TymServe™ 2100L
Network Time Server

8500-0078

User's Guide

Rev. B

(May, 1999)

COPYRIGHT NOTICE

Copyright 1998 Datum Inc All Rights Reserved

This manual is provided to assist the user in the operation and maintenance of the supplied equipment or software. It is recognized that multiple copies may be required to support even a single unit, and for this reason, permission is hereby granted to reproduce the supplied User's Guide for the purpose stated above, provided that this notice is included as part of the copy. Additional copies are also available from Datum Inc for a nominal fee.

In no case, however, does the supply of this manual or the granting of rights to reproduce the manual, grant any rights to use information contained within to reproduce the supplied equipment or software, either in whole or in part.

The equipment or software described in this manual have been developed solely at the expense of Datum Inc and are proprietary. No unlimited rights in technical data are granted. Limited rights as per DFARS 252.227-7013 shall be effective for 10 years from the copyright date.

Declaration of Conformity

Application of Council Directive(s) 89/339/EEC, 93/68/EEC

Standard(s) to which Conformity is Declared EN 50081-1, EN 50082-1, EN60950

Manufacturer's Address 6781 Via Del Oro, San Jose, CA, USA 95119-1360

Importer's Name

Importer's Address

Type of Equipment I. T. E

Model No. TymeServe™ 2100L

Serial No. Year of Manufacture 1998

I, the undersigned, hereby declare that the equipment specified above conforms to the Directive(s) and Standard(s).

Place Datum, Inc.

Leonard D Knight

(Signature)

Date January, 4 1999

Leonard D. Knight

(Full Name)

Quality Assurance Engineer

(Position)

TymServe 2100L
NETWORK TIME SERVER

TABLE OF CONTENTS

SECTION **PAGE**

CHAPTER ONE
GENERAL INFORMATION

1.0 Introduction 1-1
1.1 Unpacking 1-2

CHAPTER TWO
INSTALLATION INSTRUCTIONS

2.0 General 2-1
2.1 Quick Initial Set 2-1
2.2 Permanent Installation 2-6
2.3 Antenna Installation 2-8
2.4 Antenna/Cable Losses 2-8
2.5 Configuration Methods 2-9

CHAPTER THREE
SERVER OPERATION
AND TIME RELATED PROTOCOLS

3.0 Time Server Operation 3-1
3.1 Global Positioning System 3-1
3.2 Hierarchical Network Time Distribution Model 3-2
3.3 Time Protocol 3-3
3.4 Daytime Protocol 3-4
3.5 Simple Network Time Protocol 3-4
3.6 Network Time Protocol 3-4
3.7 NTP Authentication 3-6
3.8 Sysplex Timer 3-8

CHAPTER FOUR
COMMAND SHELL AND DESCRIPTION

4.0 General....4-1
4.1 Command Description....4-1
4.2 Network Directory....4-3
4.2.1 NTP Server Directory....4-5
4.2.1.1 NTP Broadcast Directory....4-7
4.2.2 Telenet Server Directory....4-7
4.2.3 SNMP Tools Directory....4-8
4.2.4 DHCP Tools Directory....4-9
4.2.5 TFTP Tools Directory....4-10
4.3 Timing Directory....4-11
4.3.1 Timing Utility Directory....4-14
4.3.2 GPS Tools Directory....4-16
4.4 Serial Directory....4-19
4.5 Utility Directory....4-21
4.6 Intrinsic Help....4-23

CHAPTER FIVE
SNMP CONFIGURATION AND CONTROL

5.0 General....5-1
5.1 Datum MIB II Extension....5-1
5.2 Additional Stored MIB Variables....5-1
5.3 MIB Compilation....5-2
5.4 Security....5-2
5.5 SNMPv1....5-2
5.6 MIB II Extension File....5-2
5.7 Datum Extension....5-3

CHAPTER SIX
TROUBLESHOOTING AND FAQ

6.0 General....6-1
6.1 Troubleshooting....6-1
6.2 Frequently Asked Question....6-3
6.3 Application Notes....6-5

APPENDIX A
SPECIFICATIONS AND DATA SHEET

A.0 Specification And Data Sheet A-1

APPENDIX B
INPUT/OUTPUT CONNECTORS

B.0 General B-1
B.1 Pin Descriptions B-1
B.2 Front-rear Panel Diagram B-2

APPENDIX C
FIRMWARE UPGRADE

C.0 General C-1
C.1 Typical Firmware Upgrade Procedure C-1

APPENDIX D
GLOSSARY

D.0 Glossary D-1

CHAPTER ONE
GENERAL INFORMATION

1.0 INTRODUCTION

Congratulations on your decision to purchase the TymServe 2100L. It shows you recognize the importance of maintaining accurate time within your network. Accurate in the sense that the resolution is within 1 to 10 milliseconds, depending on network installations, and the time is traceable through the Global Positioning System, GPS, satellite network to international standard time (called Universal Coordinated Time, UTC – or 'Universal Time Coordinated' in the source French). The TymServe is an ideal choice for establishing an accurate time source, and its network management facilities simplify operation.

The TymServe is a stand-alone time server that distributes time over a TCP/IP network including Internet, using the Network Time Protocol, NTP. The TymServe acts as a primary time server that broadcasts or responds to the specific time requests from client computers. In a client/server mode, NTP client sends a time request packet to the server, the server affixes its current time and returns the packet, and the client software processes the time data to adjust its local clock.

Global Positioning System

The US Department of Defense, Global Positioning System (GPS), is a constellation of 24 satellites in orbits inclined 56 degrees to the equator and orbiting the earth twice a day. These satellites transmit signals that are used by the GPS receivers to determine the position and time with great accuracy. The GPS receiver in your TymServe tracks these satellites as they pass overhead and determines the time and position from these satellites.

Server

The TymServe is classified as a Stratum One Time Server. This means that it derives its time from a Primary Time Reference, in this case GPS or its own free running internal clock, and supplies time to the clients. The TymServe is presently *not* designed to operate as a Stratum Two (or Three) Server wherein it simultaneously acts as a client deriving its time via an NTP process with a Stratum One (or Two) server and acts as a server for clients further down the hierarchy.

Client

Client NTP software comes in many different flavors, depending on the type of host and its operating system. Describing all of them and their installation procedures is beyond the scope of this server installation manual. Sources from where the NTP client software can be obtained and further advice on how to install this software is given in the reference listing in the back of this manual. Included with your TymServe is a diskette. This contains a shareware program, Dimension 4, that runs on Windows 95 or NT platforms. Instructions in the Quick Initial Setup section refer to use of this program.

Quick Initial Setup and Permanent Installation Instructions

Antenna placement and cable routing are the most troublesome aspects of installing a GPS based instrument. However, for new users it is important to get the TymServe up and running quickly and verify its operation with known client software. For this reason we recommend that new users follow the 'Quick Initial Setup' instructions as shown in Table 1-1 before proceeding with the permanent installation.

Table 1-1

Description	Quick Initial Setup	Permanent Installation
Installation Type	Place TymServe on the desk close to your laptop or desktop computers.	Mount TymServe on 19 inch rack close to TCP/IP network.
Antenna	Simply run it outside the building or set it close to the glass window with a view of the sky.	Roof top installation with mast and cable.
Setup Computer	Instructions assume using RS232 with HyperTerminal in Windows 95/NT desktop or laptop computer.	VT100 ASCII Terminal via serial interface on any compatible host or Telnet connection after the Quick Initial Setup.
Setup Commands	IP Address, Net Mask, Default Route and Time Source.	Same as Quick Initial Setup plus set of commands as described in Chapter 4 of this Manual.
Client Synchronization	'Dimension 4' SNTP client software for Windows 95/NT. This software is included in the Utility Diskette provided with the TymServe.	Download NTP Client software from the Internet Site. This software may require support from your IT group for configuration.

1.1 UNPACKING

Unpack all items in the box and inspect. If there is any damage, please contact Datum at (408) 578-4161 immediately so proper claims and/or replacement can be made. The following items should be included - see Figure 1-1.

- TymServe 2100L.
- AC Power Cord with US style wall plug.
- Antenna, 'Bullet II' or optional 'High Gain'.
- Antenna Mast, 40 cm aluminum mast threaded to screw into the bottom of the antenna.
- Mounting Bracket - for attaching mast to railing.

- 50 feet standard RG58 (Belden 8240) or optional RG 8 (Belden 9913) coaxial Cable.
- Optional adapter cable for Belden 9913 antenna cable.
- Utility Diskette - with MIB II SNMP extensions and Dimension 4 SNTP client software.
- User's Guide (this manual).

Figure 1-1 TymServe 2100L and Its Accessories

CHAPTER TWO

INSTALLATION INSTRUCTIONS

2.0 GENERAL

This section describes various steps required for Quick Initial Setup and Permanent installation of the TymServe. The configuration of 'network' and 'timing' that is performed during the Quick Initial Setup is also required for the Permanent Installation of the unit. This section also briefly covers antenna and cable installation.

2.1 QUICK INITIAL SET

- Remove and discard rubber washer from underneath the antenna terminal. Connect coaxial antenna cable directly to the bottom of the antenna, use adapter terminal for the optional Belden 9913 cables. See Figure 2-1 for typical cable connection details.

- Run the antenna cable outside the building or close to the window and place antenna on the ground or windowsill with view of the sky. Prevent any cable twists and/or damage to the antenna.

- Connect other end of the antenna cable to P3 (SMA connector) on the back of TymServe.

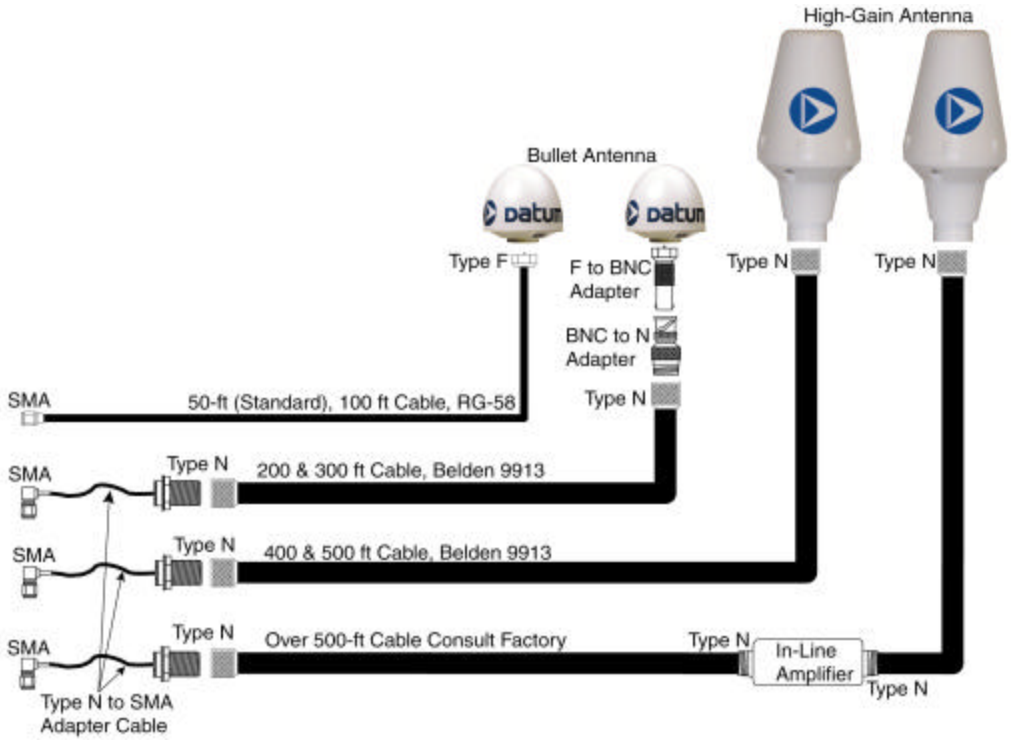


Figure 2-1 Antenna and Cable Options

- Connect setup computer to serial port B of the TymServe with a straight through RS232 serial cable. See Figure 2-5 for a typical connection setup.

- Connect the TymServe from RJ45 connector to the TCP/IP network through Ethernet 10baseT twisted pair cable. If the connection is made directly to the computer use crossed over 10baseT cable or otherwise, use and Ethernet hub for connections.

- Connect 100 - 130V / 220 – 240V, 48 - 440 Hz AC power supply to the back of TymServe and turn the power on.

- Establish a serial connection between the setup computer and the TymServe. The following example illustrates the steps performed to establish a serial connection through the HyperTerminal utility in Windows 95.

  Press `<Start> <Accessories> <HyperTerminal>`
  Double click `<Hypertrm.exe>`
  Enter the following information in the 'Connection Description' window and Press <`ok`>
  Enter <`any name`> in the 'Name' window
  Select the following information in the 'Phone Number' window and Press `<ok>`
  Select <`com Port number`> in 'Connect using'
  Enter the 'Port Settings' information in the 'Com1 Properties' window and Press `<ok>`
  'Bits per second' `<9600>`
  'Data bits' `<8>`
  'Parity' `<none>`
  'Stop bits' `<1>`
  'Flow Control' `<Xon/Xoff>`

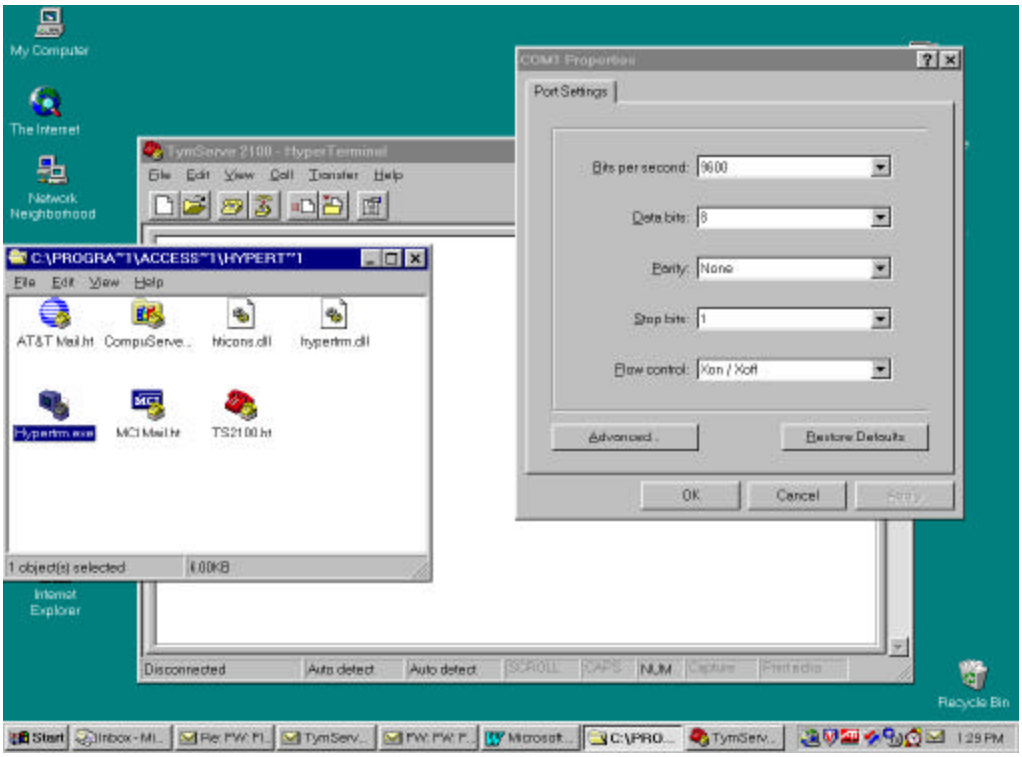


Figure 2-2 Serial Port Configuration on Host Computer

A blank screen for the 'HyperTerminal' serial connection will appear as shown in Figure 2-2. Press `<enter>` twice to see ? mark. This indicates that the serial connection with the TymServe is established and the unit is ready for initial configuration.

- Configure the following Network and Timing parameters through the serial connection. Refer to Figure 2-3 to see an example of how these parameters are entered. Make a note of these parameters for future reference and also for the permanent installation.

  Enter `<IP address>`
  Enter `<Subnet mask>`
  Enter `<Default gateway for the devices on different subnet>`
  Enter `<Timing Source (GPS -mode 6) or (Freerun -mode 1)>`

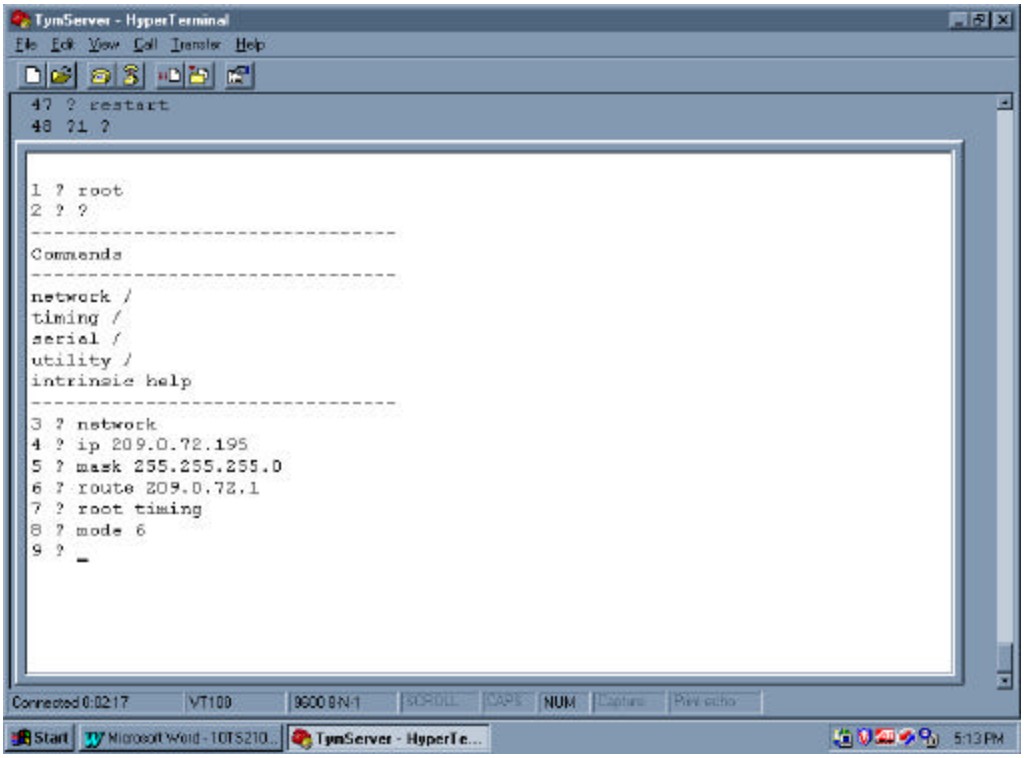


Figure 2-3 Shell through RS232 Connection

- After the initial configuration, the TymServe will acquire satellite signals. This is indicated with the 'Tracking' light on. Sometimes tracking may take longer than 15-30 minutes due to low signal strengths that are required for the GPS receiver in the TymServe. However, after the 'Tracking' light is on, it may take another 15-30 minutes for the internal oscillator to stabilize. This will be indicated with the 'Locked' light on. With all three LEDs on (Locked, Tracking, Power), the unit is ready to distribute time.

- In order to check the functionality of the Network Time Protocol, NTP, first check the Ethernet connection of the TymServe with the client computer as follows:

  At the command prompt of the client computer 'ping' to see if the TymServe is visible on the network. Here is an example of the ping command issued at the DOS level in Windows 95:

  ```
  ping <ip address of the TymServe> <Enter>
  ```

  If there is no response from the TymServe then first troubleshoot and fix the connection problem before checking the functionality of the TymServe on the network.

- Installation of Dimension 4 software is performed as follows:

  - Make separate directory for 'Dimension 4' on the hard disk of the client computer.
  - Copy 'd4time41.zip' file from the Utility Diskette to this directory.
  - Unzip this file and save the extracted files in the same directory.
  - Open 'Dimension 4' window by double clicking the 'd4.exe' file as shown in Figure 2-4.
  - Add `<ip address of the TymServe>` in the 'Server' box and Press `<ok>`

A message window will appear now indicating that the time is adjusted based on the response from the TymServe. The client computer will automatically get time after the time interval configured in the Dimension 4 'How often' field. This verifies that the TymServe is responding to the time requests from the clients. For more information regarding configuring Dimension 4, please refer to the 'Help' section included in the Dimension 4 program.

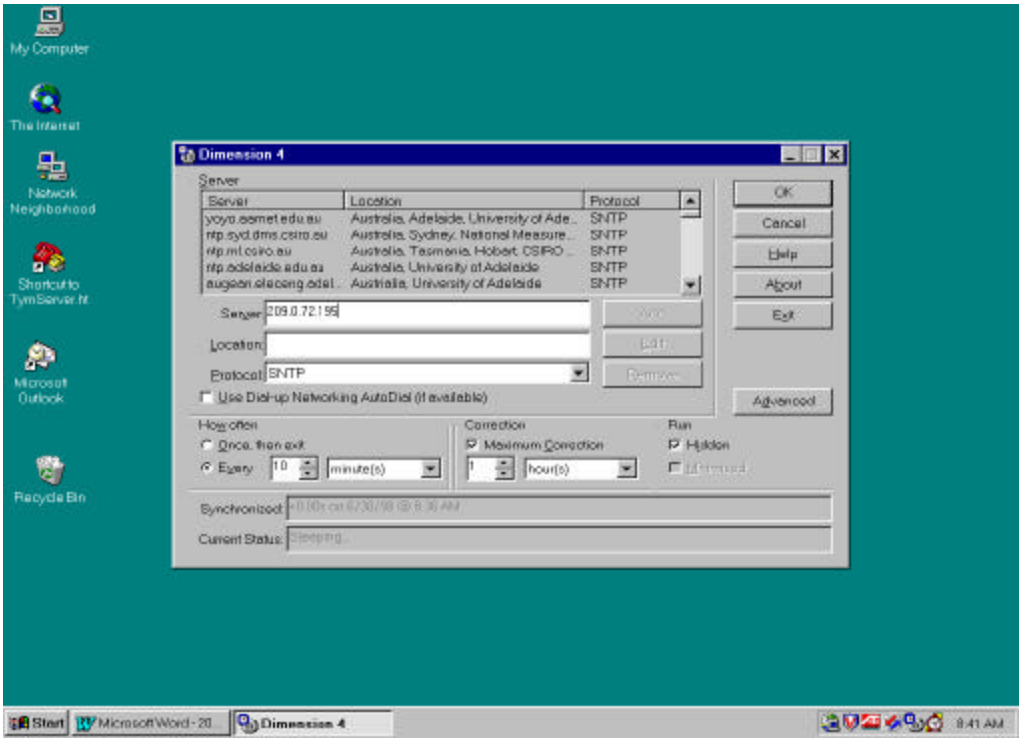


Figure 2-4 Configuration of Dimension 4

2.2 PERMANENT INSTALLATION

Once the Quick Initial Setup is done and the unit is tested ok for its functionality, it is ready for permanent installation in the rack. Disconnect the antenna cable, RS232 serial cable, 10baseT Ethernet and power cable from the unit and follow the steps listed below.

- Install and secure the TymServe in the rack with the screws.
- After the physical installation, connect the TymServe to the AC power supply.
- Connect the 10baseT twisted pair Ethernet cable from RJ45 connector of the TymServe to the network.
- Connect the GPS antenna cable to the back of TymServe. For more details please refer to the Antenna and Cable Installation in this chapter.

- Turn the power on. The 'Tracking' light will turn on in about 15-30 minutes. After this, it normally takes 15-30 minutes to 'Lock' and then it is ready to be configured.

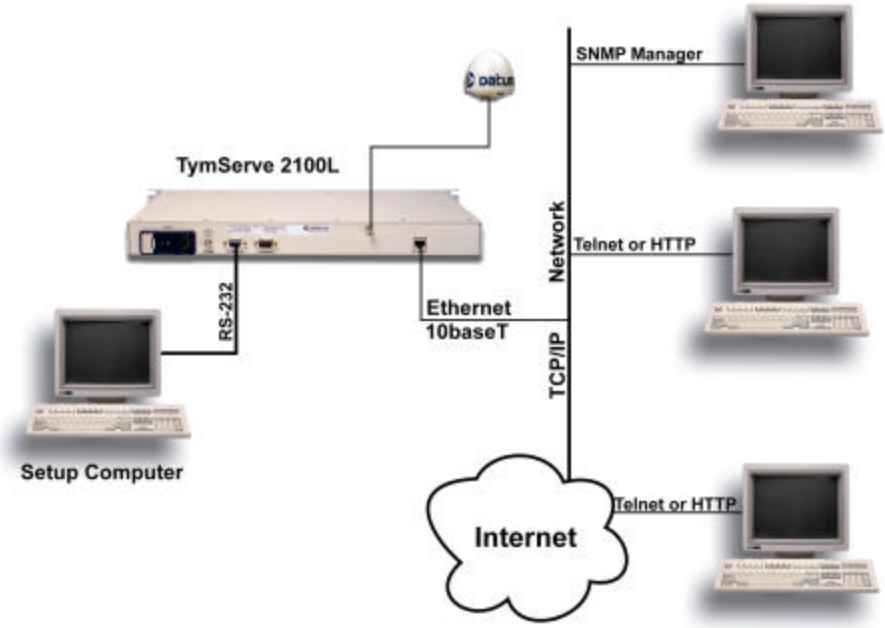


Figure 2-5 Typical Connection for TymServe 2100L

Since this manual only covers the installation and basic configuration of the TymServe, NTP Client software configuration is not included with this manual. The software and configuration details are available from the following Internet sites:

http://www.eecis.udel.edu/~ntp
http://www.microsoft.com/ntserver/tools/iclient.htm
http://cs3.ecok.edu:457/NetAdminG/BOOKCHAPTER-netadminN.about.html

- If the IP address is dynamically obtained from DHCP server, note this address for establishing the TELNET session. The TymServe is shipped from factory the DHCP option is turned off.

2.3 ANTENNA INSTALLATION

The bullet antenna provided with the unit comes with a weatherproof housing, suitable for permanent installation in an outdoor location. If the antenna has to be installed in a partially enclosed environment, it should be tested for functionality before it is permanently installed.

The antenna should be located with an unobstructed clear view of the sky for optimum tracking conditions. The antenna can receive satellite signals though glass, canvas or thin fiberglass. The antenna's operation is not affected if it is partially covered with snow, provided the snow is dry and does form a continuous ice sheet on the surface. The shape of the bullet antenna is designed to prevent accumulation of rain, snow or ice on its surface. The satellite signals cannot penetrate foliage, dense wood or metal structures.

The antenna kit also contains a antenna mast and mounting brackets. Install it outdoor with a clear view of the sky, away from the radio frequency interference. Type F connector end of the antenna cable is passed through the antenna mast and connected to the bottom of the antenna and the other end is connected to the back TymServe, P3 (SMA connector).

2.4 ANTENNA / CABLE LOSSES

Table 2-1 provides calculated losses for different types and lengths of cables used with the antenna. For reliable operation of the TymServe, the signal level at the input of the TymServe must be between 15db and 25db.

Table 2-1 GPS Cable Configuration Table

PartDescription(dB)/CableLength	**50 ft ***	**100 ft**	**200 ft**	**300 ft**	**400 ft**	**500 ft**
Standard Bullet Antenna (dB)	35	35	35	35		
Hi Gain Antenna (dB)					50	50
Internal GPS cable (dB)	-0.5	-0.5	-0.5	-0.5	-0.5	-0.5
BNC/N Adapter Cable (dB)		-0.5	-0.5	-0.5	-0.5	-0.5
Bias T (DC Block) (dB)					-1.0	-1.0
Belden 8240 Standard RG 58 (dB)	-9.5					
Belden 9913 Cable (dB)		-5.6	-11.2	-16.8	-22.4	-28.0
Gain at Receiver (dB)	**25**	**28.4**	**22.8**	**17.2**	**25.6**	**20.0**

Note: *1. Standard Cable.
2. For cable lengths greater than 500 feet contact Datum.

2.5 CONFIGURATION METHODS

There are four access methods available to perform configuration and/or status control of the TymServe. These access methods are summarized below.

RS-232 Serial Port B

The standard DTE style RS-232, DB9 (female) connector provides the preferred method of initial configuration and setup of the TymServe through a VT100 ASCII terminal using 9600, 8, N, 1 for communication parameters. Flow control is accomplished by the use of software Xon/Xoff. This method of access uses the 'command shell' explained later in this manual. This access method is not password protected. See Figure 2-3 for a typical Serial connection.

TELNET Access

To use the TELNET access, first configure the network parameters e.g., ip address, mask, default route, etc., through the RS-232 Serial Port B as explained in the initial setup. To establish a TELNET connection enter '*telnet: <ip address of the TymServe>'* at the DOS or Windows command prompt, or use any of the standard TELNET utilities. Navigating the TELNET command session is identical to the method used by the RS-232 access method.

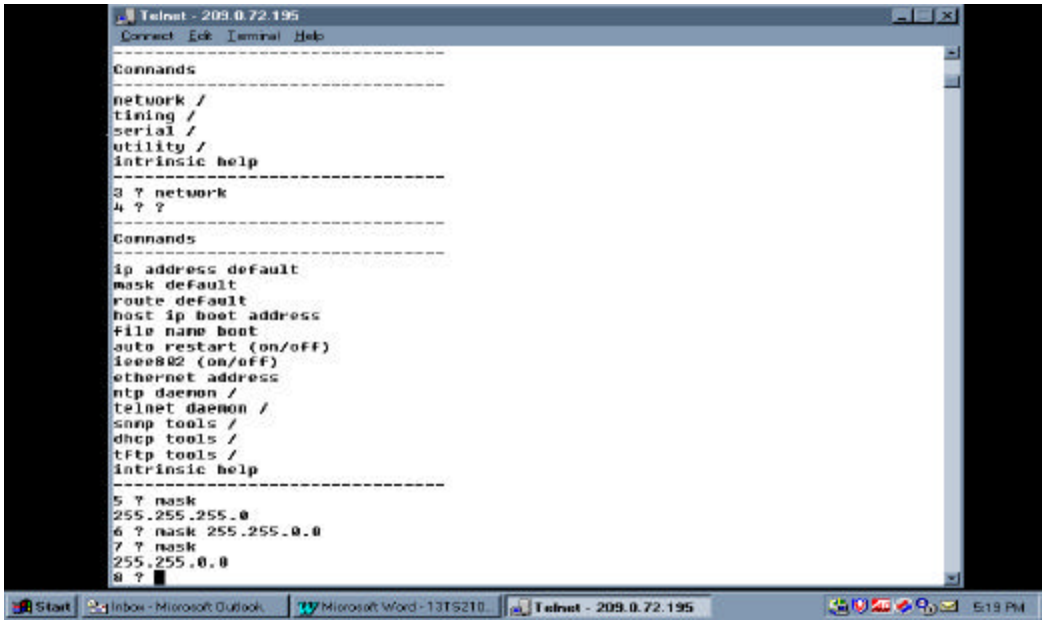


Figure 2-6 Telnet Session

TELNET interface can be password protected for security reasons. The password can only be enabled/disabled through the RS-232 serial port. If the user forgets the TELNET password, then it must be changed through the RS232 port. The TymServe allows only one TELNET session at a time. If the unit is not disconnected properly, the previous TELNET session will be timed out and disconnected after an hour. To disconnect the TELNET session use the `<exit>` command after the conclusion of the TELNET session. TELNET access uses reserved port number 23. Figure 2-6 shows an example of a typical TELNET session.

The TELNET interface can be further protected by disabling the TELNET Server daemon. Refer to `<auto>` and `<stop>` commands in Chapter 4 for more details.

SNMP Access

The TymServe provides various remote features like configuration, status and management control through the Simple Network Management Protocol, SNMP version 1 (RFC1157). In order to use SNMPv1, set and request packets. The network parameters must be configured through RS-232 or TELNET. Once the network parameters are set, the packets can be sent to configure the operating mode of the unit. Chapter 5, SNMP Configuration and Control, covers the Datum MIB Extension, MIB compilation and Security aspects of the SNMP.

Internet HTTP Access

The basic operating status of the TymServe can be viewed from the Hyper Text Markup Language (HTML) custom page over the HTTP protocol, by entering the IP address of the TymServe on the network as shown in Figure 2-7 below. The status screen also provides the time of the local host.

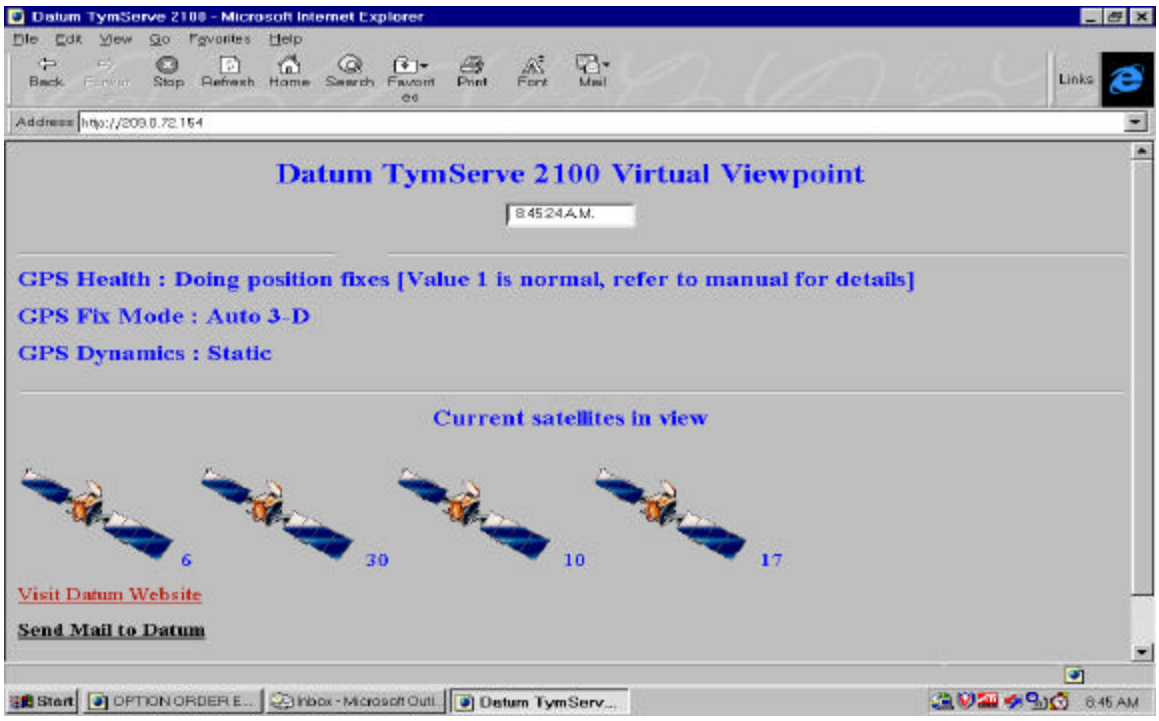


Figure 2-7 A Sample Web Page View

CHAPTER THREE

SERVER OPERATION AND TIME RELATED PROTOCOLS

3.0 TIME SERVER OPERATION

Time is distributed to the network utilizing Network Time Protocol (NTP), Simple Network Time Protocol (SNTP), Time Protocol, Daytime Protocol over TCP/IP or through Sysplex Timer over the Serial Port A. Client software is installed on the client machines before the NTP daemon can maintain the time synchronization with the TymServe.

Once the TymServe is locked with its source, it will continuously provide time even if the timing signal is lost. When the GPS time signal is lost, the 'Tracking' and 'Locked' light will turn off and the unit will run in the Freerun mode, thus maintaining the time with it own internal clock. The NTP message returned by the TymServe will indicate (via the Reference Timestamp) when it last obtained time updates from the timing signal.

The clients that need to be synchronized should be running a copy of the public domain NTP daemon or equivalent client software. If an NTP daemon is not available on your system you can obtain a copy of RFC-1119 from the Network Information Center NIC via FTP, in order to implement an NTP daemon for your system. Details of the NTP protocol and synchronization techniques are beyond the scope of this User's Guide.

The TymServe maintains the year value as a four-digit number and will have correct transition from the end of 1999 to year 2000 at the turn of the century. If the TymServe is powered down and powered back up in the year 2000 or later, it will correctly determine the year and time from the GPS satellite signal. It will also recognize the year 2000 as a leap year. In addition, the TymServe will have correct transition through the GPS week rollover event that will occur in August 1999.

3.1 GLOBAL POSITIONING SYSTEM

The Global Positioning System, GPS, is a constellation of 24 satellites in orbits inclined 56 degrees to the equator and orbiting the earth twice a day. This means that the further north you are in the Northern Hemisphere, it is more probable that satellites will be passing to the south of you. Conversely, Southern Hemisphere installations favor views of the northern sky.

The GPS receiver in your TymServe tracks these satellites as they pass overhead and determines the range of the satellite from the antenna. There are four unknowns: x, y, z, and t (or loosely latitude, longitude, altitude, and time). Knowing the range from one satellite places you on a sphere; two satellites the intersection of two spheres, a circle; three satellites two points; and, finally, four satellites yield a complete four variable solution. However, once x, y, and z are known, either because the receiver has tracked at least four satellites and positioned itself, or the user has entered a known position into the TymServe; only one satellite is required to solve for time. This is important because placement of the antenna in high rise urban areas frequently results in a restricted view of the sky, but for a timing solution this is not as much of a liability as it would be if one were trying to derive a full navigational solution.

The GPS transmission is a 1.5 GHZ (L_1 Band) spread spectrum signal. Being spread spectrum it is relatively immune to interference, however not perfectly so. High energy sources, especially those with significant in band energy, can swamp the receiver RF processing circuitry. In addition, it has proved difficult to operate GPS at power substations or in close proximity to high voltage 60 Hz sources. Datum offers an optional high gain antenna that is useful in these heavy interference situations, however, again it is recommended that the Quick Setup be pursued with the antenna away from radiating sources before dealing with such situations

3.2 HIERARCHICAL NETWORK TIME DISTRIBUTION MODEL

Network time distribution systems generally employ a hierarchical time distribution model as shown in Figure 3-1. In hierarchical systems, the primary time source clocks are considered stratum zero which includes GPS satellites and time sources at the United States Naval Observatory, USNO; National Institute of Standards and Technology, NIST; or other national time standards organizations.

The TymServe acts as Stratum one time server that derives its time from the GPS satellites and distributes this time through TCP/IP network or Sysplex Timer to the computers. The client computers may act as Stratum two time server and distribute time to Stratum three computers as shown in Figure 3-1.

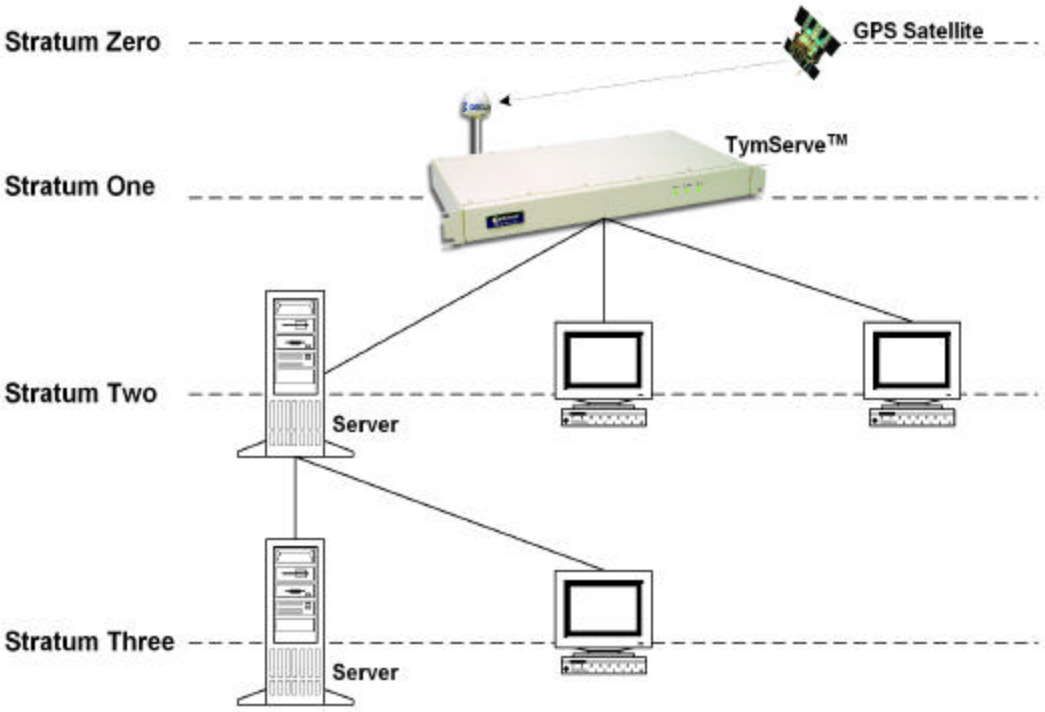


Figure 3-1

3.3 TIME PROTOCOL (RFC 868)

This protocol provides a site-independent, machine-readable date and time. The time service on the TymServe responds to the originating source the time in seconds since midnight on January 1, 1900. The time is the *number of seconds* since 00:00 (midnight) January 1, 1900 GMT, such that the time 1 is 12:00:01 am on January 1, 1900 GMT; this base will serve until the year 2036.

This protocol may be used either above the Transmission Control Protocol (TCP) or above the User Datagram Protocol (UDP). If the server is unable to determine the time, it either refuses the connection or closes it without sending any response. When used over TCP the TymServe listens for a connection on port 37. Once the connection is established, the server returns a 32-bit time value and closes the connection.

When used over UDP the TymServe listens for a datagram on port 37. When a datagram arrives, the TymServe returns a datagram containing the 32 bit time value.

3.4 DAYTIME PROTOCOL (RFC 867)

The Daytime service sends the current date and time as a *character string* without regard to the input. When used over TCP the TymServe listens for a connection on port 13. Once a connection is established the current date and time is sent out as an ASCII character string. The service closes the connection after sending the quote. When used over UDP the TymServe listens for a datagram on port 13. TymServe responds to the UDP request with the current date and time as an ASCII character string.

3.5 SIMPLE NETWORK TIME PROTOCOL (RFC 1361)

This is a simplified access protocol for servers and clients using NTP as now specified and deployed in the Internet. The access paradigm is identical to the UDP/Time client implementation. SNTP is also designed to operate in a dedicated server configuration including an integrated radio clock. SNTP uses the standard NTP timestamp format described in RFC 1305 and previous versions of that document. NTP stamps are represented as a 64-bit unsigned fixed-point number, in seconds relative to 0h on January 1, 1900.

3.6 NETWORK TIME PROTOCOL (RFC 1305 & RFC 1119)

The Network Time Protocol is used to synchronize computer clocks in the TCP/IP computer network. It provides comprehensive mechanism to access national time and frequency dissemination services, organize the time-synchronization subnet and adjust the local clocks. NTP provides accuracy of 1-10 ms, depending on the jitter characteristics of the synchronization source and network paths. NTP is a client of the User Datagram Protocol (UDP), which itself is a client of the Internet Protocol (IP)

NTP Data Format

The format of the NTP message data area, which immediately follows the UDP header, is shown below. NTP time stamps are represented as a 64 bit unsigned fixed-point number, in seconds relative to 0^h on 1 January 1900. The integer part is in the first 32 bits and the fraction part in the last 32 bits.

NTP Message Data

0			8	16	24	31
LI	VN	MODE	Stratum	Poll	Precision	
Synchronizing Distance (Root Distance) (32 Bits).						
Synchronizing Dispersion (Root Dispersion) (32 Bits).						
Reference Identifier (32 Bits).						
Reference Time Stamp (64 Bits).						
Originate Time Stamp (64 Bits).						
Receive Time Stamp (64 Bits).						
Transmit Time Stamp (64 Bits).						
Authenticator (Optional) (96 Bits).						

Leap Indicator (LI)
This is a two-bit code warning of an impending leap second to be inserted/deleted in the last minute of the current day, with bit 0 and bit 1, respectively, coded as follows:

00: No warning.
01: Last minute has 61 seconds.
10: Last minute has 59 seconds.
11: Alarm condition (clock not synchronized).

When the TymServe is first powered on i.e., before time is initially acquired from the timing signal, and when the timing parameters are changed, the alarm condition will be indicated. This condition will persist until the TymServe acquires time and should not be set again until the unit is powered off and on.

Version Number (VN)
This is a three-bit integer indicating the NTP version number. The TymServe will return the version number from the incoming NTP message.

Mode
This is a three-bit integer indicating the mode. For the TymServe this field is set to four indicating the server mode. The TymServe always operate in server mode, which means that it will synchronize clients but will never be synchronized by clients.

Stratum
This is an eight-bit integer indicating the stratum level of the local clock. For the TymServe this field is set to one indicating a primary reference.

Poll Interval
This is an eight-bit signed integer indicating the maximum interval between successive messages, in seconds to the nearest power of two. The TymServe will return the poll interval from the incoming NTP message.

Precision
This is an eight-bit signed integer indicating the precision of the local clock, in seconds to the nearest power of two. For the TymServe this field is set to -19 (minus nineteen) which is the value closest to the 1 μ sec precision of the TymServe.

Synchronizing Distance (Root Distance Version 3)
This is a 32-bit fixed-point number indicating the estimated round-trip delay to the primary synchronizing source, in seconds with fraction point between bits 15 and 16. Set to zero in the TymServe.

Synchronizing Dispersion (Root Dispersion Version 3)
Synchronizing Dispersion is a 32 bit fixed-point number indicating the estimated dispersion to the primary synchronizing source, in seconds. Root Dispersion indicates the maximum error relative to the primary reference source.

Reference Clock Identifier
This is a 32-bit code identifying the particular reference clock. In the case of stratum 1 (primary reference), this is a four-octet, left justified, zero-padded ASCII string. For the TymServe the four-octet string is dependent on the time source selected, 'GPS' for GPS and 'FREE' for Free Running Clock.

Reference Timestamp
This is the local time at which the local clock was last set or corrected, in 64-bit timestamp format. With the TymServe, the Reference Timestamp is the last time that a valid timing signal was detected. Therefore, the Reference Timestamp will indicate the time at which the timing signal was lost. When the timing signal returns, the Reference Timestamp will be updated.

Originate Timestamp
This is the local time at which the request departed the client host for the service host, in 64-bit timestamp format.

Receive Timestamp
This is the local time at which the request arrived at the service host, in 64-bit timestamp format.

Transmit Timestamp
This is the local time at which the reply departed the service host for the client host, in 64-bit timestamp format.

Authenticator
This field is used to hold a cryptochecksum if authentication has been enabled. Refer to the appendix on NTP authentication for more information about this mechanism.

3.7 NTP AUTHENTICATION

NTP authentication provides a mechanism that allows a NTP client to ensure that the timestamp received has come from a trusted source and has not been modified in transit. We have extended the authentication mechanism so that it can also be used to deny service to those clients who submit NTP timestamp requests without valid authentication information. The NTP protocol includes space for two variables related to authentication; an authentication key identifier field and a cryptochecksum field.

Authentication Mechanism

The mechanism used to generate the authentication data must be shared by the client and the server. The popular public domain implementation of NTP, known as xNTP, allows for the use of either DES (Digital Encryption Standard) or MD5 (Message Digest version 5). Due to export restrictions on cryptographic techniques, the TymServe supports only the MD5 encryption algorithm. MD5 provides an adequate level of security for NTP transmissions.

MD5 is a one-way hash function that processes the input data and produces 128 bits (16 bytes) of hash value. This cryptochecksum is then placed in the packet. Since the data itself is not encrypted, anyone could theoretically capture the packet, modify the data and put a new cryptochecksum into the packet. What makes the cryptochecksum secure is that a mutually agreed upon, secret key is loaded into the MD5 algorithm before the NTP data is loaded. This produces a cryptochecksum that cannot be reproduced without knowledge of the secret key.

Programming And Storage Of Key Identifier/Key Pair

The TymServe allows for the programming and storage of four key identifier/key pairs. Although it is possible to have over four billion keys, four are sufficient for the TymServe as it only has one level of access, requesting timestamps. While there are only four key identifier/key pairs, the key identifiers themselves can have any value from 1- 4,294,967,296. The format of the MD5 secret key is based on the approach taken by the public domain xNTP package. The key is an eight character alphanumeric string. This key identifier/key pair is stored in a flash EPROM and need only be programmed once.

Public Domain xNTP Package

A note should be made for those clients *not* using the public domain xNTP package. As defined by RFC1305, the cryptochecksum takes up 64 bits (eight bytes) in the NTP message. Because the MD5 algorithm produces 128 bits (16 bytes) of cryptochecksum, the NTP packet is enlarged by eight bytes to handle the entire cryptochecksum. As this field is the last in the packet, it should not present any difficulty.

NTP Authentication Only

The NTP Authentication only mechanism is an added feature in the TymServe and not part of the NTP specification as detailed in RFC1305. This mechanism provides a way to restrict access to the TymServe. To understand this mechanism, it is pertinent to first understand the way the NTP (RFC1305) defines the authentication process. If authentication is enabled, and a valid authentication key identifier and cryptochecksum is received, then the NTP packet is filled in and a new cryptochecksum is computed and added to the packet. The packet is then sent back to the client. However, if authentication is enabled, and an authentication failure occurs, either because the key identifier is 0 (defined as no encryption) or unrecognized, or the cryptochecksum is invalid, then the NTP packet is STILL RETURNED, but will contain no authentication data. Many of our customers have expressed an interest in somehow adapting the authentication mechanism to allow them to restrict access to the TymServe, for security or administrative purposes. The NTP Authentication ONLY mechanism provides that capability.

If NTP Authentication has been enabled, and the customer enables the NTP Authentication ONLY mode, the TymServe will discard any incoming NTP packet which does not contain both a valid key identifier (not equal to 0) and a valid cryptochecksum. In this way, the customer can limit access to the TymServe to only those clients who have been given the key identifier/secret MD5 key pair.

3.8 SYSPLEX TIMER

This is sometimes called an External Time Reference (ETR), the Sysplex Timer provides a synchronized Time-Of-Day (TOD) clock for multiple attached computers. A Sysplex is needed when two or more systems are configured in a Sysplex, which is short for a SYStem comPLEX. Only one Sysplex Timer can do the job, but, to ensure availability, companies should install a second duplex timer as a backup in case the primary timer fails.

TymServe receives input from the GPS Antenna and provides Sysplex Timer output through its serial port A. The serial port A provides an ASCII broadcast of UTC time that is often used by computers that can't or don't use NTP. Be sure that your computer is setup with the proper serial port parameters (i.e. baud rate, data bits, stop bits, parity). The serial port will start broadcasting the time only after it receives a ‘c’ or ‘C’ character, and it will stop the broadcast when it receives an ‘r’ or ‘R’ character. The Carriage Return character is transmitted on-time.

The following time information string is transmitted once per second (when started with the 'c' or 'C' character). The 'DDD' field represents 3 ASCII digits of days (001 - 366), etc. The Quality Indicator indicates the validity of the time. The Carriage Return character is transmitted on-time. The first rising edge of the Carriage Return character occurs within 200 nanoseconds after the TymServe 1 PPS signal transitions from low to high.

(SOH)DDD:HH:MM:SSQ(CR)(LF)

FIELD	DESCRIPTION
(SOH) (0x01)	ASCII Start of Header
DDD	Day of year
HH	Hours (24-hour clock)
MM	Minutes
SS	Seconds
Q	Quality Indicator (space = normal operation)
(CR) (0x0D)	ASCII Carriage Return (transmitted on-time)
(LF) (0x0A)	ASCII Line Feed

QUALITY CHAR	DESCRIPTION
space	Normal operation, time set and not flywheeling
X	Time not set yet
F	Time was set, but currently flywheeling

CHAPTER FOUR
COMMAND SHELL AND DESCRIPTION

4.0 GENERAL

Command shell is a command line interface that is accessible through Serial Port B or TELNET. It is a multiple level tree where the input is entered as a command in the form of ASCII strings typed at the command prompt. The ready state of the command shell is an ASCII '?' prompt. The specific commands available at a particular tree level can be displayed by entering a ? <*Enter*> at the ? prompt. A complete command shell tree is shown in Figure 4-1.

A CR-LF, CR, or LF sequence depending on the translation setting in the serial configuration sub-directory terminates all ASCII commands entered. The command interface interprets the input on a character-by-character basis. As a result, only enough characters to uniquely identify the command need to be entered for the command interface to recognize which action should be performed. The command interface will also accept multiple commands on a single line when they are separated by spaces. This obviates the need to press <Enter> after each command. Maximum buffer size is 128 bytes. The commands are categorized into three types:

- **Level Commands** are available at the root level and have a forward slash (/) following the command string.
- **Action Commands** show the current setting or set new parameters when executed with the corresponding parameter.
- **Intrinsic Commands** are available at any level of the system.

4.1 Command Description

The command shell is case sensitive and, therefore, commands and parameters should be entered in the same format as described in this section. The commands are divided into the following categories:

network /
timing /
serial /
utility /
intrinsic help

There may be multiple entries of the input parameters for each command. Each entry is corresponding to one of the allowable input parameters. If multiple parts are shown in the command menu then type the first part of each command. Otherwise the following parts will be treated as input parameters, which may cause some confusion beyond user's expectation. For example, if the user enters `<file name ts21.hex>` instead of `<file ts21.hex>` the TymServe will set the file name to be 'name' and 'ts21.hex' will be ignored and may cause incorrect configuration.

The commands can be accessed by the any one of the following three methods:

- *RS232* (Command Shell)
- *TELNET* (Command Shell)
- *SNMP* (SNMP Management Software Interface)

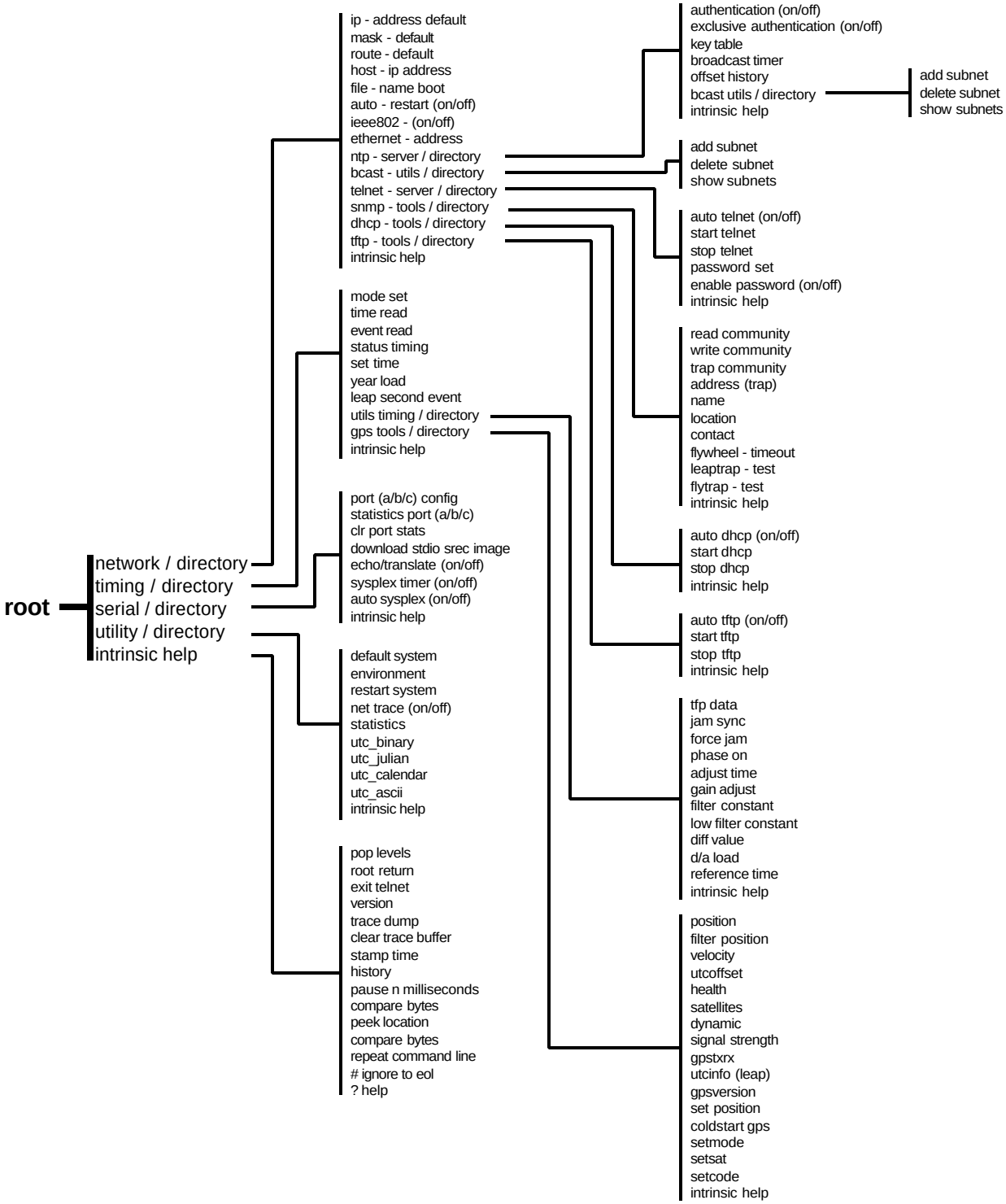


Figure 4-1 Serial/TELNET Command Tree

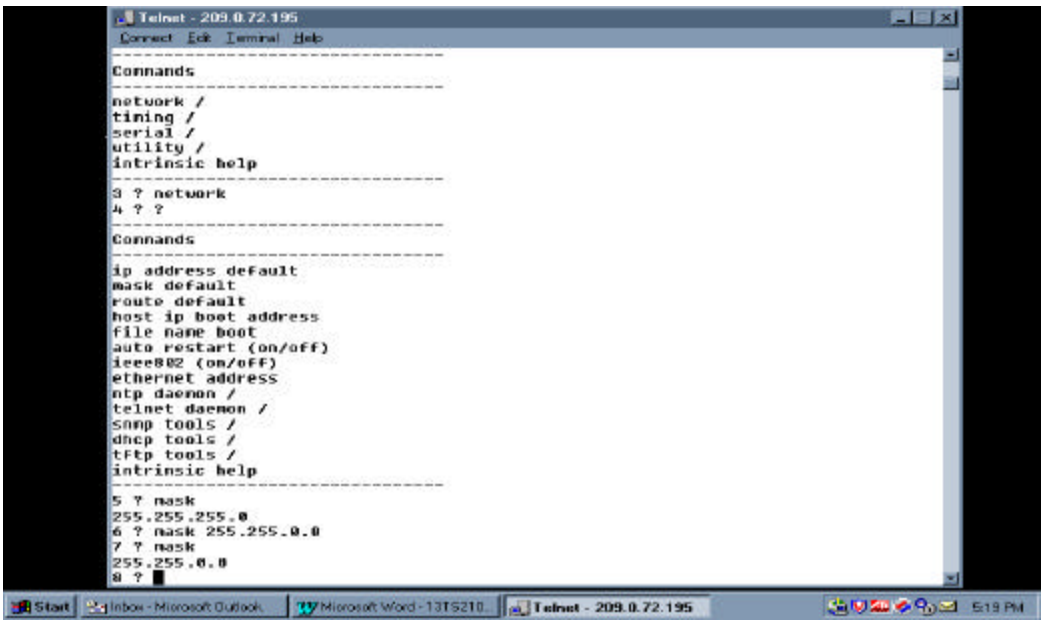


Figure 4-2

4.2 NETWORK DIRECTORY

Typing 'network <*Enter*>' under root directory enters the network directory as shown in Figure 4-2. The commands in this directory provide network configuration options. Some network server daemons and tools are available in the directory.

`ip <xxx.xxx.xxx.xxx>` *(RS232, TELNET)*

Queries or sets the network IP address of the TymServe in dotted quad notation. This variable can also be set automatically using the automatic DHCP function. If this value is changed using a TELNET session, the connection will be lost and a new connection will need to be started using the new address. The network interface will be restarted after successful storage of the new parameter in nonvolatile memory.

`mask <xxx.xxx.xxx.xxx>` *(RS232, TELNET)*

Queries or sets the network IP mask address of the TymServe in dotted quad notation. This variable can also be set automatically using the automatic DHCP function. If this value is changed using a TELNET session, the connection will be lost and a new connection will need to be started. The network interface will be restarted after successful storage of the new parameter in nonvolatile memory.

`route <xxx.xxx.xxx.xxx>` *(RS232, TELNET)*

Queries or sets the network IP default route address of the TymServe in dotted quad notation. This variable can also be set automatically using the automatic DHCP function. If this value is changed using a TELNET session, the connection will be lost and a new connection will need to be started. The network interface will be restarted after successful storage of the new parameter in nonvolatile memory.

`host <xxx.xxx.xxx.xxx>` *(RS232, TELNET, SNMP)*

Queries or sets the network TFTP server address for the TymServe in dotted quad notation to be used for downloading the new firmware. This variable can also be set automatically using the automatic DHCP function.

`file <file name>` *(RS232, TELNET, SNMP)*

Queries or sets the filename of the firmware image which will be requested during a TFTP session to download new firmware. This filename can also be set automatically using the automatic DHCP function. This is used for upgrading the FLASH EPROM, which contains the TymServe operating code. This function is not necessary for normal operation of the unit. By default the file name is set to 'ts21.hex'.

`auto <'on' or 'off'>` *(RS232, TELNET, SNMP)*

Queries or sets the auto restart mode. This function is used to control the operation of the TymServe after new firmware downloads. If this mode is enabled, the TymServe will reboot after a successful download and storage of a new version of operating firmware. This will allow the TymServe to begin using the new firmware immediately. If this mode is disabled, the TymServe must be rebooted or power cycled to load the new firmware into RAM.

Auto restart mode:
On = restart after successful firmware updates.
Off = do not restart after successful firmware updates

`ieee802 <'on' or 'off'>` *(RS232, TELNET)*

Queries or sets the network frame header type. The default type is off which means that DIX packet headers are used. 99% of TCP/IP based networks use DIX packet headers. *Do not* change this parameter unless you are certain that the packet format should be changed. If this parameter is changed improperly, the RS-232 access method will have to be used to reset this value.

Frame type parameters used by the network interface:
Off = use Ethernet DIX packet headers.
On = use IEEE 802.2 packet headers.

`ethernet` *(RS232, TELNET, SNMP)*

Queries the hardware Ethernet address. This value is used for definitions of the TymServe recorded in BOOTP or DHCP servers. This value is a unique identifier that is programmed at the factory.

4.2.1 NTP SERVER DIRECTORY

Typing 'ntp <Enter>' under network directory enters the NTP directory, which carries NTP configuration commands.

`authentication<'on' or 'off'>` *(RS232, TELNET, SNMP)*

Sets the NTP daemon up to use the standard NTP authentication mechanism defined in RFC 1305, which provides a way to restrict access to TymServe. See the NTP Authentication appendix for a complete description of this mode.

NTP authentication mode:
Off = disabled
On = enabled

`exclusive <'on' or 'off'>` *(RS232, TELNET, SNMP)*

Sets the NTP daemon up to use the standard NTP 'exclusive' authentication mode, which provides a way to further restrict access to TymServe in addition to authentication defined in RFC1305. See the NTP Authentication appendix for a complete description of this mode.

NTP 'exclusive' authentication mode:
Off = disabled.
On = enabled

`key` *(RS232, TELNET, SNMP)*

This function is used to query or set the NTP authentication key pairs. Refer to the NTP Authentication appendix or RFC1305 for complete details of the use of these keys. Due to export restrictions, only MD5 authentication is supported. Also, to maintain compatibility with the public domain xNTP implementation of the NTP protocol, only ASCII character sequences can be used as authentication keys.

The `<enter>` at the key entry prompt => will display the 1-5 keys entries in the form => keynbr key

Where: keynbr is the NTP authentication key id
key is the 1-8 character MD5 key (ASCII only).

The fields will be blank if no key is stored. New keys can be entered using the same format as shown in Figure 4-3. Entering a blank line will terminate the key entry prompt session.

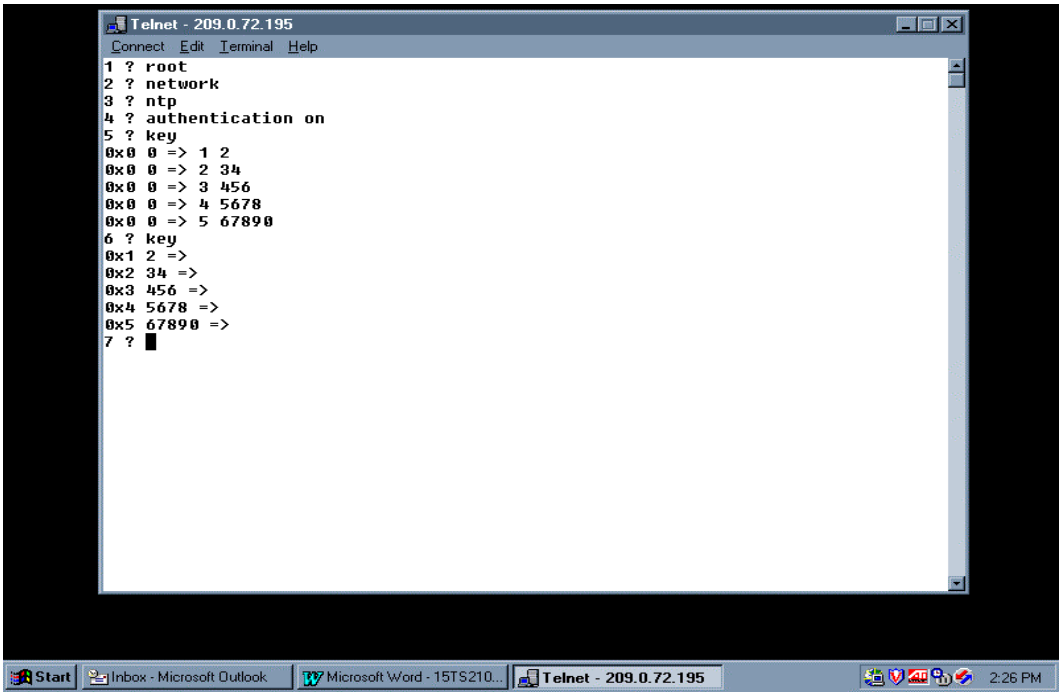


Figure 4-3

`broadcast` *(RS232, TELNET)*

Selects or queries the state of the NTP broadcast mode (NTP mode 5). If this mode is enabled, the TymServe will broadcast a NTP broadcast packet to the local subnet every specified number of seconds. This mode of operation has no impact on the standard NTP client/server mode. Regardless of the state of the NTP broadcast mode, the TymServe will respond to client request packets with server packets.

NTP broadcast mode (NTP mode 5):

0 = disable broadcast

Any Positive Integer ‘x’ = enable broadcast every ‘x’ seconds of time interval

The actual time interval used by NTP broadcast is the value that is calculated by rounding down or equal to the value of ‘x’ to the closest value of power of 2. For example input of 10 sets time to 8 which is 2 to the power 3. Therefore, the actual time interval settings are 1, 2, 4, 8, 16, 64 and so on.

`offset` *(RS232, TELNET)*

Queries an offset record of a NTP client. The TymServe creates a hash table at startup and continually adds and updates entries regarding NTP clients who submit NTP client mode packets. This function is useful for debugging purposes but is not required for normal operation. The values are based on data in the client request and do not include network latencies.

Statistic information of NTP clients

a) Packet count.
b) Maximum offset.
c) Last offset.
d) 10-sample rolling average.

4.2.1.1 NTP BROADCAST DIRECTORY

Typing 'bcast <enter>' under ntp directory enters the ntp broadcast directory. This directory contains commands that allow up to 32 broadcast addresses (subnets) to receive ntp broadcasts from the TymServe. If the broadcast command in the ntp subdirectory has been set to a non-zero value, ntp broadcast messages will be sent to the broadcast addresses entered here. Note that all new commands operate on **broadcast addresses**, not subnet addresses. This is required as different broadcast address schema exist and the programmed broadcast address **must** agree with the broadcast address for the particular subnet programmed into the gateway or router which will deliver the packets.

`add <xxx.xxx.xxx.xxx>` *(RS232, TELNET)*

Adds a ntp broadcast subnet to non-volatile storage.
NOTE: To configure ntp broadcasts to be sent on the local subnet, the value 255.255.255.255 must be used. This corresponds to the local broadcast address in the internal routing tables of the TymServe.

`delete <xxx.xxx.xxx.xxx>` *(RS232, TELNET)*

Deletes a broadcast subnet from non-volatile storage.

`Show` *(RS232, TELNET)*

Displays the currently programed subnet broadcast addresses.

4.2.2 TELNET SERVER DIRECTORY

Entering 'telnet `<enter>'` under 'network' directory gets the TELNET directory. This directory provides TELNET configuration commands.

`auto <'on' or 'off'>` *(RS232, TELNET)*

Select or query the state of the automatic TELNET server mode. If this mode is enabled, the TymServe will be ready to accept TELNET client after power up. Otherwise, no TELNET connection will be allowed. Note that only one TELNET session is allowed at a time.

`start/stop` *(RS232, TELNET)*

'start' command is used to manually start a currently disabled TELNET server. 'stop' command is used to disconnect a currently running TELNET session gracefully or disable TELNET server for the security reasons. TELNET session automatically terminates after an hour of idle time when there is no activity. The length of time for automatic termination is not configurable. The intrinsic command 'trace' can be used to view the status of a TELNET session.

`password <user selected password>` *(RS232, TELNET)*

This command is used to restrict TELNET access to the TymServe. The use of a password for TELNET access can be enabled or disabled from the same subdirectory. TELNET password is transmitted in an un-encrypted format, therefore, the security provided by this feature is just to discourage the casual users. If a password set previously is forgot, this command can be used to retrieve the password in the shell through a serial connection.

`enable <'on' or 'off'>` *(RS232, TELNET)*

This command allows the user to restrict TELNET access to the TymServe to those users who know the password. Setting of the password can be accomplished using the password command available in the same sub-directory. Parameters: 'On' or 'off' (enables or disables TELNET password).

4.2.3 SNMP TOOLS DIRECTORY

Typing 'snmp <Enter>' under network directory enters the SNMP directory. It carries SNMP configuration commands.

`read <read community name>` *(RS232, TELNET)*

Queries or sets the SNMPv1 'read community name'. The default value for this variable is the ASCII string 'public'. The input could be any ASCII string with 1-40 characters. This is an industry standard community name and represents a possible security risk, therefore, this variable should be changed.

`write <write community name>` *(RS232, TELNET)*

Queries or sets the SNMPv1 'write community name'. The default value for this variable is the ASCII string 'private.' The input could be any ASCII string with 1-40 characters. This is an industry standard community name and represents a security risk. Query displays current community name or blank line if it is not configured.

`trap <trap community name>` *(RS232, TELNET)*

Queries or sets the SNMPv1 trap community name. The default value for this variable is the ASCII string 'Datum.' The input could be any ASCII string with 1-40 characters. Query displays current community name or blank line if it is not configured.

address <xxx.xxx.xxx.xxx> *(RS232, TELNET)*

Queries or sets the ip address of the SNMPv1 management console in dotted quad format that should receive any trap messages generated by the TymServe. The default value is 0.0.0.0, which the TymServe will interpret to mean that trap messages should not be transmitted.

name <sysName> *(RS232, TELNET, SNMP)*

Queries or sets the MIB-II variable sysName value as a ASCII string. This string is stored in nonvolatile memory and is most often used to provide a unique identifier to SNMPv1 management consoles. The input could be any ASCII string with 1-40 characters. The default value for this variable is a null string (blank).

location <sysLocation> *(RS232, TELNET, SNMP)*

Queries or sets the MIB-II variable sysLocation value. The input could be any ASCII string with 1-40 characters. This string is stored in nonvolatile memory and is most often used to identify the location installation of a network device to SNMPv1 management consoles. The default value for this variable is a null string (blank).

contact <sysContact> *(RS232, TELNET, SNMP)*

Queries or sets the MIB-II variable sysContact value. The input could be any ASCII string with 1-40 characters. This string is stored in nonvolatile memory and is most often used to identify the technical or administrative contact for a particular network device to SNMPv1 management consoles. The default value for this variable is a null string (blank).

flywheel <SNMPv1 flywheeling trap> *(RS232, TELNET, SNMP)*

Queries or controls the generation of the SNMPv1 flywheeling trap. New value is between 0-86400. Where 0 indicates that a trap should not be sent and any other allowed value indicates the alarm value. Decimal number indicating the number of seconds after the reference timing signal is lost before a snmpv1 trap message will be sent.

4.2.4 DHCP TOOLS DIRECTORY

Typing 'DHCP <Enter>' enters the DHCP directory as shown in Figure 4-4, which carries commands to start or stop DHCP manually and command to enable or disable automatic DHCP when system powers up.

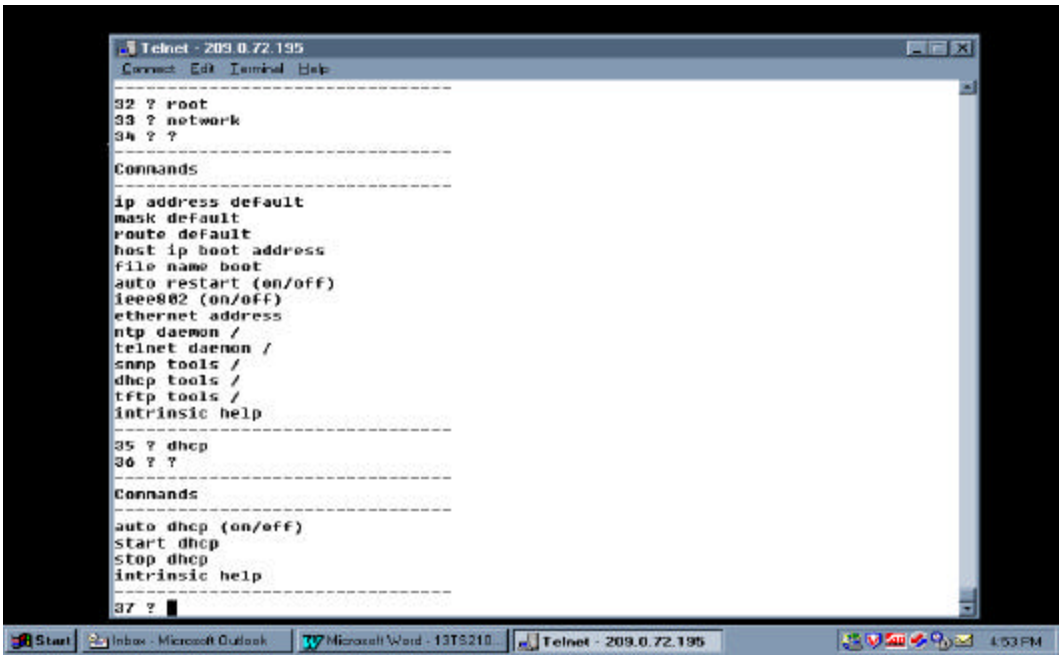


Figure 4-4

`auto <'on' or 'off'>` *(RS232, TELNET, SNMP)*

Select or query the state of the automatic DHCP mode. If this mode is enabled, the TymServe will attempt to download new network parameters from a DHCP server after every reboot. For DHCP sessions, the DHCP server must be programmed with the Ethernet address of the TymServe which can be obtained using the 'ethernet' command in the network sub-directory.

`start/stop` *(RS232, TELNET, SNMP)*

Starts or stops a DHCP session to obtain network parameters from a DHCP or BOOTP server. A DHCP session will set the ip address, network mask, and route variables. In addition, if configured on the DHCP or BOOTP server, the host and TFTP boot file name can be obtained and configured by TymServe. The network interface will be restarted after a successful DHCP session to start using the new variables. The intrinsic command 'trace' can be used to view the status and values relayed during a DHCP session.

4.2.5 TFTP TOOLS DIRECTORY

Typing `<tftp> <enter>` in the network directory enters TFTP directory as shown in Figure 4-5. It carries the TFTP commands.

`auto <'on' or 'off'>` *(RS232, TELNET, SNMP)*

Selects or queries the state of the automatic TFTP mode. If this mode is enabled, the TymServe will attempt to download new operating firmware from a TFTP server after every reboot. For TFTP transfers, the ip address, net mask, route, host, and file variables must be configured. This mode of operation is NOT recommended.

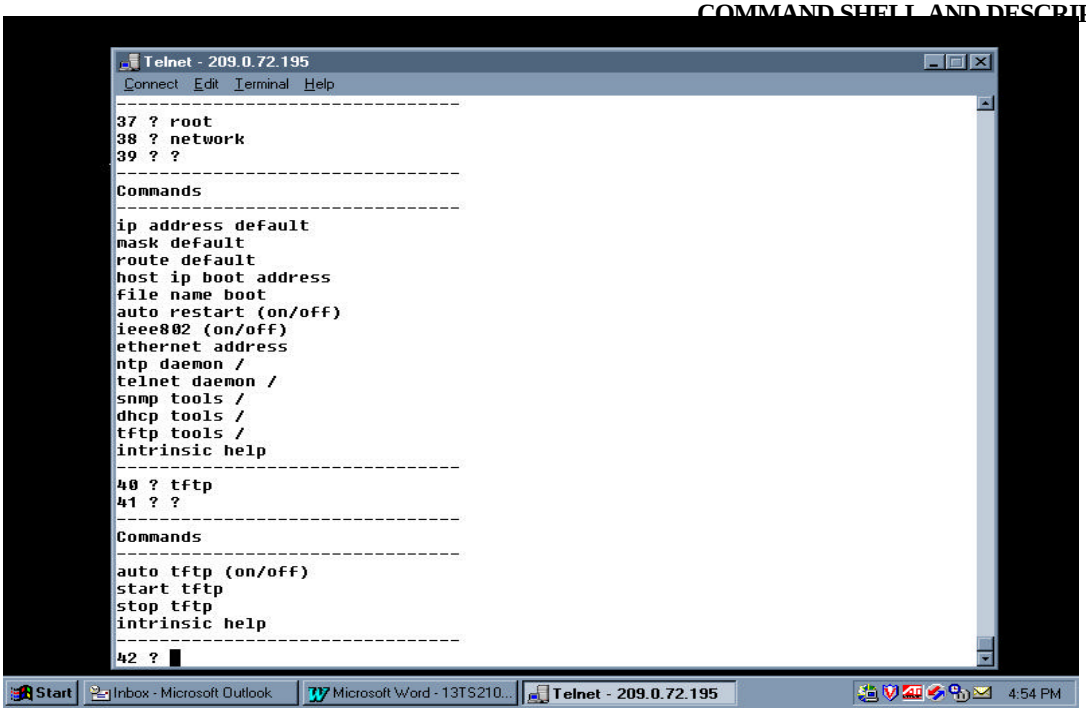


Figure 4-5

`start / stop` *(RS232, TELNET, SNMP)*

Starts or stops a TFTP session to download new operating firmware from a TFTP server. For TFTP transfers, the ip address, net mask, route, host and file variables must be configured. When a TFTP session is started, a rolling indicator will be displayed in TELNET or RS-232 sessions to indicate that a transfer is taking place. If the indicator stops moving, the intrinsic command 'trace' can be used to display the terminal status of the TFTP session. See Appendix C for more information about upgrading the firmware.

4.3 TIMING DIRECTORY

Typing 'timing <Enter>' under root directory enters the timing directory as shown in Figure 4-6. It carries commands to configure the time engine. Some timing related utility tools are available in the directory.

`mode <mode-value>` *(RS232, TELNET, SNMP)*

Selects the reference time source that will be used by the TymServe to synchronize its internal clock, where **mode-value** is one of the following

1 = Freerun

6 = GPS

`time` *(RS232, TELNET, SNMP)*

Queries the current date and time.

`event` *(RS232, TELNET, SNMP)*

Queries the captured event every 100ms. Enters '<Control-C>' to stop it. Captured event about every 100 ms.

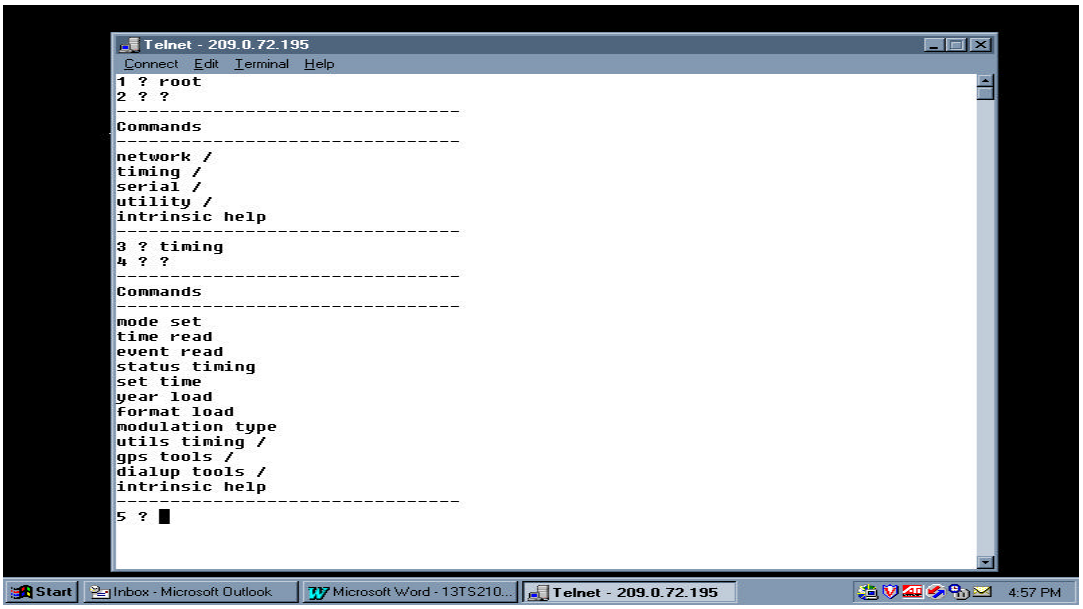


Figure 4-6

`status` *(RS232, TELNET, SNMP)*

Queries the current timing status of the TymServe and this bitmask relays information shows the internal PLL disciplining to the selected timing reference source:

'Status: 0x0v'

where 'v' has value 0 – 7, and can be represented by <Bit 2> <Bit 1> <Bit 0>

Bit 0 = 0 if receiving reference signal (Tracking)
= 1 if NOT receiving reference signal (Flywheeling)

Bit 1 = 0 if Bit 0 = 0 AND if phase difference < 2 usec in mode 6 or 5 usec in mode 0
= 1 otherwise

Bit 2 = 0 if Bit 0 = 0 AND if frequency difference < 5E8 per second (Locked)
= 1 otherwise
where phase difference is between internal oscillator and reference signal and frequency difference is between internal oscillator and reference signal

Thus

0 = 000 – Locked, where Tracking and Locked light are on (stabilized)
1 = 001 – not defined
2 = 010 – Tracking, where Tracking light is on and Locked light off (stabilizing)
3 = 011 – not defined
4 = 100 – Tracking, where Tracking light is on and Locked light off (stabilizing)
5 = 101 – not defined
6 = 110 – Tracking, where Tracking light is on and Locked light off (stabilizing)
7 = 111 – Flywheeling, where Tracking and Locked light are off (cold starting)

set <time> *(RS232, TELNET, SNMP)*

Sets the current time. While this command can be used in any mode, it is useful only for the special 'Freerun' or '1PPS' mode. The input is in one of the following format:

x.y
mm / dd / yyyy hh : mm : ss.x
yyyy ddd hh : mm : ss.x
MON dd yyyy hh : mm : ss.x
hh : mm : ss.x

where x.y
mm / dd / yyyy = month / day / year.
yyyy ddd = year number-of-day-in-the-year.
MON dd yyyy = 3-up-case-letter-month date year.
hh : mm : ss.x = hour : minute : second . second-fraction

UTC time in the format of second . second-fraction since 0:00AM on January 1, 1970, where the trailing .0 is required.

year <xxxx> *(RS232, TELNET, SNMP)*

Sets the year used by the TymServe between 1970 - 2050. While this command can be used in any of the modes, it is not useful for GPS where the year is automatically set from the GPS signal.

leap <type> <Time> *(RS232, TELNET)*

Queries and sets leap second event information. The command without parameter returns current setting of leap second event information. If UTC second count of leap second event time is the past, leap event type should be '0' (zero). If UTC second count of leap second time is the time of future, leap event type should be either '1' or '-1.

where **type** is the leap second event type:
0 = no action or cleared
1 = insertion.
-1 = deletion.
and **Time** is UTC time when leap event will occur in one of the following format:
x.y
mm / dd / yyyy hh : mm : ss.x
yyyy ddd hh : mm : ss.x
MON dd yyyy hh : mm : ss.x
hh : mm : ss.x

where x.y = second . second-fraction.
mm / dd / yyyy = month / day / year.
yyyy ddd = year number-of-day-in-the-year.
MON dd yyyy = 3-up-case-letter-month date year.
hh : mm : ss.x = hour : minute : second . second-fraction

UTC time in the format second . second-fraction since 0:00AM on January 1, 1970. The second-fraction part for leap second event should be always '0'.

Note that this command is very important to TymServe that is running in IRIG-B, Freerun or 1PPS mode as the time reference does not provide leap second information. It is user's responsibility to set leap second event so that TymServe can operate correctly in the event of leap second insertion or deletion. For TymServe that is running in GPS or IEEE1344 IRIG-B mode, the leap second event does not need to be set by this command as it is done automatically but this command can be used to query or verify leap second event.

4.3.1 TIMING UTILITY DIRECTORY

Typing 'utils <Enter>' under timing directory enters the utility timing directory as show in Figure 4-7. Commands in this directory provide utility for timing engine.

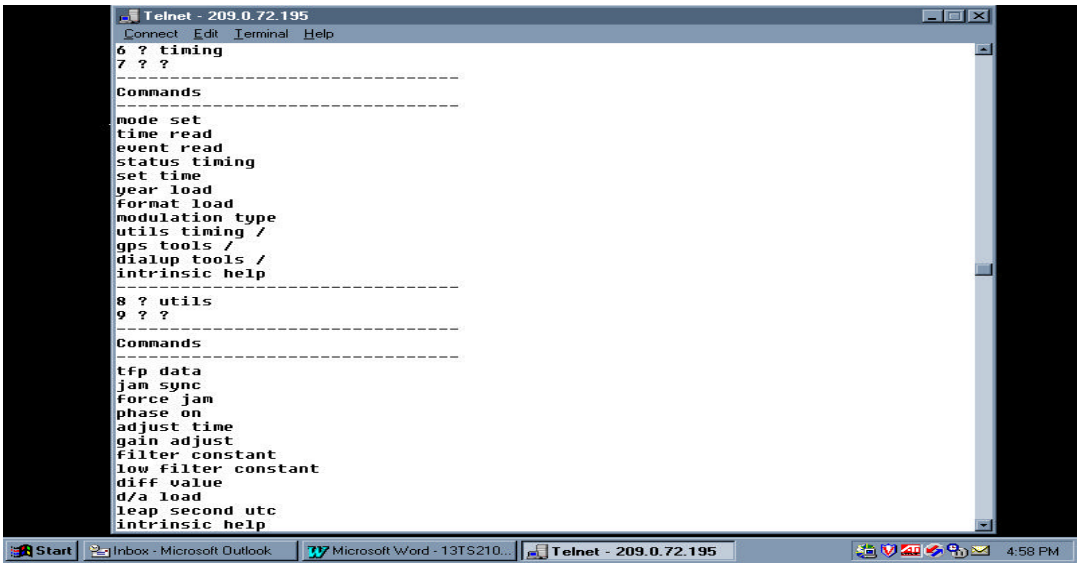


Figure 4-7

`tfp` *(RS232, TELNET)*

Queries various information from the timing co-processor on the TymServe as shown in Table 4-1.

Table 4-1

Input Parameter	Returns
0 (tfp d/a).	0xXXXX (PLL oscillator disciplining voltage in hex).
1 (tfp leap second count).	xx (current leap second count for GPS mode).
2 (firmware version of timing coprocessor).	TS21 V.vvv MM/DD/YYYY HH:MM:SS where V.vvv = major version . minor version. MM / DD / YYYY = month / day / year HH : MM : SS = hour : minute : second.
3 (current timing coprocessor selected timing mode).	Mode (this should match the value returned by the mode command).
4 (timecode mode)	Mode (this should match the value returned by the format command).
5 (gain for 10 MHz phase lock loop).	Value of gain.
6 (filter gain KM)	Value of KM.
7 (filter gain KO).	Value of KO.

`jam <'1' enable or '0' disable >` *(RS232, TELNET)*

Enables or disables automatic jam-synch.

`force` *(RS232, TELNET)*

Forces a jam-synch on next reference pps with GPS.

`phase <'0' or positive disturbance–100ns step size>`*(RS232, TELNET)*

Reads and displays phase values for ten seconds then forces a phase step of user specified magnitude such as 10, 100, or 1000. Then continues to read and display phase until a key is pressed. It reads and displays current 'undisturbanced' phase if '0' is entered. Any value as magnitude of phase disturbance ('0' as no disturbance).

`adjust <value at step size – 100ns>` *(RS232, TELNET)*

Adjusts timing engine time so that speeds up with positive value while slows down with negative value in step size 100ns. Do not change this parameter unless you are absolutely certain that it is necessary to do so. Any value (positive or negative). Step size of adjust at 100ns. Remaining adjustment time.

`gain <Gain value>` *(RS232, TELNET)*

Sets discipline filter gain value for time reference signal. Do not change this parameter unless you are absolutely certain that it is necessary to do so. The gain value range is –32767 to +32768.

`filter <value>` *(RS232, TELNET)*

Sets filter constant. Do not change this parameter unless you are absolutely certain that it is necessary to do so. The floating point value range is 0.0 – 1.0

`Low` *(RS232, TELNET)*

Current low pass filter constant. Not applicable to TymServe currently.

`diff <x>` *(RS232, TELNET)*

Sets diff value to adjust the period of the HC11 generated 1PPS. A value of '0' resets the diff to 33920 which causes the pps to be 0x2E6 clock cycles. Positive values lengthen the diff while negative values shorten the diff. The effect on diff is cumulative, except for value '0'. Do not change this parameter unless you are absolutely certain that it is necessary to do so.

`d/a <DAC control value between 0x0000 - 0xFFFF>`

(RS232, TELNET, SNMP)

Queries or sets the d/a control steering voltage that controls the TymServe oscillator value. This setting, in conjunction with the gain and constant, is used to control the frequency of the on board oscillator. The setting of this value is recommended only for advanced users who wish to control the frequency of the oscillator in the special 'freerunning' mode.

`reference` *(RS232, TELNET)*

Queries latest reference time. The reference time reflects whether the reference source is available currently or not. This command can be used to check when TymServe loses its reference source if the reference time returned is not current.

4.3.2 GPS TOOLS DIRECTORY

Typing 'gps <Enter>' under timing directory enters the gps-tools directory as shown in Figure 4-8. The commands in this directory are useful only when the units work with GPS option.

`position` *(RS232, TELNET)*

Queries the current position information of latitude and longitude in degrees and altitude in meters as reported by the GPS receiver. Until the GPS receiver starts providing position fixes, the position displayed will correspond to the factory default value of 37/-121/0, which corresponds to the position of the factory where the GPS receivers are manufactured.

`filter`

Starts a command loop that continuously requests GPS position information until a key is pressed. This command is a test function for future support of auto position sampling and is not necessary for normal operation of the TymServe.

`velocity` *(RS232, TELNET, SNMP)*

Queries the current 3-dimension velocity information as reported by GPS receiver's movement: East, North and Up in meters per second.

`utcoffset` *(RS232, TELNET, SNMP)*

Queries UTC offset in seconds as reported by the GPS receiver.

`health` *(RS232, TELNET, SNMP)*

Queries information about GPS tracking status and the operational health of the GPS receiver. Note that error code '1' is always set since the GPS receiver battery back-up is not installed.

Error code:

Bit 0 = Battery back-up failed
Bit 1 = Signal processor error
Bit 2 or 3 = Alignment error
Bit 4 = Antenna feed line fault
Bit 5 = Excessive reference frequency error.

`satellites` *(RS232, TELNET, SNMP)*

Queries the selected GPS satellites' ID number. Satellite ID Number varies with location and time.

Operation Mode: auto or manual.
Dimension in Time Calculation: 1-D, 2-D or 3-D.

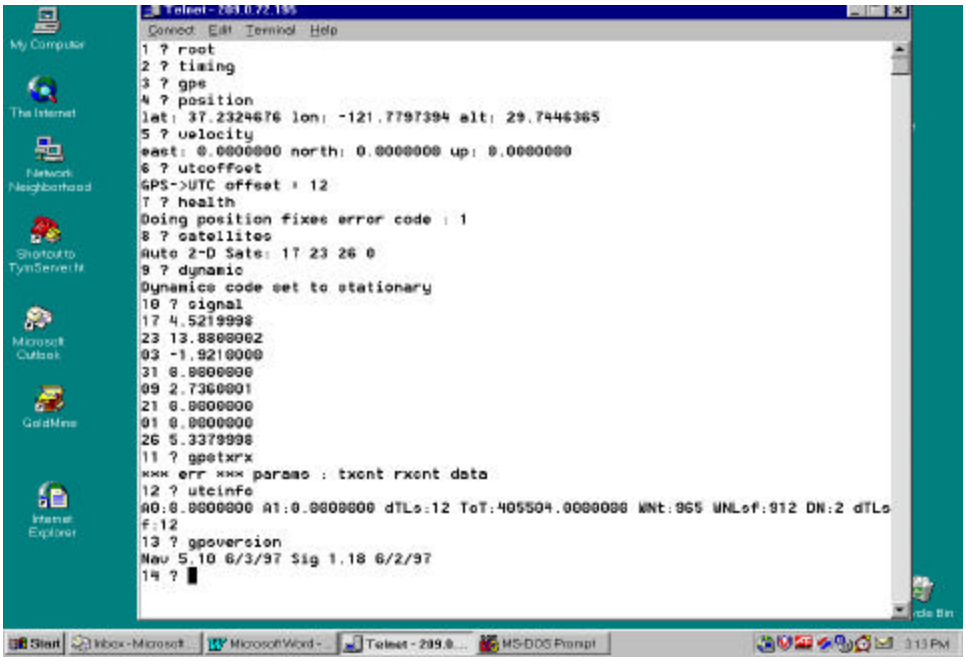


Figure 4-8

`dynamic` *(RS232, TELNET, SNMP)*

Queries dynamic code received from the GPS receiver to indicate the current movement of the receiver e.g. stationary, < 50 knots on sea, < 120 knots on land, < 800 knots in the air and etc.

`signal` *(RS232, TELNET, SNMP)*

Queries up to twelve satellite ID number / signal levels for all the satellites currently being tracked. A satellite will be selected only when its signal value is equal or great than six.

`gpstxrx` *(RS232, TELNET, SNMP)*

Queries user queried data from GPS receiver directly. The third input parameter is the packet number that GPS recognizes as a command for any query. Detailed information about packet number and its corresponding action is available in Datum's GPS Satellite Receiver Addendum User's Guide.

<transmit count> <receive count> <transmitting data>
where the parameters are sent to GPS receiver for query purpose:
<transmit count> = number of bytes in <transmitting data> field to be transmitted to GPS receiver.
<receive count> = number of bytes to expect to be received from GPS receiver as response.
<transmitting data> = packet number as data to be transmitted to GPS receiver.

The number of bytes to be received is specified by <receive count> field.

`utcinfo` *(RS232, TELNET, SNMP)*

Queries UTC time information received from GPS receiver. The actual UTC time equals GPS time minus dTLs. UTC time information is as follows:

A0 = (not used).
A1 = (not used).
dTLs = current leap second.
ToT = (not used).
WNt = current week number.
WNLsf = week number when event did occur or will occur.
DN = day of week for event.
DTLsf = future leap second to occur.

`gpsversion` *(RS232, TELNET, SNMP)*

Queries GPS software versions by GPS receiver's manufacturer, where the first half is the version and date for Navigation software and the second half is for Signal Processor software.

`Setposition` *(RS232, TELNET)*

Sets the position of the TymServe via latitude, longitude and altitude in degrees. This command is a test function for future support of auto position sampling and is not necessary for normal operation of the TymServe. Note that the TymServe is capable of determining its own position at startup without any user input.

`Coldstart` *(RS232, TELNET)*

Sends a command to the embedded GPS receiver inside the TymServe. This command is a test function for future support of auto position sampling and is not necessary for normal operation of the TymServe.

`Setmode` *(RS232, TELNET)*

Sets the position solution mode of the embedded GPS receiver inside the TymServe. This command is a test function for future support of auto position sampling and is not necessary for normal operation of the TymServe.

`Setsat` *(RS232, TELNET)*

Sets the satellite for single satellite operations or selects the overdetermined timing mode. This command is a test function for future support of auto position sampling and is not necessary for normal operation of the TymServe.

`Setdcode` *(RS232, TELNET)*

Sets the dynamics code of the embedded GPS receiver inside the TymServe. This command is a test function for future support of auto position sampling and is not necessary for normal operation of the TymServe.

4.4 SERIAL DIRECTORY

Typing 'serial `<enter>`' under root directory enters the serial directory. The serial commands are used for status or configuration of TymServe through the serial port B.

`port <port> <baud> <data-bit> <stop-bit> <parity>` *(RS232, TELNET)*

Queries or sets serial configuration for port A, B and C. If a port is current in use and its parameters are changed, the connection will be lost and a new connection with proper parameters will need to be started.

where	<port>	= A , B or C
	<baud>	= baud rate between 50 and 115200.
	<data-bit>	= data length between 5 and 8 bits.
	<stop-bit>	= stop bit length either 1 or 2.
	<parity>	= one of parity choices: none, odd, low, even and high.

`statistics` *(RS232, TELNET)*

Queries statistics information including number of transmitting and receiving counts, number of receiving breaks and drops and etc. Statistics information for selected port. Port name: A, B or C.

`clr <port 'A', 'B' or 'C'>` *(RS232, TELNET)*

Clears port statistics selected by user.

`downloads` *(RS232, TELNET)*

This function tells the TymServe to start looking for a new version of firmware to be downloaded via the rear panel RS-232 port Serial B. After issuing this command, the user should select text file download from his ASCII terminal and stream the file containing the new firmware to the TymServe. Once the command is issued, a rolling indicator will be displayed to confirm that the TymServe is waiting for a firmware download. Once the file begins streaming, the rolling indicator should continue. If instead, a string of errors are reported, check that the record terminators (i.e. cr, cr-lf, etc.) are properly matched with the settings in the serial menu. This is used for upgrading the FLASH EPROM which contains the TymServe operating code. This function is not necessary for normal operation of the unit. The file format is Motorola S-records.

`echo <port> <echo-state> <translate-state>` *(RS232, TELNET)*

Qurries or sets the state of echo and translate for port A, B, C and TELNET port. If echo is on, a character entered will be echoed on screen, otherwise it will not. While if translate is on, the carriage-return entered will be translated to new-line-feed. The translation should be coordinated with the setting of the serial for hyper-terminal, tip or whatever term emulator.

where	<port>	= A , B or C
		= telnd (for TELNET port).
	<echo-state>	= on (echo enabled).
		= off (echo disabled).
	<translate-state>	= on (translate enabled).
		= off (translate disabled).

`sysplex <'on' or 'off'>` *(RS232, TELNET)*

Transmits UTC time through the rear panel serial port when sysplex is set to 'on.' Once sysplex is on, system starts to poll the serial port A. It starts the time transmission after it receives 'c' or 'C' character via the serial port. It stops when it receives 'r' or ' RS232' character. It transmits UTC time in ASCII format once every second at the time when carriage-return is transmitted, i.e. carriage-return is transmitted on time each second.

`auto sysplex <'on' enable or 'off' disable>` *(RS232, TELNET)*

Queries and sets if the TymServe should automatically start the sysplex timer output without waiting for an input 'c' or 'C' character. This command is only useful if sysplex has been turned on. If auto is set on, the TymServe will ignore any input characters and continuously generate serial time output.

4.5 UTILITY DIRECTORY

Typing 'utility <enter>' under root directory gets the utility directory. These commands provide a set of utility commands for TymServe.

default *(RS232, TELNET)*

Sets all TymServe system parameters to default values that are hard-coded in the firmware. These parameters include (1) system state; (2) network related parameters such as IP address, net mask, default route, boot host and boot file name; (3) serial related parameters such baud rate, stop bit, parity for all ports and etc. If this command is executed while DHCP is running, network related parameters such as ip address, net mask and etc. will be restored after they are set to default values at the time when ip address lease needs to be renewed.

restart *(RS232, TELNET)*

Restarts the system.

net <'on' or 'off'> *(RS232, TELNET)*

Queries or sets the state of the network trace facility. The network trace uses promiscuous mode to dump network data to the trace buffers. This function is provided for debugging purposes and is not necessary for normal operation of the TymServe

statistics *(RS232, TELNET)*

Queries system statistics including number of ethernet transmitting, receiving and dropping, timer usage by system, DHCP binding number, and TFTP, TELNET server and NTP Server session numbers. Statistical information about network activities and system timer usage.

utc_binary *(RS232, TELNET)*

Converts input time to UTC second counts since 0:00AM on January 1, 1970. If only hh : mm : ss.x is used as input, it assumes the date is January 1, 1970.

one of following format:
mm / dd / yyyy hh : mm : ss.x
yyyy ddd hh : mm : ss.x
MON dd yyyy hh : mm : ss.x
hh : mm : ss.x

where
mm / dd / yyyy = month / day / year.
yyyy ddd = year number-of-day-in-the-year.
MON dd yyyy = 3-up-case-letter-month date year.
hh : mm : ss.x = hour : minute : second . second-fraction

UTC time in the format of second . second-fraction since 0:00AM on January 1, 1970.

`utc_julian` *(RS232, TELNET)*

Converts input time to Julian time. If only hh : mm : ss.x is used as input, it assumes the date is January 1, 1970.

one of following format:
x.y
mm / dd / yyyy hh : mm : ss.x
MON dd yyyy hh : mm : ss.x
hh : mm : ss.x

where
x.y = second . second-fraction.
mm / dd / yyyy = month / day / year.
MON dd yyyy = 3-up-case-letter-month date year.
hh : mm : ss.x = hour : minute : second . second-fraction

Julian time in the format of yyyy ddd hh : mm : ss.x, where
yyyy ddd = year number-of-day-in-the-year.

`utc_calendar` *(RS232, TELNET)*

Converts input time to Calendar time. If only hh : mm : ss.x is used as input, it assumes the date is January 1, 1970.

one of following format:
x.y
yyyy ddd hh : mm : ss.x
MON dd yyyy hh : mm : ss.x
hh : mm : ss.x

where
x.y = second . second-fraction.
yyyy ddd = year number-of-day-in-the-year.
MON dd yyyy = 3-up-case-letter-month date year.
hh : mm : ss.x = hour : minute : second . second-fraction

Calendar time in the format of mm / dd / yyyy hh : mm : ss.x, where
mm / dd / yyyy = month / day / year

`utc_ascii` *(RS232, TELNET)*

Converts input time to ASCII time. If only hh : mm : ss.x is used as input, it assumes the date is January 1, 1970.

one of following format:
x.y
mm / dd / yyyy hh : mm : ss.x
yyyy ddd hh : mm : ss.x
hh : mm : ss.x

where
x.y = second . second-fraction.
mm / dd / yyyy = month / day / year.
yyyy ddd = year number-of-day-in-the-year.
hh : mm : ss.x = hour : minute : second . second-fraction

ASCII time in the format of MON dd yyyy hh : mm : ss.x, where
MON dd yyyy = 3-up-case-letter-month date year.

4.6 INTRINSIC HELP

Intrinsic help commands are commands that can be used in any directory as a basic shell command tool. To get a list of intrinsic commands, enter <`intrinsic`>.

`pop` *(RS232, TELNET)*

Moves the command shell to the previous level. The command `<root pop> <enter>` disconnects the TELNET session.

`root` *(RS232, TELNET)*

Moves the command shell to point to the main shell directory. While this function is not necessary for operation, it can be useful for navigating the command shell.

`exit` *(RS232, TELNET)*

This command exits the TELNET client session while it is active. This command has no effect if it is executed in the serial session. To terminate the existing TELNET session from a serial session use ‘stop’ command as explained in the TELNET directory.

`version` *(RS232, TELNET)*

Displays the current firmware version and build date of the TymServe

Rev V.VVV MM/DD/YYYY HH:MM:SS.
Filename: ts21.hex
Where V.VVV = major.minor version.
MM / DD / YYYY = month / day / year, HH : MM : SS = hour : minute : second

Filename: Name of Firmware Image file

`trace` *(RS232, TELNET)*

Displays current contents of the trace buffers. These buffers contain information reported by various subsystems of the TymServe. After a restart, the buffers will contain status reports from each major initialization steps. If an error occurs, a message is written to the trace buffers whenever possible. The trace buffers are implemented in a circular buffer where newer entries will overwrite older information when the buffers are full. The clear command can be used to flush the trace buffers. If you suspect an error has occurred, use this command to query the system.

`clear` *(RS232, TELNET)*

Clears the trace buffers. While this function is not necessary for operation, it can be useful for debugging purposes.

`stamp` *(RS232, TELNET)*

Querries the time stamp of internal operating system clock which is set to zero when TymServe is powered on. Time stamp of Internal operating system clock in millisecond.

`history` *(RS232, TELNET)*

Displays last fifteen commands that have been executed.

`pause` *(RS232, TELNET)*

Waits for user specified number of milliseconds and then executes next command if it is provided in the same command line following the number. For example, 'pauses 5000 history' command waits for five seconds and then executes history command. Input any integer number as number of milliseconds to pause.

`Peek <address> <count>` *(RS232, TELNET)*

Displays memory contents in number of bytes specified by count at location specified by address. The address and count can be hex numbers but should begin with '0x.'

where <address> = starting address of memory to be displayed.
<count> = number of bytes of memory to be displayed.

Memory contents in hex at memory location specified by input parameters.

`compare <src address> <dst address> <count>` *(RS232, TELNET)*

Compares memory contents of source and destination specified by addresses. If they are the same, nothing is returned. Otherwise only the contests that are different are displayed.

where <src address> = starting address of source.
<dst address> = starting address of destination.
<count> = number of bytes to be compared.

repeat *(RS232, TELNET)*

Repeats number of times specified by user to execute command prior to repeat commands in the same command line. Any integer number for repeat in executing commands

*(RS232, TELNET)*

Provides shell command parsing a mark that indicates the end of command line.

? *(RS232, TELNET)*

Displays various commands available in current directory for use on the TymServe.

CHAPTER FIVE

SNMP CONFIGURATION AND CONTROL

5.0 GENERAL

In order to use SNMPv1 set and request packets, the network parameters must be configured using either RS-232 or telnet access methods. Once the network parameters have been set, SNMPv1 packets can be sent to configure the operating mode. Compile the ASN.1 datumMIB definition file provided on the SNMPv1 management platform. Select the variable tsTengMode to set the operating mode. The initial read community name is "public" and the initial read/write community name is "private."

Datum has designed and implemented a custom MIB extension to complement our support of the MIB-II variable set. The datumMIB provides access to data and controls specific to the TymServe and NTP service. The datumMIB extension also provides SNMPv1 traps, which are generated by the TymServe in the event of NTP leap indicator changes or extended reference time source signal losses. An ASN.1 definition of the datumMIB extension is provided on a MSDOS formatted floppy diskette. This file is suitable for compilation on any SNMP management platform which supports RFC1155 (SMI), RFC1213 (TRAPS), and RFC1215 (OBJECT MACRO DisplayString).

Regardless of plans for SNMPv1 access, these community names should be changed for security reasons. This section briefly covers the Datum MIB Extension, MIB compilation and Security aspects of the SNMP.

5.1 DATUM MIB II EXTENSION

The data available from the TymServe through SNMP is based on the MIB-II variable set which has been enhanced with a custom MIB extension to provide data unique to NTP and the TymServe. Manufacturer specific, or enterprise, MIB extensions are given a unique identifier which defines where the extension is located in the MIB tree. Datum Inc has been assigned an enterprise number of 601. Datum Inc is currently in the process of compiling an RFC to define a standardized NTP variable set and allocate space in MIB-II. Due to this ongoing process, the current MIB extension is defined as experimental (volatile). If you have any input regarding the RFC, or changes to the MIB extension, send email to info@datum.com. Please include datumMIB in the subject. A definition of the MIB extension in included in the Appendix A.

5.2 ADDITIONAL STORED MIB VARIABLES

The first 39 characters of the sysName, sysContact, and sysLocation will be stored in flash eprom and loaded during any subsequent initialization or powerup. These variables can be set through either the command interface or SNMP set requests.

5.3 MIB COMPILATION

Most SNMP management platforms are capable of reading MIB-II data. The Datum specific MIB extension needs to be added to the manager. Typically, these MIB extensions are defined using a syntax known as ASN.1. The Datum MIB extension is provided on a DOS format floppy disk and is defined using ASN.1 and includes imports from RFC1155-SMI, RFC-1212, RFC1213-MIB and RFC-1215. The user should consult their SNMP manager documentation in order to determine how to compile the Datum MIB extension into their SNMP software package.

5.4 SECURITY

Once the SNMP management software has been configured to recognize data from the TymServe, the security parameters on the manager need to be defined to match those set on the TymServe.

5.5 SNMPv1

The security parameters for SNMPv1 are based on a community name, which is a string of ASCII characters (i.e., “public”), and an IP address. The TymServe defines the IP address such that SNMPv1 packets will be accepted from any IP address which has a valid community name. The community name in SNMPv1 packets defines the level of access. The TymServe allows for the definition of two community names, one which is read-only and another which has read-write privileges. The user should consult their SNMP manager documentation in order to determine how to create an object and set the community names. Typically, the information required will be the IP address of the TymServe and the community names which were set through the command interface of the TymServe.

5.6 MIB II EXTENSION FILE

Datum MIB II extension defined in the next section is available to users in the Utility Diskette shipped with the TymServe. The MIB file name is ‘TS2100.MIB’. This file is also available through Datum’s FTP site: ftp.datum.com, in /pub/NetProducts/TymServe2100 directory.

5.7 DATUM MIB EXTENSION

```
DATUM DEFINITIONS ::= BEGIN

IMPORTS
OBJECT-TYPE
FROM RFC-1212
DisplayString
FROM RFC1213-MIB
TRAP-TYPE
FROM RFC-1215
enterprises
FROM RFC1155-SMI;

datumMIB    OBJECT IDENTIFIER ::= {enterprises 601}

bancomm     OBJECT IDENTIFIER ::= {datumMIB 1}
timing      OBJECT IDENTIFIER ::= {datumMIB 2}
austron     OBJECT IDENTIFIER ::= {datumMIB 3}
fts         OBJECT IDENTIFIER ::= {datumMIB 4}
efratom     OBJECT IDENTIFIER ::= {datumMIB 5}
experiment  OBJECT IDENTIFIER ::= {datumMIB 99}

products        OBJECT IDENTIFIER ::= {bancomm 1}

ts2000          OBJECT IDENTIFIER ::= {products 1}
ts2100          OBJECT IDENTIFIER         ::= {products 2}

version         OBJECT IDENTIFIER ::= {ts2100 1}

ntp                     OBJECT IDENTIFIER ::= {version 1}
tyming                  OBJECT IDENTIFIER ::= {version 2}
gps                     OBJECT IDENTIFIER ::= {version 3}
dialup                  OBJECT IDENTIFIER ::= {version 4}
net                     OBJECT IDENTIFIER ::= {version 5}
etc                     OBJECT IDENTIFIER ::= {version 6}
```

ntpLeapIndicator OBJECT-TYPE
SYNTAX INTEGER {
nowarning(1),
leapinsertion(2),
leapdeletion(3),
unsynchronized(4)}
ACCESS read-write
STATUS mandatory
DESCRIPTION
"NTP Leap Indicator. This is a two-bit code warning of an impending leap second to be inserted into the NTP timescale. The bits are set before 23:59 on the day of insertion and reset after 00:00 on the following day. This causes the number of seconds (rollover interval) in the day of insertion to be increased or decreased by one. In the case of primary servers the bits are set by operator intervention, while in the case of secondary servers the bits are set by the protocol. The two bits, bit 0 and bit 1, respectively, are coded as follows:

00	no warning
01	last minute has 61 seconds
10	last minute has 59 seconds
11	alarm condition(clock not synchronized)

In all except the alarm condition(11), NTP itself does nothing with these bits, except pass them on to the time-conversion routines that are not part of NTP. The alarm condition occurs when, for whatever reason, the local clock is not synchronized, such as when first coming up or after an extended period when no primary reference source is available."
::= {ntp 1}

ntpMode OBJECT-TYPE
SYNTAX INTEGER {
unspecified (1),
symactive (2),
sympassive (3),
client (4),
server (5),
broadcast (6),
reservedctl (7),
reservedpriv (8)}
ACCESS read-only
STATUS mandatory
DESCRIPTION
"NTP association mode. This is an integer indicating the association mode, with values coded as follows:
==========
0 unspecified
1 symmetric active
2 symmetric passive
3 client
4 server
5 broadcast
6 reserved for NTP control messages
7 reserved for private use
==========
***Note*:** In the Datum 2100 series, this value is currently ALWAYS set to 4 (server only)."
::= {ntp 2}

ntpStratum OBJECT-TYPE
SYNTAX INTEGER (0..255)
ACCESS read-only
STATUS mandatory
DESCRIPTION
"Current NTP stratum level. This is an integer indicating the stratum of the local clock with values defined as follows:
==========
0 unspecified
1 primary reference (e.g., calibrated atomic clock, radio clock)
2-255 secondary reference (via NTP)
==========
***Note*:** In the Datum 2100 series, this value is currently ALWAYS 1 (primary reference)."
::= {ntp 3}

ntpPrecision OBJECT-TYPE
SYNTAX INTEGER (-127..127)
ACCESS read-only
STATUS mandatory
DESCRIPTION
"Current NTP precision value. This is a signed integer indicating the precision of the various clocks, in seconds to the nearest power of two. The value must be rounded to the next larger power of two; for instance, a 50-Hz (20ms) or 60-Hz (16.17ms) power-frequency clock would be assigned the value -5 (31.25ms), while a 1000-Hz (1ms) crystal-controlled clock would be assigned the value -9 (1.95ms)."
::= {ntp 4}

ntpRefClkID OBJECT-TYPE
SYNTAX DisplayString (SIZE (1..40))
ACCESS read-only
STATUS mandatory
DESCRIPTION
"NTP Reference Clock Identifier. This is a 32 bit code identifying the particular reference clock. In the case of stratum 0 (unspecified) or stratum 1 (primary reference), this is a four-octet, left-justified, zero-padded ASCII string. While not enumerated as part of the NTP spec, the following are suggested ASCII identifiers:

DCN	DCN routing protocol
NIST	NIST public modem
TSP	TSP time protocol
DTS	Digital Time Service
ATOM	Atomic clock (calibrated)
VLF	VLF radio (OMEGA,etc.)
callsign	Generic radio
LORC	LORAN-C radionavigation
GOES	GOES UHF environment satellite
GPS	GPS UHF satellite positioning

The following ref ids are used by the 2100:

GPS	TS2100-GPS (GPS satellite)
FREE	TS2100-ALL (INTERNAL CLOCK)"

::= {ntp 5}

ntpRefTime OBJECT-TYPE
SYNTAX DisplayString (SIZE(1..40))
ACCESS read-only
STATUS mandatory
DESCRIPTION
"NTP Reference Timestamp. This is the time, in timestamp format (converted to DisplayString), when the local clock was last updated. If the local clock has never been synchronized, the value is zero."
::= {ntp 6}

ntpVersion OBJECT-TYPE
SYNTAX INTEGER (0..127)
ACCESS read-only
STATUS mandatory
DESCRIPTION
"NTP Version. This is an integer indicating the version number of the sender. NTP messages will always be sent with the current version number NTP.VERSION and will always be accepted if the version number matches NTP.VERSION. Exceptions may be advised on a case-by-case basis at times when the version number is changed.
==========
Note: The 2100 series was implemented using NTP version 3. However, the 2100 series will accept a version of 2 or 3 and return the same version number in the packet. This behavior is subject to change."
::= {ntp 7}

ntpAuthOnOff OBJECT-TYPE
SYNTAX INTEGER (0..1)
ACCESS read-write
STATUS mandatory
DESCRIPTION
"Enable or disable MD5 Authentication Mode."
::= {ntp 8}

ntpAuthOnlyOnOff OBJECT-TYPE
SYNTAX INTEGER (0..1)
ACCESS read-write
STATUS mandatory
DESCRIPTION
"Enable or disable MD5 Authentication Only Mode."
::= {ntp 9}

ntpNumberRequests OBJECT-TYPE
SYNTAX INTEGER (0..32768)
ACCESS read-write
STATUS mandatory
DESCRIPTION "This variable is a rollover counter which reflects the number of ntp packets received by the 2100. It is valid for all versions of the 2100. The counter may be set to 0."
::= {ntp 10}

```
ntpLeapChange TRAP-TYPE
        ENTERPRISE ntp
        VARIABLES {ntpLeapIndicator}
        DESCRIPTION "The trap indicates a change in state of the ntp leap indicator.  It will
        pass the new value of the          leap indicator."
::=0

tymingTimeSrcUTCOffset OBJECT-TYPE
        SYNTAX INTEGER (-11..12)
        ACCESS read-write
        STATUS mandatory
        DESCRIPTION "This variable is valid for the 2100-IRIG or the 2100-GPS when
        operating in IRIG decoder mode.  The variable represents the offset of the input
        timecode from UTC in signed hours.  The allowable values for this variable are -11
        through 12.If this variable is queried on the 2100-ACTS it will return 0."
::= {tyming 1}

tymingStatus OBJECT-TYPE
        SYNTAX DisplayString (SIZE(1..40))
        ACCESS read-only
        STATUS mandatory
        DESCRIPTION "This variable reflects the current status information on the 2100 Time
        and Frequency Processor.  This info is only available on the IRIG and GPS versions.
        The string will indicate either Tracking or Flywheeling.  This is a direct reflection of bit 0
        of the 2100  status bit 0.  Bits 1 & 2 are not relevant for network users.
        Status Register Definitions
        ================================================
        bit 0   1 = flywheeling(not locked)
                0 = locked(to selected reference)
        bit 1   1 = time offset > X microseconds
                0 = time offset < X microseconds
                    (X = 5 for IRIG | X = 2 for GPS)
        bit 2   1 = frequency offset > 5E8
                0 = frequency offset < 5E8
        ================================================"
::= {tyming 2}
```

```
tymingMode OBJECT-TYPE
        SYNTAX INTEGER {
                timecode (1),
                freerun (2),
                internal (3),
                gps (7)}
        ACCESS read-write
        STATUS mandatory
        DESCRIPTION "Operating mode of the timing engine."
::= {tyming 3}

tymingTime OBJECT-TYPE
        SYNTAX DisplayString (SIZE(1..40))
        ACCESS read-write
        STATUS mandatory
        DESCRIPTION "Time returned from the time registers."
::= {tyming 4}

tymingEventTime OBJECT-TYPE
        SYNTAX DisplayString (SIZE(1..40))
        ACCESS read-only
        STATUS mandatory
        DESCRIPTION "Time returned from the event registers."
::= {tyming 5}

tymingYear OBJECT-TYPE
        SYNTAX INTEGER
        ACCESS read-write
        STATUS mandatory
        DESCRIPTION "Year used by timing engine."
::= {tyming 6}

tymingInTcFormat OBJECT-TYPE
        SYNTAX INTEGER {
                tcIRIGA  (65),
                tcIRIGB  (66),
                tc2137   (67),
                tcNASA36 (78),
                tcXR3    (88)}
        ACCESS read-write
        STATUS mandatory
        DESCRIPTION "Format of the input timecode (i.e. B,N)."
::= {tyming 7}
```

```
tymingInTcModulation OBJECT-TYPE
        SYNTAX INTEGER {
                        dclevel (68),
                        modulated (77)}
        ACCESS read-write
        STATUS mandatory
        DESCRIPTION "AM or DC format timecode."
::= {tyming 8}

tymingOutTcFormat OBJECT-TYPE
        SYNTAX INTEGER {
                        tcIRIGB (66)}
        ACCESS read-write
        STATUS mandatory
        DESCRIPTION "Timecode Generator output format."
::= {tyming 9}

tymingVersion OBJECT-TYPE
        SYNTAX DisplayString (SIZE(1..40))
        ACCESS read-only
        STATUS mandatory
        DESCRIPTION "String containing timing engine version and
        creation date and time."
::= {tyming 10}

tymingLeapSeconds OBJECT-TYPE
        SYNTAX INTEGER (0..255)
        ACCESS read-write
        STATUS mandatory
        DESCRIPTION "Current leap second count in timing engine."
::= {tyming 11}
tymingD2a OBJECT-TYPE
        SYNTAX INTEGER (0..65535)
        ACCESS read-write
        STATUS mandatory
        DESCRIPTION "oscillator disciplining d2a value."
::= {tyming 12}

tymingKval OBJECT-TYPE
        SYNTAX DisplayString (SIZE(1..40))
        ACCESS read-write
        STATUS mandatory
        DESCRIPTION "Oscillator disciplining filter constant value."
::= {tyming 13}
```

```
tymingFlyPeriod OBJECT-TYPE
        SYNTAX INTEGER
        ACCESS read-write
        STATUS mandatory
        DESCRIPTION
        "Period in seconds of allowable flywheeling before the tymingFlywheel trap will be sent.
         If the value is set to 0, the trap will never be generated."
::= {tyming 14}

tymingFlywheel TRAP-TYPE
        ENTERPRISE tyming
        VARIABLES {ntpRefTime}
        DESCRIPTION
        "The trap is intended to provide notification of extended flywheeling events. The user
        programmable value tymingFlyPeriod is the number of seconds the 2100 may
        flywheeling before a trap is sent."
::=0

gpsPosition OBJECT-TYPE
        SYNTAX DisplayString (SIZE(1..80))
        ACCESS read-only
        STATUS mandatory
        DESCRIPTION
         "This variable returns a position fix. It is only valid on the 2100-GPS. The returned
        string will contain the latitude & longitude expressed in degrees & minutes and the
        altitude in meters. If this variable is queried on the 2100-IRIG or 2100-ACTS the
        returned value will be N/A."
::= {gps 1}

gpsVelocity OBJECT-TYPE
        SYNTAX DisplayString (SIZE(1..80))
        ACCESS read-only
        STATUS mandatory
        DESCRIPTION
         "This variable returns a velocity fix. It is only valid on the 2100-GPS. The returned
        string will contain the East-North-Up velocity expressed in meters/second. If this
        variable is queried on the 2100-IRIG or 2100-ACTS the returned value will be N/A."
::= {gps 2}
```

```
gpsUTCOffset OBJECT-TYPE
        SYNTAX INTEGER (0..127)
        ACCESS read-write
        STATUS mandatory
        DESCRIPTION
        "This variable returns the current offset between the monotonic time maintained by the
        GPS satellite constellation and UTC time.  This value is commonly referred to as the
        leap second count.  It is only valid on the 2100-GPS. This value is obtained from the
        GPS receiver but there may be a time lag between the incidence of a leap second
        correction and the capture of that correction by the 2100-GPS.  For this reason the
        user is allowed to program the leap second value which will be used until the 2100-GPS
        receives the leap second count from the GPS satellite constellation.  If this variable is
        queried on the 2100-IRIG or 2100-ACTS the value returned will be 0."
::= {gps 3}

gpsHealth OBJECT-TYPE
        SYNTAX DisplayString (SIZE(1..80))
        ACCESS read-only
        STATUS mandatory
        DESCRIPTION "Health packet from GPS."
::= {gps 4}

gpsAllinView OBJECT-TYPE
        SYNTAX DisplayString (SIZE(1..80))
        ACCESS read-only
        STATUS mandatory
        DESCRIPTION "Satellites in view packet from GPS."
::= {gps 5}

gpsMode OBJECT-TYPE
        SYNTAX INTEGER (0..20)
        ACCESS read-write
        STATUS mandatory
        DESCRIPTION "GPS Receiver Operating Mode."
::= {gps 6}

gpsDynamicsCode OBJECT-TYPE
        SYNTAX INTEGER { land (1), sea  (2), air  (3), static (4)}
        ACCESS read-write
        STATUS mandatory
        DESCRIPTION "GPS Receiver Dynamics Code."
::= {gps 7}
```

dialForceCall OBJECT-TYPE
SYNTAX INTEGER (0..1)
ACCESS read-write
STATUS mandatory
DESCRIPTION "The variable is used to either initiate a call to the NIST ACTS service or abort a call that is in progress. It is only valid on the 2100-ACTS. Reading this value will return abort(0) if offline and call(1) if online. Sending a set of this value to 0 or 1 will cause the 2100-ACTS to take the appropriate action."
::= {dialup 1}

dialCallRefTime OBJECT-TYPE
SYNTAX DisplayString (SIZE(1..40))
ACCESS read-write
STATUS mandatory
DESCRIPTION "FORMAT: MM/DD/HH. All 8 digits are required for set requests. Use \ or / as delimiters. The variable is used to read or set the call reference time. It is only valid on the 2100-ACTS. For more information on this variable, refer to the 2100 Users Manual. If this variable is queried on the 2100-GPS or 2100-IRIG it will return N/A."
::= {dialup 2}

dialCallInterval OBJECT-TYPE
SYNTAX DisplayString (SIZE(1..40))
ACCESS read-write
STATUS mandatory
DESCRIPTION "FORMAT: MM/DD/HH. All 8 digits are required for set requests. Use \ or / as delimiters. The variable is used to read or set the call interval time. It is only valid on the 2100-ACTS. For more information on this variable, refer to the 2100 Users Manual. If this variable is queried on the 2100-GPS or 2100-IRIG it will return N/A."
::= {dialup 3}

dialCallOnReset OBJECT-TYPE
SYNTAX INTEGER (0..1)
ACCESS read-write
STATUS mandatory
DESCRIPTION "The variable is used to read or set the 2100- ACTS Reset behavior. It is only valid on the 2100-ACTS. For more information on this variable, refer to the 2100 Users Manual. If this variable is queried on the 2100-GPS or 2100-IRIG it will return 0."
::= {dialup 4}

```
        dialInitString OBJECT-TYPE
                SYNTAX DisplayString (SIZE(1..40))
                ACCESS read-write
                STATUS mandatory
                DESCRIPTION "The variable is used to read or set the NIST ACTS dial prefix.  It is
                only valid on the 2100-ACTS.  For more information on this variable, refer to the 2100
                Users Manual.  If this variable is queried on the 2100-GPS or 2100-IRIG it will return
                N/A."
        ::= {dialup 5}

        dialPhonePrefix OBJECT-TYPE
                SYNTAX DisplayString (SIZE(1..40))
                ACCESS read-write
                STATUS mandatory
                DESCRIPTION "The variable is used to read or set the NISTACTS dial prefix.  It is
                only valid on the 2100-ACTS.  For more information on this variable, refer to the 2100
                Users Manual. If this variable is queried on the 2100-GPS or 2100-IRIG it will return
                N/A."
        ::= {dialup 6}

        dialPhoneNumber OBJECT-TYPE
                SYNTAX DisplayString (SIZE(1..40))
                ACCESS read-write
                STATUS mandatory
                DESCRIPTION "The variable is used to read or set the NISTACTS phone number.  It
                is only valid on the 2100-ACTS.  For more information on this variable, refer to the
                2100 Users Manual. If this variable is queried on the 2100-GPS or 2100-IRIG it will
                return N/A."
        ::= {dialup 7}

        dialServiceType OBJECT-TYPE
                SYNTAX INTEGER (0..6)
                ACCESS read-write
                STATUS mandatory
                DESCRIPTION "The variable is used to read or set the type of time service being
accessed."
        ::= {dialup 8}

        netTftpHost OBJECT-TYPE
                SYNTAX DisplayString (SIZE(1..40))
                ACCESS read-write
                STATUS mandatory
                DESCRIPTION "Ip address of tftp host machine."
        ::= {net 1}
```

```
netTftpFile OBJECT-TYPE
        SYNTAX DisplayString (SIZE(1..40))
        ACCESS read-write
        STATUS mandatory
        DESCRIPTION "File name of boot file image for tftp."
::= {net 2}

netAutoRestart OBJECT-TYPE
        SYNTAX INTEGER (0..1)
        ACCESS read-write
        STATUS mandatory
        DESCRIPTION "Enable or disable automatic restarts after downloading new
        firmware."
::= {net 3}

netRestart OBJECT-TYPE
        SYNTAX INTEGER (0..1)
        ACCESS write-only
        STATUS mandatory
        DESCRIPTION "Force a restart of the 2100."
::= {net 4}

netAutoTftp OBJECT-TYPE
        SYNTAX INTEGER (0..1)
        ACCESS read-write
        STATUS mandatory
        DESCRIPTION "Enable or disable automatic tftp downloads on power up."
::= {net 5}

netTftpSession OBJECT-TYPE
        SYNTAX INTEGER (0..1)
        ACCESS read-write
        STATUS mandatory
        DESCRIPTION "Start or stop a tftp session."
::= {net 6}

netAutoDhcp OBJECT-TYPE
        SYNTAX INTEGER (0..1)
        ACCESS read-write
        STATUS mandatory
        DESCRIPTION "Enable or disable automatic bootp on power up."
::= {net 7}
```

```
netDhcpSession OBJECT-TYPE
        SYNTAX INTEGER (0..1)
        ACCESS read-write
        STATUS mandatory
        DESCRIPTION "Start or stop a bootp session."
::= {net 8}

etcBootVersion OBJECT-TYPE
        SYNTAX DisplayString (SIZE(1..40))
        ACCESS read-only
        STATUS mandatory
        DESCRIPTION "Version and creation date and timestamp for the bootstrap firmware."
::= {etc 1}

etcOpVersion OBJECT-TYPE
        SYNTAX DisplayString (SIZE(1..40))
        ACCESS read-only
        STATUS mandatory
        DESCRIPTION "Version and creation date and timestamp for the runtime firmware."
::= {etc 2}

etcSerialNbr OBJECT-TYPE
        SYNTAX DisplayString (SIZE(1..40))
        ACCESS read-only
        STATUS mandatory
        DESCRIPTION "Unique serial number burned into each unit."
::= {etc 3}

etcImageLoc OBJECT-TYPE
        SYNTAX INTEGER
        ACCESS read-write
        STATUS mandatory
        DESCRIPTION "Start location for the image download."
::= {etc 4}

etcInfo OBJECT-TYPE
        SYNTAX INTEGER (0..65535)
        ACCESS read-write
        STATUS mandatory
        DESCRIPTION "Get or set the value of the info bit on the 2100.  Not used in standard
        product at this time."
::= {etc 5}
```

```
        etcSysplexOnOff OBJECT-TYPE
                SYNTAX INTEGER (0..1)
                ACCESS read-write
                STATUS mandatory
                DESCRIPTION "Enable or disable the sysplex first protocol output on the 2100 serial
                port."
        ::= {etc 6}

END
```

CHAPTER SIX

TROUBLESHOOTING AND FAQ

6.0 GENERAL

This guide is provided to help users to find commonly asked questions during setup and external equipment failures. The TymServe is a line-replaceable-unit (LRU) and most users will replace the entire unit with a spare unit (or return the unit to the factory for repairs) should a hard failure occur.

6.1 TROUBLESHOOTING

Tracking LED does not light up

During the initial warm up cycle it may take from fifteen minutes to an hour to track the satellites signals and turn the Tracking LED on. An unlit Tracking LED on the front panel indicates that the timing source is either not present or is unusable. Make sure that the correct timing source e.g. Mode '6' for GPS or Mode '1' for Freerun is selected in the 'Timing' directory. The Mode '6' requires the GPS signal be present at the P3 input of the TymServe. If it still does not light up, check location of antenna and the cable connections. Make sure the type and length of the cable is as specified in the Figure 2-1.

Locked LED does not light up

Once the four satellite signals with sufficient strength, typically above 6 db, are received by the TymServe, its Tracking light turns on. After this the internal clock stabilizes and the Locked LED turns on. It may take up to an hour to get the Locked LED on. If the Tracking LED is on but the Locked LED is off then its internal crystal may not be stable, please call the factory for more information.

TymServe does not respond to Ping command.

Establish a serial or TELNET connection and verify that the IP, subnet mask and default gateway addresses are entered correctly in the 'Network' directory of the TymServe. Check the Ethernet 10baseT cable connections between RJ45 connector and the hub or network. If TymServe is directly connected to the computer verify that 10baseT cable is connected through the crossed over connector or through the hub.

TymServe does not respond to NTP Queries.

Verify that the TymServe can be pinged as explained above. If TymServe can be pinged, but it doesn't respond to NTP queries, then verify that the NTP software on your computers is setup properly, and also verify that client has the correct IP address of the TymServe.

Can not make TELNET Connection

Only one TELNET connection is allowed at a time. Make sure that the TymServe can be pinged and then try again. If you still are unable to make a TELNET connection, it is possible that the previous TELNET session is not disconnected. To disconnect the previous TELNET session, access TymServe through the Serial B port and issue the 'Stop' command through the 'network', 'telnet' directory or wait until the session is timed out after an hour. The other alternative is to power cycle the unit.

Why does the position parameters change when checked at different times?

Due to Selective Availability (SA) of the satellite signals i.e. the intentional error introduced by the US Government for security reasons, the signals received by the TymServe may calculate different location reading at different times. However, the expected position parameters range between plus minus 1000m.

Can not establish Serial Connection with the TymServe?

Make sure that the connection is made with straight through serial cable with Serial Port B, not Serial Port A. Check that the configuration setting are set to VT 100 ASCII terminal using 9600, 8,N, 1 as explained in Configuration Methods, Section 2.3.

During Firmware Upgrade the TFTP session fails.

TFTP is UDP based connectionless file transfer protocol. If TFTP downloading fails it could be due to one of the following reasons:

First, TFTP server configuration is not correct. If this is the case the spinning bar that shows the progress of the download process will stop after few rotations. This means that it could not receive data from the TFTP server. Check to make sure the 'firmware file name' and 'host IP address' is entered correctly in the TymServe. Also check if the firmware file is placed in the right directory of the TFTP server.

Second, the traffic on the network is interfering with the downloading process. In this case the spinning bar will usually spin for quite some time and then stop before finishing the download process. Excessive traffic could cause packet collision and subsequently fail the download process. TymServe and host machine connected to the same network hub will greatly reduce the chances of traffic collision. Sometimes it takes three to four tries to complete the Firmware upgrade process.

6.2 FREQUENTLY ASKED QUESTION

How could we obtain NTP client software to work with TymServe?
NTP client software information and configuration details are available from the following Internet sites:

http://www.eecis.udel.edu/~ntp/
http://www.microsoft.com/ntserver/tools/iclient.htm
http://cs3.ecok.edu:457/NetAdminG/BOOKCHAPTER-netadminN.about.html

Client software and/or configuration information for the UNIX, Windows NT and Novell operating system can be downloaded from the above sites. SNTP client software is included with the hardware.

What are the main differences between SNTP and NTP clients?
SNTP is a Simple Network Time Protocol based on RFC 1361 that gets its time from the specified time servers for the machine on which it is installed. This protocol can not be configured to obtain time from alternate time server if the primary server is down. This could be called as a short version of NTP client software.

NTP is Network Time Protocol is based on RFC 1305 and 1119 which can be configured to obtain and distribute the time on the network. It has a built in algorithm to calculate the time accurately up to 1-10 milliseconds. This can be configured to obtain time from an alternate source in case the original time server fails or gets out of synchronization.

How does the TymServe handle Leap Second?
Today's clocks keep pace with each other to within two or three millionth of a second over a year's time. The earth, on the other hand, might randomly accumulate nearly a full second's error during a year. This error is added (or deleted) as a leap second from the UTC time on the last day of the June or December in any year in which the leap second is needed to keep in step with the rotation of the earth.

The GPS satellites will send notice of an upcoming leap second about 2 months in advance. The TymServe will receive this notice and as per NTP specifications will start advising clients 24 hours in advance. At the leap second event, the TymServe will add or delete the leap second from the transmitted time accordingly.

The TymServe will do the same to IEEE1344 IRIG-B signal. However, in the event of leap second, when the time source is regular IRIG-B, 1PPS or Freerun (including using ACTS), user must pre-program the leap second event with command 'leap' so that TymServe can be notified and therefore can maintain the time correctly.

Is TymServe Year 2000 compliant?
Yes. The year information in TymServe is handled in four digits and it is Year 2000 compliant. TymServe will correctly transition from the end of 1999 to year 2000.

How does TymServe handle GPS Rollover event in August 22, 1999?
TymServe is designed to correctly transition through the GPS rollover event that will occur on midnight (12:00 AM) August 22, 1999. During this event the GPS week 1023 will transition to zero and start the count over.

What level of signal strengths are required by the TymServe receiver to start Tracking?
TymServe requires four satellite signals with strengths greater than 6db to turn the Tracking LED on. After the Tracking LED is on TymServe requires only one satellite signal to maintain its time. If it looses the fourth signal the TymServe will automatically transfer to Free-run mode and will keep on providing time.

What does 'doing position fixes error code: 1' means?
This indicates that the unit does not have DC back up power supply. Our current design does not have DC back up power supply and this message may be ignored.

How do I check versions of the Firmware in TymServe?
There are two versions that need to be checked to verify that the unit has the latest firmware installed. One, enter `<version>` through the Serial or TELNET connection. This gives Firmware version of the TymServe. As of writing of this manual the latest released Firmware version is 2.2. Second, to check GPS receiver version enter `<gpsversion>` in the 'timing' and then to the 'gps' directory through the Serial or TELNET connection. This gives the firmware version of the GPS receiver. As of writing of this manual the latest GPS version is 5.1. Please check factory for the latest versions of the factory.

How many maximum numbers of computers can be hooked to the TymServe?
TymServe acts as a stand-alone time server. Average time to process the NTP request is approximately one millisecond. Therefore, TymServe can handle approximately 1000 requests per second. However, client computers running as Stratum 2 computers access TymServe in an interval of 64 to 65536 seconds as the time progresses. The number of computers to be connected to the TymServe should be determined by the customer, based on the capability of the network and the acceptable level of load on the network.

6.3 APPLICATION NOTES

Problem: Modified files are being 'lost' by the file server in a distributed computing environment.

File servers will typically compare a submitted file time stamp to that of the same named file resident on the server. Only if the submitted file is dated later will it be accepted. It is not uncommon in networks with poor synchronization, that a user may modify a file, submit it to the file server (IE. it may be part of a software 'build' process), but have it rejected (without the modifier being aware of the fact). All this due to the user's workstation having an incorrectly set clock and resulting file time stamp.

> **Solution: Setup the network with a hierarchical time distribution system based on NTP.**
>
> TymServe receives time from the Global Positioning System, GPS, satellite constellation and distributes time to workstations at a lower hierarchy using the Network Time Protocol, NTP. This is a public domain protocol that can be installed on user workstations, which then synchronize to the Stratum 1 TymServer to within 1 to 5 milliseconds. In such a well-synchronized system, files will be accurately time stamped and users can be sure that modified files will be correctly accepted and installed on the server.

Problem: Some users are denied access to the network by network security algorithms.

One common threat to network security is the so-called 'man in the middle' attack. A packet of series of packets are intercepted, perhaps modified, and then replayed by the attacker to simulate a 'good guy'. A simple protection to this attack is to provide packet filtering algorithms whereby a packet is only accept if it is very current (with respect to a secret time stamp). For instance, a filter may be established whereby packet older than 1 second is rejected. Of course, if a user workstation's clock is off by more than this filter time window, the user will be denied access.

> **Solution: See above.**

Problem: Our local Stratum 2 time server (a salvaged workstation) running in the computer vault 'hung' and before we realized the problem, we had some 'flammers' from high sources complaining of denial of service.

Install a standalone, dedicated Stratum 1 server, the TymServe in your computer vault. Many processes (we have just mentioned file transfers and security algorithms) are dependent on accurate workstation to workstation; and workstation to server synchronization. While it is possible to synchronize to remote time servers, more users are now finding this actually results in a loss of control of this important resource. One debacle where a supposedly 'free' workstation gets 'hung' is usually enough to encourage savvy IT managers to installed a dedicated TymServer. Public domain servers are fine for redundant backup, but for mission critical local area networks the primary source of time should be a TymServe.

APPENDIX A

SPECIFICATIONS AND DATA SHEET

Specifications and Data Sheet

Item	Specific	Description
Network Interface	Ethernet over 10BaseT	IEEE 802.3 specifications
	Connection	10BaseT, Twisted Pair RJ45
	Time Protocols	NTP Version 2 (RFC - 1119) NTP Version 3 (RFC - 1305) SNTP (RFC 1361) Time Protocol (RFC - 868) Daytime Protocol (RFC - 867)
	Supported Protocols	DHCP, BOOTP and TFTP MD5 Authentication (NTP) Telnet HTTP/HTML Status Page SNMP v1 w/Custom MIB II Extension
Serial Connection	Port A	DTE style RS-232 port with DB9 (male) connector for Sysplex timer. Default 9600 baud, 8 data bits, 1 start/stop bit, no parity
	Port B	DCE style RS-232 port with DB9 (female) connector for status and configuration. Default 9600 baud, 8 data bits, 1 start/stop bit, no parity (8-N-1)
Timing Accuracy	Network	1-10 milliseconds, typical
	Internal Clock	<10 microseconds, relative to UTC, GPS tracking
GPS Receiver	Channels	Six.
	Frequency	1.575 GHz, C/A Code.
	GPS	<2 microseconds, relative to UTC
Chassis	AC Power In	100-130Vac / 220-240Vac, 48 – 440Hz, 20W
	Size and Weight	1.75”H-17.00”W-9.5”D 1U Height Rack Mount, <10 lbs. (4.5Kg) 19” width with Rack Mount ears
Operating and Storage Environments	Temperature	0^{o} C to 55^{o} C *Operating* -50^{o} C to 85^{o} C *Non-Operating*
	Relative Humidity	10% - 80% *Operating* (non-condensing). 5% - 95% *Non-Operating* (non-condensing)
Mean Time Between Failure (MTBF)	Average Time	56,000 Hours

Comparison of Datum TymServe Model 2100 and 2100L

Parameter	TymServe 2100	TymServe 2100L
Inputs	GPS – Antenna / Pre-amp IRIG A, IRIG B, NASA 36 (Modulated 2:1 to 6:1) 500 mV-10Vp-p Differential TTL DC Level Shift (DCLS) 1PPS IRIG B Modulated 3:1, 3V p-p, 75Ω	GPS - Antenna
Outputs	IRIG B, Differential TTL, DCLS, 50Ω 1PPS, TTL Rising edge on-time, 50Ω 10MHz, 50Ω square wave, Sine wave option w/OCXO or Rubidium	
Input/ Output	Ethernet 10baseT connector Ethernet AUI connector Serial Port A, DTE, Sysplex Timer and Modem Serial Port B, DCE Configuration/Status	Ethernet 10baseT connector Serial Port A, DTE, Sysplex Timer Serial Port B, DCE Configuration/Status
Standard Features	Front Panel Key Pad Front Panel Display Front Panel LEDs (Locked, Tracking, Power)	 Front Panel LEDs (Locked, Tracking, Power)
Oscillators	VCXO (Standard) OCXO (Optional) Rubidium (Optional)	VCXO (Standard)
Power Supply	95 – 265VAC, 47-63 Hz -48 Volts (Optional)	95 – 265VAC, 47-63 Hz
Network Features	SNMP & MIB II NTP v2 and v3 TCP/IP, MD5 BOOTP, DHCP and TFTP TELNET HTTP to HTML Status Page NIST, ACTS and USNO	SNMP & MIB II NTP v2 and v3 TCP/IP, MD5 BOOTP, DHCP and TFTP TELNET HTTP to HTML Status Page

APPENDIX B

INPUT/OUTPUT CONNECTORS

B.0 GENERAL

The TymServe 2100L input/output connectors located on the rear panel of the unit are shown in Figure B-1. These connectors are Ethernet 10BaseT connector, GPS antenna connector, two RS-232 serial ports, and AC power entry.

B.1 PIN DESCRIPTIONS

P2 - Ethernet RJ45

Description: 8 Pin Phone Jack, Manufacturer: AMP, Part #: 555153-1

Pin Number	Description
Pin 1	TX (+).
Pin 2	TX (-).
Pin 3	RX (+).
Pin 4	N/C.
Pin 5	N/C.
Pin 6	RX (-).
Pin 7	N/C.
Pin 8	N/C.

P5 - Serial A (Data Terminal Port)

Description: 9 Pin "D" Plug, Manufacturer: AMP, Part #: 869436-1

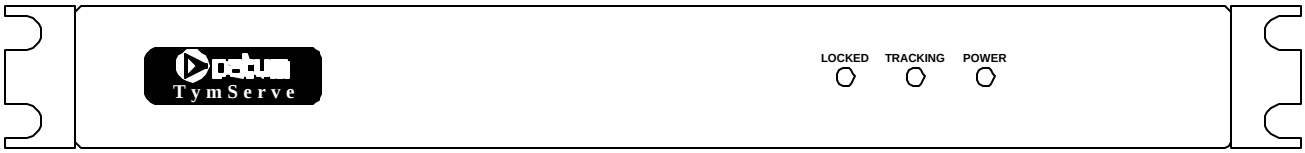
Pin Number	Description
Pin 1	RS-232 Data Carrier Detect (in).
Pin 2	RS-232 Receive Data (in).
Pin 3	RS-232 Transmit Data (out).
Pin 4	RS-232 Data Terminal Ready (out).
Pin 5	Ground.
Pin 6	RS-232 Data Set Ready (in).
Pin 7	RS-232 Request to Send (out).
Pin 8	RS-232 Clear to Send (in).
Pin 9	RS-232 Ring Indicator (in).

P6 – Serial Port B (Setup Port)

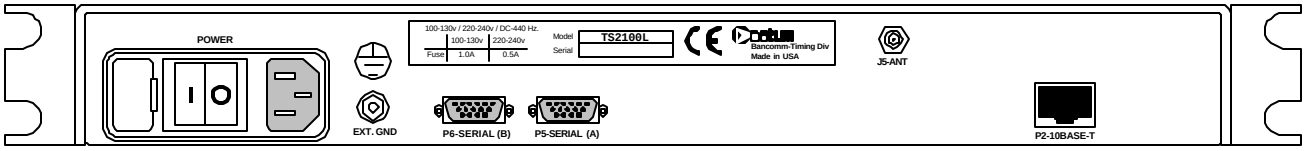
Description: 9 Pin "D" Plug, Manufacturer: AMP, Part# 869436-1

Pin Number	Description
Pin 1	N/C.
Pin 2	RS-232 Transmit Data (out).
Pin 3	RS-232 Receive Data (in).
Pin 4	N/C.
Pin 5	Ground.
Pin 6	N/C.
Pin 7	RS-232 Clear to Send (in).
Pin 8	RS-232 Request to Send (out).
Pin 9	N/C.

B.2 FRONT-REAR PANEL DIAGRAM



Front View



Rear View

Figure B-1

APPENDIX C

FIRMWARE UPGRADE

C.0 GENERAL

The program code that operates TymServe is stored in the flash or non-volatile RAM. The flexibility of the TymServe allows the program code to be updated through both local and remote connections. The program updates are obtained from the factory in an ASCII file format that can be read by the TymServe and converted to program code. The program update can be downloaded to the TymServe through any of the following means:

- Local RS232: Send the text file through connection to TymServe.
- Remote TFTP protocol to load the new image from the TFTP server through Ethernet connection to TymServe.

A 'boot' section is stored separately from the program code to allow for recovery in the event of a catastrophic error during the download of a program update. If such an event occurs, the affected TymServe can be power cycled and will recover to a point sufficient to allow a reload of the program update. The latest upgrades of firmware are available for download from the Datum Internet site at ftp.datum.com. A release notes RELEASE.TXT available in the same location provides a list of what are new and what are fixed in the latest release of firmware.

Once the program update has been successfully downloaded, it will be stored in non-volatile RAM. Next, the TymServe can be restarted, either locally or remotely, and will commence using the new program code.

C.1 TYPICAL FIRMWARE UPGRADE PROCEDURE

TymServe 2100 Firmware Update Instruction: UPGRADE.TXT
Image filename: `ts21lc.hex`
Image version: the version specified in RELEASE.TXT

This accompanies the release file containing the Motorola S-Record for the new firmware image. There are two methods of loading the new firmware image onto your TymServe. The referable method is to use TFTP (Trivial File Transfer Protocol). A second, slower method is available which allow the loading of new firmware through the serial port used to connect to the
TymServe command shell. Use of either of these methods requires that the new firmware image be downloaded to a local machine on your network. Use FTP to download the new firmware image (name of the current release shown above). The file is available from Datum's FTP server:

ftp.datum.com in the /pub/NetProducts/TymServe2100 directory

User may use FTP as 'anonymous' user or Web Browser to download the '`ts21lc.hex`' file.

A caveat is necessary here. Although this file has an extension hex it is NOT a binary file. It is ASCII and therefore, translation MUST BE ENABLED when flipping the file back and forth across UNIX and DOS machines. To be safe, simply use the command 'type ascii' before transferring the file to any machine by FTP. If you use a GUI FTP client, you must find and disable any 'auto-sensing' file filter. Once you have downloaded the new firmware image, refer to the notes below for examples of typical installation.

Some Generic Notes

1. After the firmware has been updated, the front of the TymServe will display '*** err *** Flash Env' in the front panel (if equipped) or in the trace buffer. This is normal after new firmware has been uploaded and indicates that flash environment size has changed. This has the effect of resetting all user programmable values (i.e. ip) to their default settings.

2. User will have to reenter the settings to the TymServe, like IP, subnet, gateway and host.

3. Some units will have unreadable displays, use the 4 or 9 key to change the contrast to a readable level if TymServe is equipped with front panel display and key pad.

4. If download fails through TFTP or Serial, DO NOT RESET YOUR TymServe! Once you have started the download process, the current image is erased in flash memory. However, a copy of the current image is loaded in RAM and running, which guarantees a normal operation unless it is restarted. The intrinsic command 'trace' may display data which will help you determine the source of the error. Stop the current transfer (root net tftp stop) and check the configuration data required for TFTP (ip, mask, route, host, file) and try the TFTP download again. If you still have difficulty, try using a TFTP client to download the new firmware to a test platform (Unix or windows) to determine whether the error is on the TFTP server side or the TymServe side. If you continue to experience difficulties, you could try using the serial download paradigm detailed below. In the event that a restart is required after failure in downloading the new firmware, not all is lost. A permanent 'boot' image is programmed into the TymServe. While this version of firmware does not have all the capabilities of the standard code (no SNMP, no HTTP, etc), it does have all necessary components to allow you to download a new firmware.

TFTP Download

***Note*:** If you do not have a TFTP server available, some shareware TFTP server applications are available. For instance, 16 and 32 bit versions for Microsoft(tm) Windows 3.x/95/NT are available from http://www.walusoft.co.uk for a nominal fee. If you have a Unix platform available, likely it includes a TFTP server. Contact your network administrator for details. If any case, once you have a TFTP server available, put the new firmware image in a location accessible to the TFTP server daemon. This may be accomplished with TFTP upload, FTP or even a Unix cp depending on your setup. On the TymServe, use the shell to change to the net directory (root net).

If you would like the TymServe 2100 to automatically restart after TFTP downloads, confirm that the auto mode is enabled in the 'network' submenu. (root net auto on)

Execute following commands:
root net file <filename>
root net host <your TFTP server IP in dotted quad notation>
root net TFTP start

A spinning progressing indicator should be displayed while the image is being downloaded. If the indicator spins a few times and then stops, or spins for longer than 10 minutes without stopping, an error has occurred.

Serial Download

Instructions for downloading a firmware upgrade to the TymServe using the RS232 serial connection.

***Note*:** If you are using another OS (i.e. Unix), translate the information and settings provided here into commands suitable for your system. The most likely source of error is the proper CR-LF sequences at the end of each S-Record.

Win 3.x/ NT 3.5x

1. Put the file on your host computer.

2. Connect host computer to the target TymServe using a standard RS232 cable.

3. Start Windows.

4. Go to Group Main, Control Panel, Ports and check the settings on COM1. You want to use 9600 baud, 8 data bits, 1 stop bit and no parity bits. You also want to use XON/XOFF flow control. Then close Ports and Control Panel.

5. Go to Group Accessories and start Terminal. Go to settings and choose terminal type VT100 (ANSI). Go to Settings,modem settings and make sure you are using the above 9600,8,N,1,XON/XOFF. Go to Settings,Preferences and check the box to translate outbound CR to CR-LF.

6. In the main window, hit <Enter> a few times and see if you get a ? prompt. Once you do, hit <Enter> again and see if you get a ? prompt every time you hit <Enter> or every other time. If it is not every time, check the translate outbound setting again.

7. type root <Enter> to reach the base of the command tree with
   timing/
   network/
   serial/
   etc.

8. Type serial <Enter> to go to the serial directory.

9. Type ? <Enter> and look for a command called download. This is what you are going to use.

10. Type download <Enter> and you should see a spinning wheel show up on the display. This means the TymServe is waiting for data.

11. Go to Transfers, Send text file, find the firmware file and start sending.

12. At the bottom of the terminal display, you should see a progress bar and a message saying Sending <filename>.

13. You should still see the spinning bar which indicates that the listen thread is either waiting or receiving data. If you see the display continuously displaying messages of 'invalid command'' then the translation mode is wrong.

14. This download may at least 10 minutes.

15. After the progress bar reaches 100% and disappears, the system will reboot.

16. When the system comes back up, type ver <Enter> to display the version number. The info will show the new version information.

Win 95/NT 4.0

1. Put the file on your host computer.

2. Connect host computer to the target TymServe using a standard RS232 cable.

3. Start Windows.

4. Start HyperTerminal (in Accessories menu).

5. Create new connection specifying
   Direct to COMx (x = serial port)
   9600,8,N,1
   XOFF/XON flow control
   ASCII setup (do NOT append lf to outbound text)

6. In the main window, hit <Enter> a few times and see if you get a ? prompt.

7. type root <Enter><Enter> to reach the base of the command tree with
   timing/
   network/
   serial/
   etc.

8. Type serial <Enter> to go to the serial directory.

9. Type ? <Enter> and look for a command called download. This is what you are going to use.

10. Type download <Enter> and you should see a spinning wheel show up on the display. This means the TymServe is waiting for data.

11. Go to Transfers, Send text file, find the firmware file and start sending.

12. You should still see the spinning bar which indicates that the listen thread is either waiting or receiving data. If you see the display continuously displaying messages of 'Invalid Command' then the translation mode is wrong.

13. This download may take at least 10 minutes.

14. After the download completes, the system will reboot.

15. When the system comes back up, type ver <Enter> to display the version number. The info will show the new version information.

APPENDIX D

GLOSSARY

D.0 GLOSSARY

This sections defines some of the commonly used terms related to TymServe.

Leap Year
If a year is divisible by 4 or 400 add a day to 365 days of that year. However, if the year is divisible by 100 but not by 400 then it is not a leap year. TymServe models 2000, 2100 and 2100L meet these requirements.

Leap Second
Today's scientists and engineers have perfected clocks based on a resonance in cesium atoms to an accuracy of better than one part in 10 trillion. These clocks keep pace with each other to within two or three millionth of a second over a year's time. The earth, on the other hand, might randomly accumulate nearly a full second's error during a year. This error is added (or deleted) as a leap second from the UTC time on the last day of the June or December in any year in which the leap second is needed to keep in step with the rotation of the earth.

UTC
Coordinated Universal Time. The UTC is an international, highly accurate and stable uniform atomic time system became effective January 1, 1972. Since the earth continuously changes its rate of rotation, leap seconds needs to be added or deleted to keep UTC approximately with the sun.

GPS
Global Positioning System is a constellation of 24 satellites in orbits inclined 56 degrees to the equator and orbiting the earth twice a day. This means that the further north you are in the Northern Hemisphere, more probable it is that satellites will be passing to the south of you. Conversely, Southern Hemisphere installation favors the view of the northern sky. These satellites transmit signals that are used by the GPS receivers to determine the position and time with great accuracy. The time is expressed as the number of weeks since midnight January 6, 1980 (GPS Week) plus number of seconds in the week. These two values are transmitted as binary integers from the satellites and converted into conventional date or day by the GPS receiver.

The orbits of these satellites and the offset (relative to international standard time, UTC) of their on board Cesium Atomic Clocks is precisely tracked by the US Air Force control network. Position and time correction information is uplinked from the ground control stations and maintained in the satellites in what is termed 'Ephemeris Tables'. Each satellite transmission reports the satellite's current position, GPS time, and the offset of the satellite's clock relative to international standard time, UTC. The GPS receiver in your TymServe tracks these satellites as they pass overhead and determines the range of the satellite from the antenna. There are four unknowns: x, y, z, and t (or loosely latitude, longitude, altitude, and time). Knowing the range from one satellite places you on a sphere; two satellites the intersection of two spheres, a circle; three satellites two points; and, finally, four satellites yield a complete four variable solution. However, once x, y, and z are known, either because the receiver has tracked at least four satellites and positioned itself, or the user has entered a known position into the TymServe 2100L; only one satellite is required to solve for time. This is important because placement of the antenna in high rise urban areas frequently results in a restricted view of the sky, but for a timing solution this is not as much of a liability as it would be if one were trying to derive a full navigational solution.

The GPS transmission is a 1.5 GHZ (L_1 Band) spread spectrum signal. Being spread spectrum it is relatively immune to interference, however not perfectly so. High energy sources, especially those with significant in band energy, can swamp the receiver RF processing circuitry. In addition, it has proved difficult to operate GPS at power substations or in close proximity to high voltage 60 Hz sources. Datum offers an optional high gain antenna that is useful in these heavy interference situations, however, again it is recommended that the Quick Setup be pursued with the antenna away from radiating sources before dealing with such situations.

NTP

Network Time Protocol is an protocol that provides a reliable way of transmitting and receiving the time over the TCP/IP networks. The NTP is useful to synchronize the internal clock of the computers to a common time source. The common time source could be GPS satellite network, dial-up connections, radio receivers, etc.

TELNET

TELNET is a terminal emulation application protocol that enables a user to log in remotely across a TCP/IP network to any host supporting this protocol. The keystrokes that the user enters at the computer or terminal are delivered to the remote machine, and the remote computer response is delivered back to the user’s computer or terminal.

Year 2000 Compliance

The TymServe maintains the year value as a four-digit number and will correctly transition from the end of year 1999 to beginning of year 2000 at the turn of the century. If the TymServe is powered down and powered back up in the year 2000 or later, it will correctly determine the year and time from the GPS satellite signal. It will also recognize the year 2000 as a leap year.

GPS Rollover Compliance

The GPS signal contains 10-bit binary integers for the GPS week number. This number could count only up-to week 1,023 that ends at midnight (12:00 AM) on August 22, 1999. At that moment, GPS week will go to zero and start the count over. TymServe will correctly transition through the GPS week rollover event that will occur on August 22, 1999.

NTP Client Software

NTP client software information and configuration details are at the following Internet sites:

http://www.eecis.udel.edu/~ntp/
http://www.microsoft.com/ntserver/tools/iclient.htm
http://cs3.ecok.edu:457/NetAdminG/BOOKCHAPTER-netadminN.about.html

Client software and/or configuration information for the UNIX, Windows NT and Novell operating system can be downloaded from the above sites.