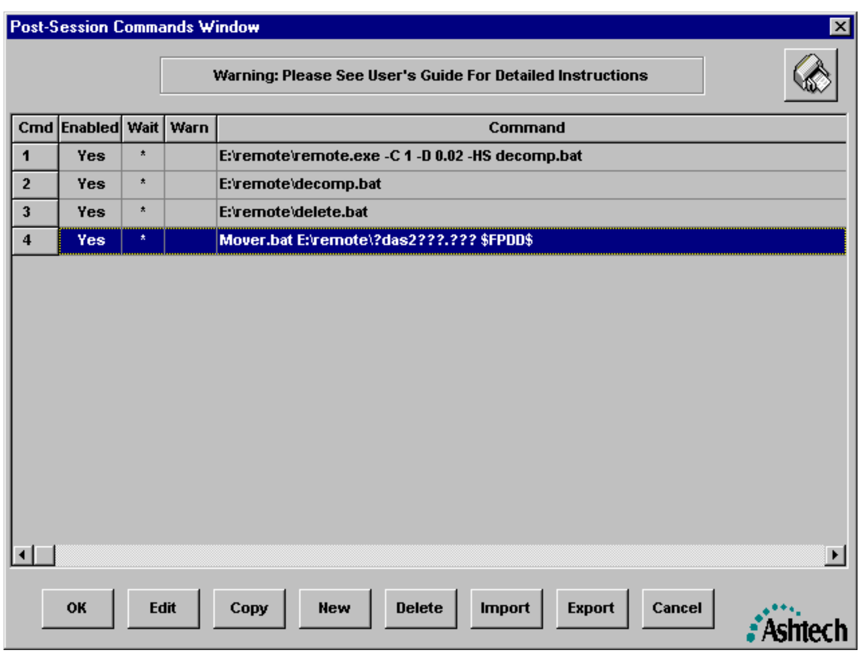


Figure 3.21 Post-Session Commands Window

This window provides the set of commands that will be launched upon completion of a *session*. The end of a *session* occurs when the “File Duration” is reached (see Section 3.1.4.10). Furthermore, this window shows the order in which the commands will be launched. Notice that each command has the components described in Table 3.2.

Table 3.2 Command Components

Column Label	Description
“Cmd”	The command number. The order of the commands specifies the order in which each post-*session* command will be launched.
“Enabled”	Indicates whether or not the command is enabled for execution at the end of the *session*.
“Wait”	Indicates whether or not the command processor of GBSS, when it is executing the post-*session* commands, is to wait for the current command to complete before launching the next command.
“Warn”	Indicates whether or not there are any detected warnings in the post-*session* command.
“Command”	The text of the post-*session* command

The following describes the functionality of each button of this window:

OK Accepts changes made to the post-session commands (including any imported commands).

Edit Begins the editing of a highlighted post-session command. Simply use the mouse cursor to select the command to be edited and then press this button. You can also begin editing a command by double-clicking on the desired command. The actual editing of a command takes place using the “Post-Session Command-Line Editor” window described in Section 3.1.8.2.

Copy Copies the highlighted post-session command to the end of the set of commands and begins editing that command by launching the “Post-Session Command-Line Editor” window described in Section 3.1.8.2. Up to 100 post-session commands are supported by GBSS.

New Creates a new command at the end of the list of post-session commands. After creating the new command you immediately begin editing the new command using the “Post-Session Command-Line Editor” window described in Section 3.1.8.2. Up to 100 post-session commands are supported by GBSS.

Delete Deletes the highlighted post-session command.

Import Upon pressing this button, you will be presented with a file selection window in which you will enter and/or browse for the name of the file containing post-session commands to be imported.

Import Upon pressing this button, you will be presented with a file selection window in which you will enter and/or browse for the name of the file in which to write the exported post-session commands.

Cancel Cancels all changes to the post-session commands.

The order of the commands can be changed through this window by selecting (with the mouse) the command to be moved and then dragging that command to the desired location within the list of commands. Again, GBSS supports up to 100 post-session commands.

The wait flag indicates whether or not the command directly following the current command will be launched immediately after launching the current command or only after the current command has completely finished its execution. That is, the command wait feature is used to enforce a dependency of one command upon another, in terms of when they are launched. The wait flag is changed through the “Post-Session Command-Line Editor” window (see Section 3.1.8.2) that is launched by selecting the desired command and pressing the Edit Button.

Whenever GBSS detects any warnings associated with a particular command-line, the column labeled “Warn” for the command contains the value “Yes”. When GBSS detects no errors with the Post-Session Command, the “Warn” entry for that command is blank. However, one should not be lulled into a false sense of security with respect to the lack of any warning indication. Specifically, GBSS can only perform limited checks on the commands. Furthermore, some checks cannot occur until the command is actually launched. Therefore, the command may contain errors that GBSS cannot detect. To determine the rationale for the warnings detected by GBSS one should highlight the offending command, press the “Edit” button and then press the “Verify” button of the displayed “Post-Session Command-Line Editor” window.

During run-time, GBSS records information about the post-session commands in the output Log file when it is enabled (see Section 3.1.6.1). This information will include the fully expanded post-session command or the rationale as to why the post-session command failed. The post-session commands will only launch if, at run-time, there are no warnings in the set of enabled post-session commands. Additionally, GBSS will terminate the set of post-session commands upon receiving an error from the operating system on any command.

If you enabled the GBSS Log file (see Section 3.1.6.1), information on all post-session commands will be written to that file. The Log file can be very helpful in testing your entered post-session commands. That is, when testing your post-session commands, it is suggested that you enable the Log file and use a simulation file (see Sections 4.6 and 4.7). The rationale for this suggestion is that the simulation file will “play back” faster than real-time and allow you to see the commands execute sooner than if you waited for the *session* to end in a live connection. Again, the log file will contain the fully expanded commands for those commands that properly executed and any errors detected for those commands that failed to launch.

3.1.8.2 Post-Session Command-Line Editor Window

The Post-Session Command-Line Edit window, Figure 3.22, allows you to edit post-session commands and perform basic checks of the commands.

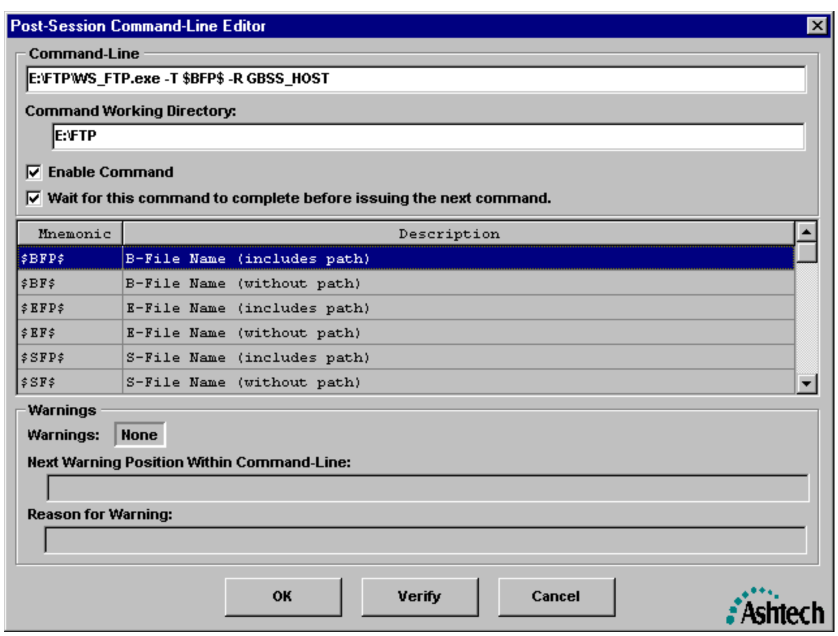


Figure 3.22 Post-Session Command-Line Edit Window

There are basically four parts of a command-line command:

1. the command-line command text,
2. the working directory of the command,
3. the command enable/disable indication, and
4. the wait-for-complete indication.

When entering the text of the command, you are free to use the GBSS command-line mnemonics. The valid mnemonics are listed in the scroll box of the window. Mnemonics are placeholders for actual values that can only be definitively determined at the completion of a *session*. For example, the B-File mnemonic, inclusive of the directory in which the file is stored, is "BFP". When the post-session command is entered, the placeholder for the B-File name and directory is entered as BFP. Just before the command is executed at the end of a *session*, GBSS substitutes the mnemonic with the actual name of the B-File, including the directory in which it is stored.

The working directory is the directory you want passed to Windows as the working directory of the command. For most calls it is the same directory as the directory in which the program you are calling is stored. In the above example, the program is stored in the directory "E:\FTP"; as such, we set this directory as the working directory. All text entered into this field for the working directory is treated verbatim. GBSS does not interpret/translate any mnemonics found in the entry of the working directory field.

The "Enable Command" checkbox is used to enable/disable selected commands. This feature is particularly useful when you are testing your post-session commands because you can disable the commands that are not directly part of your test.

The wait-for-complete indication is used to indicate whether or not GBSS will wait for the current command to complete before launching the next command in the set of post-session commands. Checking this checkbox allows you to specify the dependency of later commands upon the completion of the current command. If there are no later dependencies, then the checkbox labeled "Wait for this command to complete before issuing the next command" does not need to be checked. However, if you want to ensure that each command is launched only after the completion of the last command, make sure that the wait-for-complete checkbox is checked on each command.

The lower half of this window displays any warnings detected by GBSS as a result of pressing the OK or Verify Buttons. Again, the verification that GBSS performs on the entered commands is limited. The lack of a negative indication regarding a command does not imply that it is error free. When GBSS detects an error, it will attempt to show the location of the violation in the text box labeled "Next Warning Position Within Command-Line". The location is indicated using 3 carat symbols (that is, "^^^") placed before the problematic entry. The rationale for the warning will be displayed in the text box labeled "Reason for Warning". Figure 3.23 provides an example of the window with a warning indication:

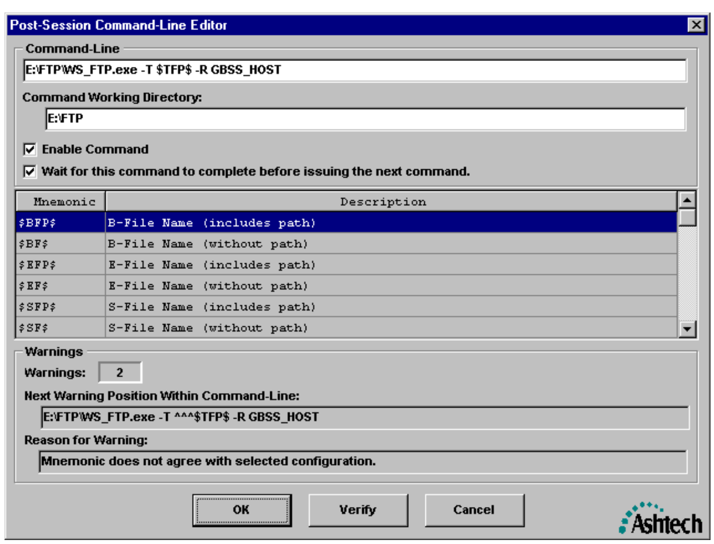


Figure 3.23 Post-Session Command-Line Editor

Notice in the above example that the position of the warning appears to be the mnemonic TFP.

Currently there are only four warnings displayed by GBSS. These are:

1. Epansion of mnemonics causes command-line length to exceed 256-character maximum.
2. Mnemonic does not agree with selected configuration.
3. Badly formed or unrecognized command-line mnemonic
4. Invalid working directory for the program

The first warning occurs when the expanded form of the command-line causes the command-line to exceed 256 characters (that is, 256 characters is the DOS maximum per command). The second warning occurs when the current configuration of GBSS conflicts with the command-line mnemonic. For example, your command-line uses a mnemonic which represents Trap Files but you have not selected Trap Files for output by GBSS in the "Configuration | Output Files" menu. The third warning occurs when GBSS does not recognize an entered mnemonic (that is, the characters between two, and including, the "$" characters). The final warning occurs when you specify a working directory that does not exist on disk.

Note that even though a command contains a warning, GBSS permits the acceptance and storage of that command. If by run-time the cause of that warning is not corrected and that command is enabled, GBSS will cancel all post-session commands until the set of enabled commands are completely warning free. Additionally, GBSS can detect other warnings/errors at run-time. When any such warnings/errors are detected, GBSS will display the rationale for the warning/error in the Diagnostic Message Window, echo that warning/error in the log file, and terminate any and all post-session commands for the current session.

Most of the mnemonics are clearly defined in the "Description" part of the mnemonic table of the Post-Session Command-Line Editor window. However, Table 3.3 describes some of the special mnemonics:

Table 3.3 Special Mnemonics

Mnemonic	Description
$$	Expands to the $ character
$\/$	Toggles On and Off the translation of the backslash character to the forward slash character. For example, suppose that the full primary directory path, without the drive, is \GPS_Data\May98\Day12". The post-session command: C:\Utl\GetFiles.exe $NDFPDD$ would translate to C:\Utl\GetFiles.exe \GPS_Data\May98\Day12 While the post-session command: C:\Utl\GetFiles.exe $\/$$NDFPDD$$\/$ would translate to C:\Utl\GetFiles.exe /GPS_Data/May98/Day12 This backslash to forward slash translation is useful when interfacing with UNIX.

Please note that mnemonics can be "chained" together to form a single command-line token. For example, assume that the current session is session 'C' on Jan 9, 1998 for site REMD, then the following post-session command:

C:\GPPS\Makeufil.exe BFP U$SITE$$S$$YY$.$DDD$

would translate to:

C:\GPPS\Makeufil.exe D:\GPS_DATA\Jan98\Day09\BREMDC98.009 UREMDC98.009

3.1.8.3 GBSS and Post-Session Commands

The purpose of this section is to provide miscellaneous information regarding the Post-Session command feature of GBSS. Before continuing with the discussion, it is important for the reader to understand what it means for a command to be "completed". The completion of a command is dependent upon the perspective from which you view the command. That is, from the perspective of the operating system (that is, Windows) a command is viewed complete when the program actually finishes. However, there are cases where the command will be completed from the GBSS perspective even though it has not completed from the perspective of Windows. For example, if the Post-Session Command-Line Editor window dialog item "Wait for this command to complete before issuing the next command" was NOT checked for a command, when GBSS launches the command it will not wait for Windows to return a “complete” indication before launching the next command. From the perspective of GBSS the command has completed. However, from the perspective of Windows, the command may still be executing.

When GBSS reaches the end of a “File Duration” it begins launching the Post-Session Commands. The command being executed, or any errors encountered while attempting to start the command, will be sent to the Diagnostic Message Window and the Log file. That is, if you encounter errors in your post-session commands, ensure that the Log file is turned on (see Section 3.1.6.1) and then run GBSS again. Any errors will be reported in the Log file and in the Diagnostic Message Window. Successful command launches will be displayed in the Post-Session Command Summary Window, in the Diagnostic Message Window, and in the Log file. The time at which the command was launched (in seconds of GPS week) will be reported to the Diagnostic Message Window and the Log file.

It is important to note that when GBSS is operating in the Simulation, Playback or Auto-Playback modes and the end of a File Duration is reached, all post-session commands will be completed, from the perspective of GBSS, before continuing the simulation/playback. Additionally, regardless of the “Wait” flag on each command, each command will be launched in a wait-for-complete mode (that is, as if the “Wait” flag was set for each command).

The following lists some miscellaneous topics and warnings related to the Post-Session Command feature:

1. GBSS will permit 100 post-session commands.
2. GBSS cannot verify the complete correctness of any Post-Session Command. As such, users should fully checkout their post-session commands and the impacts of these commands upon other components of their system before allowing the post-session commands to become part of operational environments.
3. Users should fully checkout their post-session commands before allowing them in operational environments. The checkout should include implementing the commands in the GBSS environment.
4. Command-line sequences which take more than the "File Duration" to complete (from both the perspective of GBSS and Windows) should not be used. That is, GBSS will not launch the commands of a given *session* if, from the perspective of GBSS, the commands from the previous *session* have not completed. Furthermore, users should ensure that their post-session commands (from the perspective of Windows) complete before the subsequent "File Duration" period is over. Failure to do so could cause GBSS to launch more processes than your computer can handle -- IT IS UP TO YOU, THE USER, TO MAKE SURE THAT THIS DOES NOT HAPPEN.
5. Programs that access the communications port to which GBSS is connected must not be used.
6. When the user manually terminates a *session* (either by "Disconnecting" or exiting GBSS), the post-session commands for that *session* will not be launched.
7. If there are any commands that have not completed, from the perspective of GBSS, at the manual termination of a *session* (that is, commands from the *session* just prior to the one just terminated are still running), then GBSS will automatically terminate those commands.
8. When the Auto-Playback feature of the GBSS is invoked, GBSS will wait for all post-session commands to finish (from the perspective of GBSS) before continuing the simulation into the next *session*.
9. When a *session* is terminated due to a system failure (such as a power failure), GBSS will not launch the post-session commands for the terminated *session*. Note: if GBSS is restarted within the same *session* timeframe, the files before the failure will be renamed and GBSS continues to store data in the files named for the current *session*. At the completion of this *session*, GBSS will execute the post-session commands on the files for the current *session* (that is, these will not include the renamed files). It is recommended that the user provide an Uninterruptible Power Supply (UPS) for the computer for any mission-critical uses of GBSS to prevent any problems with lost data.
10. When the post-session commands for a *session* are terminated due to a system failure (such as a power failure), GBSS will not restart those commands on the restart of GBSS. The commands will, however, be started at the completion of

the subsequent *session*. It is recommended that the user provide an Uninterruptible Power Supply (UPS) for the computer for any mission-critical uses of GBSS to prevent any problems with lost data.

11. It is recommended that you not use post-session commands with embedded spaces in the paths. For examples, many programs are automatically installed in the "\Program Files" directory. While GBSS has been tested with and does operate properly with paths containing spaces, this author has seen the Windows command processor become confused with paths containing spaces. For this reason, it is recommended that you not use paths with embedded spaces (that is, copy or move programs to paths that do not contain spaces). Additionally, you can assist the command interpreter by placing quotes around the executable program name or batch file name (including path) in the command-line text. For example, C:\Program Files\Ashtech\GBSS\Utls\AshFTPMD.exe contains a space. To assist the interpreter, enclose the entire program path in quotes: for example, "C:\Program Files\Ashtech\GBSS\Utls\AshFTPMD.exe". You **should not**, however, place quotes around the working directory of the post-session command.
12. When testing your post-session commands, it is suggested that you enable the Log file and use a simulation file (see Sections 4.6 and 4.7). The rationale for this suggestion is that the simulation file will "play back" faster than real-time and allow you to see the commands execute sooner than if you waited for the session to end in a live connection. Again, the log file will contain the fully expanded commands for those commands that properly executed and any errors detected for those commands that failed to launch.
13. MS-DOS Batch files can be called from GBSS as well. When batch files are called from within GBSS, Windows creates a command interpreter window and runs the batch file in that environment. You must, however, clearly state in the text of the command-line call that the file being executed is a batch file. You do this by appending the file type to the execution command. For example, suppose you had a batch file named "RENAMER.BAT" in the directory C:\BATCHES. Your command-line text should look something like the following:

    C:\BATCHES\RENAMER.BAT

    Notice that the ".BAT" extension has been added.
14. Intrinsic MS-DOS functions cannot be directly called from the command-line of GBSS. Intrinsic MS-DOS functions have no associated .EXE file on your hard drive. Examples of these functions are "del", "rename", "copy" and "mkdir". If you wish to use an intrinsic MS-DOS function post session, you must embed it in a batch file and execute the batch file using a post-session command.

3.1.8.4 Post-Session Command-Line Examples

In this section, several post-session command examples are presented. Many assumptions are made in these examples and will, therefore, not necessarily work verbatim in your environment. These assumptions include such things as hard drive letters and directories in which files are stored. Thus, if you wish to use the examples, you must tailor them to your computer/system environment.

3.1.8.4.1 Trap File Auto-Playback Example

The Automatic playback feature was specifically designed to support the Post-Session Command-Line feature. This is why GBSS accepts the GPS week and Seconds of GPS week for the start of the Trap File in the automatic playback mode (see Section 4.7). The Post-Session Command mnemonics GPW and GPS represent the GPS week and the Seconds of GPS week, respectively. The following shows an example call that could be entered into GBSS using the GBSS Post-Session Commands feature to invoke the Automatic Playback feature of GBSS:

    D:\SCNDGBSS\GBSS.exe –P GPW GPS TFP

When GBSS is called, it employs the configuration contained in the files GBSS.INI and GBSS.SES. To set features used during the automatic playback which are different than those normally run for GBSS, one would need to make a copy of the files GBSS.exe, GBSS.ses and GBSS.ini in a different directory such as the SCNDGBSS directory above. Again, the reason is that the configuration is stored in the files GBSS.INI and GBSS.ses. By copying the files to a new directory, you are essentially making an independent configuration of GBSS. You will then need to set the configuration of the copied GBSS using its configuration windows (that is, thereby setting the independent configuration). You do not need additional sentinel keys to run additional copies of GBSS.

For the purposes of this discussion, assume that the normal GBSS.exe and GBSS.ini files are in the C:\Program Files\Ashtech\GBSS. Further assume that you make copies of the files GBSS.exe and GBSS.ini in the directory D:\SCNDGBSS. You should then launch (that is, execute) the copy of GBSS in the D:\SCNDGBSS directory and configure the program as desired using the normal menus of GBSS. After changing the settings of the copy as desired, exit GBSS (this will save the configuration). When you initiate the Automatic Playback feature, be sure to call the one in the D:\SCNDGBSS directory. In this example, you configure your normal copy of GBSS (that is, the one in the directory C:\Program Files\Ashtech\GBSS) to call the second copy of GBSS at the completion of the *session* via the following Post-Session Command:

D:\SCNDGBSS\GBSS.exe –P GPW GPS TFP

Obviously, the working directory of this command should be D:\SCNDGBSS. This will have the primary copy of GBSS (that is, the one in the C:\Program Files\Ashtech directory) call the second copy of GBSS (that is, the one in the D:\SCNDGBSS directory), which will use its own independent configuration (that is, stored in the file D:\SCNDGBSS\GBSS.INI). Two independent copies of GBSS will be running simultaneously: one copy will be communicating with a receiver and one will be in playback mode.

You might ask why would you want GBSS to call a copy of itself? Suppose that your objective was to create 30-second epoch data and 1-second epoch data files from the same receiver and you wanted to store that data into different directories. To do this, you would have the primary copy of GBSS communicate directly with the GPS receiver requesting it to output 1-second data. Further, this copy should be set to output a Trap File. The second copy should then be configured to output the data to a different directory (via the Configuration | Output Files menu) and have its epoch filtering set to 30 seconds (see Section 3.1.4.12). Finally, you should have the primary copy of GBSS configured to call the second copy of GBSS (via Post-Session Commands) as shown in the example of this Section.

3.1.8.4.2 Directory Creation on and Transfer to an FTP Server Example

Appendix C documents a utility program supplied with GBSS with the name of ASHFTPMD.EXE. This utility program was specifically designed to create directories on an FTP server (which grants permission to do so). We have also discussed the use of WS_FTP professional, an FTP program used to transfer files to FTP servers (which grants permission to do so). In this example, we use the program ASHFTPMD to create directories on the remote FTP server, if those directories do not already exist, and then transfer GBSS collected files to those directories. Furthermore, on the target FTP server we wish to store raw Ashtech receiver file data in one directory while storing the RINEX counterparts in another directory.

We will assume that the directory in which ASHFTPMD.EXE and FTP95PRO.EXE are stored is "C:\FTP". As such the working directory of each post-session command is "C:\FTP". With these assumptions consider the following post-session commands, each of which are created on a "wait for complete" basis.

1. c:\ftp\ashftpmd.exe ftp.server.name usrid passwd /pub/ashtech/$YYYY$/MM_DD
2. c:\ftp\ashftpmd.exe ftp.server.name usrid passwd /pub/rinex/$YYYY$/MM_DD
3. c:\ftp\ftp95pro.exe –i ashtech.ini-p ash_u -s local:BFP -d ash_u:/pub/ashtech/$YYYY$/MM_DD/BF
4. c:\ftp\ftp95pro.exe –i ashtech.ini -p ash_u -s local:EFP -d ash_u:/pub/ashtech/$YYYY$/MM_DD/EF
5. c:\ftp\ftp95pro.exe –i ashtech.ini -p ash_u -s local:SFP -d ash_u:/pub/ashtech/$YYYY$/MM_DD/SF
6. c:\ftp\ftp95pro.exe –i ashtech.ini -p ash_u -s local:$ROFP$ -d ash_u:/pub/rinex/$YYYY$/MM_DD/ROF
7. c:\ftp\ftp95pro.exe –i ashtech.ini -p ash_u -s local:$RNFP$ -d ash_u:/pub/rinex/$YYYY$/MM_DD/RNF

In each post-session command, the mnemonics $YYYY$, MM, DD are used. In each case, these mnemonics represent the time of the *session* as 4-digit year, 2-digit month, and 2-digit day of the month, respectively. For example, suppose that the *session* of data occurs on December 14, 1999, then when launching the post-session commands GBSS will translate first post-session command as follows:

1) c:\ftp\ashftpmd.exe ftp.server.name usrid passwd /pub/ashtech/1999/12_14

This command will launch the program ASHFTPMD and create the directory "/pub/ashtech/1999/12_14" on the FTP server. Our intent with this command is to create a directory on the FTP server in which to store the raw Ashtech-formatted files. Again, that directory will only be created if it does not already exist on the FTP server. Of course, this assumes that the parameters passed to ASHFTPMD define the correct FTP server, user name, and password and that the FTP server grants that “user name” permission to create directories. Throughout the remainder of this example, we will assume that the FTP server grants the permissions needed and that the account information is correct.

The second command is translated by GBSS as follows:

2) c:\ftp\ashftpmd.exe ftp.server.name usrid passwd /pub/rinex/1999/12_14

This command will launch the program ASHFTPMD and create the directory “/pub/rinex/1999/12_14” on the FTP server. Our intent with this command is to create a directory in which to store the GBSS created RINEX files. Again, that directory will only be created if it does not already exist on the FTP server.

In the remaining post-session commands of this example, our intent is to use the program FTP95PRO.EXE to transfer files to the FTP server (please consult the manual for that program for details on its command-line structure and options). In each of these commands, we push the appropriate files to the directories previously created. To do this we use several mnemonics, which are described in Table 3.4.

Table 3.4 Post-Session Mnemonics

Mnemonic	Description
BFP	The name of the B-File created by GBSS, including full drive and path to that file. It is important to stress that we are referring to the name and path of the file created by GBSS and not the file itself.
BF	The name of the B-File created by GBSS, but does not include any drive or path information. It is important to stress that we are referring to the name of the file created by GBSS and not the file itself.
EFP	The name of the E-File created by GBSS, including full drive and path to that file. It is important to stress that we are referring to the name and path of the file created by GBSS and not the file itself.
EF	The name of the E-File created by GBSS, but does not include any drive or path information. It is important to stress that we are referring to the name of the file created by GBSS and not the file itself.
SFP	The name of the S-File created by GBSS, including full drive and path to that file. It is important to stress that we are referring to the name and path of the file created by GBSS and not the file itself.
SF	The name of the S-File created by GBSS, but does not include any drive or path information. It is important to stress that we are referring to the name of the file created by GBSS and not the file itself.
$ROFP$	The name of the RINEX Observation File created by GBSS, including full drive and path to that file. It is important to stress that we are referring to the name and path of the file created by GBSS and not the file itself.
ROF	The name of the RINEX Observation File created by GBSS, but does not include any drive or path information. It is important to stress that we are referring to the name of the file created by GBSS and not the file itself.
$RNFP$	The name of the RINEX Navigation File created by GBSS, including full drive and path to that file. It is important to stress that we are referring to the name and path of the file created by GBSS and not the file itself.
RNF	The name of the RINEX Navigation File created by GBSS, but does not include any drive or path information. It is important to stress that we are referring to the name of the file created by GBSS and not the file itself.

Let us suppose that the file is *session* B (RINEX *session* ‘2’) on December 14, 1999. Further, let us suppose that the Ashtech sub-directory structure has been enabled in GBSS and that the files of interest are stored in the Primary Data Directory having a root directory of “E:\GPSDATA”. The third command would be translated by GBSS as follows:

3) c:\ftp\ftp95pro.exe –i ashtech.ini -p ash_u -s local:E:\GPSDATA\DEC99\DAY14\BSITEB99.348 -d ash_u:/pub/ashtech/1999/12_14/BSITEB99.348

Using the same supposition, the sixth post-session command would be translated as follows:

6) c:\ftp\ftp95pro.exe –i ashtech.ini -p ash_u -s local: E:\GPSDATA\DEC99\DAY14\SITE3482.99O -d ash_u:/pub/rinex/1999/12_14/SITE3482.99O

We have shown the translation for the remaining post-session commands because they are similar to those already described.

3.1.8.4.3 A Remote Receiver and a Network Directory Example

In this example, we seek to copy all data stored by GBSS to a network drive. Further, it is desired to have GBSS, using another program, dial a remotely located GPS receiver to retrieve its data, convert that data to RINEX, and then store the downloaded and converted data onto the network drive. To accomplish this, the following steps are to be performed through GBSS post-session commands at the completion of each "File Duration":

1) GBSS will call CGREMOTE.EXE, an Ashtech program for remote receiver control, to dial a remote GPS receiver and download its data.
2) GBSS will call a batch file created by CGREMOTE.EXE to decompress the data downloaded from step 1 into Ashtech formatted GPS data files.
3) GBSS will delete, via a batch file, the receiver image file downloaded in step 1.
4) GBSS will move, via a batch file, the data files created from step 2 to the same directory in which GBSS is storing data from a local receiver.
5) GBSS will call XYZASHRX.EXE to convert the data files from step 4 to their associated RINEX counterparts.
6) GBSS will copy the created RINEX files of step 5 and those created by GBSS (through its connection to a local GPS receiver) to the target network directory.

Throughout this example, we make the following assumptions:

1) The reader is familiar with the utility programs, and their command-line parameters, called through these post-session commands (for example, CGREMOTE.EXE and XYZASHRX.EXE).
2) The reader is familiar with MS-DOS batch command files.
3) The directory in which REMOTE.EXE is stored is E:\REMOTE.
4) The directory in which XYZASHRX.EXE is stored is "C:\Program Files\Ashtech\GBSS\UTILS".
5) All of the GBSS output files of interest (that is, the files to be transferred to the shared network drive) are stored in the GBSS Primary Output Data Directory.
6) The target network drive for all of the files of interest is drive "E:". The target directory structure on that drive will be \Refdata.*YY*\Month.*mmm*\Day.*dd*, where, *YY* is the 2-digit year of the data, *mmm* is the 3-character month of the data, and *dd* is the 2-digit day of the month.
7) The File duration used by GBSS maps one-to-one with the sessions programmed in the receiver. This ensures that the session codes for Ashtech formatted data files generated by GBSS will be the same as those generated by the remote receiver (via CGREMOTE.EXE).

The following lists the post-session commands for this example, each of which are created on a "wait for complete" basis. With each, we list the GBSS post-session command and its associated working directory.

C1) `e:\Remote\cgremote.exe -C 1 -D 2 -S -HS DECOMP.BAT`
Working Directory: `e:\Remote`

C2) `e:\Remote\DECOMP.BAT`
Working Directory: `e:\Remote`

C3) `e:\Remote\DELETE.BAT`
Working Directory: `e:\Remote`

C4) `e:\Remote\MOVE1.BAT $FPDD$`
Working Directory: `e:\Remote`

C5) `"C:\Program Files\ASHTECH\GBSS\UTILS\XYZAshRx.exe" -I $FPDD$\BCVB2$S$$YY$.$DDD$ $FPDD$\ECVB2$S$$YY$.$DDD$ $FPDD$\SCVB2$S$$YY$.$DDD$ -S 1 -C 0`

Working Directory: `C:\Program Files\ASHTECH\GBSS\UTILS`

C6) `E:\REMOTE\Copier.bat $FPDD$ $YY$ $MMM$ $DD$`
Working Directory: `e:\Remote`

We will now describe each step in more detail.

Step 1:

There are no mnemonics from GBSS required to call CGREMOTE.EXE. We simply use the normal command-line format of CGREMOTE to dial the remote receiver, download receiver image files, and create the batch file needed to decompress the image files.

Step 2:

We call the batch file created under step 1 to decompress the receiver image files downloaded under step 1. The decompressed files will be the normal Ashtech processing data files (for example, B-, E-, and S-Files). Notice that we have included the ".BAT" extension when calling the batch routine. This is required by the operating system and should be included with any post-session command call to a batch file.

Step 3:

We call the batch command file DELETE.BAT to delete the downloaded receiver image file, as it is no longer needed. Notice that in the post-session command that we again include the ".BAT" extent of the batch file. The contents of the called batch file are as follows:

REM FILE: DELETE BAT

cd

cd remote

del rcvb2???.???

This is a particularly noteworthy batch file in that it contains an MS-DOS intrinsic command (see Section 3.1.8.3). Intrinsic MS-DOS commands cannot be directly called from a GBSS post-session command. Windows permits GBSS, however, to launch a batch file that contains these intrinsic commands. This is why we did not call the "del" command directly from the post-session command.

Step 4:

GBSS calls the batch command file MOVE1.BAT to move the data files created under step 1. The contents of the batch file are as follows:

REM FILE: MOVE1.BAT

REM %1 = TARGET DIRECTORY

move ?CVB2???.??? %1

This is another particularly noteworthy batch file in that it also contains an MS-DOS intrinsic command (see Section 3.1.8.3). Intrinsic MS-DOS commands cannot be directly called from a GBSS post-session command. Windows permits GBSS, however, to launch a batch file that contains these intrinsic commands. Also to be noted is the use of argument passing into a batch file (that is, the use of the "%1"). For explanations of this parameter passing mechanism, please consult other references on MS-DOS batch programming.

Step 5:

GBSS calls the program XYZAshRx.EXE to convert the Ashtech-formatted files of step 4 to RINEX. In this post-session command we use the GBSS mnemonics listed in Table 3.5.

Table 3.5 GBSS Mnemonics

Mnemonic	Description
$FPDD$	The drive and directory of the current GBSS Primary Data Output Path.
S	The session code of the files to be converted to RINEX.
YY	The 2-digit year of the files to be converted to RINEX.
DDD	The 3-digit day of the year of the files to be converted to RINEX.

Through this post-session command we construct the full name of the files needed to convert the B-, E-, and S-Files to the RINEX Observation and Navigation data files. It is important to note that through the mnemonics of the post-session command feature, you can create tailored directory and file names.

Step 6:

GBSS calls another batch file to copy the RINEX files of step 5 and those created by GBSS (through its connection to a local GPS receiver) to the target network drive/directory. The content of the batch file called is as follows:

```
REM  FILE: COPIER.BAT

REM  %1 = GBSS Data path
REM  %2 = 2-digit Year
REM  %3 = Month
REM  %4 = 2 Digit Day
REM
REM  SYNTAX: copier.bat $FPDD$ $YY$ $MMM$ $DD$
REM
mkdir e:\RefData.%2
mkdir e:\RefData.%2\Month.%3
mkdir e:\RefData.%2\Month.%3\Day.%4
copy %1\*.* e:\RefData.%2\Month.%3\Day.%4
del /Q %1\*.*
```

The batch file first attempts to make the target directory on the network drive. Notice that if the target directory already exists, the mkdir commands (which is also an MS-DOS intrinsic command) will fail but will not cause the batch file execution to terminate. Next the batch file calls an MS-DOS intrinsic command to copy all of the data from the source directory (that is, the GBSS Primary Data Output Path) to the target network drive. The batch file then deletes (in the no confirmation asking, or quiet, mode) the files in the GBSS Primary Data Output Path.

As you can see by the documentation of the batch file (that is, the text immediately following "REM" on the first eight lines of the file) that there are four arguments passed to this batch file. These arguments relate, in order, to the parameters expressed in the last post-session command.

3.1.8.4.4 RINEX Session Rename and Push to an FTP Server Example

The primary objectives of this example are:

1. Renaming the RINEX files created by GBSS such that their session codes precisely match those of the session codes for the Ashtech B-, E-, and S-Files; and

2. Upload all of the data files created by GBSS, including the renamed RINEX files, to an FTP server.

Throughout this example, we make the following assumptions:

1. The reader is familiar with the utility programs, and their command-line parameters, called through these post session commands (for example, FTP95PRO.EXE).
2. The reader is familiar with MS-DOS batch command files.
3. The directory in which the batch file is stored is C:\BATCHES.
4. The target directory structure on the remote FTP server will be /cors/data/mmmYY/Day*dd*, where, *mmm* is the 3-character month of the data, *YY* is the 2-digit year of the data, and *dd* is the 2-digit day of the month.
5. Permission to create directories and store files on the remote FTP server has been granted for the accounts we are using.

The following lists the post-session commands for this example, each of which are created on a "wait for complete" basis. With each, we list the GBSS post-session command and its associated working directory.

C1) `C:\BATCHES\RxAshSes.Bat $FPDD$ $SITE$ $S$ $RXS$`
Working Directory: `C:\BATCHES`

C2) `C:\BATCHES\RxAshSes.Bat $FSDD$ $SITE$ $S$ $RXS$`
Working Directory: `C:\BATCHES`

C3) `c:\ftp\ashftpmd.exe ftp.ourserver.com usrname passwd /cors/data/$MMM$$YY$/DAY$DD$`
Working Directory: `C:\FTP`

C4) `c:\ftp\ftp95pro.exe –i ashtech.ini -p condorST -s local:$BFP$ -d condorST:/cors/data/$MMM$$YY$/DAY$DD$/$BF$`
Working Directory: `C:\FTP`

C5) `c:\ftp\ftp95pro.exe –i ashtech.ini -p condorST -s local:$BFP$ -d condorST:/cors/data/$MMM$$YY$/DAY$DD$/$EF$`
Working Directory: `C:\FTP`

C6) `c:\ftp\ftp95pro.exe –i ashtech.ini -p condorST -s local:$BFP$ -d condorST:/cors/data/$MMM$$YY$/DAY$DD$/$SF$`
Working Directory: `C:\FTP`

C7) `C:\BATCHES\UPLOAD.BAT $S$ $DDD$ $YY$ $FPDD$ /cors/data/$MMM$$YY$/DAY$DD$/`
Working Directory: `C:\BATCHES`

We will now describe each step in more detail.

Step 1:

GBSS calls the batch command file RXASHSES.BAT to rename the RINEX files stored in the GBSS Primary Data Directory such that their session codes to exactly match that of the B-, E- and S-Files. The contents of the batch file is as follows:

```
REM %1 = THE DIRECTORY IN WHICH THE RINEX FILE TO BE RENAMED RESIDES

REM %2 = THE 4-CHARACTER SITE NAME OF THE FILE
REM %3 = THE B-FILE SESSION CODE
REM %4 = THE GBSS-GENERATED RINEX SESSION CODE
REM
rename %1\%2???%4.??? %2???%3.???
```

Here we are calling the MS-DOS intrinsic command "rename", as such we have placed it into a batch file. Notice in the post-session command that we are using the mnemonics `$S$` and `$RXS$`. "`$S$`" is the mnemonic for the session code of the Ashtech processing B-, E-, and S-Files and "`$RXS$`" is the mnemonic for the session code of the RINEX files.

Step 2:

GBSS calls the batch command file RXASHSES.BAT to rename the RINEX files such that their session codes to exactly match that of the B-, E- and S-Files. We call the same batch file as was called under step 1 but are performing the operation for the files stored in the GBSS Secondary Data Directory.

Step 3:

GBSS calls the utility program ASHFTPMD.EXE to create a directory on the remote FTP server. This particular utility program will only attempt to create the directory if it does not already exist on that server.

Steps 4 to 6:

GBSS calls FTP95PRO.EXE to push the B-, E- and S-files created by GBSS. For example, in step 4 we use the mnemonic BFP which contains the full name and path of the B-File created by GBSS. We also use the mnemonic BF which contains only the name of the B-File (and not any drive or path information).

Step 7:

GBSS calls the batch file name UPLOAD.BAT to push the renamed RINEX files to the remote FTP server. The contents of the batch file are as follows:

```
REM %1 -- BFILE SESSION CODE
REM %2 -- SOURCE DAY OF YEAR (3-DIGIT)
REM %3 -- SOURCE 2-DIGIT YEAR
REM %4 -- SOURCE PATH
REM %5 -- DESTINATION PATH
C:\FTP\FTP95PRO.exe –i ashtech.ini -p condorST -s local:%4\CNDR%2%1.%3o -d condorST:%5
C:\FTP\FTP95PRO.exe –i ashtech.ini -p condorST -s local:%4\CNDR%2%1.%3n -d condorST:%5
```

In this case, we did not need to use a batch file to perform the operations but did so for the instructive purposes of the example. We are simply pushing the RINEX Observation and Navigation files to the remote FTP server using FTP95PRO.EXE. Please pay particular attention to the mnemonic used for the session code in the post-session command. “S” Is the mnemonic for the session code of the Ashtech processing B-, E-, and S-Files. Here we are using this mnemonic because we have already renamed the RINEX session codes to exactly match that of the B-, E- and S-Files.

3.1.9 External Modules Configuration

Built into GBSS is a real-time interfacing capability specifically designed to provide external programs with real-time data collected through GBSS. This allows you to develop applications using GPS data without concern for the details associated with communicating with the different receiver variants and the file management associated with archival of that data.

Next we will present a conceptual view of the interface mechanism. Most users of GBSS will not be concerned with this conceptual view. However, the explanation will help explain the rationale behind the configuration needs and configuration approach within GBSS as it relates to the external programs (termed external modules).

Later, we will present the configuration approach through an example. We will use the GBSS Met module as our example. This module was designed to facilitate the collection and archival of meteorological data using GBSS, a GPS receiver, and a meteorological sensor.

3.1.9.1 Conceptual View of the External Module Interface

Conceptually, this interface has the appearance depicted in Figure 3.24.

Figure 3.24 External Module Interface

GBSS manages all that is needed to set-up for and initiate the transfer of data from the GPS receiver. Also, GBSS, through its normal configuration, can archive and manage various data collected from the receiver. When configured to do so, GBSS can initialize and communicate through an interface to external programs.

There are a variety of ways to configure the relationship between one or more external programs, a single or multiple copies of GBSS, and one or more GPS receivers. The simplest form is depicted above. That is, a single external program communicating with a single copy of GBSS with a single receiver. Consider the second general form depicted in Figure 3.25.

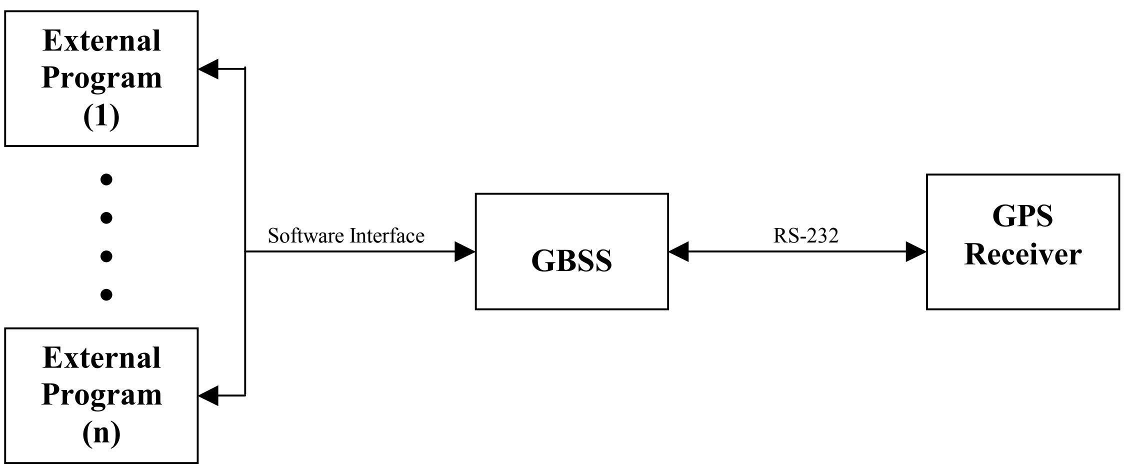


Figure 3.25 Interface with Multiple External Programs

In the above-depicted configuration, there are many applications (or external modules) interfacing with a single copy of GBSS over a single software interface. Each external program, through this interface, can send commands to and receive data from GBSS. Although the figure gives the appearance that the external programs can communicate with one another through this interface, they cannot: the developers of the external applications would need to build their own inter-program interface to do so.

The third general form is depicted in Figure 3.26.

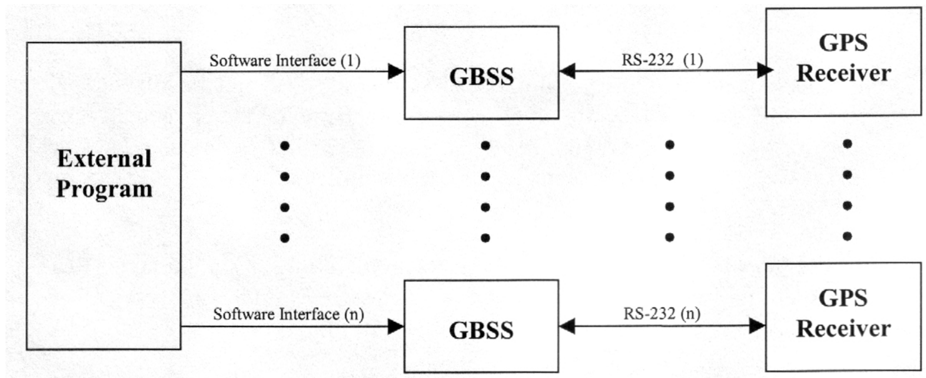


Figure 3.26 Third General Form for External Interface

In the above figure, we depict one external program interfacing with many copies of GBSS, each of which is communicating with a single receiver. To be clear, GBSS communicates with a single receiver: a single copy does not communicate with multiple receivers simultaneously. When configured in this third form, each copy of GBSS must originate from its own copy in its own directory on disk. That is, one must copy GBSS.exe and its configuration files (GBSS.INI and GBSS.SES) to another directory, one for each multiple copy desired. The rationale for this copying is that the configuration for a single copy of GBSS is stored in its configuration files (GBSS.INI and GBSS.SES). Launching multiple copies of GBSS from the same directory location would cause a contention between these copies in terms of their configuration. Thus, each copy of GBSS could be configured independently by placing these files in separate directories.

It is particularly important to note that in the third form (shown in the above figure) that there are multiple interfaces to the single external program, one for each copy of GBSS. These interfaces are truly independent, each of which originates from a single copy of GBSS.

Presented above are the general forms of the interfacing mechanism. Combinations of the above forms can be exploited to take full advantage of the data made available from GBSS through this interface. However, with the flexibility comes complexity. This complexity is manifested into a requirement that might confuse the novice user. We have sought to reduce that complexity to a few key configuration items. The most notable of which is the requirement that users provide an interface name. This name does nothing more than identify the interface to both GBSS and its associated external program(s). That is, the interface identifier, which is simply an 8-character text string of your choice, must be supplied to both GBSS (through its external module configuration) and to all external programs using data from that single copy of GBSS. Further, each copy of GBSS must have its own interface identifier. That is, if two copies of GBSS are given the same interface name, **then** one copy of GBSS will report an error and disallow any transactions over that interface.

3.1.9.2 Configuration Description Through an Example

As was stated earlier, we will present the configuration approach through an example. We will use the GBSS Met module, which was designed to facilitate the collection and archival of meteorological data using GBSS, a GPS receiver, and a meteorological sensor. Presented here, will be the configuration from the GBSS perspective, and we will not describe the configuration requirements of the GBSS Met Module. For a detailed description of the GBSS Met Module configuration process, please consult the appropriate documentation for that program.

GBSSMet depends upon GBSS providing it with real-time data. Because GBSS and the Meteorological Module (that is, **GBSSMet**) have been designed to allow multiple copies of each program to be running at one time, one must specify which copy(ies) of **GBSSMet** runs with a single copy of GBSS (the rationale for which was described in Section 3.1.9.1). GBSS does this by essentially naming a real-time data pipe. This same pipe name is then used in the Meteorological Module.

To configure the real-time outputs for GBSS, one must run GBSS and select the “Configuration | Real-Time Interface” menu item. Upon selecting this menu option, you will be presented with the menu shown in Figure 3.27.

:

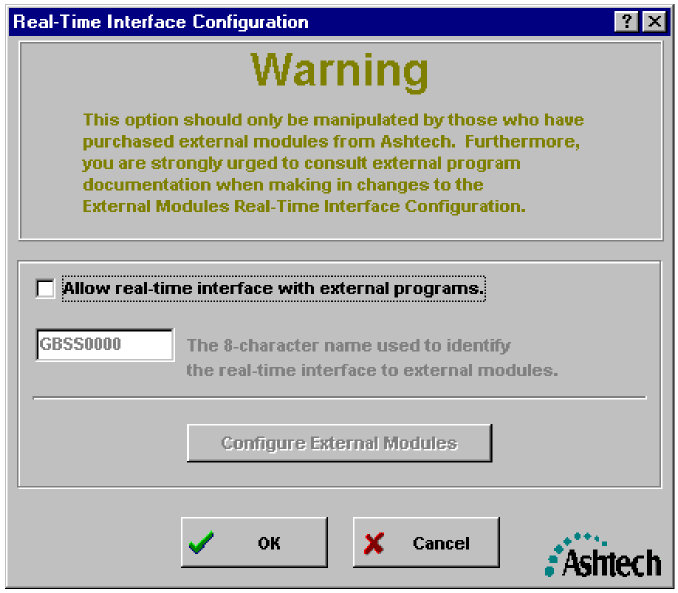


Figure 3.27 Real-Time Interface Configuration Window

To enable the real-time interface to external program modules (such as GBSSMet), you must select the checkbox labeled "Allow real-time interface with external programs". Once you select this checkbox, the 8-character pipe identification field and the "Configure External Modules" button become active. The 8-character pipe identification is used to identify the data stream from this copy of GBSS to external modules. For example, if you will be running multiple copies of GBSS on a single computer and wish to provide real-time data to external modules, then each copy of GBSS should use its own 8-character pipe name. You can choose the names, but make each one different and do NOT use any space or other special characters. For most installations, the default pipe name is sufficient.

For most installations of the Meteorological Module, you will simply need to set the checkbox labeled as "Allow real-time interface with external programs". That is, set this checkbox and then press the 'OK' button. However, there are times when a more detailed configuration is required. This is described below.

By pressing the "Configure External Modules" button, you will be given an opportunity to provide GBSS with information about the external program module. Upon pressing this button, you will be presented a window similar to Figure 3.28.

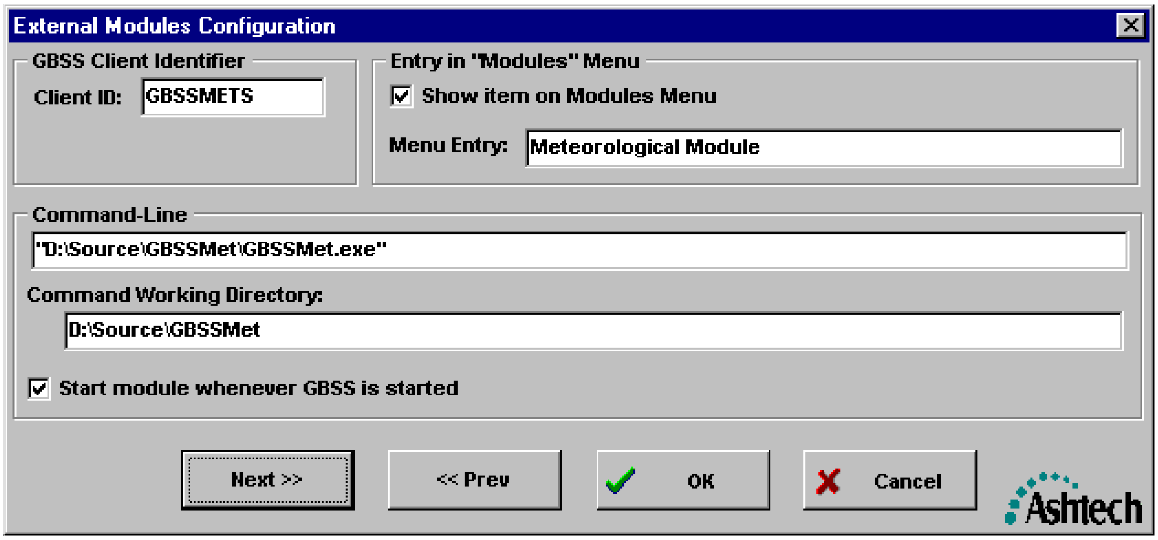


Figure 3.28 External Modules Configuration Window

Each copy of GBSS will support up to ten external program modules. The "Next >>" and "Prev <<" buttons let you scan through the configuration of each of the ten modules.

Certain external modules are specifically identified to GBSS by a "Client ID". This ID is used primarily to prevent multiple copies of the same program from automatically being launched by GBSS. The developer of the external module defines the ID. For the case of the Meteorological Module, the Client ID is "GBSSMETS". The entry **is** case sensitive. Please note that this discussion assumes that the external program being called implements the interface to GBSS. That is, one could, through this external modules feature, have GBSS launch an external program that does not implement the GBSS real-time interface. However, because that program is not communicating with GBSS through this interface, GBSS will continuously assume that it failed to launch the external program. This could lead to GBSS inadvertently launching many copies of the external program. Therefore, it is suggested that this mechanism be used only for external programs that fully implement the GBSS interface.

The area of the menu labeled "Entry in 'Modules' Menu" allows you to specify whether or not you want an entry in the Modules item of the GBSS main menu. That is, once you "allow real-time interface with external programs", a "Modules" entry will be added to the GBSS Main menu as shown in Figure 3.29.

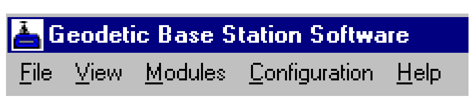


Figure 3.29 Modules Addition to Main Menu

By configuring an "Entry in [the] 'Modules' Menu", you then are specifying whether or not you want an entry in the "Modules" menu. The above example would add the following to the Modules Menu (Figure 3.30).

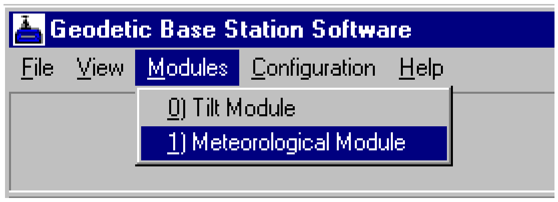


Figure 3.30 Added Modules

The “Command-Line” area of the External Modules Configuration window allows you to specify how GBSS is to launch the external module. You MUST provide **BOTH** the Command-Line and the Working Directory. Failure to do so could result in erroneous behavior of the external module. When supplying the text of the command-line (not the working directory), you should use quotation characters to clearly state path names with spaces in them. Notice, in the example provided, that the entire command-line is quoted. This is because we are quoting the entire full path to the executable Meteorological Module program. When the command-line is used to call an external program with additional command-line options, only the entire full path of the executable program should be quoted (that is, the quotation should not include the options, unless the options of that program requires some quotation of parameters).

The meaning of the checkbox labeled as “Start module whenever GBSS is that GBSS will launch the external module whenever GBSS itself is started. Additionally, when you leave this configuration screen and the Real-Time Interface Configuration Screen and this checkbox is checked, GBSS will immediately attempt to launch the external module.

Please note that whenever GBSS starts an external module, it writes a message to its Diagnostic Message window indicating the success or failure to launch that module. When having difficulty launching an external module, you should consult the Diagnostic Message window to determine why.

Again, it is important to stress that the use of this external module interface/start-up mechanism be restricted to external programs that fully implement the GBSS interface. The interface is defined such that GBSS is aware of any external modules that may be communicating over that interface. Upon starting an external program, GBSS awaits a response from that module, through its interface, indicating the successful start-up of that module. The lack of a response, among other responses, is deemed an erroneous condition. Failure to comply with this warning could lead to erroneous behavior of GBSS and/or any external programs called through this mechanism.

3.2 Configuring GBSS for Auto-Startup

GBSS can be setup to automatically connect to the GPS receiver once your operating parameters have been set. When this feature is enabled, the base station software will automatically startup and connect to the GPS receiver when Windows is started. This is particularly useful, for example, when it is desired that data logging be continued immediately after a power failure. To accomplish this, the base station software must first be configured to your specifications. Connect GBSS to your GPS receiver and verify that all parameters are set correctly. Then, complete the instructions that follow.

During the installation of GBSS, you were asked if you wanted GBSS to automatically startup each time Windows is started. If you answered yes to that question then most of the manual steps described herein have already been done for you.

To set GBSS to automatically start when Windows starts, follow these steps:

1. Determine the drive and folder in which `GBSS.EXE` is stored (this can be done by using the Windows 95 or NT “Start | Find | Files or Folders” feature).
2. Add GBSS.EXE to the Windows Start-Up folder:
    a. Press the Windows 95 or Windows NT “Start” button.
    b. Select “Settings”
    c. Select “Taskbar…”
        i. When the “Taskbar Properties” window is shown, select the “Start Menu Programs” Tab

ii. Press the “Add” button.

iii. For the “Command line” enter the complete path of `GBSS.EXE` found under step 1 followed by “-C”. For examples, you could enter the following:

C:\Program Files\Ashtech\GBSS –C (NOTE: there is a space before the –C).

It is important to note that on some computer platforms in this automatic connection mode that GBSS may attempt to access the hardware (for example, the software sentinel or the communication ports) before Windows has fully initialized that hardware and/or pertinent device drivers. This is due to the multithreaded nature of Windows NT. In other words, Windows NT may launch all initialization and “Startup” applications simultaneously upon the start of Windows NT. It is therefore possible (and probable) that GBSS will finish initializing and begin searching for the software sentinel drivers before Windows has finished loading. Because of this, we may need to instruct GBSS to delay sometime before attempting to utilize the sentinel drivers (that is, to give NT enough time to complete the initialization of the drivers). We instruct GBSS to delay using the command-line -C parameter. This delay parameter instructs GBSS to wait before attempting to access the hardware and device drivers. The delay is entered as shown in the following example:

C:\Program Files\Ashtech\GBSS –C 45 (NOTE: there is a space before the –C and before 45).

In the above example, GBSS.EXE would wait 45 seconds before attempting to establish a connection with its sentinel and communication ports. Whenever the –C parameter is specified without the delay parameter GBSS, by default, waits 30 seconds. It is permissible to specify wait periods less than 30 seconds.

It is recommended that you find the minimum -C delay parameter that your computer can use and then set the value to 10 seconds above that value. You can tell when you have fallen below the minimum. That is, upon boot-up of your computer and the automatic startup, GBSS informs you that it cannot locate/initialize the sentinel (and the sentinel key is present) or it cannot open the specified communication port (and the specified port is a valid port on your computer).

There are some special automatic startup concerns for Windows NT machines. These start-up concerns are detailed in Section 2.6. Failure to address the special start-up concerns may prevent GBSS from automatically starting with each start of Windows NT.

4

RUNNING THE GEODETIC BASE STATION SOFTWARE

4.0 RUNNING GBSS - OVERVIEW

This chapter provides a description of GBSS once it has been configured. Before actually running GBSS, you will need to have installed the software sentinel key and then have configured GBSS. Chapter 3 provides a description of the configuration process.

4.1 Connecting To and Disconnecting From the Receiver

GBSS can be connected to a receiver through one of two primary means:

1. Manually connecting to the receiver through the "File | Connect" menu options; or
2. Automatically connecting to the receiver as is described in Section 3.2.

To manually connect to the receiver, select the File menu option from the menu bar of GBSS. This will call the drop-down menu shown in Figure 4.1.

Figure 4.1 Manual Connect Drop-Down Menu

From this menu, simply select "Connect". Once connected to a receiver, you can manually disconnect by selecting "Disconnect" from this same menu. Also, once connected you cannot alter the configuration of GBSS. That is, GBSS is designed such that it is pre-configured and then you connect to the receiver. Chapter 3 describes how to configure GBSS.

Again, you can have GBSS automatically connect to the receiver each time Windows is started. See Section 3.2 for details on how to configure GBSS to auto-connect to the receiver in this way.

In either case, you must physically connect the Ashtech receiver to the computer using an Ashtech NULL modem RS-232 cable. Numerous message errors may be reported if you fail to use a cable that meets or exceeds the specification of Ashtech cables.

4.2 Main Display Window

Figure 4.2 labels each area of the main window and provides an example of the main GBSS display:

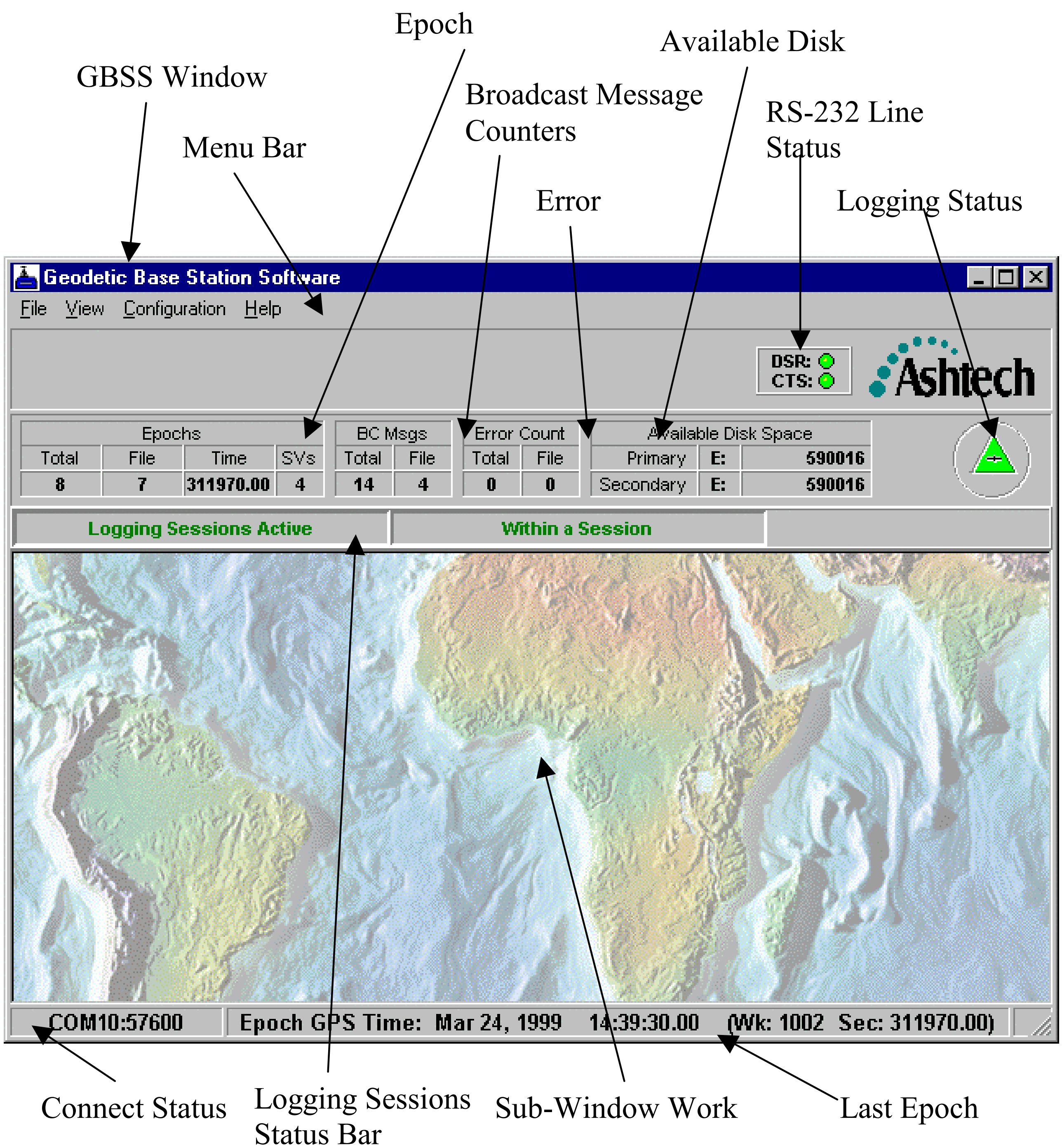


Figure 4.2 GBSS Main Window

With the exception of the "Main Window Title" and the "Menu Bar" items, the sub-sections which follow describe each of the above labeled items.

4.2.1 Epoch Counters

During the logging of receiver data, the Epoch Counter area of the main window will display information pertaining to the current epoch of data. Depending upon the selected File Duration option (see Section 3.1.4.10), files are closed and new ones are opened at a specified interval. As such, it is sometimes beneficial to know the epoch status since connecting to the receiver as well as the epoch status in relation to the current file set. To this end, the epoch counters are labeled as "Total" and "File". The "Total" counter shows the total epochs received since the connection to the receiver was established and the "File" counter shows the number of epochs written to the files of the current file set since they were opened.

The "Time" field shows the time (in seconds of GPS week) of the epoch data received from the receiver. The "SVs" field shows the number of satellites for which there is data in the epoch obtained from the receiver.

4.2.2 Broadcast Message Counters

These Broadcast Message Counters keep track of the number of satellite broadcast messages (also known as satellite navigation messages or nav. messages) received from the receiver since the connection to the receiver was established and since the start of the current file set. The "Total" and "File" fields have similar meaning as those of the Epoch Counters (see Section 4.2.1) but pertain to the Broadcast messages.

4.2.3 Error Counters

The Error Counters keep track of the number of messages that have been received from the GPS receiver that were in error (that is, their checksums were invalid). The "Total" and "File" fields have similar meaning as those of the Epoch Counters (see Section 4.2.1) but pertain to the number of erroneous messages received.

It is important to note that while GBSS may report the detection of errors, it will not write erroneous data to the Ashtech observation files (B-, E- and S-Files). That is, the receiver sends GBSS several packets of data per epoch. GBSS will save (and write to the observation files) all packets that are valid and discard any that are not. It is also important to note that even invalid packets are written to the Trap File (see Section 3.1.4.5).

4.2.4 Available Disk Space

The Available Disk Space area is divided into primary and secondary disk drive space. These drives pertain to the drives specified for the Primary and Secondary Data directories (see Sections 3.1.4.3 and 3.1.4.4). That is, the specified Primary and Secondary Data directories also contain drive letters. The drive letters of these two directories, even if they are the same, are used as the primary and secondary drives of this display. It is important to note that GBSS also has Primary and Secondary Compression directories. GBSS makes no effort to display the available disk space of drives specified by these two directories (unless they are duplicated by the drives specified for the Primary and Secondary Data directories). In some situations this might encourage a user to place the compression directories on the same drives as the data directories.

The available disk space is shown in units of Kilobytes (that is, the number of bytes divided by 1024).

4.2.5 RS-232 Line Status Indicators

The RS-232 Line Status Indicators are located at the upper right of the GBSS window and are used to provide a positive indication of the RS-232 status lines. They are only active when GBSS is connected to an Ashtech receiver (that is, just because a receiver is connected to the computer does not imply that GBSS recognizes the receiver – you must instruct GBSS to connect to the receiver before these status lines become active). Normally, both the DSR and CTS indicators are green. Typically, when the DSR is green and the CTS is not, then the connection is failing at the receiver side of the RS-232 connection. Otherwise, the connection failure is most likely near (or along the line) where the computer connects to the RS-232 cable.

Please note that if you disable the CTS/RTS hardware handshaking (see Section 3.1.1.3) then the CTS indicator will not illuminate. Likewise, if you disable the DTR/DSR hardware handshaking (see Section 3.1.1.4) then the DTR indicator will not illuminate.

4.2.6 Logging Status Icon

The logging status Icon changes color and flashes based upon the current communication status of GBSS. The Icon contains no color when GBSS is not currently connected to the receiver. Once GBSS is connected to the GPS receiver (see Sections 3.2, 4.1, 4.6 and 4.7) the Icon will be in one of three states:

1. Green Triangle
2. Yellow Flashing Triangle
3. Red Flashing Triangle

It is important to remember that the color of the icon relates only to the current status (that is, it does not consider the historical performance of the communication with the receiver beyond the past few epochs of data). The green triangle (solid green) implies that GBSS is currently operating without any obvious problems. The Yellow Flashing Icon is displayed when 4% of the recent bytes received from the receiver cannot be interpreted by GBSS. The Red Flashing Icon is displayed when 30% of the recent bytes received from the receiver cannot be interpreted by GBSS or there is an unexpected change in the RS-232 line status (such as a break in the RS-232 line). GBSS is not designed to interpret all possible messages from an Ashtech receiver. As such, you may have special configuration needs that cause data to be output to GBSS from the receiver. For this reason, GBSS will not terminate based upon any value of invalid bytes received from the receiver. Furthermore, you may have configured the GPS receiver to output messages that GBSS does not interpret. In this case, GBSS will report either the Yellow or Red flashing icon, will be recording B- and E- files, and, assuming that the Trap File feature is turned on, will be recording all data (including the unrecognized messages) to the Trap File.

GBSS can be configured to play a sound file when a yellow flashing Logging Status Icon is displayed. It can be configured to play a different sound file when a red flashing Logging Status Icon is displayed. Refer to Section 3.1.6.3 for more details on configuring GBSS to play sound files. This feature is especially useful for continuous operation when the monitor has been turned off (to conserve power). For these situations the audible alarm rather then the flashing indicator provide the first clue of any problems.

4.2.7 Connect Status

The lower status bar of the GBSS main window contains status information, among which is the connection status. When GBSS is not connected to a receiver (and not simulating a receiver), this area of the window displays “Disconnected”. When GBSS is connected to a live receiver, this area of the window shows “COMx:*nnnnn*”, where x is the communication port of the receiver and *nnnnn* is the BAUD being used to communicate with the receiver. When GBSS is in its simulation or play back modes, the connect status shows “Simulated” while the simulation is in progress and “Disconnected” when the simulation completes.

4.2.8 Epoch Time Display

The lower status bar of the GBSS main window contains status information, among which is the GPS time of the last epoch received. When GBSS is not connected to a receiver (or simulating a receiver), this area of the display is left blank. Again, the GPS time of the epoch is displayed in a Gregorian fashion and should not be confused with UTC (that is, the time displayed is in the GPS system and not UTC).

4.2.9 Sub-Window Display Area

The sub-window display area of the main window is used to display all of the status windows described under Section 4.3. You are free to move all of the sub-windows within this area. Moving a window beyond the display region of this area will cause scroll bars within this area to be displayed. These scroll bars will allow you to “scroll” through the display area.

4.2.10 Logging Sessions Status Bar

The logging sessions status bar will be visible only when the Logging Sessions/Session Programming feature of GBSS has been enabled (see Section 3.1.5). When visible, the text at the left of the status bar (that is, “Logging Sessions Active”) will flash. The text to the right of the active indicator will either contain the words “Within a Session” (as shown in the screen capture of Section

4.2) or “Not Within a Session” as shown in Figure 4.3.

Figure 4.3 Logging Sessions Status Bar

GBSS changes the text according to the last epoch received from the receiver. Whenever the current GPS time, as determined from the last received epoch, falls within a *logging session*, the text displayed shows the “Within a Session” state. Conversely, when a received epoch time falls outside of the *logging sessions*, the text displayed shows the “Not Within a Session” state. Please note the distinction between the epoch time and the current GPS time. That is, GBSS determines the state of the *logging session* status bar using the most recent epoch received from the receiver and not the current time of the GPS constellation.

Also notice that the text of the logging session status bar appears to use the term *Session* and not *Logging Session* (the distinction is detailed in Section 3.1.5). However, no inference to the term *Session* is implied. That is, when “Within a Session” or “NOT Within a Session” is displayed, it is meant to imply that the data received falls inside or outside of, respectively, a *logging session*.

4.3 Status and Display Sub-Windows

GBSS provides the following sub-windows that can be displayed while connected to a receiver:

1. Geodetic Position Window
2. Earth-Centered Earth-Fixed Position Window
3. Channel Summary Window
4. Diagnostic Messages Window
5. Logging Summary Window
6. Post-Session Command Summary Window
7. Time Display Window

Each of these windows is accessed via the “View” menu selection from GBSS and then selecting the appropriate display. You are free to resize all of these windows as desired. To return any window to its default size, with the exception of the Diagnostic Message window, simply right-click within the display area of the window and chose the “Window Size to Default” option. Section 4.2.9 describes moving these windows within the area provided by the main window.

4.3.1 Geodetic Position Window

Figure 4.4 provides an example of the Geodetic Position window:

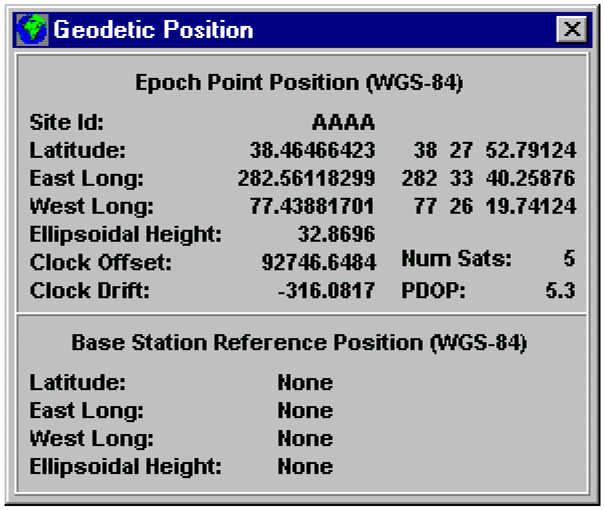


Figure 4.4 Geodetic Position Window

Notice that the window is divided into two parts: 1) the epoch point position and 2) the base station reference position. The upper portion (that is, that which is labeled "Epoch Point Position") contains information obtained from the most recent point position solution provided by the receiver in the "$PASHR,PBN" message. The lower portion of the window contains the site position information that was passed to the receiver based on user-supplied values (see Section 3.1.3). As shown in the above example, when a latitude, longitude, or height component of the site position has a zero value, this lower portion of the window will display "none" (implying that no position was entered) for that component. Figure 4.5 shows the same wi ndow with a position having been entered through the Configuration | Site Data window:

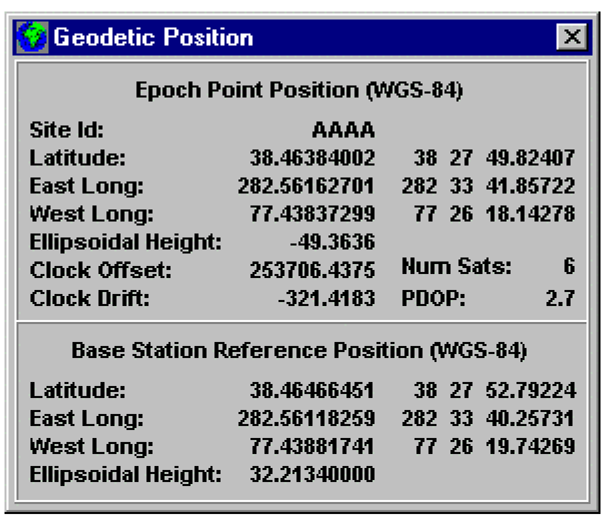


Figure 4.5 Configuration Site Data Window

As with all other windows in the program, GBSS will show the units of a displayed field by resting the cursor over that field.

4.3.2 Earth-Centered Earth-Fixed Position Window

Figure 4.6 provides an example of the Earth-Centered Earth-Fixed Position display window:

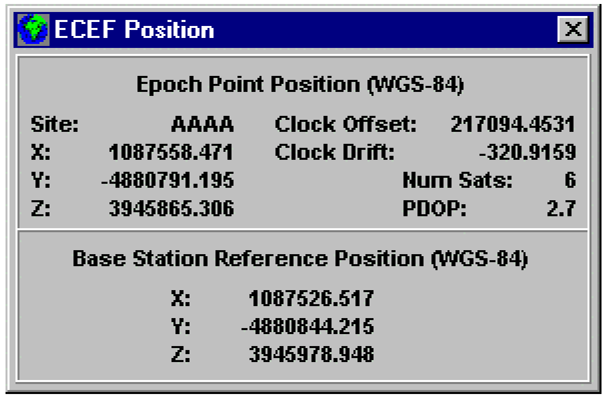


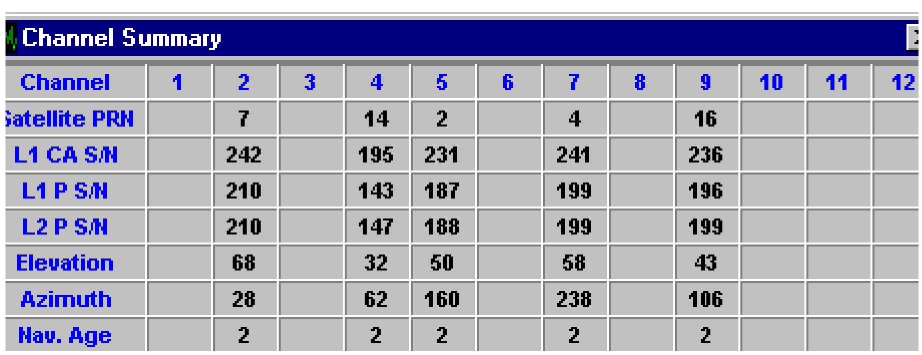
Figure 4.6 ECEF Position Window

Like that of the Geodetic Position window, this window is divided into two parts: 1) the epoch point position and 2) the base station reference position. The upper portion (that is, that which is labeled "Epoch Point Position") contains information obtained from the most recent point position solution provided by the receiver in the "$PASHR,PBN" message. The lower portion of the window contains the site position information that was passed to the receiver (see Section 3.1.3). The receiver ultimately requires that the X, Y, and Z components be converted to latitude, longitude, and height. As such, when a latitude, longitude, or height component of the site position has a zero value, this lower portion of the window will display "Partial" (implying that only a partial position was entered) for that all components. When all components have a zero value, then all will be displayed as "none" (that is, no fixed position entered).

As with all other windows in the program, GBSS will show the units of a displayed field if you rest the cursor over that field for a couple of seconds.

4.3.3 Channel Summary Window

Figure 4.7 is an example display of the Channel Summary window from a Z-12 receiver:



Channel Summary

Channel	1	2	3	4	5	6	7	8	9	10	11	12
Satellite PRN		7		14	2		4		16			
L1 CA S/N		242		195	231		241		236			
L1 P S/N		210		143	187		199		196			
L2 P S/N		210		147	188		199		199			
Elevation		68		32	50		58		43			
Azimuth		28		62	160		238		106			
Nav. Age		2		2	2		2		2			

Figure 4.7 Channel Summary Window 1

The information in this window is displayed based upon channel numbers, which are shown across the top of the window. The row labeled "Satellite PRN" is used to show which satellite, by its Pseudo-Random Noise number, is being tracked on each channel. Obviously, empty columns imply that no satellite is being tracked on that channel. The rows labeled with "L1 C/A S/N", "L1 P S/N", and "L2 P S/N" display the signal- to-noise value reported by the receiver for that satellite on the L1 C/A Code tracking loop, the L1 P-Code tracking loop, and the L2 P-Code tracking loop, respectively. The “Elevation” row displays the elevation (degrees) of the satellite being tracked. The “Azimuth” row displays the azimuth (degrees) of the satellite. The “Nav. Age” column indicates how long ago, in minutes, that a navigation message for that satellite was received.

Figure 4.8 shows an example of this window when connected to an Ashtech Z-18 receiver:

Channel Summary

Channel	1	2	3	4	5	6	7	8	9	10	11	12
Satellite PRN	16		24		7				4	2		18 R
L1 CA S/N	49		39		51				52	51		49
L1 P S/N	36		21		40				42	38		49
L2 P S/N	36		21		40				42	39		50
Elevation	45		26		67				62	47		60
Azimuth	102		234		40				244	162		68
Nav. Age	0		0		0				0	0		99

Channel	13	14	15	16	17	18	19	20	21	22	23	24
Satellite PRN	17 R	3 R			9 R	4 R						
L1 CA S/N	33	51			32	50						
L1 P S/N	32	51			31	51						
L2 P S/N	40	48			33	50						
Elevation	13	60			11	49						
Azimuth	112	56			272	178						
Nav. Age	99	99			99	99						

Figure 4.8 Channel Summary Window 2

Notice in the above example screens that the GLONASS satellites are displayed in dark red and have the letter R after their satellite numbers. Also notice that, in this display, the GLONASS satellites have a navigation age of 99. This is because at the time of the screen capture, GBSS had not received the navigation message for these satellites from the GPS receiver. Please note that the 99 is not restricted to just the GLONASS satellites: that is, whenever any navigation message has not been received from the GPS receiver, the Nav. Age will be 99.

As with all other windows in the program, GBSS will show the units of a displayed field if you rest the cursor over that field for a couple of seconds.

4.3.4 Diagnostic Messages Window

The Diagnostic Message Window, Figure 4.9, displays real-time textual messages that provide information on the current activities of GBSS. Most messages are time tagged with the seconds of GPS week of the most recent epoch. These time tags are enclosed in the [and] characters. In the example that follows, GBSS received its first SNAV message at 324530.0.

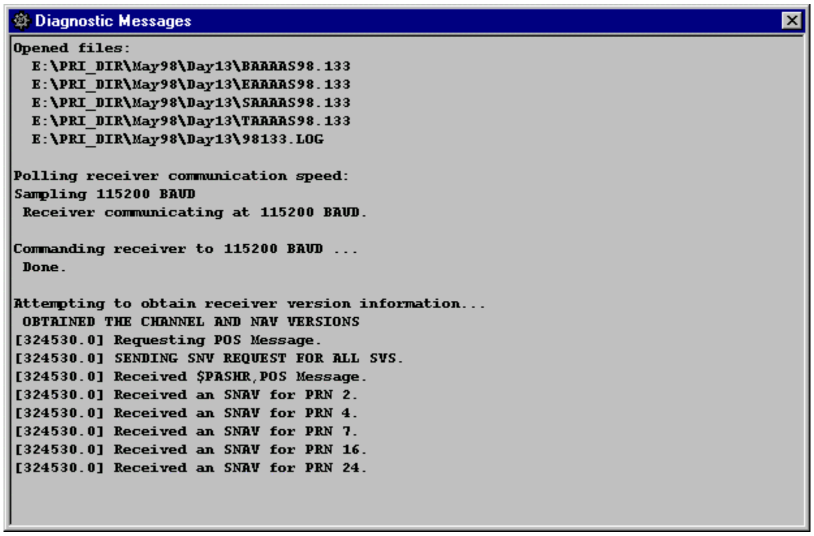


Figure 4.9 Diagnostic Messages Window

The amount of messages displayed through this window is controlled by the "Display verbose diagnostic messages" checkbox on the Configuration | Other Options menu (see Section 3.1.6).

4.3.5 Logging Summary Window

The Logging Summary window, Figure 4.10, provides a summary of the data being logged from the receiver. The following shows an example of this window when there are no files selected for output to the secondary data directory.

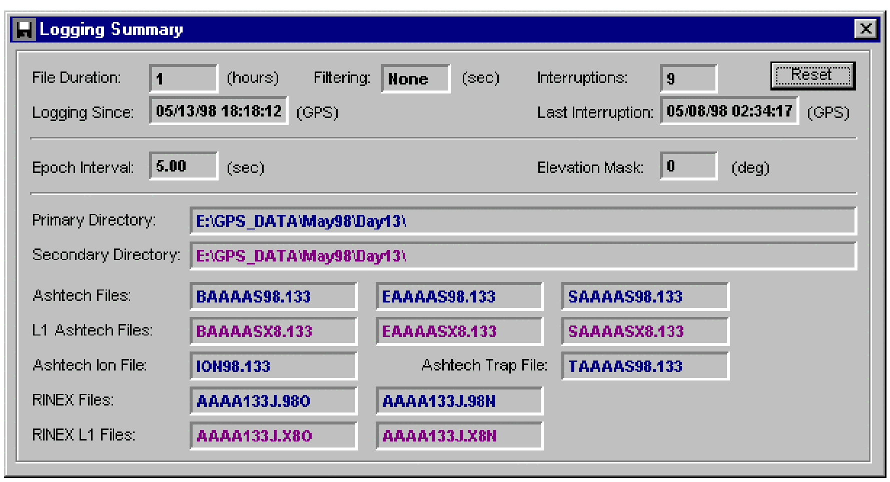


Figure 4.10 Logging Summary Window 1

In the example shown in Figure 4.11, both the primary and secondary directories have data being written to them.

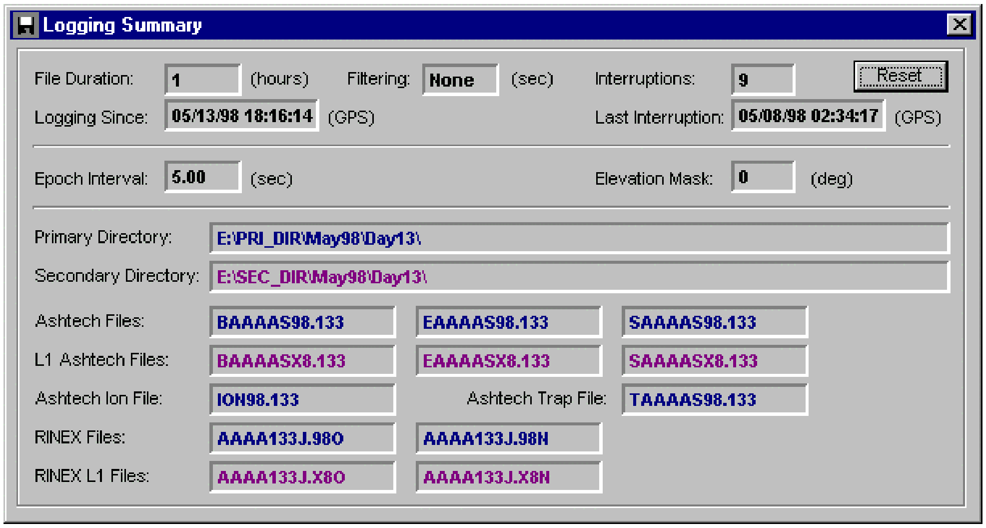


Figure 4.11 Logging Summary Window 2

The text of the paths and the names of the output files are color coded to show which files are targeted for which directories. Those files that are being written to the primary data directory will be displayed in navy blue. Those that are being written to the secondary directory are displayed in purple. For an explanation of the output paths and the names of the files see Chapter 3.

Table 4.1 lists the remaining parameters in this window:

Table 4.1 Remaining Parameters

Display Field	Description
File Duration	This field merely echoes the value set on the Configuration | Output Files menu (see Section 3.1.4.10).
Filtering	This field merely echoes the value set on the Configuration | Output Files menu for Epoch Filtering (see Section 3.1.4.12).
Epoch Interval	This field merely echoes the value set on the Configuration | Receiver Settings menu (see Section 3.1.2). Please note that if GBSS is in its Passive mode (see Section 3.1.2.1) or the “Command recording interval and elevation mask” checkbox of the Receiver Settings menu is unchecked, then the value displayed here will be “UNKN” (for unknown).
Elevation Mask	This field merely echoes the value set on the Configuration | Receiver Settings menu (see Section 3.1.2). Please note that if GBSS is in its Passive mode (see Section 3.1.2.1) or the “Command recording interval and elevation mask” checkbox of the Receiver Settings menu is unchecked, then the value displayed here will be “UNKN” (for unknown).
Logging Since	Displayed in this field is the corrected CPU GPS time (see Section 3.1.7) when the current connection to the receiver was established.
Interruptions and Last Interruption	Displayed in this area is the total number of interruptions since the “Reset” button on this window was pressed. GBSS keeps track of interruptions by periodically storing data into a specific file. When a connection is correctly terminated (for example, as described in Section 4.1 or when the program terminates normally), the data in that file is removed. If your computer fails and is restarted, GBSS can detect that residual data and determine, within +/- 5 minutes, when the interruption occurred. Please note that while this window displays only the last interruption, the Log File (see Section 3.1.4.5) keeps track of every interruption.

4.3.6 Post-Session Command Summary Window

The Post-Session Command Summary window provides a description of the Post-Session commands as they are executed. This window will show each command as it executes and will display the full text of the command (that is, the mnemonics will be replaced by their translated run-time values). Please note that the fully expanded text will also be written to the Diagnostic Message Window (see Section 4.3.4) and, if enabled, to the Log file (see Section 3.1.6.1).

Figure 4.12 provides an example of the Post-Session Command Summary window.

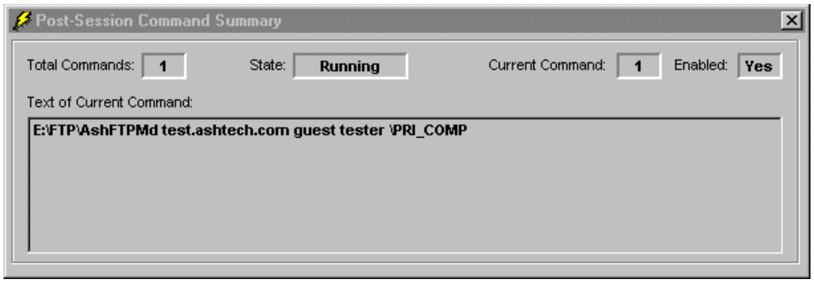


Figure 4.12 Post-Session Command Summary Window

4.3.7 Time Display Window

Figure 4.13 shows an example of the Time Display window:

Figure 4.13 Time Display Window

In order to understand this window, you need to be familiar with the concept of the "corrected CPU GPS time", which is described in Section 3.1.7. All of the times displayed in this window are related to the CPU time (that is, not epoch time). The time displayed with the "Raw" label is the raw (that is, unmodified) time from your computer's clock. The time displayed with the "GPS" label is the corrected CPU GPS time. The time displayed as "Local" is the corrected CPU GPS time plus the value of the "Local Time Zone". You are free to change the value of the local time zone but those changes do not take effect until the "Set Zone" button is pushed.

It is important to note that the update of this window is considered to be of low priority. As a result, when receiving data from a receiver configured to output epoch data frequently (say an epoch interval of 2 seconds or less), and depending on the speed of your computer, the update of this display may appear sporadic.

4.4 Terminal Window

The Geodetic Base Station Software is equipped with a terminal window that allows you to send commands to the GPS receiver and view the responses. This window is particularly useful for sending special commands to the GPS receiver (You can also send special start-up commands to the GPS receiver when GBSS is started – see Section 3.1.2.6). In order to use this feature, GBSS must be in its active mode (see Section 3.1.2.1) and communicating with an Ashtech receiver (that is, you must have selected "File | Connect"). To launch this window, select File | Terminal menu option and you will be presented with a window similar to Figure 4.14.

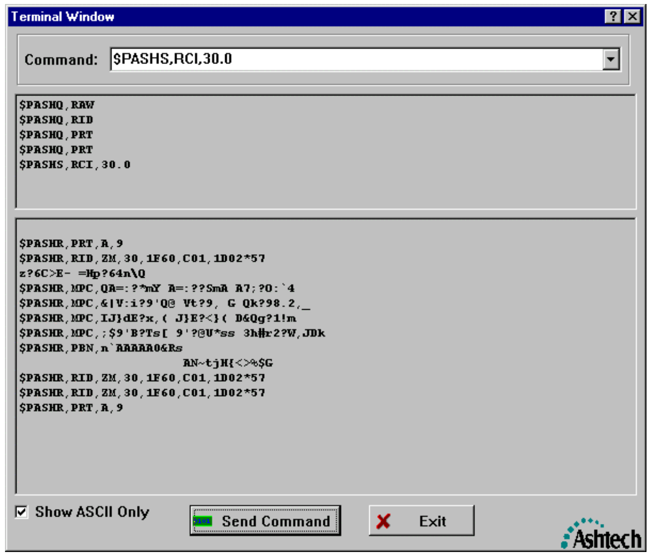


Figure 4.14 Terminal Window

In the upper area of the window (that is, the area labeled with "Command"), you can enter the command you wish to send to the receiver or you can use the drop-down list to select a message template. If you select a message template, you may need to modify it before sending the command to the receiver. For example, the template for the RCI command is "$PASHS,RCI,xxx.xx". In this case, you will need to change the "xxx.xx" part of the command to contain a valid epoch interval before sending the command. Entered commands are sent using the "Send Command" Button. To determine valid commands, you must consult the appropriate receiver manual. You do not need to supply the <CR><LF> (that is, carriage return line feed) sequences required at the end of the messages; GBSS will add those for you.

The upper middle portion of the window displays the commands that you have sent to the receiver. This area will also show the commands sent to the receiver, and they will be shown as they are sent, when the file upload capability has been enabled (see Section 3.1.2.6).

The lower middle portion of the window displays the data being received from the Ashtech receiver. Notice that in the above example that most of the data is not in a normal human readable form. This is because, in this example, the display is showing epoch data received from the receiver. This epoch data, like that of some other message data, is communicated in a binary form. GBSS will show only the ASCII portions of those messages when the "Show ASCII only" checkbox is checked. When this checkbox is unchecked, the non-ASCII data is shown in a hexadecimal representation of the data (that is, "0x' followed by the hexadecimal number of the character being represented).

It is important to note that GBSS will continue to log all epoch data normally while the terminal window is active (unless you issue the command, using this window, to stop the receiver from sending epoch data). Also, if the Trap File capability is enabled (see Section 3.1.4.5), all bytes received from the receiver will be output to the Trap File. Finally, this window is a lower priority than the tasks that handle the actual data received from the receiver. As a result, when receiving data from a receiver configured to output epoch data frequently (say an epoch interval of 2 seconds or less), and depending on the speed of your computer, the update of this display may appear sporadic and may appear as if GBSS is loosing data. However, the display update is at a low priority and may, depending upon the system load, not have enough time to display all of the data.

4.5 Uploading a Command File to the Receiver

There are two separate points in the operation of GBSS where a command file may be uploaded to the receiver. The first is when GBSS established the connection with the GPS receiver (as described in Section 3.1.2.6). In the second case, a command file may be sent by GBSS to the receiver at any time during data logging (GBSS must be in its Active Mode). This file can be used to provide the GPS receiver with information and configuration data that is specific to your particular needs at that time. For example, you can create a file that contains the commands necessary to cause the GPS receiver to start sending differential corrections from port C at a 3-second rate.

The format of the upload file is described in Appendix B.

4.6 Running a Simulated Connection to a Receiver

As stated in Section 3.1.4.5, GBSS is capable of creating communication Trap Files. A Trap File can be played back through GBSS in such a way that simulates a connection to the receiver. GBSS can perform this play back in one of two modes: “Play Back” and “Simulation”. Neither mode simulates the exact timing of the data received. They are designed simply to allow GBSS to interpret the data as if it were actually received from a receiver. Obviously, any commands issued by GBSS when operating in these modes are in vain as there is no receiver actually connected to GBSS. It should also be obvious that setting the epoch interval on the Configuration | GPS Receiver menu (see Section 3.1.2.2) has no impact in the data received from the Trap File. That is, GBSS does “simulate” a GPS receiver but can only do so by using the data that is available in the Trap File.

The Play Back mode is started by selecting “File | Play Back” from the GBSS menu bar. The Simulation mode is started by selecting “File | Simulation”. After selecting either of these, you will be provided with a file selection window through which you can specify the file to be played back through GBSS. Once started, you can terminate either mode by terminating the program or selecting “File | Disconnect” from the GBSS menu bar.

Before beginning the playback or simulation, GBSS attempts to determine the start time of the Trap File data (for proper output file naming and *logging session* purposes). Trap Files generated by GBSS version 3.0.00, and higher, contain a special time stamp header, earlier versions do not. If GBSS cannot locate this header in the Trap File, you will be prompted to enter a start time of the Trap File data. An example of this window is shown in Figure 4.15.

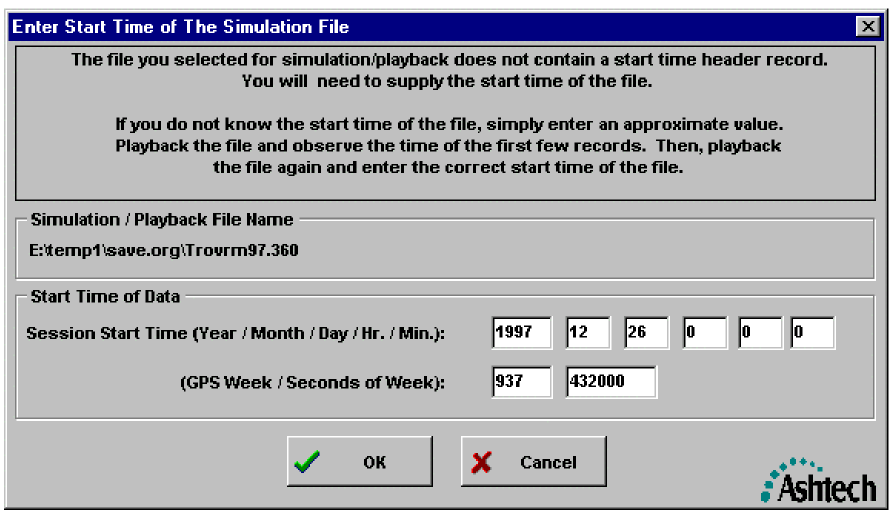


Figure 4.15 Simulation Start Time Window

When this window is displayed, GBSS will gain as much information as possible about the start time of the Trap File based upon the name of the file. However, that information is usually limited to the nearest day. You should enter the correct start time of the Trap File. If you do not know the exact start time, run the program in Simulation Mode (that is, not Play Back Mode), enter an approximate time, press OK, and note the time of the first epoch retrieved from the file. You may need to wait until receiving both an epoch and a navigation message to determine the GPS week of the data, but you should used the time associated with the first epoch received. Then you can terminate the simulation, re-run in either the Simulation or the Play Modes, and enter the correct start time of the file.

The Play Back and Simulation modes only differ in the speed at which the file is played back trough GBSS. In the Simulation Mode, you are provided with slide bar, which controls the speed of the simulation. The slide bar, Figure 4.16, is found in the tool bar (which is just below the program menu bar) and is used to control the speed of the simulation. The Play Back mode provides no user control of the speed at which the file is played back through GBSS.

Figure 4.16 Simulation Speed Slide Bar

GBSS is capable of re-playing Ashtech receiver output data trapped by other programs (such as ProComm or SmartComm). In these cases, you MUST specify the type of receiver that the file was created from in the “Configuration | Receiver” menu (see Section 3.1.2.7). Failure to do so could result in incorrectly interpreted data. To specify the correct receiver type you may need to place GBSS into the Passive mode (see Section 3.1.2.1).

Section 3.1.4.5 of this document describes the format of the Trap File, as well as how to concatenate several Trap Files.

4.7 Automatic Playback from the Command-Line

Sections 4.6 and 4.7 of this manual both describe simulated connections to GPS receivers. As Section 4.6 states, the simulation/playback feature of GBSS allows one to replay a Trap File using other features/configuration values of GBSS. The Automatic Playback feature enables one to playback a Trap File through a command-line call. When GBSS is executed in this fashion, it will automatically playback the file and will terminate upon completion of the playback. This feature, in conjunction with the normal data logging, can be used to create output files with various GBSS controlled attributes (such as epoch intervals). For example, one could, in normal data logging, collect 1-Hz. data in a Trap File and then use the Auto-playback feature to created B-Files with 30-, 20-, 15-, 10-, 5-, and 1-second epochs.

The Automatic Playback feature is invoked as shown in the following example command-line call:

GBSS –P 944 226320 D:\GPSData\TARC1K97.041

The syntax of this call is as follows:

`GBSS -P` *GPSWk GPSWkSec TrapFName*

Where, *GPSWk* The GPS Week of the start of the Trap File specified by *TrapFName*.

GPSWkSec The Seconds of GPS Week of the start of the Trap File specified by *TrapFName*.

TrapFName The name of the Trap File to be used in the playback.

Section 3.1.8.4.1 provides an example of the Automatic Playback feature as it is used in conjunction with the Post-Session Command feature. Please review this section carefully if you wish to use this feature.

A

FILE NAMING APPROACH

Almost every Ashtech file is named based upon some value of GPS time. Because GBSS opens many files before connecting to the receiver, it needs a good approximation of the current GPS time (that is, it cannot ask the GPS receiver for the current GPS time). For this reason the “Configure | GPS Time” menu is provided (see Section 3.1.7) to allow GBSS to generate a corrected CPU GPS time. Using the corrected CPU GPS time, GBSS can generate the names of data files.

GBSS will create the file names based upon this corrected CPU GPS time of the computer (not the receiver provided time) but files are closed based upon the time stored within the GPS data received from the receiver. The "File Duration" parameter of the “Configuration | File Outputs” Menu (see Section 3.1.4.10) specifies the duration (in hours) of the files during the logging. A value of -1 disables the File Duration option. When output files reach their duration, they are closed and files corresponding to the next *session* are automatically opened. Each time the output files reach their "duration" (based upon the GPS time embedded in the data received from the receiver) they are closed and the files for the next *session* are opened. (Please see Section 3.1.5 for a distinction between the terms *session* and *logging session).*

Again, the computation for file closure is based upon the Time of GPS week from the data received. That is, if the duration is one hour, file closures will occur on the hour boundary (that is, the first set of files after a Connection to the receiver is established may not contain one whole hour of data).

In each of the following sections, the phrase “corrected CPU GPS time” is used to refer to the CPU time corrected to GPS time described in Section 3.1.7.

Additionally, GBSS avoids overwriting files with the same name and does not append to existing files (with the exception of Log files). This is particularly important for cases where GBSS is terminated indirectly (such as for a power failure). When GBSS is restarted after one of these indirect terminations, it may generate output file names for files that are already on disk (for example, GBSS is terminated in the middle of a File Duration and is restarted within the same File Duration). When this occurs, GBSS attempts to rename the files already on disk. It does this by appending a ‘Z’ to the tail end of disk resident file name. If there is already a file on disk with this ‘Z’ at the end of the file name (that is, GBSS has already been through this re-name process at least once), GBSS will attempt the rename process using the letter ‘Y’. This rename process will continue until either GBSS finds an available name or all letters of the alphabet are used (in which case GBSS will start to overwrite older files with the same name).

Under Windows 95, GBSS alters the renaming process described above because Windows 95 does not permit file name extensions greater than three characters. Under Windows 95, rather than appending the rename character to the end of the file, GBSS replaces the last digit of the year in the file name with the rename character.

Finally, GBSS supports three file-naming modes (see Section 3.1.4.2). As was Section 3.1.4.2 details, these naming modes are indistinguishable when the “File Duration” parameter (see Section 3.1.4.10) is set to –1 or between 1 and 84. However, when the “File Duration” parameter specifies a duration that is sub-hour, the file naming convention must be changed to support numerous files opened and closed within an hour (that is, a single letter session code is not enough).

The classic Ashtech file names follow an 8.3 name format:

tnnnnsyy.ddd

where,	t	is the file type (B, E, S, or T),
	nnnn	is the 4-character station name,
	s	is the session code,
	yy	is the last two digits of the year, and
	ddd	is the day of the year.

In this classic form, the session code is usually a letter. Under this approach, and with a File Duration of 1.0 hours, the session code can start at ‘A’ (that is, for the 0 hour of the day), increment to ‘B’ on the next session (that is, for hour 1 of the day), and continue with each session until ‘X’ is reached (that is, for hour 23 of the day). However, when the File Duration is less than 1.0

hour, this form of human-readable/discernable file names can no longer be used. Therefore, GBSS provides three file-naming modes:

1. Classic GBSS Names
2. Classic Ashtech Names and Extended RINEX names
3. Extended for both Ashtech and RINEX Names

As can be seen, there are two choices containing “classic” naming and two choices where “extended” naming is available.

Under the “classic” naming form (with the File Duration less than one hour) the normal file naming convention does not apply in that the ‘s’ (that is, session) and ‘ddd’ (that is, day of year) codes become incrementors. At the start of logging, the session code ‘s’ becomes ‘A’ and ‘ddd’ becomes ‘000’. As the File Duration is reached, the files are closed and new ones are opened with the ‘s’ set to ‘A’ and the ‘ddd’ becomes ‘001’. With each File Duration, this process continues until ‘s’ is set to ‘A’ and ‘ddd’ is set to ‘999’. At this point, the increment causes a rollover, where ‘s’ is set to ‘B’ and ‘ddd’ is reset to ‘000’. This rollover process continues until ‘s’ reaches ‘Z’ and ‘ddd’ reaches ‘999’. At this point, each time ‘ddd’ reaches ‘999’ it is simply reset to ‘000’.

Under the “extended” naming form (with the File Duration less than one hour) the session code ‘s’ becomes a three-character field. It basically has the form *hmm*, where *h* is the hour code (that is, ‘A’ to ‘X’) and *mm* represents the start minute of the hour. Because the File Duration parameter has a lower limit of 0.05 (that is, 3 minutes), independent file names can be generated.

In the sections that follow, the “classic” naming modes are detailed. When and “extended” naming form is selected, the *session* code of the file name changes as described above.

A.1 Ashtech Subdirectory Naming Approach

The “Use Ashtech Subdirectory Structure” Checkbox of the “Configuration | File Outputs” menu (see Section 3.1.4.1) indicates whether or not the Ashtech subdirectory naming structure will be employed. When checked, subdirectories, from those specified in the Primary and Secondary directories (both data and compression), will automatically be created and filled. When unchecked, the created files will be stored in the directories specified as Primary and Secondary. The Ashtech subdirectories are of the form:

mmmYY\DAY*dd*

where,	mmm	is the 3-character month of the year
	YY	is the last two digits of the year; and
	dd	is the day of the current month

Example 1:The corrected CPU GPS time indicates that the day is Oct 28, 1997, the “Use Ashtech Subdirectory Structure” Checkbox is checked, and the Primary Subdirectory is `E:\SITE1.DAT`. GBSS would create the following Primary directory:

E:\SITE1.DAT\Oct97\DAY28

All of the data slated for output to the primary directory on Oct. 28, 1997 will be placed in that primary directory.

Example 2:The corrected CPU GPS time indicates that the day is Sept. 7, 1997, the “Use Ashtech Subdirectory Structure” Checkbox is NOT checked, and the Secondary Subdirectory is `C:\GPSDATA\GEO1.` GBSS would place all of the data for every day in the following directory:

C:\GPSDATA\GEO1

A.2 B-, E-, S- and Trap File Naming Approach

GBSS depends upon several sources of information when generating Ashtech file names. The Ashtech files are named using the following naming convention:

tnnnnsyy.ddd

where,	t	is the file type (B, E, S, or T),
	nnnn	is the 4-character station name,
	s	is the *session* code,
	yy	is the last two digits of the year, and
	ddd	is the day of the year.

For example, a file named BRK12C97.233 is an Ashtech B-File for site RK12 taken in *session* C of day 233 of 1997. Using this convention, GBSS creates the names of B-, E-, S- and Trap Files. The 4-character station name is obtained from the "Configuration | Site / Site Name" menu (see Section 3.1.3.1). The *session* code depends upon the "File Duration" parameter of the "Configuration | File Outputs" menu (see Section 3.1.4.10) and upon the corrected CPU GPS time. The corrected CPU GPS time is also used to determine the year and day of the year.

It should be clear that GBSS creates the file names based upon this corrected CPU GPS time of the computer (not the receiver). However, files are closed based upon the time stored within the GPS data received from the receiver. In naming the files, the *session* will be based upon the hour of the corrected CPU GPS time and the "File Duration" parameter of the "Configuration | File Outputs" Menu (see Section 3.1.4.10). For example, if the File Duration is set to one hour, then the *session* code will relate to the hour in which the file is opened (A = hour 0, B = hour 1, C = hour 2, etc..). With each file closure, the *session* will be incremented. When the File Duration is set to 24 hours, the *session* for every file will be "0" (that is, zero).

Sessions are closed, and new ones opened, based upon the GPS time tags of the data being received from the receiver. GBSS uses the correct CPU GPS time only for file naming purposes.

Again, GBSS avoids overwriting files with the same name and does not append to existing files (with the exception of Log files). This is particularly important for cases where GBSS is terminated indirectly (such as for a power failure). When GBSS is restarted after one of these indirect terminations, it may generate output file names for files that are already on disk (for example, GBSS is terminated in the middle of a File Duration and is restarted within the same File Duration). When this occurs, GBSS attempts to rename the files already on disk. It does this by appending a 'Z' to the tail end of disk resident file name. If there is already a file on disk with this 'Z' at the end of the file name (that is, GBSS has already been through this re-name process at least once), GBSS will attempt the rename process using the letter 'Y'. This rename process will continue until either GBSS finds an available name or all letters of the alphabet are used (in which case GBSS will start to overwrite older files with the same name).

Under Windows 95, GBSS alters the renaming process described above because Windows 95 does not permit file name extensions greater than three characters. Under Windows 95, rather than appending the rename character to the end of the file, GBSS replaces the last digit of the year in the file name with the rename character.

Additionally, when recording dual-frequency and single frequency data (as described in Section 3.1.4.5) GBSS distinguishes between the two types of file sets by altering the convention for the L1 only data files. Here, GBSS simply alters the century digit of the year to an 'X'. For example, suppose the dual-frequency B-File name is BASHTA99.312, the corresponding single frequency file name would be BASHTAX9.312.

Finally, there is a distinction between the terms *session* and *logging session* used throughout this documentation. The distinction is detailed in section 3.1.5.

A.3 RINEX File Naming Approach

GBSS used the same information that was used to generate the B-file name to generate the RINEX output file names.

For files of 1-hour duration or greater, the RINEX files are named using the following naming convention:

nnnnddds.yyt

where,	nnnn	is the 4-character station name,
	ddd	is the day of the year,
	s	is the RINEX *session* code,
	yy	is the last two digits of the year, and
	t	is the file type (N for Navigation and O for Observation).

For files of less than 1-hour duration, the RINEX files are named using the following naming convention if one of the two the Extended RINEX names check boxes is checked as described in Section 3.1.4.2:

nnnnddds.yyt

where,	nnnn	is the 4-character station name,
	ddd	is the day of the year,
	h	is the hour of the start time of the file (A=0…X=23)

mm	is the minute of the start time of the file
yy	is the last two digits of the year, and
t	is the file type (N for Navigation and O for Observation).

For example, a file named RK122333.97O is a RINEX Observation file taken in *session* C of day 233 of 1997. Notice that in this case the *session* was translated from a C to the number 3. This is because by convention, 1-hour RINEX *sessions* start with the number '1' and work to '9', then roll over to 'A', and then increment from 'A'. That is, the Ashtech B-File names for Ashtech *sessions* A to I map to RINEX *sessions* 1 to 9 and Ashtech *sessions* J – Z map to RINEX *sessions* A to S. Both the Ashtech and RINEX file names use the '0' *session* code (that is, the number zero) to denote an entire 24-hour data set.

Using this convention, GBSS creates the names of the RINEX Observation and Navigation files. The 4-character station name is obtained from the "Configuration | Site / Site Name" menu (see Section 3.1.3.1). The *session* code depends upon the "File Duration" parameter of the "Configuration | File Outputs" menu (see Section 3.1.4.10) and upon the corrected CPU GPS time. The corrected CPU GPS time is also used to determine the year and day of the year.

Again, GBSS avoids overwriting files with the same name and does not append to existing files (with the exception of Log files). This is particularly important for cases where GBSS is terminated indirectly (such as for a power failure). When GBSS is restarted after one of these indirect terminations, it may generate output file names for files that are already on disk (for example, GBSS is terminated in the middle of a File Duration and is restarted within the same File Duration). When this occurs, GBSS attempts to rename the files already on disk. It does this by appending a 'Z' to the tail end of disk resident file name. If there is already a file on disk with this 'Z' at the end of the file name (that is, GBSS has already been through this re-name process at least once), GBSS will attempt the rename process using the letter 'Y'. This rename process will continue until either GBSS finds an available name or all letters of the alphabet are used (in which case GBSS will start to overwrite older files with the same name).

Under Windows 95, GBSS alters the renaming process described above because Windows 95 does not permit file name extensions greater than three characters. Under Windows 95, rather than appending the rename character to the end of the file, GBSS replaces the last digit of the year in the file name with the rename character.

Additionally, when recording dual-frequency and single frequency RINEX data (as described in Section 3.1.4.5) GBSS distinguishes between the two types of file sets by altering the convention for the L1 only data files. Here, GBSS simply alters the century digit of the year to an 'X'. For example, suppose the dual frequency RINEX file name is ASHT3121.99O, the corresponding single frequency file name would be ASHT3121.X9O.

Finally, there is a distinction between the terms *session* and *logging session* used throughout this documentation. The distinction is detailed in section 3.1.5.

A.4 ION File Naming Approach

The Ionospheric Model file names are of the following form:

IONyy.ddd

where	yy	is the last two digits of the year, and
	ddd	is the day of the year.

For example, a file named ION97.233 is an ionospheric model file for day 233 of 1997.

Like the data file names, the compression file names depend upon the "File Duration" parameter (see Section 3.1.4.10) and the current corrected CPU GPS time (see Section 3.1.7).

A.5 LOG File Naming Approach

The GBSS Log file names are of the following form:

yyddd.LOG

where,	yy	is the last two digits of the year, and
	ddd	is the day of the year.

For example, a file named 97233.LOG is a GBSS Log file for day 233 of 1997. In the year 2000 the file name would begin with '00'.

Like the data file names, the compression file names depend upon the “File Duration” parameter (see Section 3.1.4.10) and the current corrected CPU GPS time (see Section 3.1.7).

A.6 Compression File Naming Approach

Compression file names are of the following form for files of 1-hour duration or greater:

syyddd.ZIP

where, s — is the *session* code,
yy — is the last two digits of the year, and
ddd — is the day of the year.

Compression file names are of the following form for files of less than 1-hour duration:

hmmyyddd.ZIP

where, h — is the hour of the start time of the contained files (A=0…X=23)
mm — is the minute of the start time of the contained files
yy — is the last two digits of the year, and
ddd — is the day of the year.

For example, a file named C97233.ZIP is a compression file for *session* C of day 233 of 1997. A file named A3002233.ZIP is a compression file for a *session* that began at hour 0, minute 30 of day 233 of 2002.

Like the data file names, the compression file names depend upon the “File Duration” parameter (see Section 3.1.4.10) and the current corrected CPU GPS time (see section 3.1.7).

Some users have expressed the desire for GBSS to employ a different compression file naming approach. Please be clear that through the post-session command-line feature (see Section 3.1.5) you have complete control over file names created using PKZIP or PKZIPC. Furthermore, you have complete control over the target directories of the output compression files and the input files compressed by these programs (including any files that may not have been generated using GBSS).

A.7 NMEA Output File Naming Approach

NMEA Message output (or capture) file names of the following form:

syyddd.NMA

where, s — is the *session* code,
yy — is the last two digits of the year, and
ddd — is the day of the year.

For example, a file named C97233.NMA is a NMEA message capture file for *session* C of day 233 of 1997.

Like the data file names, the NMEA capture file names depend upon the “File Duration” parameter (see Section 3.1.4.10) and the current corrected CPU GPS time (see section 3.1.7).

A.5 LOG File Naming Approach

The GBSS Log file names are of the following form:

yyddd.LOG

where, yy — is the last two digits of the year, and
ddd — is the day of the year.

For example, a file named 97233.LOG is a GBSS Log file for day 233 of 1997. In the year 2000 the file name would begin with ‘00’.

Like the data file names, the compression file names depend upon the “File Duration” parameter (see Section 3.1.4.10) and the current corrected CPU GPS time (see Section 3.1.7).

A.6 Compression File Naming Approach

Compression file names are of the following form for files of 1-hour duration or greater:

syyddd.ZIP

where, s is the *session* code,
yy is the last two digits of the year, and
ddd is the day of the year.

Compression file names are of the following form for files of less than 1-hour duration:

hmmyyddd.ZIP

where, h is the hour of the start time of the contained files (A=0…X=23)
mm is the minute of the start time of the contained files
yy is the last two digits of the year, and
ddd is the day of the year.

For example, a file named C97233.ZIP is a compression file for *session* C of day 233 of 1997. A file named A3002233.ZIP is a compression file for a *session* that began at hour 0, minute 30 of day 233 of 2002.

Like the data file names, the compression file names depend upon the "File Duration" parameter (see Section 3.1.4.10) and the current corrected CPU GPS time (see section 3.1.7).

Some users have expressed the desire for GBSS to employ a different compression file naming approach. Please be clear that through the post-session command-line feature (see Section 3.1.5) you have complete control over file names created using PKZIP or PKZIPC. Furthermore, you have complete control over the target directories of the output compression files and the input files compressed by these programs (including any files that may not have been generated using GBSS).

A.7 NMEA Output File Naming Approach

NMEA Message output (or capture) file names of the following form:

syyddd.NMA

where, s is the *session* code,
yy is the last two digits of the year, and
ddd is the day of the year.

For example, a file named C97233.NMA is a NMEA message capture file for *session* C of day 233 of 1997.

Like the data file names, the NMEA capture file names depend upon the "File Duration" parameter (see Section 3.1.4.10) and the current corrected CPU GPS time (see section 3.1.7).

B

UPLOAD FILE FORMAT

Sections 3.1.2.6 and 4.5 of this document describe how a file can be uploaded to an Ashtech receiver. This section describes the required format of those files.

The general format of the Upload File is ASCII text. However, some messages sent to an Ashtech receiver may contain binary data. Almost every message sent to an Ashtech receiver requires the <CR><LF> (carriage return, line feed) sequence at the end of the message. For these reasons we need a means of entering binary data into the upload file. Additionally, there are cases where a pause in uploading data is required before sending the remaining data. To accommodate all of these needs, the upload is sent through an interpreter imbedded in GBSS. This interpreter sends all ASCII text until it meets with a special escape character. After these escape characters will be other characters that describe the information to be sent or the processing to take place. Once the escape processing is handled, GBSS will resume sending the ASCII data in the file.

The special escape character is '\'. If this character is followed by a "+" character, then the escape sequence is considered to be delay processing. A delay sequence is of the form \+*time*+, where *time* is the number milliseconds the interpreter should delay before continuing with the upload. If the character following the escape character is '0' (that is, the zero character), then GBSS is to translate the octal escape sequence into a binary character to be sent. For example, \015 is the octal representation of the carriage return character. To send a '\' character, the character immediately following the '\' must be a '\'. Finally, the carriage returns (or any line feed characters) that are normally part of an ASCII file are not sent as part of the uploaded data, to send these characters you must use the octal representation of the characters.

The following provides an example upload file:

$PASHS,RCI,5.0\015\012
\+100+
$PASHQ,RID,\015\012
\+100+
$PASHQ,,RAW\015\012
\+100+
$PASHS,RCI,15.0\015\012
\+1000+

The interpreter translates the above file as follows:

```
$PASHS,RCI,5.0<CR><LF>

wait 100 milliseconds
$PASHQ,RID<CR><LF>
wait 100 milliseconds
$PASHQ,RAW<CR><LF>
wait 100 milliseconds
$PASHS,RCI,15.0<CR><LF>
wait 1000 milliseconds
```

C

UTILITY PROGRAM ASHFTPMD

C.1 Introduction

Program ASHFTPMD.EXE was developed by The XYZ's of GPS, Inc. in direct support of the Ashtech Geodetic Base Station Software (GBSS).

ASHFTPMD.EXE is a 32-bit Windows program that utilizes WS FTP Pro (developed and sold by Ipswitch, Inc.) to create directories on a remote (that is, host) computer. More specifically, ASHFTPMD was designed to support the Ashtech Geodetic Base Station Software (GBSS) by allowing one, via the GBSS Post-Session Command feature, to create directories on a remote system and then use WS FTP Pro to transfer collected GPS data to those remote system directories via FTP.

C.2 System Requirements

ASHFTPMD imposes the following requirements (but does not necessarily report any errors if the requirements are not met):

1. Windows 95 or Windows NT on and Intel 486 compatible or higher;
2. At least 4 Mbytes memory;
3. At least 1 Mbyte disk space in order to store the program;
4. WS FTP Pro (Version 4.5 or higher); and
5. Underlying socket connection to the Internet.
6. A GBSS Software Sentinel Key

Notice that a software sentinel key protects ASHFTPMD. When distributed with Asthech's GBSS, ASHFTPMD will use the same sentinel key as is provide with GBSS.

C.3 Using ASHFTPMD

Again, ASHFTPMD was designed to create directories on remote systems, if they do not already exist, using the File Transfer Protocol (or FTP). Before continuing one must understand that ASHFTPMD and WS_FTP Professional are two independent software programs. Each one of these programs is designed to connect to some FTP site and perform some operations via FTP commands. In order to perform their operations, both of these programs must connect to the remote (or host) computer. Almost all FTP host sites require some form of login. WS FTP Pro allows one to specify information needed to fully establish a connection with a host system. Among the information specified is the host name (normally some form of an Internet address), the user login name, and a password for that user. Likewise, ASHFTPMD needs this same information. WS FTP stores the needed information as part of its initialization file (or .INI file). However, that information is encrypted and, therefore, unavailable to ASHFTPMD. In short, both WS FTP Pro and ASHFTPMD will require the same entries for the host name, user name, and user password. Both programs will require that entry.

The command-line call to create a directory using ASHFTPMD is as follows:

`ASHFTPMD` *host userid password tgtdir* [-L *logfname*], where

host	is the host name of the target system on which ASHFTPMD will make a directory.
userid	is the user ID for logging into the designated host.
password	is the password associated with userid.
tgtdir	is the target directory on the host computer.
logfname	which is optional, is the name of the file in which to log all FTP transactions using this program.

Please note that the call assumes that the underlying Internet connection has been established and that there is a WinSock-like interface running and available to ASHFTPMD (such as an Internet dialer available under Windows NT.

ASHFTPMD will only attempt to create directories that do not exist. This is particularly useful when used in conjunction with GBSS. That is, you can repeatedly pass ASHFTPMD the same target directory structure (such as the Ashtech sub-directory structure for GBSS that is day of the month based) with each *session* closure. ASHFTPMD will only attempt to create the directory if that directory is not present. ASHFTPMD allows you to specify directories that are more layers deep that currently on the remote system: that is, ASHFTPMD will attempt to create every non-existing sub-path directory in the entire path.

C.4 Troubleshooting

The following lists the error messages output by ASHFTPMD:

1) Message: Memory Allocation Error
"Cannot create memory needed to manage the FTP connection!"

Rationale:The above message is displayed because Windows 95 or NT has lead ASHFTPMD to believe that too much memory was being used. To clear the problem, try exiting any background tasks or threads under your control and then re-run ASHFTPMD.

2) Message: `ASHFTPMD Connection Failure"`
Connection failure status information:"
Return Code = #"
Error Code = #"
Message = *string*"
Line = *string*"

Rationale:The above message is displayed because ASHFTPMD cannot establish a connection with the remote system. This message will most often occur when you do not provide enough information to complete the transaction.

3) Message: AS`HFTPMD Change Directory Failure"`

Failure status information:"
Target Dir = *target directory*\n"
Return Code = #"
Error Code = #"
Message = *string*"
Line = *string*"

Rationale:The above message is displayed because ASHFTPMD cannot perform the change directory command. Notice that the target directory of the command is displayed.

4) Message: `ASHFTPMD Create Directory Failure"`

Failure status information:"
Target Dir = *target directory*\n"
Return Code = #"
Error Code = #"
Message = *string*"
Line = *string*"

Rationale:The above message is displayed because ASHFTPMD cannot get FTP to create a new directory. Notice that the target directory of the command is displayed.

D

UTILITY PROGRAM GNSS2GPS

D.1 Introduction to GNSS2GPS

Program GNSS2GPS.EXE was developed by The XYZ's of GPS, Inc. in direct support of the Ashtech Geodetic Base Station Software (GBSS).

GNSS2GPS was developed as a 32-bit Windows program that creates GPS-only Ashtech data files from Ashtech GPS/GLONASS observation and navigation Files. This program specifically supports the Ashtech Geodetic Base Station Software (GBSS). The program is invoked via command-line parameters.

D.2 System Requirements

GNSS2GPS imposes the following requirements (but does not necessarily report any errors if the requirements are not met):

7. Windows 95 or Windows NT on and Intel 486 compatible or higher;
8. At least 1 Mbytes memory;
9. At least 1 Mbyte disk space in order to store the program;
10. A GBSS Software Sentinel Key

Notice that a software sentinel key protects GNSS2GPS. When distributed with Asthech's GBSS, GNSS2GPS will use the same sentinel key as is provide with GBSS.

D.3 Using GNSS2GPS

Again, GNSS2GPS was designed to create GPS only Ashtech files from Ashtech GPS/GLONASS receiver data files (B- and E-Files). At the time this documentation was published, GNSS2GPS was tested on Ashtech GG-24 and Z18 receivers.

The syntax of the command-line call to create GPS only data files from GPS/GLONASS data files is as follows:

GNSS2GPS [-B *src_bfile tgt_bfile*] [-E *src_efile tgt_efile*]

To create a GPS only B-File from a GPS/GLONASS B-file, use the –B option followed by a source (*src_bfile*) and target (*tgt_bfile*) B-Files. To create a GPS only E-File from a GPS/GLONASS E-file, use the –E option followed by a source (*src_efile*) and target (*tgt_efile*) E-Files. You can, in a single run, create both B- and E-files by using both the B- and –E options.

Examples:

gnss2gps -B BREMDA98.093 C:\GPSOnly.DAT\BREMDA98.093

Creates the GPS Only B-File "C:\GPSOnly.DAT\BREMDA98.093" from the source GPS/GLONASS B-File "BREMDA98.093" stored in the current working directory.

gnss2gps -E EREMDA98.093 C:\GPSOnly.DAT\EREMDA98.093

Creates the GPS Only E-File "C:\GPSOnly.DAT\EREMDA98.093" from the source GPS/GLONASS E-File "EREMDA98.093" stored in the current working directory.

gnss2gps -B BREMDA98.093 BREMDAv8.093 -E EREMDA98.093 EREMDAv8.093

Creates the GPS Only B-File "BREMDAv8.093" from the source GPS/GLONASS B-File "BREMDA98.093". Also creates the GPS Only E-File "EREMDAv8.093" from the source GPS/GLONASS E-File "EREMDA98.093". The input and output directories are the current working directory.

E

UTILITY PROGRAM XYZAshRx

E.1 INTRODUCTION TO XYZAshRX

XYZAshRx.EXE, written by The XYZ's of GPS, Inc., was designed to convert Ashtech raw GPS observation files to the Receiver INdependent EXchange format (RINEX). The RINEX format is described completely in Appendix F. The program has been designed to operate on a Windows 95 or NT platform. To facilitate a wide variety of users, XYZAshRx can be configured and run through normal Windows Graphical User Interfaces (GUI) or launched directly from the command-line. That is, a user can use the conversion program just like most other Windows programs or the converter program can be executed, without need of human intervention, from other programs (such as the Ashtech Geodetic Base Station Software), from a DOS command line, or from batch files.

XYZAshRx currently supports the Ashtech GPS and GPS/GLONASS receivers. Data are output in the RINEX Version 2.0 format. The specification for this version of RINEX came from Dr. Werner Gurtner's paper, "RINEX: The Receiver Independent Exchange Format Version 2", Revised in July of 1998, reproduced in Appendix F.

E.1.1 Minimum System Requirements

XYZAshRx requires the target platform to be a Windows 95 or Windows NT based computer. While XYZAshRx requires less than one megabyte of memory to run, Windows 95 and NT impose higher minimums. You should consult the appropriate Microsoft documentation to determine the minimum system requirements for Windows.

XYZAshRx requires less than 2 megabytes of disk space. However XYZAshRx creates ASCII output files from your Raw Ashtech Observation files. As a general rule of thumb ASCII RINEX files require approximately 1.5 times the space required by Raw Ashtech Observation Files.

E.1.2 Demo Versions

There are two basic configurations of XYZAshRx: fully operational and demonstration versions. This document applies to both configurations. Demonstration versions, which are freely distributed over the Internet or provided on diskette without accompanying sentinel keys, will not be nearly as capable as the operational versions. For example, demonstration versions will only create RINEX output files of 100 epochs or fewer. This document will not seek to delineate the specific differences between the demonstration and the operational versions. It is important to note, however, that the installation instructions documented herein apply to both configurations.

E.2 INSTALLATION OVERVIEW

XYZAshRx is currently distributed as part of GBSS. The program will be installed in the "Utils" sub-directory of GBSS. For example, if during the installation of GBSS, you chose that GBSS be installed into the "D:\Program Files\ASHTECH\GBSS" directory, then XYZAshRx would be installed into the directory "C:\Program Files\ASHTECH\GBSS\Utils".

The automatic installation of GBSS does not put XYZAshRx into the Windows search path. One can add XYZAshRx to the Windows search path by right clicking on "My Computer", selecting the "Properties" menu item, selecting the "Environment" tab, selecting the "Path" system variable and then editing the "Value" field to include the directory in which XYZAshRx is stored.

Additionally, the automatic installation of GBSS does not put an icon onto the desk top nor in the Start Menu bar. To create a program icon on the desktop, simply find the program file XYZAshRx.EXE using "My Computer", right click and drag the icon to the desktop and select "Create Shortcut Here". For details on how to add a program to the Start Menu, consult Windows Help searching on "Start Menu" and selecting the topic dealing with adding menu items to the Start Menu.

Finally, XYZAshRx is protected by a software sentinel key (the same key as is used by GBSS). The software sentinel key is installed by attaching the end of the sentinel key labeled ↑COMPUTER↑ to a parallel printer port of your computer. Tighten the screws of the sentinel key to connect the key securely to your computer. If a printer was connected to your computer, attach that cable to the sentinel. If the sentinel cannot be installed because of an obstruction behind the computer, you can place the sentinel key later in the parallel sequence (for example, you could attach the sentinel key to a DB-25 male to DB-25 female cable which is connected to your computer's parallel port).

E.3 RUNNING XYZAshRx

XYZAshRx creates RINEX Version 2.0 files from Raw Ashtech Observation files. The program can be manually instructed to convert files or can be called directly from the command-line. This latter method allows programs to invoke the converter without the need of human intervention. Throughout the remainder of this documentation, we will call the former approach the Manual/GUI approach and the latter will be called the Command-Line approach.

Both the Manual/GUI and the Command-Line approaches rely upon configuration information contained in the INI file associated with the converter program. This INI file is called "XYZAshRx.INI" and is stored in the same directory as the main program. Changes to this configuration file occur whenever the user changes configuration using the Manual/GUI approach. That is, when desiring to use the Command-Line mode, you should pre-set the configuration using the Manual/GUI approach. Section E.3.1 will describe the Manual/GUI approach to using XYZAshRx and Section E.3.2 will describe the Command-Line approach to using XYZAshRx.

It is important to re-state the fact that the configuration information is stored in the file "XYZAshRx.INI" which is stored in the same directory as the program file (that is, "XYZAshRx.EXE"). When you desire an alternative configuration, for example to support two GBSS installations simultaneously, you should copy the following files to another directory:

XYZAshRx.INI

XYZAshRx.EXE

XYZAshRx.BMP

By doing this, you create an independent copy of the .INI file. Launching the program XYZAshRx.EXE in this copied directory causes the program to use the INI file in that directory. This copy would utilize a configuration that is independent of the other copy (that is, the master copy) of XYZAshRx on your computer. In contrast, launching two copies of the XYZAshRx from the same directory access the same INI file. The configuration stored at the end of the runs of the programs is highly dependent upon which copy of the program terminated first. Needless to say, you are strongly advised not to launch more than one copy of XYZAshRx from the same directory (unless both copies will use the exact same configuration information). Launching several copies of XYZAshRx, each originating from its own directory, is perfectly acceptable and encouraged.

E.3.1 Manual/GUI Approach

Upon starting XYZAshRx without command-line parameters, the screen shown in Figure E.1 is displayed.

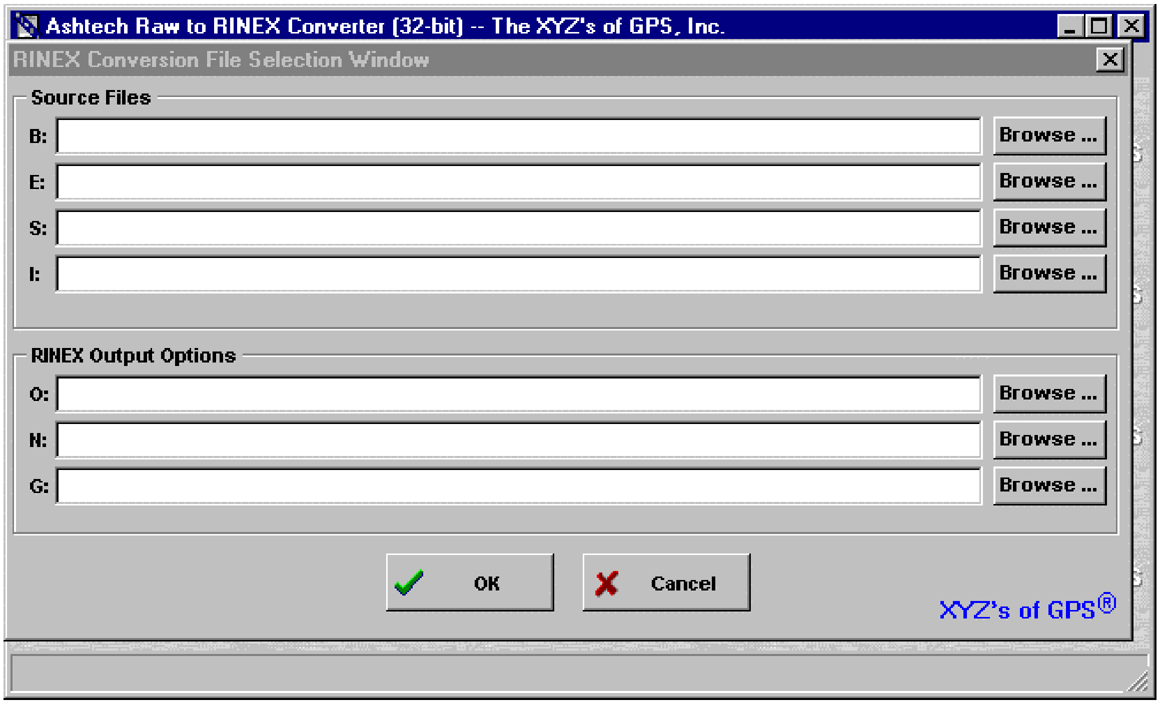


Figure E.1 Ashtech Raw to RINEX Window 1

Presented above are the Main Window of the program and the File Selection Window. The File Selection Window is presented automatically at the start of the program as a means of saving time and will be described in more detail in Section E.3.1.1. For now, we will assume that the File Selection Window has been closed to facilitate the description of the program's main window. The main window of the program is shown in Figure E.2.

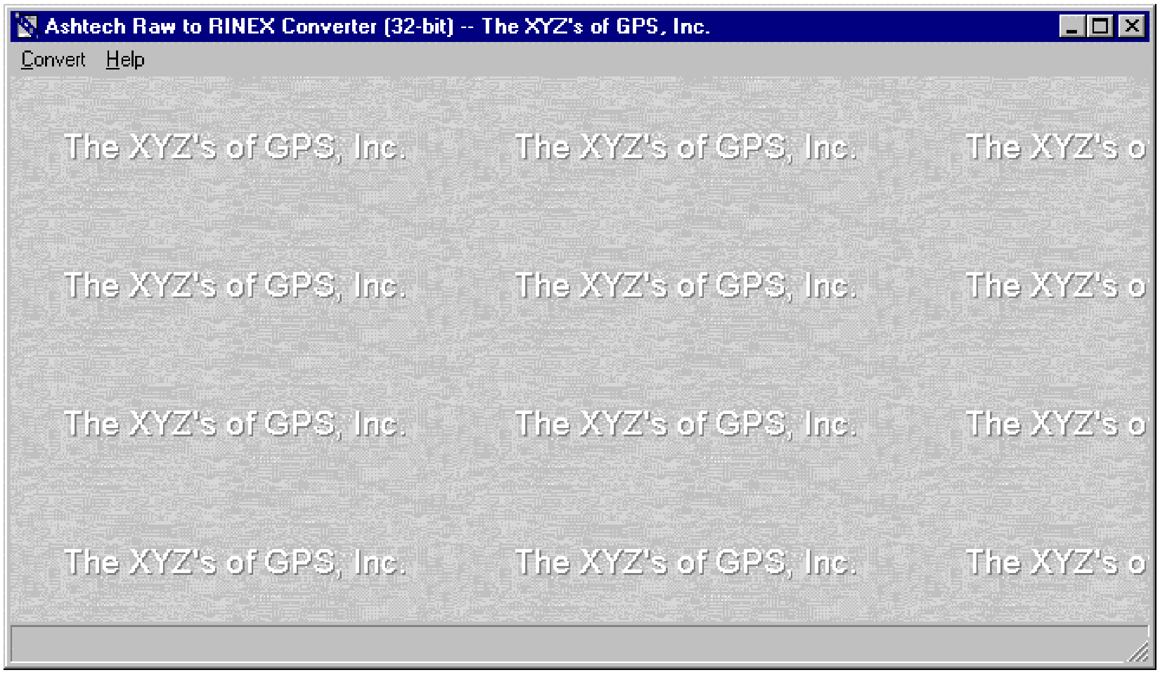


Figure E.2 Ashtech Raw to RINEX Window 2

Notice the program has only two main menu items. When you select the “Convert” menu item you will be presented with the following drop-down menu, Figure E.3.

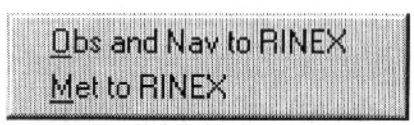


Figure E.3 OBs and Met Drop-Down

That is, you can only “Convert” “To RINEX” using XYZAshRx. Upon selecting the “Obs and Nav to RINEX” option, you will be presented with the File Selection Window presented in Section E.3.1.1. Upon selecting the “Met to RINEX” option, you will be presented with the RINEX Meteorological File Selection” Window presented in Section E.3.1.2. Through both of these windows, you will select the input and output files of a single conversion run.

When you select the “Help” menu item from the main menu, you will be presented with the command line help drop-down menu, Figure E.4.

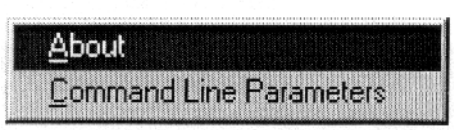


Figure E.4 Command Line Help Drop-Down

When you select the “About” menu item XYZAshRx will display a window containing information about the program. In response to selecting the “Command Line Parameters” item XYZAshRx will display a window describing how to invoke the Command-Line Mode of the program. The command-line mode is further documented in Section E.3.2 of this manual.

E.3.1.1 File Selection Window

The File Selection Window is displayed in response to the “Convert | To RINEX” selection from the program’s main menu. Figure E.5 provides an example of the File Selection Window:

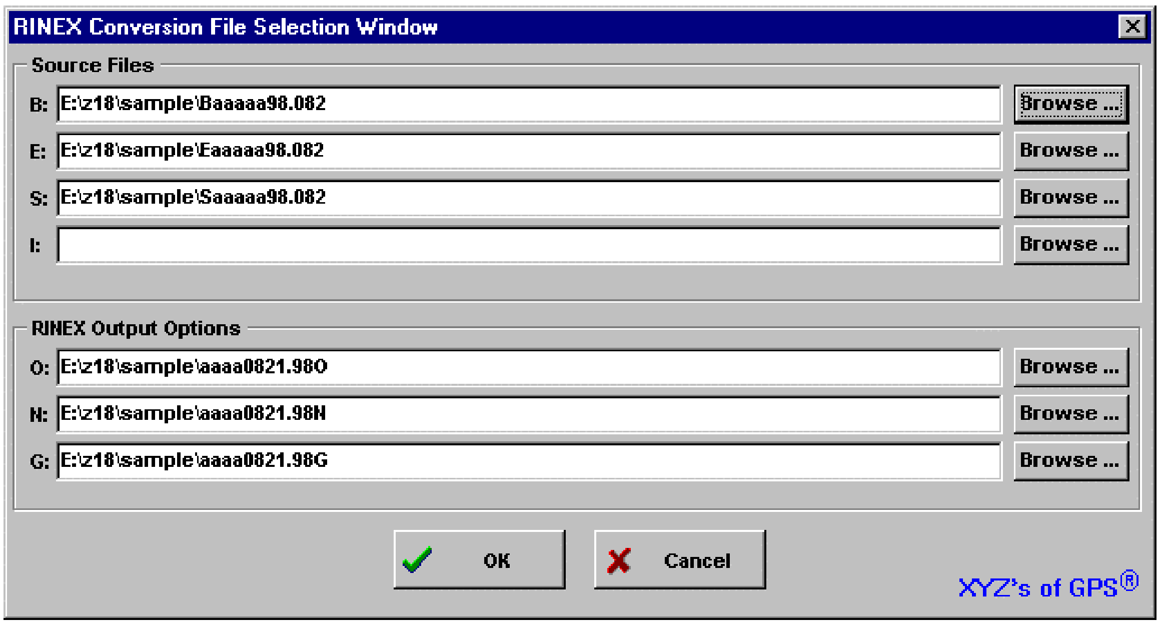


Figure E.5 RINEX Conversion File Selection Window

Table E.1 describes the fields in this window.

Table E.1 Edit Fields

Label	Description	Required
There are seven edit fields in this window. The first four correspond to the following input file names		
B:	Input Ashtech formatted B-File (raw observations)	Yes
E:	Input Ashtech formatted E-File (broadcast orbits)	Yes
S:	Input Ashtech formatted S-File (site file)	No
I:	Input Ashtech formatted I-File (ionospheric data)	Yes/No
The latter 3 edit fields correspond to the following output file names		
O:	Output RINEX Observation File	No/Yes
N:	Output RINEX GPS Navigation File	No/Yes
G:	Output RINEX GLONASS Navigation File	No/Yes

XYZAshRx seeks to help with the file naming by keying off of the source B-File. Each time the edit field of source B-File is changed and contains a valid B-File name (that is, the file exists on disk), XYZAshRx will automatically fill in the other input file names for the files that exist on disk and will automatically create the output file names. In the sample window above, the user used the "Browse…" button of the B-File to select input B-File and XYZAshRx found associated E- and S-Files on the disk (but not an associated I-File) and automatically named the output RINEX Observation and Navigation files.

In the input file table above, notice that the S- and I-Files are optional. When these files are not specified XYZAshRx automatically makes changes necessary to output header data in the associated RINEX files. In particular, when the S-File is specified, XYZAshRx sets certain default values, for the output RINEX header, based on data contained in the S-File. When the S-File name is not provided, the site related header data is obtained from and saved to the configuration file. The RINEX header data affected by the presence of the S-File are as follows:

Station Name
Observer's Name
Receiver Serial Number
Antenna Serial Number
Antenna Offsets (Slant, Radius, and Delta Vertical)

Obviously, the above information would be obtained from the S-File if it were present.

Similarly, the presence of the I-File affects the header of the output RINEX Navigation File. For GLONASS RINEX conversions, the I-File is required. When the I-File is present, XYZAshRx places the ionospheric header data (containing the model alphas and betas) and the leap second header data (that is, the data needed to determine the time delta between GPS time and UTC) into the RINEX Navigation File. When the I-File is not present, these header components will not be output to the RINEX Navigation File.

Notice that in the output file table above, the output RINEX Navigation and Observation files appear to be both optional and required. This is because at least XYZAshRx requires one of the output files.

Once the file names are entered, press OK and a window allowing you to edit/change the header data is displayed (See Section E.3.1.1.1)

E.3.1.1.1 RINEX Header Data Edit Window

The RINEX Header Data Edit Window is displayed in response to pressing the OK button on the File Selection Window (See Section E.3.1.1). Figure E.6 provides an example of this window:

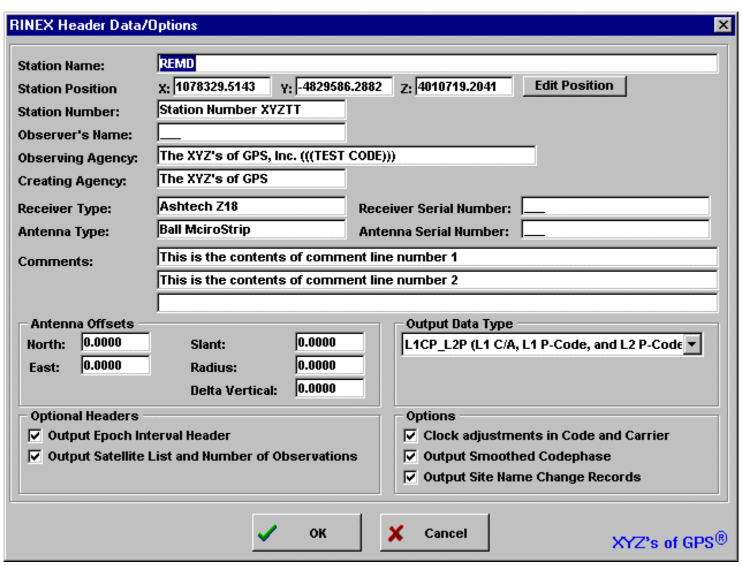


Figure E.6 RINEX Header Data Edit Window

With the exception of the “Output Data Type” the data of each field comes from either the Site File or from the saved configuration data of the program (that is, stored in the INI file of the program). When the Site File is not used as input to the program, all of the data for this window, with the exception of the “Output Data Type”, comes from the saved configuration data. When the Site File is used as input, the following fields come from that file:

Station Name
Observer’s Name
Receiver Serial Number
Antenna Serial Number
Antenna Offsets (Slant, Radius, and Delta Vertical)

The “Output Data Type” field is used to specify which RINEX observation fields will be output to the RINEX Observation file. The “Output Data Type” can be one of those listed in Table E.2.

Table E.2 Output Data Types

Selection	Output Data Types
L1	L1 C/A Carrier and Code (includes Doppler)
L1CP_L2P	L1 C/A, L1 P-Code and L2 P-Code Carrier and Code (includes Doppler)
CDPHASE	L1 C/A Codephase only
L1CP	L1 C/A and L1 P-Code Carrier and Code (includes Doppler)
L1C_L2P	L1 C/A and L2 P-Code Carrier and Code (includes Doppler)

It should be clear that some selections would not be available for some receiver types. For example, if the input data is from a G-12 receiver (a C/A L1 only Code and Carrier receiver), the selectable output data types for the receiver will not include any of the P-Code or L2 observables. By default, the selection will be set to output all possible observation types for the input receiver type.

The receiver position (labeled as “Station Position” on the window) will come from either the B-File data or from a user entered position. This decision is made using the Site Position Window which is accessed by pressing the “Edit Position” button on this window. For further details on the Site Position Window see Section E.3.1.1.2.

Remember the changes entered into this RINEX Header Data Edit Window will be saved as part of the program’s configuration data (that is, stored in the program’s .INI file). In this way, the header data need only be configured once. Minor changes to the output header data can then be made during each subsequent run of the program.

E.3.1.1.2 RINEX Site Position Window

The RINEX Site Position Window permits you to select the source of the position data output as part of the RINEX header data. This window is accessed from the RINEX Header Data Edit Window (see Section E.3.1.1.1). Figure E.7 is an example of the RINEX Site Position Window:

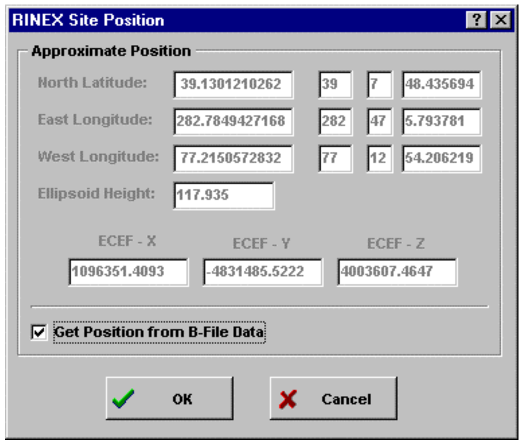


Figure E.7 RINEX Site Position Window

The RINEX Site Position Window allows you to specify the position of the receiver to be output as part of the RINEX header data. The RINEX specification indicates that this position need only be approximate and provides no specifics on the required accuracy of the approximation. The position written as part of the header data can come from a manually entered position or from a weighted average of the position and PDOP data of the B-File. To specify that the approximate position is to come from the weighted average of the B-File position data, check the "Get Position form B-File Data" checkbox. Otherwise, uncheck the box and enter the approximate position in the window.

Like the RINEX Header Data Edit Window, the data entered into this window will be saved as part of the configuration information for the program. If the "Get Position form B-File Data" checkbox is checked, then XYZAshRx will obtain the approximate position from B-File on the current and later runs.

E.3.1.2 RINEX Meteorological Files

Before continuing, it must be noted that the meteorological option of the RINEX converter is not available unless the presence of a software sentinel key coded for the GBSS Meteorological module can be found.

To convert collected meteorological data to RINEX using the manual approach simply launch the program XYZAshRx (with no command-line options). On doing so, you will be provided with the main program screen. Using the main menu of the program select the "Convert" menu option and then select the "Met to RINEX" sub-menu option, Figure E.8.

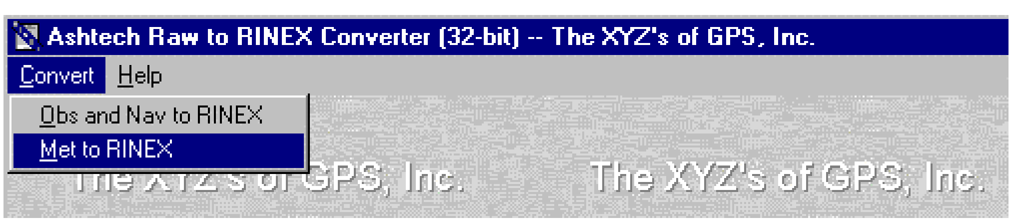


Figure E.8 Met to Rinex Option

Selecting this sub-menu option will provide you with a screen that looks similar to Figure E.9.

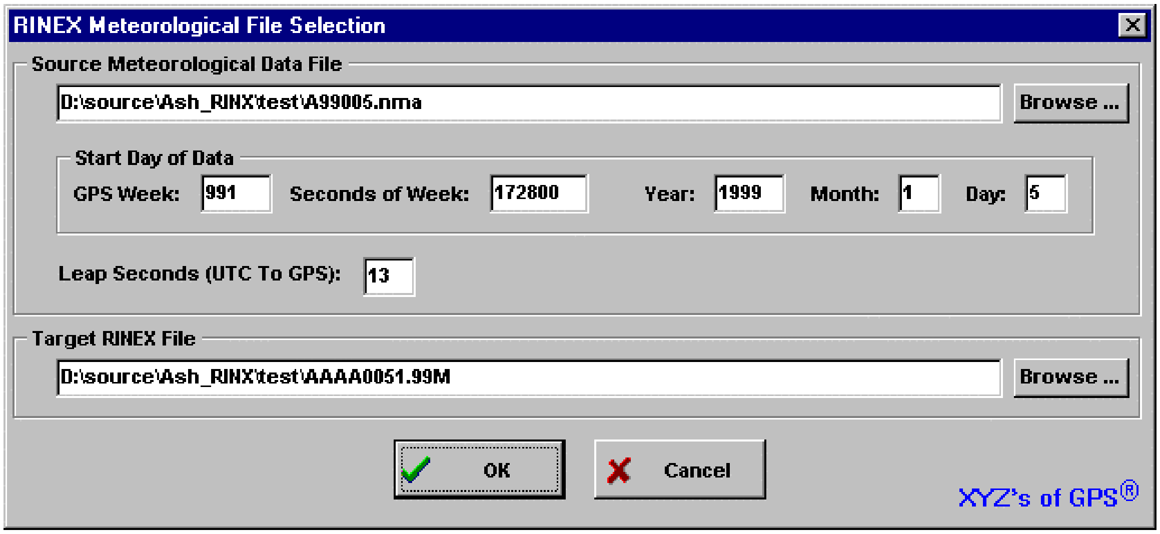


Figure E.9 RINEX Meteorological File Selection Window

The primary components of this screen are:

1. The source meteorological data file name.
2. Start day of the source meteorological data file.
3. Leap seconds for converting UTC to GPS times.
4. The target RINEX meteorological data file name.

Upon completing the edits, press the "OK" button to enter/edit the header data related to the output RINEX meteorological data (see Section A.2.5).

E.3.1.2.1 Specifying the Source Meteorological Data File

To select the source meteorological data file, simply type its name in the first prompt field provided or use the "Browse" button next to that field. By pressing the "Browse" button, you will be provided with a file selection window similar to that available in other Windows type programs.

The file type you will be looking for will normally have a ".NMA" extent. This is because the file is actually created by GBSS and stored in the Ashtech NMEA file format. This is important because you must instruct GBSS to store the NMEA file (via its "Configuration | File Outputs" menu options). Furthermore GBSS must be configured to store the XDR and GXP NMEA messages. This is because the XDR message will contain the meteorological sensor data and the GXP message will contain the time tag associated with the XDR message. Please note that XYZAshRx assumes that the Meteorological data is sampled near the time stated in the GXP message. That said, it should be clear that the accuracy of the time tag as it relates to the XDR message is +/- the rate at which the GXP messages are being output.

E.3.1.2.2 Start Day of the Meteorological Data File

Two key facts must be considered when converting the NMEA capture file created by GBSS: 1) the NMEA GXP message contains only time of day time tags and 2) the NMEA GXP message contains UTC time tags. In this Section, we address item 1; item 2 is addressed in Section E.3.1.2.3.

Because the NMEA GXP message (that is, the message used to time tag the meteorological data) contains only time of day, we must resolve the start day of the data file. It is for this reason that the "Start Day of Data" prompts are provide in the file selection window. You can specify the start day either through GPS time or through a Gregorian time. When you specify the GPS time, you need only be accurate to the day in which the data starts. If you specify a seconds of GPS week which falls within the day (that is, it does not specify a time that is on the exact start of the day) the program will be able to resolve the actual start time of the file. That is, simply specify a time that falls within the correct day in which the start of the file belongs. If the file contains data from multiple days of data, the program will be able to resolve those day changes as well.

If the input meteorological data file name is in the Ashtech file-naming format, the program will be able to resolve the start time of the data without needing your assistance (but still giving you the chance to override the start time of the file). Once you specify the name of the file, the program will parse that name and determine if it fits the proper naming convention and, if so, it will fill in the fields with the start time of the file. Otherwise the program will default to a start time based upon the CPU clock (which you should override if it not correct).

E.3.1.2.3 Leap Seconds: UTC to GPS Conversion

Two key facts must be considered when converting the NMEA capture file created by GBSS: 1) the NMEA GXP message contains only time of day time tags and 2) the message contains UTC time tags. In this Section, we address item 2; item 1 is addressed in Section E.3.1.2.2.

The RINEX standard requires that the time tags of the meteorological data be in GPS time. The NMEA GXP message (that is, the message used to time tag the meteorological data) contains UTC time stamps. As such, we must convert the times to GPS. The "Leap Seconds" data entry field of the window allows you to specify the delta between GPS and UTC. As of January 8, 1999, there are 13 leap seconds between UTC and GPS. That is, UTC + 13 = GPS time.

The value of the "Leap Seconds" field will be saved between runs of the program. Furthermore it will be used during the command-line conversion of meteorological data (see Section A.3).

E.3.1.2.4 Specifying the Output RINEX Meteorological Data File

The output file name is specified in one of 4 ways:

1. The input meteorological data file name complies with the Ashtech naming convention;
2. You cursor through or edit any of the "Start Day of Data" edit fields;
3. You manually enter the output file name; and
4. You use the "Browse" button next to the edit field pertaining to the output file name.

When approaches 1 or 2 are used, the output file name will comply with the RINEX file naming convention (for details on the file naming convention, see the GBSS User's Manual).

To select the source meteorological data file, simply type its name in the first prompt field provided or use the "Browse" button next to that field. By pressing the "Browse" button, you will be provide with a file selection window similar to that available in other Windows type programs.

E.3.1.2.5 Entering the RINEX Header Data

After pressing the "OK" button of the window in which you specified the source and target files, you will be provided with a window similar to Figure E.10.

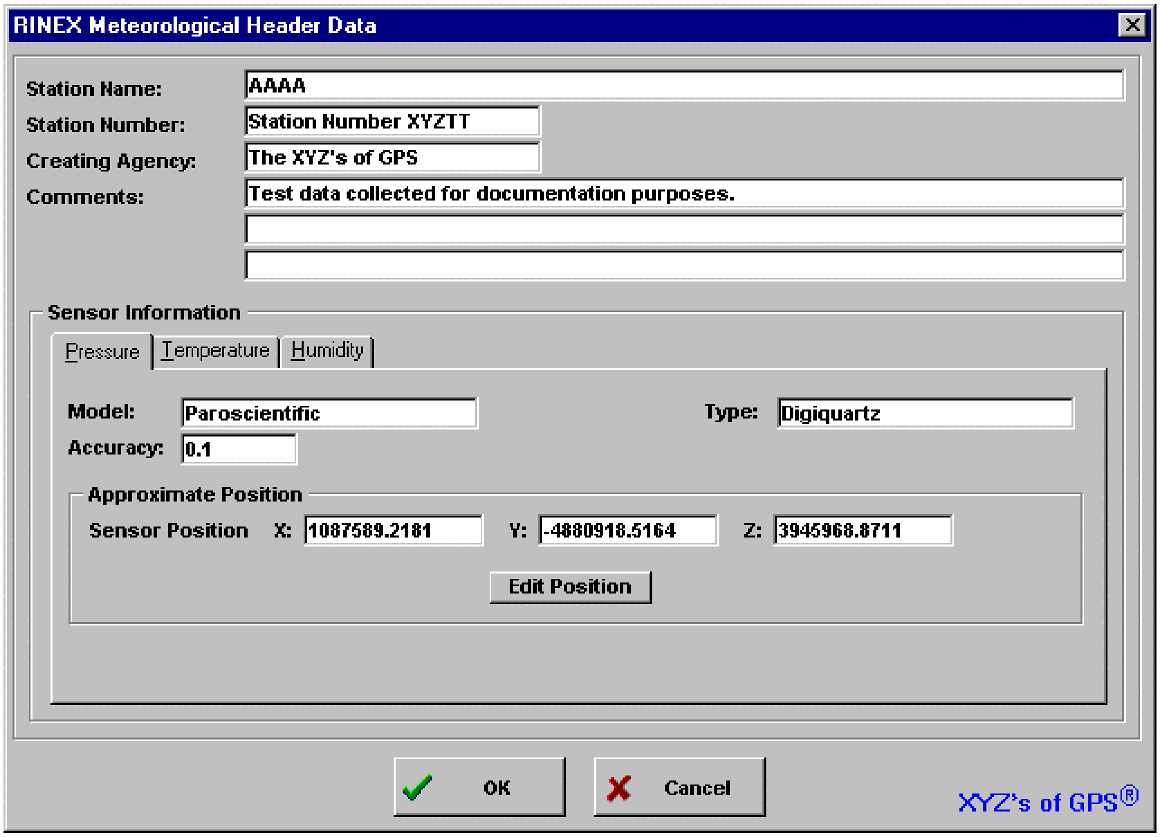


Figure E.10 RINEX Meteorological Header Data Window

In this window you can edit all of the data related to the header data of the output RINEX meteorological file. To obtain the specifics of the meanings of each of these files it is suggested that you consult the RINEX standard documentation.

The data of this window will be saved between runs of the program. Furthermore, this data will be used in the command-line mode when creating the output file.

E.3.2 Command-Line Approach

The Command-Line approach to running XYZAshRx is invoked by calling the program with the command-line parameters listed in this section. The Command-Line approach permits automatic conversion to RINEX of raw Ashtech data files without human intervention. In this way, programs or batch files can automatically convert Ashtech data files to RINEX.

Sections E.3.1.1.1, E.3.1.1.2, and E.3.1.2.5 discuss information that is written to the RINEX header when using the Manual/GUI approach. This RINEX header information is also written to the output files when using the Command-Line approach but it is assumed that you have pre-configured the program with the desired information. The pre-configuration is performed by using the Manual/GUI method described in Sections E.3.1.1.1, E.3.1.1.2, and E.3.1.2.5. Once the configuration has been entered using the menus of Sections E.3.1.1.1, E.3.1.1.2 and E.3.1.2.5 and you have exited the Manual/GUI run of the program, the configuration will be saved and usable from the Command-Line approach.

Before continuing, it is important to re-state that the S-File and I-File inputs are optional. However, the I-File is required when converting GPS/GLONASS data files. As is discussed in Section E.3.1.1, the absence or presence of the S-File and/or the I-File affects header data that is output to both the RINEX Navigation and Observation Files. Please consult Section E.3.1.1 for the specifics in this area as they also apply under the Command-Line approach.

The general form of the command-line call to XYZAshRx is as follows:

XYZASHRX.EXE -I *bfile efile* [*sfile*] [*ionfile*] [*options*]

Notice that you are minimally required to use the “-I” option and specify the names of the B- and E-Files (parameters *bfile efile,* respectivel*y*). All items enclosed in “[“ and “]” are optional. For example, the names of the S-File and I-File are optional. When specifying the input file names, it is assumed that either the current working directory contains the source data files or the full path to the files is specified as part of each file name. For example, the following call assumes that the input B-, E- and S- Files come from the current working directory:

Example 1:

XYZAshRx -I BN102A97.323 EN102A97.323 SN102A97.323

In the following example the B- file comes from the directory D:\BDATA and the S-file comes from the directory E:\EDATA (notice that no S-File is specified):

Example 2:

XYZAshRx -I D:\BDATA\BN102A97.323 E:\EDATA\EN102A97.323

The list of options for the program can be combined in any order and are as follows:

-O *obsfile*
-N *navfile*
-T *path*
-S *smooth*
-C *change*

When neither the “–N” nor the “-O” options are specified, XYZAshRx will automatically create the names of the output RINEX Navigation and Observation files. Additionally in this case, XYZAshRx will place these output files in the current working directory. When either the “–N” or the “–O” parameters are specified, XYZAshRx will only output the associated file type. Some examples are listed below.

Example 3:

XYZAshRx -I BN102A97.323 EN102A97.323 SN102A97.323

The converter uses the files BN102A97.323, EN102A97.323, and SN102A97.323, from the current working directory, as the input B-, E-, and S-Files, respectively. Because no -O or -N parameters were specified, XYZAshRx will automatically name the RINEX observation and navigation output files. The output files will be placed in the same directory as that of the B-File, which is the current working directory in this case.

Example 4:

XYZAshRx -I BN102A97.323 EN102A97.323 –O OBS.DAT –N NAV.DAT

The converter uses the files BN102A97.323 and EN102A97.323 from the current working directory as the input B- and E-Files, respectively. Because no S-FILE was specified, the converter will use the site related data stored in the program's configuration file for site related data. Because the -O parameter was specified, the converter will output the Observation data to the file OBS.DAT. The -N parameter specifies that the Navigation RINEX data will be output to the file NAV.DAT. The output files will be placed in the same directory as that of the B-File, which is the current working directory in this case.

The “-T” parameter allows you to specify the target path of the output RINEX files. Some examples are listed below.

Example 5:

XYZAshRx -I BN102A97.323 EN102A97.323 SN102A97.323 –T D:\RINEX.OUT

The converter uses the files BN102A97.323, EN102A97.323, and SN102A97.323, from the current working directory, as the input B-, E-, and S-Files, respectively. Because no -O or -N parameters were specified, XYZAshRx will automatically name the RINEX observation and navigation output files but will place these files in the directory D:\RINEX.OUT.

Example 6:

XYZAshRx -I BN102A97.323 EN102A97.323 –O OBS.DAT –N NAV.DAT –T E:\RINEX.OUT

The converter uses the files BN102A97.323 and EN102A97.323 from the current working directory as the input B- and E-Files, respectively. Because no S-FILE was specified, the converter will use the site related data stored in the program's configuration file for site related data. Because the -O parameter was specified the converter will output the Observation data to the file OBS.DAT. The -N parameter specifies that the Navigation RINEX data will be output to the file NAV.DAT. The output files will be placed in the directory E:\RINEX.OUT.

The "-S" option allows you to specify whether the converter will output smoothed codephase measurements. Permissible values for this parameter are 0 and 1; the value 1 directs the program to output the smoothed codephase contained in the Ashtech B-File. When the –S option is not specified, XYZAshRx will apply the smoothing as if "–S 1" were specified on the command-line. The –C option is used to specify whether or not the site name change records are to be written to the output RINEX Observation file. Permissible values for this parameter are 0 and 1; the value 1 directs the program to output the site name change records. When the –C option is not specified, XYZAshRx will output site name change records as if "–C 1" were specified on the command-line. Some examples are listed below.

Example 7:

XYZAshRx -I BN102A97.323 EN102A97.323 SN102A97.323 –S 1

The converter uses the files BN102A97.323, EN102A97.323, and SN102A97.323, from the current working directory, as the input B-, E-, and S-Files, respectively. Because no -O or -N parameters were specified, XYZAshRx will automatically name the RINEX observation and navigation output files. The output files will be placed in the same directory as that of the B-File, which is the current working directory in this case. Additionally, the -S parameter specifies that the converter will output Ashtech smoothed codephase observations.

Example 8:

XYZAshRx -I BN102A97.323 EN102A97.323 -p D:\RINEX.OUT -C 0

The converter uses the files BN102A97.323 and EN102A97.323 from the current working directory, as the input B- and E-Files, respectively. Because no -O or -N parameters were specified, XYZAshRx will automatically name the RINEX observation and navigation output files but will place these files in the directory D:\RINEX.OUT. Additionally, the -C parameter specifies that there will be no (that is 0) site change records written to the output RINEX observation file.

To convert a meteorological file (stored in a NMEA capture file) from the command-line, use the following form to call XYZAshRx:

XYZAshRx -M *inmetfile outmetfile* [*gpswk gpswksec*]

where,

inmetfile	The name of the input meteorological file. If no path is provided, it is assumed that the file will come from the current working directory.
outmetfile	The name of the output RINEX meteorological file. If no path is provided, it is assumed that the file will be output to the same directory in which the input file is stored.
gpswk	Specifies the start day of the input file. This field allows you to specify the GPS week component of time.

`gpswksec` Specify the start day of the input file. This field allows you to specify the seconds of GPS week component of time.

Notice that the *gpswk* and *gpswksec* parameters are optional. However, if you specify either then you must specify both, even if *gpswksec* is zero. If you choose not to specify the start time of the file, the program will use the current CPU time to determine the start day of the data (you are strongly encouraged to avoid this approach).

All other parameters used in the conversion, such as leap seconds and header data (see Section E.3.1.2), will use values maintained from the last GUI run of the program.

The following provides an example:

XYZAshRx A99005.NMA N1020051.99M 991 176752

In this example the converter uses the file A99005.NMA from the current working directory as the input meteorological data file. The output RINEX meteorological data file will be placed in the file N1020051.99M. Furthermore the converter will use the GPS time 991, 176752 (that is, the GPS week and seconds of GPS Week, respectively) to designate the start day of the source meteorological data. Notice that 176752 does not fall on an exact day boundary. This is acceptable because the source data is time stamped with time of day. Therefore, the specified GPS time need only be accurate to the day in which the data falls as the converter will resolve the time of day within that day. Again, the value stored for the leap seconds parameter from the last GUI run of the program will be used to convert UTC time tags to GPS time (see Section E.3.1.2).

Finally, please notice that using the command-line approach allows you to use the post-session commands of GBSS to automatically convert your meteorological data to RINEX when GBSS ends a collection session. The following is an example of a GBSS post-session command:

Command:

C:\Program Files\ASHTECH\GBSS\UTILS\XYZAshRx.exe -M NFP $SITE$$DDD$$S$.$YY$M GPW GPS

Working directory of the command:

C:\Program Files\ASHTECH\GBSS\UTILS

In the above example, the name of the NMEA capture file (including its full path) is passed to the conversion program using the GBSS mnemonic `$NFP$`. The output file name is created by concatenating several mnemonics and other characters. That is, we concatenate **`$SITE$`**, **`$DDD$`**, **`$S$`**, "." (that is, the period character), **`$YY$`**, and "**`M`**" (that is, the character M) to get **`$SITE$$DDD$$S$.$YY$M`**. The mnemonics and their meanings are as follows:

`$SITE$`	The four-character site name of the file.
`$DDD$`	The three-digit day of the year.
`$S$`	The single-character session code of the file.
`$YY$`	The two-digit year of the session.

Because we did not specify the directory of the output file, XYZAshRx uses the same directory as the input file.

Finally, we have also used the mnemonics for the time of the start of the session. Specifically, we have used `$GPW$` and **`$GPS$`** for GPS Week for the start of the session and seconds of GPS week for the start of the session, respectively.

RINEX FILE FORMATS

RINEX VERSION 2

RINEX: The Receiver Independent Exchange Format Version 2

Werner Gurtner, Astronomical Institute, University of Berne

(Revision, April 1993)

(Clarification December 1993)

(Doppler Definition: January 1994)

(PR Clarification: October 1994)

(Wlfact Clarification: February 1995)

(Event Time Frame Clarification: May 1996)

(Minor errors in the examples A7/A8: May 1996)

(Naming convention for compressed met files; January 1997)

(Continuation line clarifications: April 1997)

(GLONASS Extensions: April 1997)

(Met sensor description and position records: April 1997)

(Wavelength factor clarifications: April 1997)

(Error in example A12: CORR TO SYSTEM TIME, April 1997)

(Redefinition of sv clock params in GLONASS Nav Mess Files: March 1998)

(Naming conventions for compressed RINEX obs files: March 1998)

(GPS week: No roll-over, continuous number: March 1998)

(Error in compressed DOS file naming convention: July 1998)

(Table A13 contained blank satellite identifiers: Sept 1998)

(Discrepancy between Tables A5 and A9 removed: Sept 1998)

INTRODUCTION

First Revision

This paper is a revised version of the one published by W. Gurtner and G. Mader in the CSTG GPS Bulletin of September/October 1990. The main reason for a revision is the new treatment of anti-spoofing data by the RINEX format (see chapter 7). Chapter 4 gives a recommendation for data compression procedures, especially useful when large amounts of data are exchanged through computer networks. In Table A3 in the original paper the definition of the "PGM / RUN BY / DATE" navigation header record was missing, although the example showed it. The redefinition of AODE/AODC to IODE/IODC also asks for an update of the format description. For consistency reasons we also defined a Version 2 format for the Meteorological Data files (inclusion of a END OF HEADER record and an optional MARKER NUMBER record).

* The slight modification (or rather the definition of a bit in the Loss *

* of Lock Indicator unused so far) to flag AS data is so small a change   *

* that we decided to NOT increase the version number!   *

Later Revisions:
* URA Clarification (10-Dec-93):

The user range accuracy in the Navigation Message File did not contain a definition of the units: There existed two ways of interpretation: Either the 4 bit value from the original message or the converted value in meters according to GPS ICD-200. In order to simplify the interpretation for the user of the RINEX files I propose the bits to be converted into meters prior to RINEX file creation.

* GLONASS Extensions:

In March 1997 a proposal for extensions to the current RINEX definitions based on experiences collected with GLONASS only and mixed GPS/GLONASS data files was circulated among several instrument manufacturers and software developers. The results of the call for comments have been worked into this document. A separate document (glonass.txt) summarizes just the necessary extensions.

* A blank satellite identifier is allowed in pure GPS files only

* Met sensor description and position records were added to facilitate the precise use of met values.

* Description and examples for wavelength factors and their temporary changes (bit 1 of LLI) clarified.

* The RINEX documentation distributed in spring 1997 contained definitions for the GLONASS satellite clock offset and drift with the intention to have them defined identically to the GPS values. Unfortunately the GLONASS Interface Document consulted had a sign error in one of the formulae.

The values should be stored into the RINEX file as -TauN, +GammaN, -TauC.

The original definition asked for -TauN, -GammaN, +TauC. See paragraph 8.2.

To avoid problems with files created with the original definitions a real valued version number (2.01) has been introduced for GLONASS nav mess files.

* IGS decided to use the Hatanaka compression scheme for RINEX observation files. Below the corresponding RINEX file name conventions are included as recommendations. The DOS naming (extension .yyE) was wrongly set to .yyY in the March 1998 version of the document.

* GPS week: The GPS week number in all RINEX files is a continuous number not affected by the 1024 roll-over, it runs from 1023 over 1024 to 1025 etc.

* A descrepancy between the definition of the header line fields of met sensor description and position in Table A5 and the example in Table A9 was removed.

The latter was correct.

The Philosophy Of Rinex

The first proposal for the "Receiver Independent Exchange Format" RINEX has been developed by the Astronomical Institute of the University of Berne for the easy exchange of the GPS data to be collected during the large European GPS campaign EUREF 89, which involved more than 60 GPS receivers of 4 different manufacturers. The governing aspect during the development was the following fact:

Most geodetic processing software for GPS data use a well-defined set of observables:

- the carrier-phase measurement at one or both carriers (actually being a measurement on the beat frequency between the received carrier of the satellite signal and a receiver-generated reference frequency).

- the pseudorange (code) measurement, equivalent to the difference of the time of reception (expressed in the time frame of the receiver) and the time of transmission (expressed in the time frame of the satellite) of a distinct satellite signal.

- the observation time being the reading of the receiver clock at the instant of validity of the carrier-phase and/or the code measurements.

Usually the software assumes that the observation time is valid for both the phase AND the code measurements, AND for all satellites observed.

Consequently all these programs do not need most of the information that is usually stored by the receivers: They need phase, code, and time in the above mentioned definitions, and some station-related information like station name, antenna height, etc.

GENERAL FORMAT DESCRIPTION

Currently the format consists of four ASCII file types:

1. Observation Data File
2. Navigation Message File
3. Meteorological Data File
4. GLONASS Navigation Message File

Each file type consists of a header section and a data section. The header section contains global information for the entire file and is placed at the beginning of the file. The header section contains header labels in columns 61-80 for each line contained in the header section. These labels are mandatory and must appear exactly as given in these descriptions and examples.

The format has been optimized for mimimum space requirements independent from the number of different observation types of a specific receiver by indicating in the header the types of observations to be stored. In computer systems allowing variable record lengths the observation records may then be kept as short as possible. The maximum record length is 80 bytes per record.

Each Observation file and each Meteorological Data file basically contain the data from one site and one session. RINEX Version 2 also allows to include observation data from more than one site subsequently occupied by a roving receiver in rapid static or kinematic applications.

If data from more than one receiver has to be exchanged it would not be economical to include the identical satellite messages collected by the different receivers several times. Therefore the Navigation Message File from one receiver may be exchanged or a composite Navigation Message File created containing non-redundant information from several receivers in order to make the most complete file.

The format of the data records of the RINEX Version 1 Navigation Message file is identical to the former NGS exchange format.

The actual format descriptions as well as examples are given in the Tables at the end of the paper.

Definition Of The Observables

GPS observables include three fundamental quantities that need to be defined: Time, Phase, and Range.

TIME:

The time of the measurement is the receiver time of the received signals. It is identical for the phase and range measurements and is identical for all satellites observed at that epoch. It is expressed in GPS time (not Universal Time).

PSEUDO-RANGE:

The pseudo-range (PR) is the distance from the receiver antenna to the satellite antenna including receiver and satellite clock offsets (and other biases, such as atmospheric delays):

PR = distance +

c * (receiver clock offset - satellite clock offset + other biases)

so that the pseudo-range reflects the actual behavior of the receiver and satellite clocks. The pseudo-range is stored in units of meters.

See also clarifications for pseudoranges in mixed GPS/GLONASS files in Chapter 8.1.

PHASE:

The phase is the carrier-phase measured in whole cycles at both L1 and L2. The half-cycles measured by sqaring-type receivers must be converted to whole cycles and flagged by the wavelength factor in the header section.

The phase changes in the same sense as the range (negative doppler). The phase observations between epochs must be connected by including the integer number of cycles. The phase observations will not contain any systematic drifts from intentional offsets of the reference oscillators.

The observables are not corrected for external effects like atmospheric refraction, satellite clock offsets, etc.

If the receiver or the converter software adjusts the measurements using the real-time-derived receiver clock offsets dT(r), the consistency of the 3 quantities phase / pseudo-range / epoch must be maintained, that is the receiver clock correction should be applied to all 3 observables:

Time(corr) = Time(r) - dT(r)

PR(corr) = PR(r) - dT(r)*c

phase(corr) = phase(r) - dT(r)*freq

DOPPLER:

The sign of the doppler shift as additional observable is defined as usual:

Positive for approaching satellites.

The Exchange Of Rinex Files

We recommend using the following naming convention for RINEX files:

ssssdddf.yyt ssss: 4-character station name designator

ddd: day of the year of first record

f: file sequence number within day

0: file contains all the existing

data of the current day

yy: year

t: file type:

O: Observation file

N: Navigation file

M: Meteorological data file

G: GLONASS Navigation file

To exchange RINEX files on magnetic tapes we recommend using the following tape format:

- Non-label; ASCII; fixed record length: 80 characters; block size: 8000
- First file on tape contains list of files using above-mentioned naming conventions

When data transmission times or storage volumes are critical we recommend ompressing the files prior to storage or transmission using the UNIX “compress" und "uncompress" programs. Compatible routines are available on VAX/VMS and PC/DOS systems, as well.

Proposed naming conventions for the compressed files:

System	Obs files	GPS Nav Files	GLONASS Nav Files	Met Files
UNIX	ssssdddf.yyO.Z	ssssdddf.yyN.Z	ssssdddf.yyG.Z	ssssdddf.yyM.Z
VMS	ssssdddf.yyO_Z	ssssdddf.yyN_Z	ssssdddf.yyG_Z	ssssdddf.yyM_Z
DOS	ssssdddf.yyY	ssssdddf.yyX	ssssdddf.yyV	ssssdddf.yyW

Proposed naming conventions for observation files compressed using the Hatanaka file compression scheme:

System	Obs files
UNIX	ssssdddf.yyD.Z
VMS	ssssdddf.yyD_Z
DOS	ssssdddf.yyE

References for the Hatanaka compression scheme: See for example ftp://igscb.jpl.nasa.gov/igscb/software/rnxcmp/docs/

IGSMails 1525,1686,1726,1763,1785

Rinex Version 2 Features

The following section contains features that have been introduced for RINEX Version 2.

Satellite Numbers:

Version 2 has been prepared to contain GLONASS or other satellite systems' observations. Therefore we have to be able to distinguish the satellites of the different systems: We precede the 2-digit satellite number with a system identifier.

snn	s:	satellite system identifier
	G or blank :	GPS
	R:	GLONASS
	T:	Transit
	nn:	PRN (GPS), almanac number (GLONASS) or two-digit Transit satellite number

Note: G is mandatory in mixed GPS/GLONASS files (blank default modified in April 1997)

Order of the Header Records

As the record descriptors in columns 61-80 are mandatory, the programs reading a RINEX Version 2 header are able to decode the header records with formats according to the record descriptor, provided the records have been first read into an internal buffer.

We therefore propose to allow free ordering of the header records, with the following exceptions:

The "RINEX VERSION / TYPE" record must be the first record in a file

The default "WAVELENGTH FACT L1/2" record (if present) should precede all records defining wavelength factors for individual satellites

The "# OF SATELLITES" record (if present) should be immediately followed by the corresponding number of "PRN / # OF OBS" records. (These records may be handy for documentary purposes. However, since they may only be created after having read the whole raw data file we define them to be optional.

Missing Items, Duration of the Validity of Values

Items that are not known at the file creation time can be set to zero or blank or the respective record may be completely omitted. Consequently items of missing header records will be set to zero or blank by the program reading RINEX files. Each value remains valid until changed by an dditional header record.

Event Flag Records

The "number of satellites" also corresponds to the number of records of the same epoch followed. Therefore it may be used to skip the appropriate umber of records if certain event flags are not to be evaluated in detail.

Receiver Clock Offset

A large number of users asked to optionally include a receiver-derived clock offset into the RINEX format. In order to prevent confusion and redundancy, the receiver clock offset (if present) should report the value that has been used to correct the observables according to the formulae under item 1. It would then be possible to reconstruct the original observations if necessary. As the output format for the receiver-derived clock offset is limited to nanoseconds the offset should be rounded to the nearest nanosecond before it is used to correct the observables in order to guarantee correct reconstruction.

ADDITIONAL HINTS AND TIPS

Programs developed to read RINEX Version 1 files have to verify the version number. Version 2 files may look different (version number, END OF HEADER record, receiver and antenna serial number alphanumeric) even if they do not use any of the new features.

We propose that routines to read RINEX Version 2 files automatically delete leading blanks in any CHARACTER input field. Routines creating RINEX Version 2 files should also left-justify all variables in the CHARACTER fields.

OS, and other, files may have variable record lengths, so we recommend to first read each observation record into a 80-character blank string and decode the data afterwards. In variable length records, empty data fields at the end of a record may be missing, especially in the case of the optional receiver clock offset.

RINEX UNDER ANTISPOOFING (AS)

Some receivers generate code delay differences between the first and second frequency using cross-correlation techniques when AS is on and may recover the phase observations on L2 in full cycles. Using the C/A code delay on L1 and the observed difference it is possible to generate a code delay observation for the second frequency.

Other receivers recover P code observations by breaking down the Y code into P and W code.

Most of these observations may suffer from an increased noise level. In order to enable the postprocessing programs to take special actions, such AS-infected observations are flagged using bit number 2 of the Loss of Lock Indicators (that is their current values are increased by 4).

GLONASS EXTENSIONS

RINEX Observation file

Time System Identifier

RINEX Version 2 needs one major supplement, the explicit definition of the time system:

GLONASS is basically running on UTC (or, more precisely, GLONASS system time linked to UTC(SU)), that is the time tags are given in UTC and not GPS time.

In order to remove possible misunderstandings and ambiguities, the header records "TIME OF FIRST OBS" and (if present) "TIME OF LAST OBS" in GLONASS and GPS observation files _can_, in mixed GLONASS/GPS observation files _must_ contain a time system identifier defining the system that all time tags in the file are referring to: "GPS" to identify GPS time, "GLO" to identify the GLONASS UTC time system. Pure GPS files default to GPS and pure GLONASS files default to GLO.

Format definitions see Table A1.

Hence, the two possible time tags differ by the current number of leap seconds.

In order to have the current number of leap seconds available we recommend to include a LEAP SECOND line into the RINEX header.

If there are known non-integer biases between the "GPS receiver clock" and "GLONASS receiver clock" in the same receiver, they should be applied. In this case the respective code and phase observations have to be corrected, too (c * bias if expressed in meters).

Unknown such biases will have to be solved for during the post processing . The small differences (modulo 1 second) between GLONASS system time, UTC(SU), UTC(USNO) and GPS system time have to be dealt with during the post-processing and not before the RINEX conversion. It may also be necessary to solve for remaining differences during the post-processing.

Pseudorange Definition

The pseudorange (code) measurement is defined to be equivalent to the difference of the time of reception (expressed in the time frame of the receiver) and the time of transmission (expressed in the time frame of the satellite) of a distinct satellite signal.

If a mixed-mode GPS/GLONASS receiver refers all pseudorange observations to one receiver clock only, the raw GLONASS pseudoranges will show the current number of leap seconds between GPS time and GLONASS time if the receiver clock is run ning in the GPS time frame - the raw GPS pseudoranges will show the negative number of leap seconds between GPS time and GLONASS time if the receiver clock is running in the GLONASS time frame

In order to avoid misunderstandings and to keep the code observations within the format fields, the pseudoranges must be corrected in this case as follows:

PR(GPS) := PR(GPS) + c * leap_seconds   if generated with a receiver clock running in the GLONASS time frame

PR(GLO) := PR(GLO) - c * leap_seconds   if generated with a receiver clock running in the GPS time frame to remove the contributions of the leap seconds from the pseudoranges.

"leap_seconds" is the actual number of leap seconds between GPS and GLONASS (UTC) time, as broadcast in the GPS almanac and distributed in Circular T of BIPM.

More Than 12 Satellites Per Epoch

The format of the epoch / satellite line in the observation record part of the RINEX Observation files has only been defined for up to 12 satellites per epoch. We explicitly define now the format of the continuation lines, see table A2.

RINEX Navigation Files for GLONASS

As the GLONASS navigation message differs in contents from the GPS message too much, a special GLONASS navigation message file format has been defined.

The header section and the first data record (epoch, satellite clock information) is similar to the GPS navigation file. The following records contain the satellite position, velocity and acceleration, the clock and frequency biases as well as auxiliary information as health, satellite frequency (channel), age of the information.

The corrections of the satellite time to UTC are as follows:

GPS   : Tutc = Tsv - af0 - af1 *(Tsv-Toc) - ... - A0 - ... - leap_sec

GLONASS: Tutc = Tsv + TauN - GammaN*(Tsv-Tb)   + TauC

*** In order to use the same sign conventions for the GLONASS corrections

as in the GPS navigation files, the broadcast GLONASS values are stored as:

-TauN, +GammaN, -TauC.

The time tags in the GLONASS navigation files are given in UTC (that is _not_ Moscow time or GPS time).

Filenaming convention: See above.

REFERENCES

Evans, A. (1989): "Summary of the Workshop on GPS Exchange Formats."

Proceedings of the Fifth International Geodetic Symposium on Satellite Systems, pp. 917ff, Las Cruces.

Gurtner, W., G. Mader, D. Arthur (1989): "A Common Exchange Format for GPS Data." CSTG GPS Bulletin Vol.2 No.3, May/June 1989, National Geodetic Survey, Rockville.

Gurtner, W., G. Mader (1990): "The RINEX Format: Current Status, Future Developments." Proceedings of the Second International Symposium of Precise Positioning with the Global Positioning system, pp. 977ff, Ottawa.

Gurtner, W., G. Mader (1990): "Receiver Independent Exchange Format Version 2." CSTG GPS Bulletin Vol.3 No.3, Sept/Oct 1990, National Geodetic Survey, Rockville.

RINEX VERSION 2 FORMAT DEFINITIONS AND EXAMPLES

0 (in L2): Single frequency instrument|

Table F 1 Observation Data File - Header Section Description

Header Label (Columns 61-80)	Description	Format
RINEX VERSION/TYPE	ormat version (2) File type ('O' for Observation Data) - Satellite System: blank or 'G': GPS 'R': GLONASS T': NNSS Transit	I6,14X, A1,19X A1,19X
PGM / RUN BY/ DATE	Name of program creating current file - Name of agency creating current file - Date of file creation	A20, A20, A20,
COMMENT	Comment line(s)	A60
MARKER NAME	Name of antenna marker	A60
MARKER NUMBER	Number of antenna marker	A20
OBSERVER / AGENCY	Name of observer / agency	A20,A40
REC # / TYPE / VERS	Receiver number, type, and version 3A20 (Version: for example Internal Software Version)	3A20
ANT # / TYPE	Antenna number and type	2A20
APPROX POSITION XYZ	Approximate marker position (WGS84	3F14.4
ANTENNA: DELTA H/E/N	-Antenna height: Height of bottom surface of antenna above marker - Eccentricities of antenna center relative to marker to the east and north (all units in meters)	3F14.4
WAVELENGTH FACT L1/2	- Wavelength factors for L1 and L2 1: Full cycle ambiguities 2: Half cycle ambiguities (squaring)0 (in L2): Single frequency instrument - Number of satellites to follow in list for which these factors are valid. 0 or blank: Default wavelength factors for all satellites not contained in such a list. - List of PRNs (satellite numbers with system identifier) Repeat record if necessary	2I6, I67(3X,A1,I2)|

Table F 1 Observation Data File - Header Section Description (continued)

Header Label (Columns 61-80)	**Description**	**Format**
# / TYPES OF OBSERV	- Number of different observation types stored in the file - Observation types If more than 9 observation types: Use continuation line(s) : Pseudorange using C/A-Code on L1 P1, P2: Pseudorange using P-Code on L1,L2 D1, D2: Doppler frequency on L1 and L2 T1, T2: Transit Integrated Doppler on 150 (T1) and 400 MHz (T2) Observations collected under Antispoofing are converted to "L2" or "P2" and flagged with bit 2 of loss of lock indicator (see Table A2). Units : Phase : full cycles Pseudorange : meters Doppler : Hz Transit : cycles The sequence of the types in this record has to correspond to the sequence of the observations in the observation records	I6 9(4X,A2) 6X,9(4X,A2) C1
INTERVAL	Observation interval in seconds	I6
TIME OF FIRST OBS	- Time of first observation record (4-digit-year, month,day,hour,min,sec) | - Time system: GPS (=GPS time system) GLO (=UTC time system) Compulsory in mixed GPS/GLONASS files Defaults: GPS for pure GPS files GLO for pure GLONASS files	5I6,F12.6, 6X,A3
|TIME OF LAST OBS	- Time of last observation record (4-digit-year, month,day,hour,min,sec) - Time system: GPS (=GPS time system) GLO (=UTC time system) Compulsory in mixed GPS/GLONASS files Defaults: GPS for pure GPS files GLO for pure GLONASS files	5I6,F12.6 6X,A3
LEAP SECONDS	Number of leap seconds since 6-Jan-1980 Recommended for mixed GPS/GLONASS files	I6
*|# OF SATELLITES	Number of satellites, for which observations are stored in the file	I6
*|PRN / # OF OBS	PRN (sat.number), number of observations |3X,A1,I2,9I6|* for each observation type indicated in the "# / TYPES OF OBSERV" - record. If more than 9 observation types: Use continuation line(s) 6X,9I6 This record is (these records are) repeated for each satellite present in the data file	3X,A1,I2,9I6 6X,9I6
|END OF HEADER	| Last record in the header section.	60X
Records marked with * are optional		

The following observation types are defined in RINEX Version 2: L1, L2: Phase measurements on L1 and L2

Table F 2 Observation Data File - Data Record Description

Obs. Record	Description	Format
EPOCH/SAT OR EVENT FLAG	Epoch: year (2 digits), month, day,hour,min,sec Epoch flag 0: OK 1: power failure between previous and current epoch >1: Event flag Number of satellites in current epoch List of PRNs (sat. numbers with system identifier, see 5.1) in current epoch Rceiver clock offset (seconds, optional) If more than 12 satellites: Use continuation line(s) If EVENT FLAG record (epoch flag > 1): Event flag: 2: start moving antenna 3: new site occupation (end od kinem. data) (at least MARKER NAME record follows) 4: header information follows 5: external event (epoch is significant, same time frame as observation time tags) 6: cycle slip records follow to optionally report detected and repaired cycle slips (same format as Observation redords; slip instead of observation; LL1 and signal strengthe blank) “Numver of satellites” contains number of records to follow (0 for event glags 2,5)	
OBSERVATIONS	Observation rep. within record for each obs. type (same seq as given in header) LL1 Signal strength If more than 5 obaservation types (=80 char): continue observations in next record. This record is (these records are) repeated for each satellite given in EPOCH/SAT-record. Observations: Phase: units in whole cycles of carrier Code: Units in meters Missing observations are written as 0.0 or blanks. Loss of lock indicator (LL!). Range: 0-7 0 or blank: OK or not known Bit 0 set: Lost lock between previous and current observation: cycle slip possible Bit 1 set: Opposite wavelength factor to the one defined for the satellite by a previous WAVELENGTH FACT L1/2 line. Valid for the current epoch only. Valid for the current epoch onlyh. Bit 2 set: Observation under antispoofing (may suffer from increased noise) Bits 0 and 1 for pahse only. Signal stength projected into interval 1-9: 1: minimum possible signal strength 5: tthreshold for good S/N ratio 9: maximum possible signal sgtrength 0 or blank: not known, don’t care	

Table F 3 Navigation Message File - Header Section Description

Header Label (Coumns 61-80)	Description	Format
RINEX VERSION/TYPE	- Format Version (2) - File type ('N' for Navigation data)	I6,14X, A1,19X
PGM/RUN BY/DATE	- Name of program creating current file - Name of agency creating current file - Date of file creation	A20, A20, A20
COMMENT*	Comment line(s)	A60
ION ALPHA*	Ionosphere parameters A0-A3 of almanac (page 18 of subframe 4)	2X,4D12.4
ION BETA*	Ionosphere parameters B0-B3 of almanad	2X,4D12.4
DELTA-UTC: A0,A1,T,W*	Almanac parameters to compute time in UTC (page 18 of subframe 4) A0,A1: Terems of polynomial T: Reference time for UTC data W: UTS reference week number. Continuous number, not mod (1024)	3X,2D19.12, 2I9
Leap seconds*	Delta time due to leap seconds	I6
End of header	Last record in the header section	60X
* Records marked with * are optional		

Table F 4 Navigation Message File - Data Record Description

OBS. Record	Description	Format
PRN/EPOCH/SV CLK	Satellite PRN number Epoch: Toc - Time of Clock year (2 digits) month day hour minute second SV clock bias (seconds) SV clock drift (sec/sec) SV clock drift rate (sec/sec^2)	I2, 5I3, F5.1, 3D19.12
Broadcast Orbit - 1	IODE Issue of Data, Ephemeris Crs (meters) Delta n (radians/sec) M0 (radians)	3X,4D19.12
Broadcast Orbit - 2	Cuc (radians) e Eccentricity (radians) Cus (radians) sqrt (A) (sqrt(m))	3X,4D19.12
Broadcast Orbit -3	Toe Time of Ephemeris (sec of GPS week) Cic ((radians) OMEGA ((radians) CIS ((radians)	3X,4D19.12

Table F 4 Navigation Message File - Data Record Description (continued)

OBS. Record	Description	Format
Broadcast Orbit - 4	i0 (radians) Crc (radians) omega (radians) OMEGA DOT (radians/sec)	3X,4D19.12
Broadcast Orbit - 5	i0 (radians/sec Codes on L2 channel GPS week # (to go with TOE) (continuous number, not mod (1024)! L2 P data flag	3X,4D19.12
Broadcast Orbit - 6	SV accuracy (meters) SV health (MSB only) TGD (seconds) IODC Issue of Data, Clock	3X,4D19.12
Broadcast Orbit - 7	Transmission time of message (sec of GPS week, derived for example from Z-count in Hand Over Word (HOW) spare spare spare	3X,4D19.12

Table F 5 Meteorological Data File - Header Section Description

Header Label (Columns 61-80)	Description	Format
RINEX version/type	Format version (2) File type (M for meteorological data)	I6,14X,A1,39X
PGM/Run By/ Date	Name of program creating current file Name of agency creating current file Date of file creation	A20,A20,A20
Comment*	Comment line(s)	A60
Marker Name	Station name (preferably identical to marker name in the associated Observation file)	A60
Marker number	Station number (preveravbly identical to marker number in the associated Observation file)	A20
#/Types of observ	Number of different observation types stored in the file Observation types The follofing meteorological observation types are defined in RINEX Version 2: PR: Pressure (mbar) TD: Dry temperature (deg Celsius) HR: Rekative humidity (percent) ZW: Wet zenith path delay (millimeters) (for WVR data) The sequence of the types in this record must correspond to the sequence of the measurements in the data records. If more than 9 types of observation types are being used, use continuation lines with format (6X, 9(4X,A2)	I6, 9(4X,A2)
Sensor Mod/Type/Acc	Description of the met sensor Model (manufacturer) Type Accuracy (same units as obs values) Observation type Record is repeated for each observation type found in #/Types of observ record	A20,A20,6X,F7.1,4X, A2,1X

Table F 5 Meteorological Data File - Header Section Description (continued)

Header Label (Columns 61-80)	Description	Format
Sensor pos XYZ/H	Approximate position of the met sensor Geocentric coordinates X,Y,Z (ITRF or WGS-84) Ellipsoidal height H Observation type Set X,Y,Z to zero if not known. Make sure H refers to ITRF or WGS-84! Record required for barometer, recommended for other sensors.	3F14.4,1F14.4,1X,A2, 1X
End of header	Last record in the header section.	60X

Table F 6 Meteorological Data File - Data Record Description

Obs. Record	Description	Format
Epoch/Met	Epoch in GPS time (not local time) Year (2 digits), month, day, hour, min,sec Met data in the same sequence as given in the header More than 8 met data types: Use continuation lines	6I3 mF7.1 4X,10F7.1,3X

Table F 7 Meteorological Data File - Header Section Description

Header Label (Columns 61-80)	Description	Format
RINEX version/type	Format version (2) File type (M for meteorological data)	I6,14X,A1,39X
PGM/Run By/ Date	Name of program creating current file Name of agency creating current file Date of file creation	A20,A20,A20
Comment*	Comment line(s)	A60
Marker Name	Station name (preferably identical to marker name in the associated Observation file)	A60
Marker number	Station number (preveravbly identical to marker number in the associated Observation file)	A20
#/Types of observ	Number of different observation types stored in the file Observation types The follofing meteorological observation types are defined in RINEX Version 2: PR: Pressure (mbar) TD: Dry temperature (deg Celsius) HR: Rekative humidity (percent) ZW: Wet zenith path delay (millimeters) (for WVR data) The sequence of the types in this record must correspond to the sequence of the measurements in the data records. If more than 9 types of observation types are being used, use continuation lines with format (6X, 9(4X,A2)	I6, 9(4X,A2)

Table F 7 Meteorological Data File - Header Section Description (continued)

Header Label (Columns 61-80)	Description	Format
Sensor Mod/Type/Acc	Description of the met sensor Model (manufacturer) Type Accuracy (same units as obs values) Observation type Record is repeated for each observation type found in #/Types of observ record	A20,A20,6X,F7.1,4X, A2,1X
Sensor pos XYZ/H	Approximate position of the met sensor Geocentric coordinates X,Y,Z (ITRF or WGS-84) Ellipsoidal height H Observation type Set X,Y,Z to zero if not known. Make sure H refers to ITRF or WGS-84! Record required for barometer, recommended for other sensors.	3F14.4,1F14.4,1X,A2, 1X
End of header	Last record in the header section.	60X

Table F 8 Meteorological Data File - Data Record Description

Obs. Record	Description	Format
Epoch/Met	Epoch in GPS time (not local time) Year (2 digits), month, day, hour, min,sec Met data in the same sequence as given in the header More than 8 met data types: Use continuation lines	6I3 mF7.1 4X,10F7.1,3X

Table F 9 Observation Data File Example

```
----|---1|0---|---2|0---|---3|0---|---4|0---|---5|0---|---6|0---|---7|0---|---8|

     2              OBSERVATION DATA    M (MIXED)           RINEX VERSION / TYPE
BLANK OR G = GPS,  R = GLONASS,  T = TRANSIT,  M = MIXED    COMMENT
XXRINEXO V9.9       AIUB                22-APR-93 12:43     PGM / RUN BY / DATE
EXAMPLE OF A MIXED RINEX FILE                               COMMENT
A 9080                                                      MARKER NAME
9080.1.34                                                   MARKER NUMBER
BILL SMITH          ABC INSTITUTE                           OBSERVER / AGENCY
X1234A123           XX                  ZZZ                 REC # / TYPE / VERS
234                 YY                                      ANT # / TYPE
  4375274.       587466.      4589095.                      APPROX POSITION XYZ
         .9030         .0000         .0000                  ANTENNA: DELTA H/E/N
     1     1                                                WAVELENGTH FACT L1/2
     1     2     6   G14   G15   G16   G17   G18   G19      WAVELENGTH FACT L1/2
     4    P1    L1    L2    P2                              # / TYPES OF OBSERV
    18                                                      INTERVAL
  1990     3    24    13    10   36.000000                  TIME OF FIRST OBS
                                                            END OF HEADER
 90  3 24 13 10 36.0000000  0  3G12G 9G 6                            -.123456789
  23629347.915           .300 8         -.353    23629364.158
  20891534.648          -.120 9         -.358    20891541.292
  20607600.189          -.430 9          .394    20607605.848
 90  3 24 13 10 50.0000000  4  4
   1     2     2   G 9   G12                              WAVELENGTH FACT L1/2
  *** WAVELENGTH FACTOR CHANGED FOR 2 SATELLITES ***        COMMENT
      NOW 8 SATELLITES HAVE WL FACT 1 AND 2!                COMMENT
                                                            COMMENT
 90  3 24 13 10 54.0000000  0  5G12G 9G 6R21R22                      -.123456789
  23619095.450      -53875.632 8    -41981.375    23619112.008
  20886075.667      -28688.027 9    -22354.535    20886082.101
  20611072.689       18247.789 9     14219.770    20611078.410
  21345678.576       12345.567 5
  22123456.789       23456.789 5
 90  3 24 13 11  0.0000000  2
                            4  1
```

Table F 9 Observation Data File Example (continued)

```
              *** FROM NOW ON KINEMATIC DATA! ***             COMMENT
 90  3 24 13 11 48.0000000  0  4G16G12G 9G 6                         -.123456789
  21110991.756       16119.980 7     12560.510    21110998.441
  23588424.398     -215050.557 6   -167571.734    23588439.570
  20869878.790     -113803.187 8    -88677.926    20869884.938
  20621643.727       73797.462 7     57505.177    20621649.276
                            3  4
A 9080                                                      MARKER NAME
9080.1.34                                                   MARKER NUMBER
         .9030         .0000         .0000                  ANTENNA: DELTA H/E/N
```

```
          --> THIS IS THE START OF A NEW SITE <--           COMMENT
 90  3 24 13 12  6.0000000  0  4G16G12G 6G 9                         -.123456987
  21112589.384       24515.877 6     19102.763 3  21112596.187
  23578228.338     -268624.234 7   -209317.284 4  23578244.398
  20625218.088       92581.207 7     72141.846 4  20625223.795
  20864539.693     -141858.836 8   -110539.435 5  20864545.943
 90  3 24 13 13  1.2345678  5  0
                            4  1
```

```
       (AN EVENT FLAG WITH SIGNIFICANT EPOCH)              COMMENT
 90  3 24 13 14 12.0000000  0  4G16G12G 9G 6                         -.123456012
  21124965.133       89551.30216     69779.62654  21124972.2754
  23507272.372     -212616.150 7   -165674.789 5  23507288.421
  20828010.354     -333820.093 6   -260119.395 5  20828017.129
  20650944.902      227775.130 7    177487.651 4  20650950.363
                            4  1
```

```
            *** ANTISPOOFING ON G 16 AND LOST LOCK           COMMENT
 90  3 24 13 14 12.0000000  6  2G16G 9
                 123456789.0      -9876543.5
                         0.0            -0.5
                            4  2
```

```
           ---> CYCLE SLIPS THAT HAVE BEEN APPLIED TO       COMMENT
                THE OBSERVATIONS                            COMMENT
 90  3 24 13 14 48.0000000  0  4G16G12G 9G 6                         -.123456234
  21128884.159      110143.144 7     85825.18545  21128890.7764
  23487131.045     -318463.297 7   -248152.72824  23487146.149
  20817844.743     -387242.571 6   -301747.22925  20817851.322
  20658519.895      267583.67817    208507.26234  20658525.869
                            4  4
         ***   SATELLITE G 9   THIS EPOCH ON WLFACT 1 (L2)  COMMENT
         *** G 6 LOST LOCK AND THIS EPOCH ON WLFACT 2 (L2)  COMMENT
                (OPPOSITE TO PREVIOUS SETTINGS)             COMMENT

----|---1|0---|---2|0---|---3|0---|---4|0---|---5|0---|---6|0---|---7|0---|---8|
```

Table F 10 Navigation Message File - Example

```
----|---1|0---|---2|0---|---3|0---|---4|0---|---5|0---|---6|0---|---7|0---|---8|

     2              N: GPS NAV DATA                         RINEX VERSION / TYPE
XXRINEXN V2.0       AIUB                12-SEP-90 15:22     PGM / RUN BY / DATE
EXAMPLE OF VERSION 2 FORMAT                                 COMMENT
     .1676D-07   .2235D-07  -.1192D-06  -.1192D-06          ION ALPHA
     .1208D+06   .1310D+06  -.1310D+06  -.1966D+06          ION BETA
     .133179128170D-06  .107469588780D-12   552960       39 DELTA-UTC: A0,A1,T,W
     6                                                      LEAP SECONDS
                                                            END OF HEADER
 6 90  8  2 17 51 44.0 -.839701388031D-03 -.165982783074D-10  .000000000000D+00
     .910000000000D+02  .934062500000D+02  .116040547840D-08  .162092304801D+00
     .484101474285D-05  .626740418375D-02  .652112066746D-05  .515365489006D+04
     .409904000000D+06 -.242143869400D-07  .329237003460D+00 -.596046447754D-07
     .111541663136D+01  .326593750000D+03  .206958726335D+01 -.638312302555D-08
     .307155651409D-09  .000000000000D+00  .551000000000D+03  .000000000000D+00
     .000000000000D+00  .000000000000D+00  .000000000000D+00  .910000000000D+02
     .406800000000D+06
13 90  8  2 19  0  0.0  .490025617182D-03  .204636307899D-11  .000000000000D+00
     .133000000000D+03 -.963125000000D+02  .146970407622D-08  .292961152146D+01
    -.498816370964D-05  .200239347760D-02  .928156077862D-05  .515328476143D+04
     .414000000000D+06 -.279396772385D-07  .243031939942D+01 -.558793544769D-07
     .110192796930D+01  .271187500000D+03 -.232757915425D+01 -.619632953057D-08
    -.785747015231D-11  .000000000000D+00  .551000000000D+03  .000000000000D+00
     .000000000000D+00  .000000000000D+00  .000000000000D+00  .389000000000D+03
     .410400000000D+06

----|---1|0---|---2|0---|---3|0---|---4|0---|---5|0---|---6|0---|---7|0---|---8|
```

Table F 11 Meteorological Data File - Example

```
----|---1|0---|---2|0---|---3|0---|---4|0---|---5|0---|---6|0---|---7|0---|---8|

     2              METEOROLOGICAL DATA                     RINEX VERSION / TYPE
XXRINEXM V9.9       AIUB                3-APR-96 00:10      PGM / RUN BY / DATE
EXAMPLE OF A MET DATA FILE                                  COMMENT
A 9080                                                      MARKER NAME
     3    PR    TD    HR                                    # / TYPES OF OBSERV
PAROSCIENTIFIC      740-16B                       0.2    PR SENSOR MOD/TYPE/ACC
HAENNI                                            0.1    TD SENSOR MOD/TYPE/ACC
ROTRONIC            I-240W                        5.0    HR SENSOR MOD/TYPE/ACC
        0.0           0.0           0.0       1234.5678 PR SENSOR POS XYZ/H
                                                            END OF HEADER
 96  4  1  0  0 15  987.1   10.6   89.5
 96  4  1  0  0 30  987.2   10.9   90.0
 96  4  1  0  0 45  987.1   11.6   89.0

----|---1|0---|---2|0---|---3|0---|---4|0---|---5|0---|---6|0---|---7|0---|---8|
```

Table F 12 Glonass Navigation Message File - Header Section Description

Header Label (Columns 61-80)	**Description**	**Format**
RINEX Version/Type	Format version (2.01) File type (G = GLONASS nav mess data)	F9.2,11X, A1,39X
Pgm/Run By/Date	Name of program creating current file Name of agency creating current file Date of file creation (dd-mmm-yy hh:mm)	A20, A20, A20
Comment*	Comment line(s)	A60
Corr to system time*	Time of reference for system time corr (year,month,day) Correction to system time scale (sec) to correct GLONASS system time to UTC (SU) (-TauC)	3I6,3X,D19.12
Leap seconds*	Number of leap seconds since 6 Jan 1980	I6
End of header	Last record in the header section	60X
*Records marked with * are optionsl		

Table F 13 Glonass Navigation Message File - Data Record Description

OBS. RECORD	DESCRIPTION	FORMAT
PRN/EPOCH/SV CLK	- Satellite almanac number - Epoch of ephemerides (UTC - year (2 digits) - month - day - hour - minute - second -SV clock bias (sec) (-TauN) - SV relative frequency bias (+GammaN) - message frame time (sec of day UTC)	I2 5I3 F5.1 D19.12D19.12 D19.12,
Broadcast Orbit - 1	-Satellite position X (km) velocity X dot (km/sec) X acceleration (km/sec) frequency number (1-24)	3X,4D19.12
Broadcast Orbit - 2	-Satellite position Y (km) velocity Y dot (km/sec) Y acceleration (km/sec) frequency number (1-24)	3X,4D19.12
Broadcast Orbit - 3	-Satellite position Z (km) velocity Z dot (km/sec) Z acceleration (km/sec) frequency number (1-24)	3X,4D19.12

Table F 14 GLONASS Navigation Message File - Example

```
----|---1|0---|---2|0---|---3|0---|---4|0---|---5|0---|---6|0---|---7|0---|---8|

     2.01           GLONASS NAV DATA                        RINEX VERSION / TYPE
ASRINEXG V1.1.0 VM  AIUB                19-FEB-98 10:42     PGM / RUN BY / DATE
STATION ZIMMERWALD                                          COMMENT
  1998     2    16    0.379979610443D-06                    CORR TO SYSTEM TIME
                                                            END OF HEADER
 3 98  2 15  0 15  0.0 0.163525342941D-03 0.363797880709D-11 0.108000000000D+05
    0.106275903320D+05-0.348924636841D+00 0.931322574615D-09 0.000000000000D+00
   -0.944422070313D+04 0.288163375854D+01 0.931322574615D-09 0.210000000000D+02
    0.212257280273D+05 0.144599342346D+01-0.186264514923D-08 0.300000000000D+01
 4 98  2 15  0 15  0.0 0.179599039257D-03 0.636646291241D-11 0.122400000000D+05
    0.562136621094D+04-0.289074897766D+00-0.931322574615D-09 0.000000000000D+00
   -0.236819248047D+05 0.102263259888D+01 0.931322574615D-09 0.120000000000D+02
    0.762532910156D+04 0.339257907867D+01 0.000000000000D+00 0.300000000000D+01
11 98  2 15  0 15  0.0-0.559808686376D-04-0.272848410532D-11 0.108600000000D+05
   -0.350348437500D+04-0.255325126648D+01 0.931322574615D-09 0.000000000000D+00
    0.106803754883D+05-0.182923507690D+01 0.000000000000D+00 0.400000000000D+01
    0.228762856445D+05 0.447064399719D+00-0.186264514923D-08 0.300000000000D+01
12 98  2 15  0 15  0.0 0.199414789677D-04-0.181898940355D-11 0.108900000000D+05
    0.131731816406D+05-0.143945598602D+01 0.372529029846D-08 0.000000000000D+00
    0.171148715820D+05-0.118937969208D+01 0.931322574615D-09 0.220000000000D+02
    0.135737919922D+05 0.288976097107D+01-0.931322574615D-09 0.300000000000D+01

----|---1|0---|---2|0---|---3|0---|---4|0---|---5|0---|---6|0---|---7|0---|---8|
```

RINEX Version 2.20

Modifications to Accommodate Low Earth Orbiter Data

****

Werner Gurtner <gurtner@aiub.unibe.ch>

Lou Estey <lou@unavco.ucar.edu>

April 12, 2001

Summary

In order to accommodate GPS data from LEO satellites some moderate modifications of the RINEX Version 2.10 format are proposed:

- New header records for
  - marker / vehicle characterization
  - antenna position and orientation
  - increased precision of the phase observations
- Two new observation types

- LA for L1 phase derived from C/A code tracking
- SA for SNR of L1 phase derived from C/A code tracking
- CH for the receiver channel numbers

- File naming recommendations
  - format for hourly files
  - format for 15-minute files

The modifications have been accepted for LEO data used in the IGS LEO Pilot Project by the IGS LEO Working Group.

The modifications are only necessary for RINEX files containing LEO data. Some of the modifications could be a valuable extension for ground-based GPS data, too.

We are looking for comments about the implementation of the modifications for ground-based data.

Introduction

The current RINEX Observation File definition does not fully accommodate data from low earth orbiters equipped with GPS or

GPS/GLONASS receivers.

The following deficiencies have been defined:

1) The RINEX file should show the nature of the antenna marker /

receiver carrier, that is if the receiver/antenna are mounted on a

satellite or another non-typical "marker"

2) Antenna position w/r to the center of mass should be stored

3) Antenna boresight in body-fixed coordinate system has to be known

4) Attitude data: It is proposed to put satellite attitude data into a separate file

5) Precision (resolution) of the data fields. As the format is fixed (that is, F14.3), the resolution of 0.001 cycles = 0.2 mm might not fulfill all future requirements.

6) Some receivers track satellites on more than one antenna or assign more than one channel to the same satellite (on the same or on different antennas). The RINEX obs format does not allow for more than one antenna at the same time. Data on more than one channel to the same satellite can be stored, however, the channel information gets lost.

7) It was noted that for some applications both possible phase observations on the first frequency, L1(CA) as well as L1(P1), should be stored in the RINEX files.

Proposed Modifications

In order to indicate the nature of the marker (spaceborne receiver) a new MARKER TYPE header record has been defined as deefined in Table F 1.

Table F 1 Proposed Marker Type Header

Marker Type	Curently defined are: GEODETIC: Earth-fixed, high-precision monumentation NON_GEODETIC: Earth-fixed, low-precision monumentation SPACEBORNE: Orbiting space vehicle AIRBORNE: Aircraft, balloon, etc. WATER_CRAFT: Mobile water craft GROUND_CRAFT: Mobile terrestrial vehicle FIXED_BUOY: "Fixed" on water surface FLOATING_BUOY: Floating on water surface FLOATING_ICE: Floating ice sheet, etc. BALLISTIC: Rockets, shells, etc. ANIMAL: Animal carrying a receiver HUMAN: Human being Record required except for GEODETIC and NON_GEODETICC marker types.	A20,40X
Attributes other than GEODETIC and NON_GEODETIC will tell the user program that the data were collected by a moving receiver. The inclusion of a "start moving antenna" record (event flag 2) into the data body of the RINEX file is therefore not necessary. Event flags 2 and 3 are still necessary to flag alternating kinematic and static phases of a moving receiver, however. The following new optional header fields have additionally been defined:		
ANTENNA: DELTA X/Y/Z*	Antenna position in body fixed coordinate system (meters)	3F14.4
ANTENNA: B.SIGHT XYZ*	Boresight of antenna: Direction os the antenna axis towards the GPS satellites. Corresponds to the vertical axix on earth-bound antenna. Unit vector in body-fixed coord. system.	3F14.4
OBS SCALE FACTOR	Factor to divide stored observations with before use. Number of observation types involved List of observation types Repeat record if different factors are applied to different observation types.	I6,I6,8(4X,A2)

Attributes other than GEODETIC and NON_GEODETIC will tell the user program that the data were collected by a moving receiver. The inclusion of a "start moving antenna" record (event flag 2) into the data body of the RINEX file is therefore not necessary. Event flags 2 and 3 are still necessary to flag alternating kinematic and static phases of a moving receiver, however.

The following new optional header fields have additionally been defined:

The OBS SCALE FACTOR record allows for example, to store phase data with 0.0001 cycles resolution if the data was multiplied by SCALE FACTOR 10 before being stored into RINEX file.

As the antenna position data could be given in a body-fixed coordinate system with the origin not coinciding with the satellite's center of mass we have to define an additional header line with the current center of mass coordinates:

Table F 1 Additional Header Line

*CENTER OF MASS: XYZ	Current center of mass of vehicle in body-fixed coordinate system (meters) (same system as used for attitide)	3F14.4 *

In order to distinguish between L1(CA) and L1(P1) we define LA and SA as new observation types (Table F 2).

Table F 2 Types of Observation

|# / TYPES OF OBSERV | -	Number of different observation types stored in the file - Observation types If more than 9 observation types: Use continuation line(s) The following observation types are defined in RINEX Version 2.20: L1, L2: Phase measurements on L1 and L2, L1 being derived from P (Y) code LA: Phase measurements on L1 derived from C/A code tracking. Use L1 if origin is not known. C1: Pseudorange using C/A-Code on L1 P1, P2: Pseudorange using P-Code on L1,L2 D1, D2: Doppler frequency on L1 and L2 T1, T2: Transit Integrated Doppler on 150 (T1) and 400 MHz (T2) S1, S2: Raw signal strengths or SNR values as given by the receiver for the L1,L2 phase observations SA; Raw signal strength or SNR for LA CH: Receiver channel number |	I6, 9(4X,A2) 9(4X,A2) |6X,9(4X,A2)

A future C/A code-derived phase observation on the second frequency could be called LB, with the associated signal strength or SNR value called SB.

As there are new header records and a new observation type defined, we will have to increase the RINEX version number from 2.10 to 2.20.

Signal strength data can be stored into the RINEX file using the S1 (or SA) and S2 observation types (the format F14.3 is good enough to handle all requirements regarding the maxumum range and precision).

Receiver channel numbers can be stored as special "observations" CH.

Lowest channel number allowed is 1 (re-number channels beforehand, if necessary). In case of a receiver using multiple channels for one satellite the channels could be packed with two digits each right-justified into the same data field, order corresponding to the order of the observables concerned. Format according to (<5-nc>(2X),<nc>I2.2,'.000'), nc being the number of channels.

Restriction: Not more than 5 channels and channel numbers <100.

Examples: 0910.000 for channels 9 and 10

010203.000 for channels 1, 2, and 3

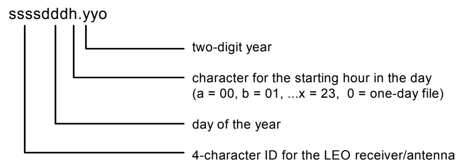
---- F14.3 ---

We propose to convert the "open-loop data" into standard phases and signal strengths before reformating the data into RINEX files for data exchange.

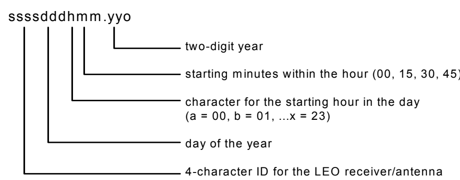
This proposal does not handle data collected by one receiver and multiple antennas. In this case the data have to be stored in separate RINEX files (one per antenna).

FILENAME RECOMMENDATIONS

We propose to use the standard file naming scheme for LEO POD tracking data by defining a 4-character code for each LEO receiver or receiver/antenna combination. The RINEX data should be distributed in daily files. Faster data exchange could use hourly files, if necessary.



For the 15-minutes high-rate ground station tracking data we recommend the following extended filenames:



Consequences for Ground-Based Data Files

Basically no changes are mandatory for ground-based GPS/GLONASS observation data files. However, the MARKER TYPE header record could be a valuable addition to the current version 2.10 to better characterize the marker/vehicle on which the receiver has been mounted.

Depending on the feedback especially from receiver manufacturers the LA and CH observation types could either be recommended to be used by ground-based receivers, too or to be restricted to space-borne receivers, only.

RINEX reader programs should be prepared to at least skip uninterpreted header records, modify LA into L1 or skip LA, and apply the OBS SCALE factor if present.

Implementation Plan

The proposed modifications have been accepted by the International GPS Service LEO Working Group in February/March 2001 to be used within the IGS LEO Pilot Project for space-borne GPS data.

We are now requesting comments from the wider GPS/GLONASS community to what extent the proposed modifications, especially the MARKER TYPE, OBS SCALE FACTOR header records and the LA and CH observables, should be used by non-space applications. We also like to get comments about the proposed MARKER TYPE classes.

Depending on the results of the poll the modifications will be restricted to MARKER TYPE = SPACEBORNE or opened to all classes.

Example

```
----|---1|0---|---2|0---|---3|0---|---4|0---|---5|0---|---6|0---|---7|0---|---8|
     2.20           OBSERVATION DATA    GPS                 RINEX VERSION / TYPE
TESTRINEX           TEST                07-AUG-00 00:44:01  PGM / RUN BY / DATE
DATE EXAMPLE OF A LEO FILE                                  COMMENT
CHAMP                                                       MARKER NAME
00039B                                                      MARKER NUMBER
SPACEBORNE                                                  MARKER TYPE
                    GFZ                                     OBSERVER / AGENCY
Main                Blackjack           2.0.9               REC # / TYPE / VERS
POD_Antenna         unknown                                 ANT # / TYPE
       -1.4880         0.0000       -0.3928                 ANTENNA: DELTA X/Y/Z
        0.0000         0.0000       -1.0000                 ANTENNA: B.SIGHT XYZ
    10.000                                                  INTERVAL
     1     1     0                                          WAVELENGTH FACT L1/2
     9    LA    L2    C1    P1    P2    L1    SA    S1    S2# / TYPES OF OBSERV
     1     3    LA    L1    L2                              OBS SCALE FACTOR
  2000     8     7     0     0    0.0000000                 TIME OF FIRST OBS
                                                            END OF HEADER
 00  8  7  0  0  0.0000000  0  7  4 25  7 19 16 11  1
  -5369354.141    -4183922.404    21470111.462    21470111.757    21470114.496
  -5369354.140         530.000         167.000         148.000
  -8742813.507    -6812576.531    22348706.474    22348706.684    22348709.446
  -8742813.505         370.000          93.000          75.000
   4733764.783     3688647.110    23786696.400    23786695.320    23786700.309
   4733764.786         199.000          25.000          32.000
  14033524.436    10935414.320    25844592.349    25844601.444    25844634.056
  14033524.414          80.000           4.000          11.000
  -3120775.023    -2431769.610    19832389.792    19832389.336    19832392.787
  -3120775.022         839.000         448.000         394.000
   9556410.416     7446601.916    25710269.924    25710274.075    25710260.983
   9556410.413          75.000           6.000           9.000
   1348505.455     1050780.864    20687372.600    20687372.514    20687375.419
   1348505.456         699.000         295.000         254.000
 00  8  7  0  0 10.0000000  0  6  4 25  7 19 16  1
  -5478891.715    -4269276.375    21449267.180    21449267.588    21449270.210
```

```
  -5478891.713        531.000        169.000        151.000
  -8920710.764   -6951197.790   22314853.801   22314854.143   22314856.548
  -8920710.762        374.000         94.000         77.000
   4965160.004    3868954.973   23830729.282   23830729.763   23830733.822
   4965160.003        199.000         26.000         30.000
  14382375.072   11207149.974   25910977.098   25910973.565   25910995.669
  14382375.062         80.000          5.000         10.000
  -3028365.514   -2359762.233   19849974.840   19849974.231   19849977.901
  -3028365.512        834.000        444.000        384.000
   1555567.947    1212128.236   20726775.295   20726775.003   20726778.132
   1555567.949        691.000        284.000        241.000
 00  8  7  0  0 20.0000000  0  6  4 25  7 19  1 16
  -5582917.443   -4350335.385   21429471.814   21429472.290   21429474.716
  -5582917.474        530.000        167.000        153.000
  -9096209.152   -7087949.770   22281457.702   22281457.735   22281460.023
  -9096209.152        377.000         99.000         77.000
   5199718.403    4051727.634   23875364.448   23875364.327   23875368.479
   5199718.399        195.000         24.000         31.000
  14730992.695   11478728.367   25977317.415   25977316.632   25977320.259
  14730992.715         79.000          5.000          9.000
   1766473.629    1376470.294   20766909.285   20766909.047   20766912.143
   1766473.629        683.000        282.000        236.000
  -2930120.571   -2283207.759   19868670.277   19868669.701   19868673.215
  -2930120.571        827.000        438.000        367.000

----|---1|0---|---2|0---|---3|0---|---4|0---|---5|0---|---6|0---|---7|0---|---8|
```

INDEX