



PNT-440

Resilient Multi-Source Clock Module

User Guide

R001

© Copyright 2024 VIAVI Solutions Inc. All rights reserved. VIAVI and the VIAVI logo are trademarks of VIAVI Solutions Inc. ("VIAVI"). All other trademarks and registered trademarks are the property of their respective owners. No part of this guide may be reproduced or transmitted, electronically or otherwise, without written permission of the publisher.

Reproduction and distribution of this guide is authorized for US Government purposes only.

VIAVI is a trademark of VIAVI Solutions in the United States and other countries. Microsoft, Windows, Windows CE, Windows NT, MS-DOS, Excel, Word and Microsoft Internet Explorer are either trademarks or registered trademarks of Microsoft Corporation in the United States and/or other countries. All trademarks and registered trademarks are the property of their respective companies.

Patented as described at www.viavisolutions.com/patents.

Every effort was made to ensure that the information in this manual was accurate at the time of printing. However, information is subject to change without notice, and VIAVI reserves the right to provide an addendum to this manual with information not available at the time that this manual was created.

Specifications, terms, and conditions are subject to change without notice. The provision of hardware, services, and/or software are subject to the VIAVI standard terms and conditions, available at www.viavisolutions.com/terms.

About this User Guide

This prefix explains how to use this User Guide and includes the following topics:

- “Purpose and scope” on page iv
- “Assumptions” on page iv
- “Related Information” on page iv
- “Conventions” on page iv
- “Safety and compliance information” on page vi
- “Technical assistance” on page viii

Purpose and scope

This manual is intended to help you use the capabilities of the PNT-440 Resilient Multi-Source Clock Module.

This manual includes task-based instructions that describe how to configure, use, and troubleshoot the test capabilities available on your instrument assuming it is configured and optioned to support the capabilities.

Assumptions

This manual is intended for novice, intermediate, and experienced users who want to use their instrument effectively and efficiently. We are assuming that you have basic computer experience and are familiar with basic telecommunication concepts, terminology, and safety.

Related Information

This manual is application-oriented and contains information about using these instruments to test service carried on each of the listed networks. It includes an overview of testing features, instructions for using the instruments to generate and transmit traffic over a circuit, and detailed test result descriptions. This manual also provides contact information for VIAVI's Technical Assistance Center (TAC).

Conventions

This manual uses conventions and symbols, as described in the following tables.

Table 1 Text formatting and other typographical conventions

Item(s)	Example(s)
Buttons, keys, or switches that you press or flip on a physical device.	Press the On button. – Press the Enter key. – Flip the Power switch to the on position.
Buttons, links, menus, menu options, tabs, or fields on a PC-based or Web-based user interface that you click, select, or type information into.	Click Start – Click File > Properties . – Click the Properties tab. – Type the name of the probe in the Probe Name field.
Directory names, file names, and code and output messages that appear in a command line interface or in some graphical user interfaces (GUIs).	<code>\$NANGT_DATA_DIR/results</code> (directory) – <code>test_products/users/defaultUser.xml</code> (file name) – <code>All results okay.</code> (output message)

Table 1 Text formatting and other typographical conventions (Continued)

Item(s)	Example(s)
Text you must type exactly as shown into a command line interface, text file, or a GUI text field.	Restart the applications on the server using the following command: <code>\$BASEDIR/startup/npui_init restart</code> Type: <code>a:\set.exe</code> in the dialog box.
References to guides, books, and other publications appear in <i>this typeface</i> .	Refer to <i>Newton's Telecom Dictionary</i> .
Command line option separators.	<code>platform [a b e]</code>
Optional arguments (text variables in code).	<code>login [platform name]</code>
Required arguments (text variables in code).	<code><password></code>

Table 2 Symbol conventions

	This symbol indicates a note that includes important supplemental information or tips related to the main text.
	This symbol represents a general hazard. It may be associated with either a DANGER, WARNING, CAUTION, or ALERT message. See Table 3 for more information.
	This symbol represents an alert. It indicates that there is an action that must be performed in order to protect equipment and data or to avoid software damage and service interruption.
	This symbol represents hazardous voltages. It may be associated with either a DANGER, WARNING, CAUTION, or ALERT message. See Table 3 for more information.
	This symbol represents a risk of explosion. It may be associated with either a DANGER, WARNING, CAUTION or ALERT message. See Table 3 for more information.
	This symbol represents a risk of a hot surface. It may be associated with either a DANGER, WARNING, CAUTION, or ALERT message. See Table 3 for more information.
	This symbol represents a risk associated with fiber optic lasers. It may be associated with either a DANGER, WARNING, CAUTION or ALERT message. See Table 3 for more information.
	This symbol, located on the equipment, battery, or the packaging indicates that the equipment or battery must not be disposed of in a land-fill site or as municipal waste, and should be disposed of according to your national regulations.

Table 3 Safety definitions

Term	Definition
DANGER	Indicates a potentially hazardous situation that, if not avoided, <i>will</i> result in death or serious injury. It may be associated with either a general hazard, high voltage, or other symbol. See Table 2 for more information.
WARNING	Indicates a potentially hazardous situation that, if not avoided, <i>could</i> result in death or serious injury. It may be associated with either a general hazard, high voltage, or other symbol. See Table 2 for more information.
CAUTION	Indicates a potentially hazardous situation that, if not avoided, could result in minor or moderate injury and/or damage to equipment. It may be associated with either a general hazard, high voltage, or risk of explosion symbol. See Table 2 for more information. When applied to software actions, indicates a situation that, if not avoided, could result in loss of data or a disruption of software operation.
ALERT	Indicates that there is an action that must be performed in order to protect equipment and data or to avoid software damage and service interruption.

Safety and compliance information

The following sections describe the safety and compliance information for the PNT-440.

California Proposition 65

California Proposition 65, officially known as the Safe Drinking Water and Toxic Enforcement Act of 1986, was enacted in November 1986 with the aim of protecting individuals in the state of California and the state's drinking water and environment from excessive exposure to chemicals known to the state to cause cancer, birth defects or other reproductive harm.

For the VIAVI position statement on the use of Proposition 65 chemicals in VIAVI products, see the Hazardous Substance Control section of the VIAVI Policies & Standards web page.

Federal Communications Commission (FCC)

The equipment was tested and found to comply with the limits for a Class A digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide

reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference, in which case you will be required to correct the interference at your own expense.

The authority to operate this equipment is conditioned by the requirements that no modifications be made to the equipment unless the changes or modifications are expressly approved by VIAVI.

Bands of Operation

The equipment operates on the following bands.

Table 4 Bands of operation

Receiver	Frequency Range
GNSS	1164 MHz to 1300 MHz
	1559 MHz to 1610 MHz
STL	1610 MHz to 1626.5 MHz

Product Environmental Compliance

VIAVI is committed to compliance with all applicable laws and regulations controlling the use of hazardous substances in its products, as well as the disposal of equipment (including batteries) and waste packaging. For details, see the VIAVI Policies & Standards web page or contact the VIAVI WEEE Program Management team at Global.WEEE@ViaviSolutions.com.

EU REACH

Article 33 of EU REACH regulation (EC) No 1907/2006 requires product suppliers to provide information when a substance included in the list of Substances of Very High Concern (SVHC) is present in a product above a certain threshold.

For information about the presence of REACH SVHC in VIAVI products, see the Hazardous Substance Control section of the VIAVI Policies & Standards web page.

Additional standards compliance

The equipment meets the following standards and requirements:

- Installation Category (Over Voltage Category) II under IEC 60664-1

- Pollution Degree 2 Category under IEC 62368-1 Safety Requirements for Electrical Equipment for Measurement, Control, and Laboratory Use

Technical assistance

If you require technical assistance, call 1-844-GO-VIAVI. For the latest TAC information, go to <https://support.viavisolutions.com>.



Table of Contents

About this User Guide		iii
	Purpose and scope	iv
	Assumptions	iv
	Related Information	iv
	Conventions	iv
	Safety and compliance information	vi
	California Proposition 65	vi
	Federal Communications Commission (FCC)	vi
	Bands of Operation	vii
	Product Environmental Compliance	vii
	EU REACH	vii
	Additional standards compliance	vii
	Technical assistance	viii
Chapter 1	Introduction	1
	About the PNT-440	2
	Optional accessories	3
	Customer-provided equipment	3
	General safety precautions	3
	Surge Protector	3
	Grounding	4
	Power connections	4
	Environmental conditions	4
	Dimensions	5
	PNT-440 device connections	5
Chapter 2	Installation	7
	Overview	8
	Pre-installation	8
	Powering the unit	8
	Connectors	9
	16-pin Hirose I/O connector	10

USB I/O connector	12
Quick start	12
Chapter 3 GPSCon Utility	15
Description	16
Using GSPCon	16
Setting options	16
Sending manual commands to the receiver	18
Using the mouse in the Graph display	19
Exporting graphics	20
Interpreting the Data	21
Chapter 4 SCPI Commands	23
Introduction	24
General SCPI commands	24
Quick Start commands	24
*IDN?	24
HELP?	25
GPS Subsystem	25
GPS:SATellite	27
GPS:SATellite:TRAcking:COUnT?	28
GPS:SATellite:VISible:COUnT?	28
NMEA Support	28
GPS:GPGGA <int> [0,255]	29
GPS:GGASat <int> [0,255]	30
GPS:GPGLL <int> [0,255]	30
GPS:GPRMC <int> [0,255]	31
GPS:GPGSA <int> [0,255]	31
GPS:GPGSV <int> [0,255]	32
GPS:GXGGA <int> [0,255]	32
GPS:GXRMC <int> [0,255]	32
GPS:POWGPS <int> [0,255]	33
GPS:POWTLV <int> [0,255]	33
GPS:SENDNmea <NMEA-0183 GGA Message String RMC Message String >	33
GPS:SOURce:EXTNMEA <AUTO ON OFF>	33
GPS:SOURce:STATE?	34
GPS:SOURce:STATE:value?	34
GPS:STL:DYNAMICmodel?	34
GPS:STL:FWver:SDPM?	34
GPS:STL:FWver?	34
GPS:STL:GEOLOCation <STATIC DYNAMIC>	34
GPS:STL:GEOLOCation?	35
GPS:STL:GUESSPOSition <BOOTSTRAP TRUEPOS MANUAL>	35
GPS:STL:GUESSPOSition?	35
GPS:STL:SN?	35
GPS:STL:PJLTGGA <int> [0,255]	36
GPS:STL:PJLTPOS <int> [0,255]	36

GPS:STL:PJLTTPPS <int> [0,255]	37
GPS:STL:PJLTRMC <int> [0,255]	37
GPS:STL:PJLTSTAT <int> [0,255]	38
GPS:STL:PJLTSVI <ON OFF>	38
GPS:STL:PJLTVEL <int> [0,255]	39
GPS:STL:PPSFilter:PFOM <int> [1,9]	40
GPS:STL:PPSFilter:TFOM <int> [1,9]	40
GPS:STL:SN?	41
GPS:STL:TRUEPOSition <double> <ll.llllllll,lll.llllllll,h.hh>	41
GPS:STL:SUBScription:ENDdate?	41
GPS:STL:SUBScription:FEATure?	41
GPS:STL:SUBScription:STARTdate?	42
GPS:STL:SUBScription:STATus?	42
GPS:STL:SUBScription:VALID?	42
GPS:XYZSPeed <int> [0,255]	42
GPS:GPZDA <int> [0,255]	43
GPS:PASHR <int> [0,255]	43
GPS:PJLTS <int> [0,255]	44
GPS:DYNAMic:MODE <int> [0,8]	45
GPS:DYNAMic:MODE 8 (Automatic Dynamic Mode)	46
GPS:DYNAMic:MODE?	47
GPS:DYNAMic:STATe?	47
GPS:REFerence:ADELay <float> <s ns> [-32767ns,32767ns]	47
GPS:REFerence:PULse:SAWtooth?	48
GPS:RESET ONCE	48
GPS:TMODE <ON OFF RSTSURV>	48
GPS:SURVey ONCE	49
GPS:SURVey:DURation <sec>	49
GPS:SURVey:VARIANCE <mm^2>	49
GPS:HOLD:POSition <int, int, int>	50
GPS:SURVEY:STATUS?	50
GPS:INITial:DATE <yyyy,mm,dd>	50
GPS:INITial:TIME <hour,min,sec>	51
GPS:RTC:DATE <yyyy,mm,dd>	51
GPS:RTC:TIME <hour,min,sec>	51
GPS:SYSTem:SElect [GPS SBAS QZSS GAL BD ^ GLO]	52
GPS:SYSTem:LBAND [L1 L2 L3 L5 ALL]	52
GPS:SYSTem:LBAND:AGCgain?	53
GPS:SYSTem:LBAND:AGCPnorm?	53
GPS:SYSTem:LBAND:SVID?	53
GPS:SYSTem:LBAND:CNO?	53
GPS:SYSTem:BANDTRACK [L1 L2 L3 L5 ALL]	53
GPS:SYSTem:TIMESOURce <GPS GAL BD AUTO>	54
GPS:SYSTem:TIMESOURce:STATe?	55
GPS:SYSTem:ETA <ON OFF>	55
GPS:SYSTem:ETA?	55
GPS:SYSTem:UPDATERATE <Hz> [1,5,10,20]	55
GPS:SPOOF?	55
GPS:HEIGHT?	55

GPS:HEIGHT:MSL?	56
GPS:HEIGHT:GPS?	56
GPS:POLarity <LOW HIGH>	56
GPS:POSition?	56
GPS:POSition:ECEF?	56
GPS:CNOthres <int> [1,40]	57
GPS:FOV <int> [3,85]	57
GPS:STATus?	57
GPS:STATus:ETA?	57
GPS:STATus:STRing?	57
GPS:PASSthru <GNSS RSRSCPI STL TOD CSAC RSRGPS MCE RSRDEB UG>	58
GPS:PASSthru:TIMEout <ON OFF>	58
GPS:PASSthru:COUNt <int> <s> [1,86400]	58
GPS:TYPE?	59
GPS:TYPE:MODE <AUTO UBLOX SEPTentrio STL>	59
GPS:SOURce:SElect <AUTO STL GNSS>	59
GPS:FILTer:SElect <STL UBX>	59
GPS:FILTer:NMEA <int> [0,255]	59
GPS:GNSS:MODE <ON OFF>	59
GPS:GNSS:RESET ONCE	60
GPS:GNSS:POWer <ON OFF>	60
GPS:GNSS:FWver?	60
GPS:GNSS:TYPE?	60
GPS:STL:RESET <ONCE WARMUP HOLD OFF>	60
GPS:STL:RESET:AUTO <int> [0,604800]	61
GPS:STL:SCPI <ASCII> <command>	61
GPS:STL:ISP ON	61
GPS:STL:BPMProcessed?	61
GPS:STL:BPMReceived?	61
GPS:STL:PFOM?	62
GPS:STL:TFOM?	62
GPS:STL:EXTCLOCK <ON OFF>	62
GPS:STL:AGC <int> [500,2700]	62
GPS:STL:CNO?	63
GPS:STL:CNO:MINTHReshold <float> [1.0,100.0]	63
GPS:STL:DOPpler:MINimum <double> [-36000.000000,0.0]	63
GPS:STL:DOPpler:MAXimum <double> [0.0,36000.000000]	63
GPS:STL:CLOCKmodel <double> [1E-20,1.0]	64
GPS:STL:CLOCKmodel:FAST <double> [1E-20,1.0]	64
GPS:STL:CLOCKmodel:MEDIUM <double> [1E-20,1.0]	65
GPS:STL:SUBScriptioN?	65
GPS:STL:SUBScriptioN:AUTHKEY <string>	65
GPS:STL:SERVo:LOOP <ON OFF>	67
GPS:POWTLV <ms> [200,1000]	67
GPS:PJLTV <int> [0,255]	68
GPS:STL:SERVo:INDOOR <OFF ON>	68
GPS:FWver?	69
GPS:JAMlevel?	69

GPS:JAMI?	69
GPS:SYSTem:LBAND:MODE <ON OFF>	69
GPS:SYSTem:LBAND:MODE?	69
GPS:SYSTem:NOTCHFilter <id>[0-3], <mode>[0-2], <ctFrqkHz>[1100000-1700000], <bwkHz>[30-1600].	69
GPS:SYSTem:UPDATERATE <Hz> [1,5,10,20]	70
GPS:ANTPower <ON OFF>	70
GPS?	70
PTIME Subsystem	70
PTIME:DATE?	70
PTIME:TIME?	70
PTIME:TIME:STRing?	71
PTIME:TINTerval?	71
PTIME:OUTput <ON OFF>	71
PTIME:LEAPsecond?	71
PTIME:LEAPsecond:PENDING?	72
PTIME:LEAPsecond:ACCumulated?	72
PTIME:LEAPsecond:DATE?	72
PTIME:LEAPsecond:DURation?	72
PTIME?	72
SYNChronization Subsystem	73
SYNChronization:HOLDOver:DURation?	74
SYNChronization:HOLDOver:INITiate	74
SYNChronization:HOLDOver:RECOvery:INITiate	74
SYNChronization:HOLDOver:TUNCertainty:PREDicted?	74
SYNChronization:SOURce:MODE <GPS EXTERNAL STL PRIORITY>	75
SYNChronization:SOURce:PRIORITY <STL GPS EXTERNAL>	75
SYNChronization:SOURce:PRIORITY:RESET DEFAULT	76
SYNChronization:SOURce:PRIORITY:TOP <STL GPS EXTERNAL>	76
SYNChronization:SOURce:MODE:EDGE <NEGative POSitive>	76
SYNChronization:SOURce:STATE?	77
SYNChronization:SOURce:STATE:VALUE?	77
SYNChronization:HOLDOver:STATE?	77
SYNChronization:TINTerval?	77
SYNChronization:TINTerval:THReshold <int> [50,2000].	77
SYNChronization:IMMEdiate	78
SYNChronization:FEEstimate?	78
SYNChronization:LOCKed?	78
SYNChronization:OUTput:1PPS:RESET <ON OFF>	78
SYNChronization:OUTput:1PPS:WIDTH <int> <ms us> [100us, 500ms]	78
SYNChronization:HEAlth?	79
SYNChronization:HEAlth:VERBOse?	80
SYNChronization:HEAlth:HISTory?	80
SYNChronization:HEAlth:HISTory:RESet	81
SYNChronization:HEAlth:HISTory:VERBOse?	81
SYNChronization:HEAlth:ALARMDelay:HOLDOver <int> [0,604800]	81
SYSTem:COMMunicate:DLOAD:TIMEout <int>[1,3600]	81
SYNChronization:TINTerval:RAW?	81
SYSTem:FwConfig?	81

SYSTem:GETCFGCurrent?	81
SYSTem:GETCFGDefault?	82
SYSTem:RESETCFGtodefault.	82
SYSTem:ID:MODELname?	82
SYSTem:ID:SN:BOX?	82
SYSTem:ID:SN?	82
SYSTem:ID:UID?	82
SYSTem:ID?	82
SYSTem:STATus:GPSDO?	82
SYNChronization?	82
DIAGnostic Subsystem	83
DIAGnostic:ROSCillator:EFControl:RELative <float> [-100.0, 100.0]	83
DIAGnostic:ROSCillator:EFControl:ABSolute <float> [0.0, 5.0]	83
DIAGnostic:LIFetime:COUNt?	84
DIAGnostic:LIFetime:SECond?	84
DIAGnostic?	84
CSAC Subsystem	84
CSAC:RS232?	84
CSAC:STeer?	85
CSAC:TYPE?	85
CSAC:LOCKed?	85
CSAC:STATus?	85
CSAC:ALarm?	86
CSAC:MODE?	86
CSAC:CONTRast?	87
CSAC:LASer?	87
CSAC:TCXO?	87
CSAC:SIGnal?	87
CSAC:HEATpackage?	87
CSAC:TEMP?	88
CSAC:SN?	88
CSAC:FWver?	88
CSAC:LIFetime?	88
CSAC:DDScenter?	88
CSAC:TECcontrol?	88
CSAC:STeer:LATch ONCE	88
CSAC:POWer <ON OFF>	89
CSAC?	89
RFOUTput Subsystem	90
RFOUTput:MODE <ON OFF>	90
RFOUTput:POWer <float> [-200.0, 200.0]	91
RFOUTput:DCBlock <ON OFF>	91
RFOUTput:RESET ONCE	91
RFOUTput:ISP ON	91
RFOUTput:SCPI <ASCII> <command>	92
RFOUTput:SIM:RESTART ONCE	92
RFOUTput:SIM:HOLDover <ON OFF LIMIT>	92
RFOUTput:SIM:HOLDover:LIMIT <int> [5, 86400]	92
RFOUTput:SIM:LNAV:SElect <AUTO USER LIVE>	92

RFOUtpuT:STATE?	93
RFOUtpuT:TINT?	93
RFOUtpuT:FEE?	93
RFOUtpuT:INTGPS:PPSERRor?	93
RFOUtpuT:INTGPS:POSERRor?	93
RFOUtpuT:INTGPS:VPOSERRor?	93
RFOUtpuT:EFC?	93
RFOUtpuT:VIN?	94
RFOUtpuT:EXTGPS:PPSERRor?	94
RFOUtpuT:EXTGPS:POSERRor?	94
RFOUtpuT:EXTGPS:VPOSERRor?	94
RFOUtpuT:FGPAdiag?	94
RFOUtpuT:LOCKcode?	94
RFOUtpuT:LOCKcode:VALue?	95
RFOUtpuT:HEAlth?	95
RFOUtpuT:HEAlth:VALue?	96
RFOUtpuT:FWver?	96
RFOUtpuT:ERRORcode:VALue?	96
RFOUtpuT:ERRORcode?	96
RFOUtpuT:SIM:POStion <OFF lat, lon, alt>.	96
MEASURE Subsystem	96
MEASure:APOWersupply?	96
MEASure:BPOWersupply?	96
MEASure:SUPPLY5VOLT?	97
MEASure:POWersupply:V33?	97
MEASure:POWersupply:VOSCillator?	97
MEASure:POWersupply?	97
MEASure?	97
System subsystem.	97
SYSTem:COMMUnicate:SERial:ECHO <ON OFF>	98
SYSTem:COMMUnicate:SERial:PROmpt <ON OFF>	98
SYSTem:COMMUnicate:SERial:BAUD <9600 19200 38400 57600 115200>	98
SYSTem:COMMUnicate:SERial:FAST <ON OFF>	99
SYSTem:COMMUnicate:GNSS:BAUD <9600 19200 38400 57600 115200>	99
SYSTem:COMMUnicate:GNSS:MODE <ON OFF>	99
SYSTem:COMMUnicate:RSRSCPI:BAUD <9600 19200 38400 57600 115200>	100
SYSTem:COMMUnicate:RSRSCPI:MODE <ON OFF>	100
SYSTem:COMMUnicate:STL:BAUD <9600 19200 38400 57600 115200>	100
SYSTem:COMMUnicate:STL:MODE <ON OFF>	100
SYSTem:COMMUnicate:TOD:BAUD <9600 19200 38400 57600 115200>	101
SYSTem:COMMUnicate:TOD:MODE <ON OFF>	101
SYSTem:COMMUnicate:CSAC:BAUD <9600 19200 38400 57600 115200>	101
SYSTem:COMMUnicate:CSAC:MODE <ON OFF>	102
SYSTem:COMMUnicate:RSRGPS:BAUD <9600 19200 38400 57600 115200>	102
SYSTem:COMMUnicate:RSRGPS:MODE <ON OFF>	102
SYSTem:COMMUnicate:MCE:BAUD <9600 19200 38400 57600 115200>	102

SYSTem:COMMunicate:MCE:MODE <ON OFF>	103
SYSTem:POWer <SLEEP DSLEEP PDOWN DPDOWN>	103
SYSTem:COMMunicate:DLOAD <STL GNSS TRANscoder>	103
SYSTem:STATus?	104
SYSTem:FACToryReset ONCE	104
SYSTem:CPURESET	104
SYSTem:ISP	104
SERVO Subsystem	104
SERVo:FASTlock <int> [1,20]	105
SERVo:FALEngth <int> [100,20000]	106
SERVo:COARSedac <int> [0,255]	106
SERVo:MODE <SLOW MEDIum FAST AUTO>	106
SERVo:MODE:FASTDURation <int> [60, 604800]	107
SERVo:MODE:MEDDURation <int> [60, 604800]	107
SERVo:STATe?	107
SERVo:DACGain <float> [0.001, 10000]	107
SERVo:EFCScale <float> [0.0,500.0]	108
SERVo:EFCScale:MEDIum <float> [0.0,500.0]	108
SERVo:EFCScale:FAST <float> [0.0,500.0]	108
SERVo:EFCDamping <int> [2,4000]	108
SERVo:EFCDamping:MEDIum <int> [2,4000]	108
SERVo:EFCDamping:FAST <int> [2,4000]	109
SERVo:SLOPe <NEG POS>	109
SERVo:TEMPCOmpensation <float> [-4000.0, 4000.0]	109
SERVo:AGINGcompensation <float> [-10.0,10.0]	109
SERVo:PHASECOrrrection <float> [-500.0,500.0]	110
SERVo:PHASECOrrrection:MEDIum <float> [-500.0,500.0]	110
SERVo:PHASECOrrrection:FAST <float> [-500.0,500.0]	110
SERVo:1PPSOffset <int> <ns> [-5000000,5000000]	110
SERVo:SLOPe <NEG POS>	110
SERVo:QUlet <ON OFF>	111
SERVo:DBGMode <int> [0, 255]	111
SERVo:TRACe <int> [0,255]	111
SERVo?	112
Chapter 5 Firmware Upgrade	113
Introduction	114
Putting the device into In-Circuit Programming (ISP) mode	114
Downloading the firmware	114
Using the JLTerm programming terminal	114
Verifying the firmware update	116

Introduction

This chapter discusses the following topics:

- [“About the PNT-440” on page 2](#)
- [“Optional accessories” on page 3](#)
- [“Customer-provided equipment” on page 3](#)
- [“General safety precautions” on page 3](#)
- [“Dimensions” on page 5](#)
- [“PNT-440 device connections” on page 5](#)

About the PNT-440

The PNT-440 module is a resilient multi-source clock module designed for seamless integration into:

- Intelligence, surveillance, and data analysis platforms
- Satellite communication (SATCOM) infrastructure
- Field-deployable systems supporting mission-critical operations
- High-reliability communication and networking environments

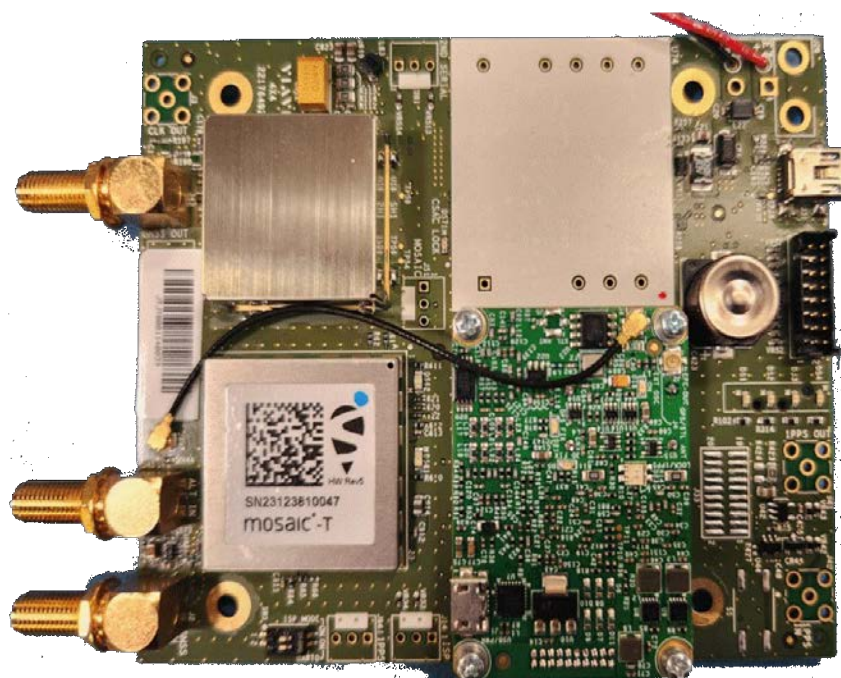
The PNT-440 contains a multi-constellation cable GNSS receiver, a Low Earth Orbit (LEO) Iridium satellite receiver, and a timing servo capable of synchronizing either GNSS or LEO timing signals. It then reformulates a “GPS – like” signal which is then fed into the Satellite Terminal, providing timing and static position information.

The PNT-440 provides:

- Integrated SecureTime altGNSS/eGNSS services:
 - altGNSS LEO Iridium STL (L-band)
 - eGNSS GEO Inmarsat (L-band)
 - altGNSS GEO Inmarsat (L-band)
- External Cesium clock backup option
- Unique anti-spoofing detection and mitigation via NMA support on all GNSS constellations
- Capability to retrofit legacy GPS clocks using μ PNTTranscoder

Figure 1 shows the PNT-440 device.

Figure 1 PNT-440 device



Optional accessories

Table 1 describes the optional accessories available.

Table 1 PNT-440 optional accessories

Part Number	Description
36-80724	Micro mini USB 2.0 mini B 1 meter cable
21-80722	Iridium/GNSS Puck Antenna – Tallysman 33-2643A
22181618	STL Antenna - Iridium Helical Active Antenna – Tallysman 33-HC610-28
22181619	GNSS Multi-Band L1/L2/L5 Antenna – Tallysman VSP6337L
22181620	GNSS Antenna Magnetic Mount – For use with GNSS Multi-Band L1/L2/L5 antenna (22181619)

Customer-provided equipment

The following equipment is needed to install the PNT-440:

- Windows PC (required to support firmware upgrades and monitoring/control)
- Frequency/time measurement equipment as required for evaluation
- 12 VDC power supply (see [“Powering the unit” on page 8](#) for more information)
- Applicable cables (see [“Connectors” on page 9](#) for more information)

General safety precautions

The following general safety precautions must be observed during all phases of operation of this instrument. Failure to comply with these precautions or with specific warnings elsewhere in this manual violates the safety standards of design manufacture, and intended use of the instrument. VIAVI assumes no liability for the customer's failure to comply with these requirements.

Surge Protector

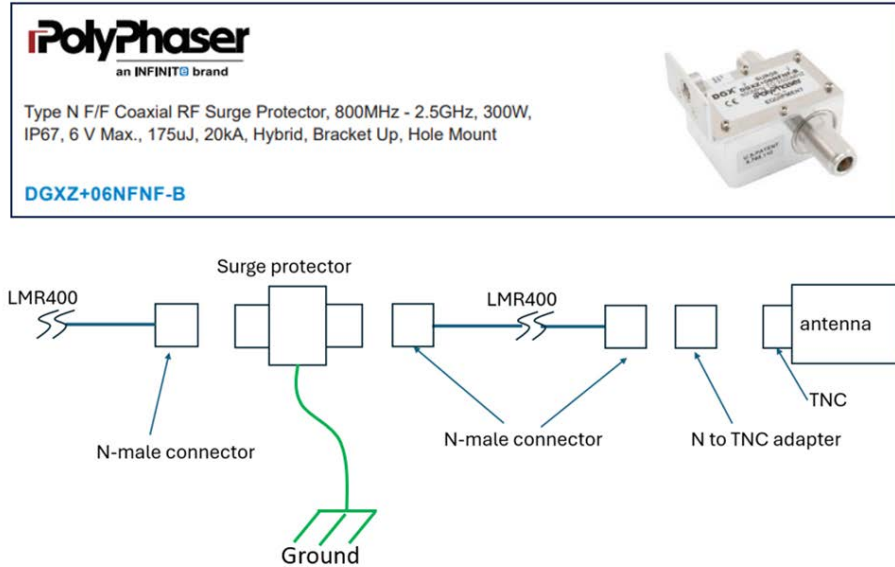
Use a surge protector with the following requirements:

- DC pass voltage threshold supports antenna power
- Clamp voltage low enough to protect downstream equipment
- Frequency response includes L-band

Surge protectors offer various installation methods. Follow the specific installation instructions provided with the surge protector.

Figure 2 shows an example for a 5V antenna power feed.

Figure 2 Surge protector example



Grounding

To avoid damaging the sensitive electronic components of the PNT-440 always make sure to discharge any built-up electrostatic charge to a good ground source, such as power supply ground. This should be done before handling the circuit board or anything connected to it (i.e. the GNSS antenna.)

Power connections

Make sure to connect the DC power to the device following the polarity indicated in [“Powering the unit” on page 8](#). Do not reverse the power pins as this will cause serious damage to the circuit board.

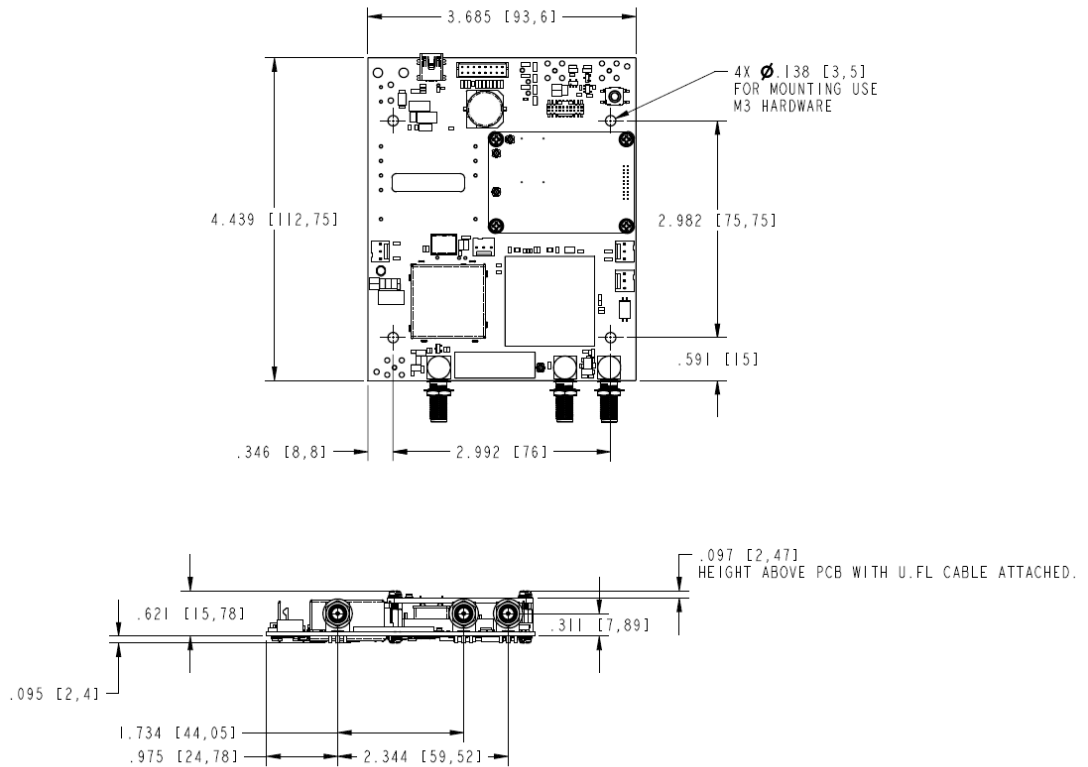
Environmental conditions

This instrument is intended for indoor use. It is designed to operate at a maximum relative non-condensing humidity of 95% and at altitudes of up to 50,000 meters. Refer to the specifications tables for the ac mains voltage requirements and ambient operating temperature range.

Dimensions

Figure 3 shows the physical dimensions of the device.

Figure 3 PNT-440 dimensions

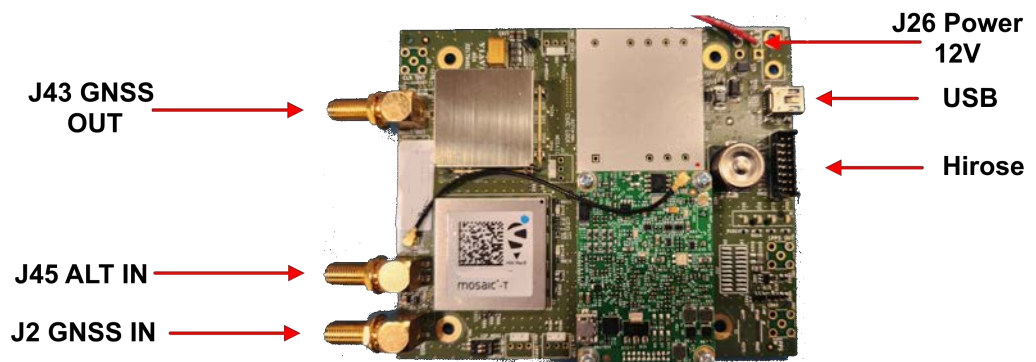


PNT-440 device connections

The following sections describe the PNT-440 device connections.

Figure 4 shows the connectors for an PNT-440.

Figure 4 PNT-440 device connectors



[Table 2](#) describes the PNT-440 device.

Table 2 PNT-440 device connections

Port Name	Description
GNSS OUT (J43)	SMA Output. Sources a GNSS L1 signal out to live GNSS receiver (internally generated GNSS signal.)
ALT In (J45)	SMA input. Input for Iridium L1 Sat band.
GNSS In (J2)	SMA input for a GNSS antenna. Supplies 3.3 VDC bias to power external GNSS active antenna.
DC 12V Power In (J26)	DC Power Input
Mini-USB	USB device port. Provides serial connection to host/PC for primary control.
Hirose	16-pin RS-232 Level Serial I/O. General I/O for various ports Including RS-232 / RS-422 / TTL-Serial / In-System-Programming and serial select control.

Installation

The following topics are discussed in this chapter:

- [“Overview” on page 8](#)
- [“Pre-installation” on page 8](#)
- [“Powering the unit” on page 8](#)
- [“Connectors” on page 9](#)
- [“Quick start” on page 12](#)

Overview

The following sections describe how to install the PNT-440 device.

Pre-installation

Perform the following steps before installing the receiver.

- 1 Determine STL antenna location.



NOTE

For indoor installations, locate the antenna near, or as close as possible, or within sight of a window, skylight, or outside wall.

- 2 Determine the STL receiver location and whether the rack mount kit is needed.
- 3 Install the RF antenna cable from the antenna location to the receiver location: 50 Ohm cable with less than 10 dB loss at 1626 MHz, SMA-M connectors both ends.
The typical cable loss for 100 ft @ 1626 MHz:
 - LMR 400 6.0 dB (preferred)
 - LMR 240 11.4 dB (preferred)
 - RG214 .7 dB (preferred)
 - LMR 200 15 dB (short cables only)



NOTE

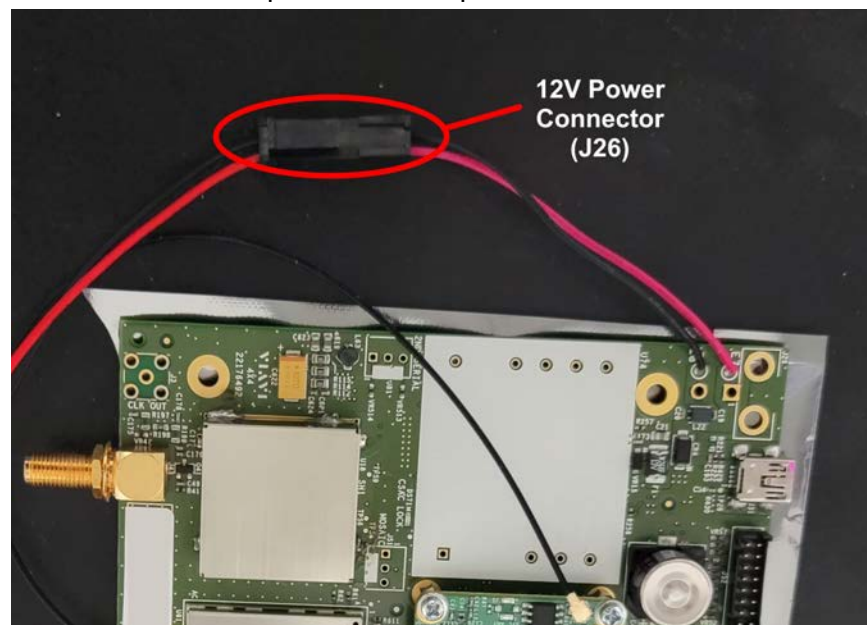
LMR 400 and similar “stiff” cables usually require a flexible RF tail extension to not over stress the SMA connection.

- 4 Have the client GNSS receiver available at the site.
- 5 Have an Uninterrupted Power Supply (UPS) available for the STL receiver.
- 6 Have a laptop ready for the integration verification (optional, but recommended.)

Powering the unit

Power to the unit is supplied by connecting the 12V DC power supply to the 12V power connector, as shown in [Figure 5](#).

Figure 5 12 VDC Power Input Connector pin locations

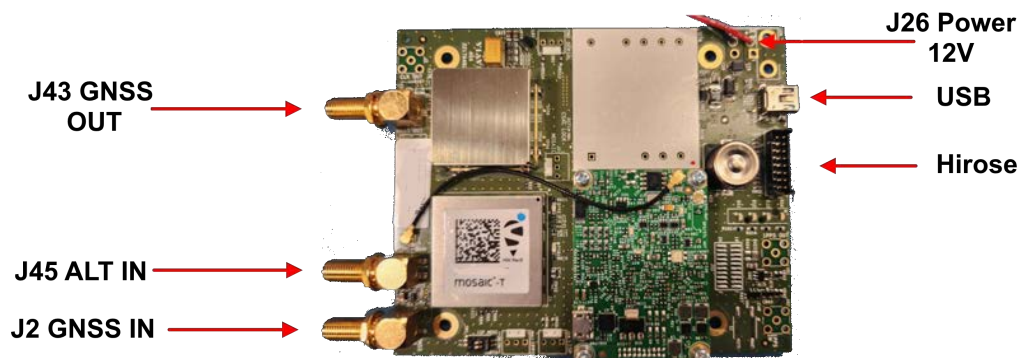


Connectors

The following sections describe the PNT-440 device connections and indicators.

[Figure 6](#) shows the connectors for an PNT-440.

Figure 6 PNT-440 connectors



[Table 3](#) describes the PNT-440 device connections.

Table 3 PNT-440 device connections

#	Port Name	Description
J43	GNSS OUT	SMA Output. Sources a GNSS L1 signal out to live GNSS receiver (internally generated GNSS signal).
J45	ALT IN	SMA Input. Input for Iridium L1 Sat band.
J2	GNSS IN	SMA input for a GNSS antenna. Supplies 3.3VDC bias to power external GNSS active antenna.

Table 3 PNT-440 device connections

#	Port Name	Description
JP2	DC 12V Power In	DC Power Input
J31	Mini-USB Serial Com	USB device port. Provides serial connection to host/ PC for primary control.
J52	Hirose 16-pin RS-232 Level Serial	General I/O for various ports Including RS-232 / RS-422 / TTL-Serial / In-System-Programming and serial select control

16-pin Hirose I/O connector

Table 4 describes the 16-pin Hirose I/O (J52) connector pin information.

Table 4 16-pin Hirose I/O (J52) connector pin information

Pin	Signal	Direction	Type	Function
1	TX-NMEA-P	OUT	Diff RS-422 120-ohm	RS-422 Serial Port TX Interface from PCBA towards external system for UART0. This Port can also be used to update firmware ISP-Mode.
2	TX-NMEA-N	OUT	Diff RS-422 120-ohm	RS-422 Serial Port TX Interface from PCBA towards external system for UART0. This Port can also be used to update firmware ISP-Mode.
3	RX-NMEA-P	IN	Diff RS-422 120-ohm	RS-422 Serial Port RX from external system towards PCBA for UART0. This Port can also be used to update firmware ISP-Mode. Mutually Exclusive with TTL Serial Port. Connect only RS-422 OR TTL Serial Port.
4	RX-NMEA-N	IN	Diff RS-422 120-ohm	RS-422 Serial Port RX from external system towards PCBA for UART0. This Port can also be used to update firmware ISP-Mode. Mutually Exclusive with TTL Serial Port. Connect only RS-422 OR TTL Serial Port.
5	GND	—	Signal return ground	Connect to system ground.
6	GND	—	Signal return ground	Connect to system ground.

Table 4 16-pin Hirose I/O (J52) connector pin information

Pin	Signal	Direction	Type	Function
7	LOCK_OK	OUT	TTL CMOS 3.3V	Dual function output: Indicates system ready/locked. Also indicates possible other events pending. Poll BIST parameter for event info.
8	ARM_UART_TXD	OUT	TTL CMOS 3.3V	TTL Serial Port TX Interface from PCBA towards external system for UART0. This Port can also be used to update firmware ISP-Mode.
9	ARM_UART_RXD	IN	TTL CMOS 3.3V	TTL Serial Port RX Interface from external system towards PCBA for UART0. This Port can also be used to update firmware ISP-Mode. Mutually Exclusive with RS-422 Diff. Connect only RS-422 OR TTL Serial Port.
10	ENTER_ISP#	IN	TTL CMOS (PU 4.7k)	Pull Low for programming flash from UART0, float otherwise
11	ENTER_UART3_ISP#	OUT	TTL CMOS (PU 4.7k)	Pull Low for programming flash from UART3, float otherwise
12	TXD_232	OUT	RS-232 SIGNAL LEVEL	RS-232 Serial Port TX Interface from PCBA towards external system for UART3. This Port can also be used to update firmware ISP-Mode
13	RXD_232	IN	RS-232 SIGNAL LEVEL	RS-232 Serial Port RX Interface from external system towards PCBA for UART3. This Port can also be used to update firmware ISP-Mode. Mutually Exclusive with USB-Serial port. Connect only RS-232 OR USB Serial Port.
14	NC	—	—	Reserved. Do not connect.
15	GND	—	—	Connect to system ground.
16	GND	—	—	Connect to system ground.

USB I/O connector

Table 5 describes the mini-USB I/O (J31) connector pin information.

Table 5 Mini-USB (J31) connector pin information

Pin	Signal	Direction	Type	Function
1	USB V5	IN	Debug Power Input	Provides 5B DC to PCBA but is nOT used to power any curcuits.
2	D-	IN/OUT	Diff USB Serial Signal	USB-Serial Interface to/from UART3. This Port can also be used to update firmware ISP-Mode. Mutually Exclusive with RS-232 port. Connect only RS-232 OR USB Serial Port.
3	D+	IN/OUT	Diff USB Serial Signal	USB-Serial Interface to/from UART3. This Port can also be used to update firmware ISP-Mode. Mutually Exclusive with RS-232 port. Connect only RS-232 OR USB Serial Port.
4	NC	—	—	Do not connect.
5	GND	—	Ground	—
6	SHIELD GND	—	Shell	—
7	SHIELD GND	—	Shell	—
8	SHIELD GND	—	Shell	—
9	SHIELD GND	—	Shell	—

Quick start

Perform the following procedure to connect the PNT-440 for operation.

To connect the PNT-440 for operation

- 1 Connect the GNSS antenna to **GNSS In**.
- 2 (Optional) Connect the STL/LEO antenna to **ALT IN**.
- 3 Connect the PNT-440 to a PC using the **Mini USB** connector.
- 4 Connect DC power to **DC 12V Power In**.

5 On the connected PC, open the **GPSCon** utility to confirm:

- a** **GNSS Status** achieves **3D fix**.
- b** **GPSDO Status** goes to **LOCKED**.
- c** **STL/LEO Status** achieves **3D Fix** (if applicable.)

See [“GPSCon Utility” on page 15](#) for more information about using the GPSCon application.

GPSSCon Utility

The following topics are discussed in this chapter:

- [“Description” on page 16](#)
- [“Using GSPCon” on page 16](#)
- [“Using GSPCon” on page 16](#)
- [“Interpreting the Data” on page 21](#)

Description

GPSCon – VIAVI Edition is a free program for the monitoring and control of the PNT-440 Receiver. It communicates with the receiver using the SCPI command set. The program can be downloaded from:

<https://www.viavisolutions.com/en-us/software-download/gpscon-controller-software>.

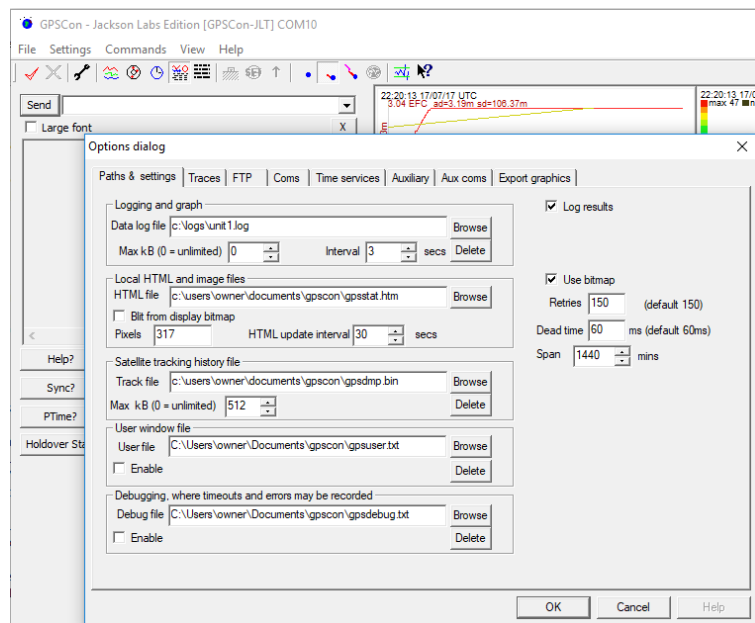
Using GPSCon

The GPSCon utility has a help file that should be consulted in order to get the full functionality of this utility. Only a few of the features and commands are mentioned in this appendix for convenience.

Setting options

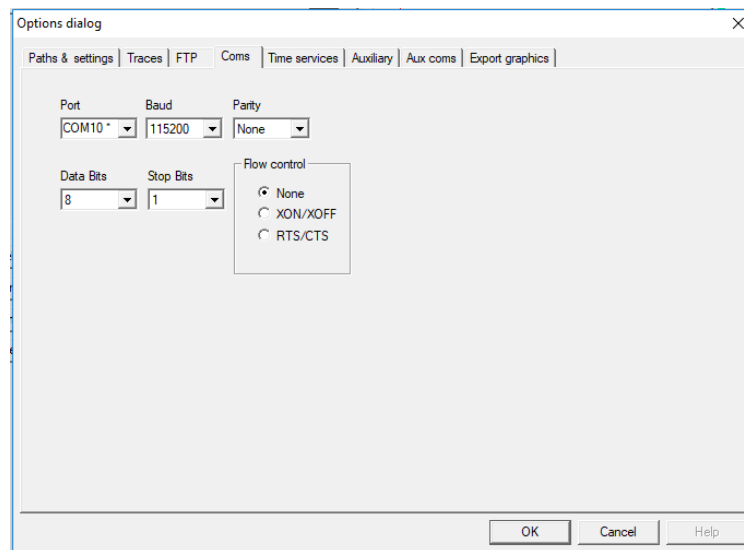
To set up the options for your GPSCon session, press the wrench icon  under the menu bar, or select Settings / Options on the menu. The window shown in Figure 7 will appear. You can select from the tabs which options you wish to set.

Figure 7 Options window



Communication parameters

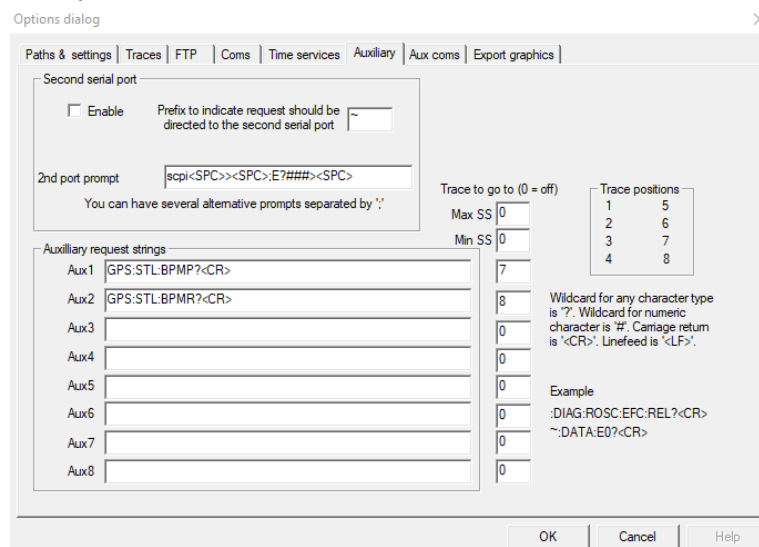
Before you can use GPSCon, you must set the communication parameters for your system. Open the dialog box by pressing the wrench icon , then select the **Coms** tab. The Options window opens on the Coms tab, as shown in Figure 8.

Figure 8 Coms parameters tab

Available COM ports for your particular computer are indicated with a “*” symbol next to the COM port number in the drop-down menu. You can only select COM ports with this adjacent symbol. Select the correct COM port for your computer’s serial port connection and set the baud rate to 115200, parity to None, Data Bits to 8 and Stop Bits to 1. Set Flow Control to “None”. Once you have configured the communication parameters, press the “OK” button to close the window.

Auxiliary parameters

After pressing the wrench icon , you can select the **Auxiliary** tab to configure auxiliary measurements. [Figure 9](#) shows an example of an auxiliary measurement.

Figure 9 Auxiliary parameters tab

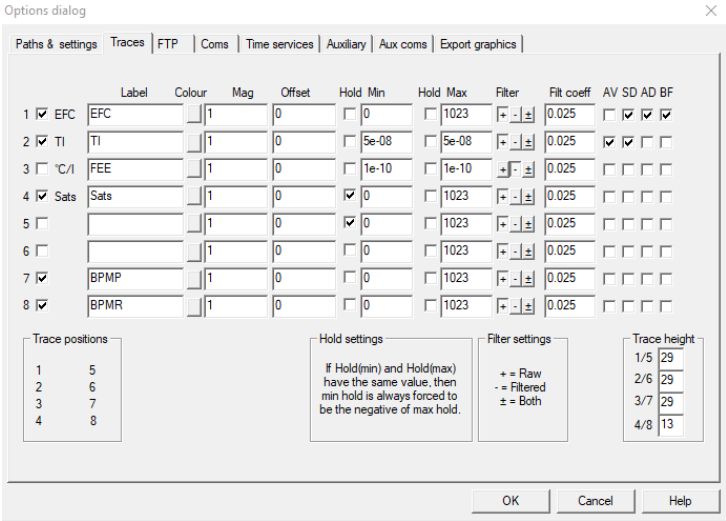
In the above example, the **Aux1** request string has been set to `GPS:STL:BPMP?<CR>` and the **Aux2** request string has been set to `GPS:STL:BPMP?<CR>`. The **Trace to go to** is set to trace positions 7 and 8 respectively. See the **Trace Position** diagram for the arrangement of the trace positions in the trace window. In this example, the bursts

process and received obtained from the GPS:STL:BPMP? and GPS:STL:BPMPR? queries will be plotted in trace position 7 and 8.

Traces parameters

After pressing the wrench icon  , you can select the **Traces** tab and configure the trace labels and vertical plot ranges. [Figure 10](#) shows an example of the trace parameter tab.

Figure 10 Traces parameters tab

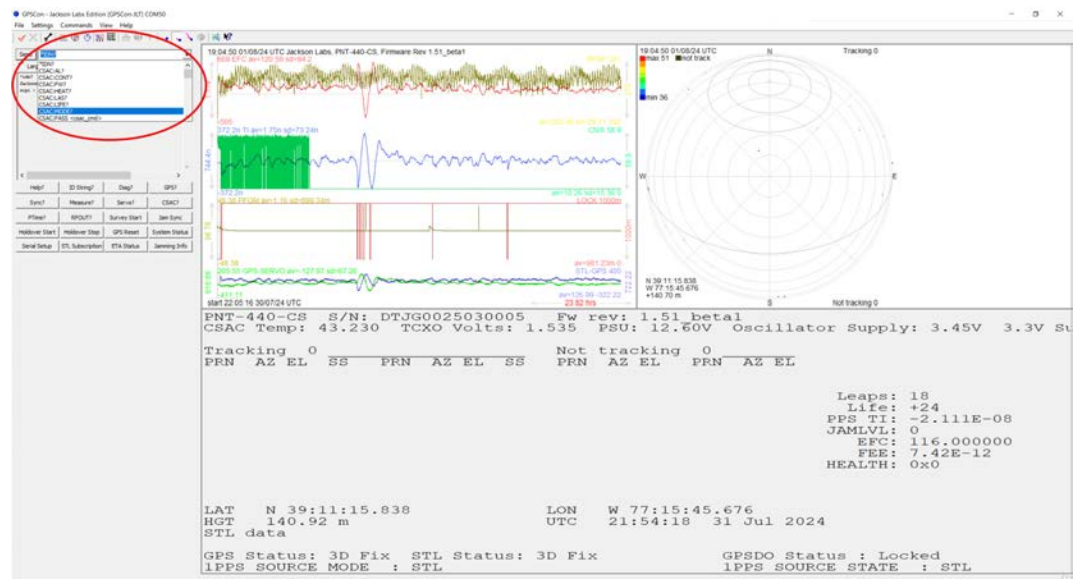


The labels and parameters are completed by default for traces 1 through 5. Any of the eight traces can be replaced by auxiliary traces as described in [“Auxiliary parameters”](#) on [page 17](#). Press the **Help** button for a full description of each option in the Traces tab.

Sending manual commands to the receiver

You can send SCPI commands manually by using the drop-down box in the upper left of the main window, as shown in [Figure 11](#).

Figure 11 Sending manual commands



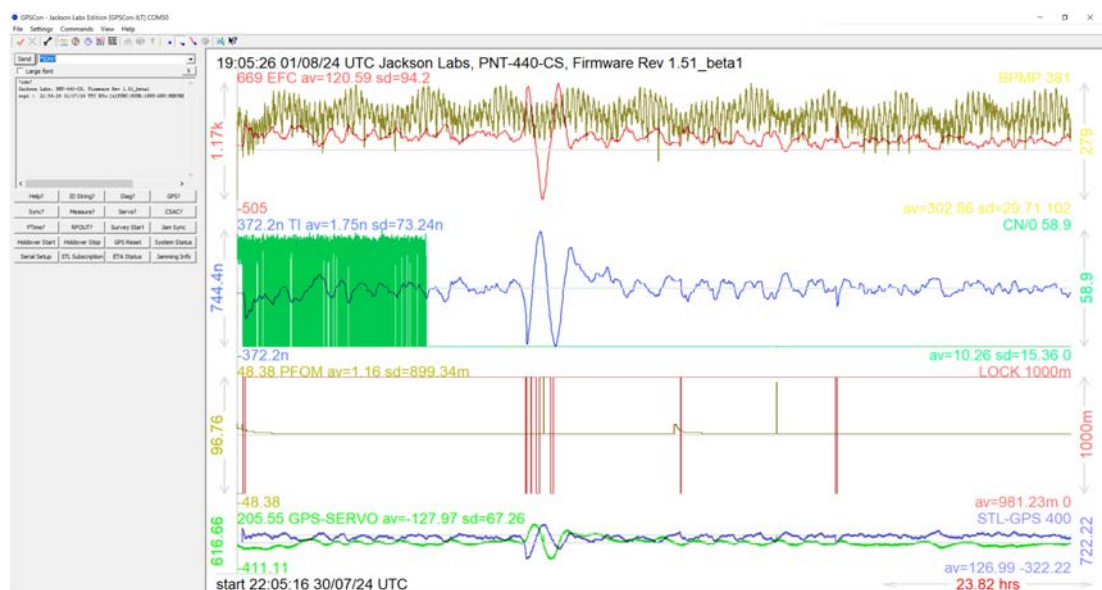
Once a command is selected, press **Send** to send it to the PNT-440 device. You can also send common commands by clicking on the buttons below the message window. Hover over the buttons to see the exact command that is sent.

NOTE
Ensure the selected command is supported by the PNT-440 device.

Using the mouse in the Graph display

Figure 12 shows the Graph display. The default view in GPSCon is **All**, which you can select with the **View/All** menu option. To see a larger view of the graph, select the **View/Graph** menu option.

Figure 12 Graph display



You can set the horizontal range of the graph using the mouse. Perform the following steps to set the horizontal range.

- 1 Set the start time by clicking on the point that marks the left side of the curve to be magnified.
- 2 Set the stop time by right-clicking on the point that marks the right most portion of the magnified curve.



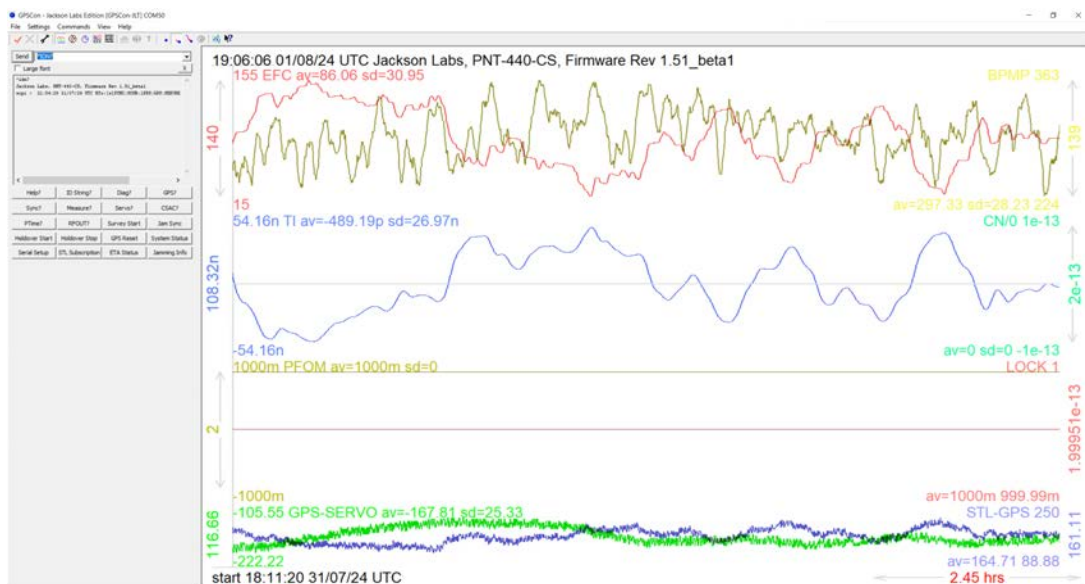
NOTE

The Set and Stop times can be removed by double-clicking anywhere on the graph.

- 3 Return to the “Fit to window” view by double left-clicking on the curve.

Figure 13 shows the extended graph display.

Figure 13 Expanded graph display



When you have locked the start and stop time using the mouse, you can scroll left or right through the data:

- To scroll to a later time, use Shift + Left click
- To scroll to an earlier time, use Shift + Right click

Exporting graphics

The settings which control the export function are contained in the “Export graphics” tab in the Options dialog.

Export allows you to create an image file of the graph and/or the satellite trails map. You can select which you want by specifying a file for the Graph path and/or Map path. If you export the graph, you have the option to export only that which is currently visible, or to export the graph which is a plot of the entire logfile contents. Use the checkbox “Export all graph data” to make this choice.

You may select a size of the exported images in X and Y. The file format may be .BMP, .JPG, .GIF, or .PNG. Your settings will be stored and will be the default next time you open this dialog.

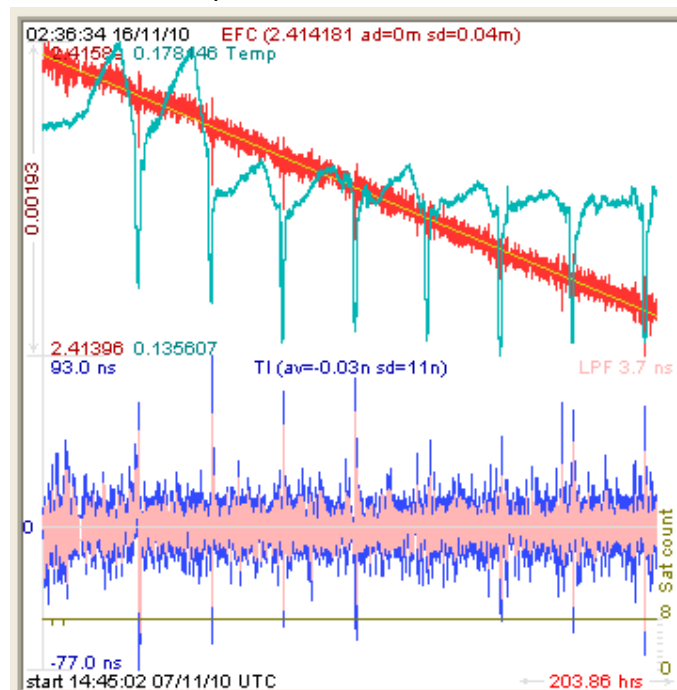
If you choose to export the graph, you can override the TI max setting in force on the screen display by entering a non-zero value into the 'Override TI' control. A value of zero causes the export to take the same setting if any as the screen display.

- To export automatically on a timed basis, enter a non-zero value in seconds to choose an export time interval.
- To manually export in accordance with the settings, press the 'Export' button.

Interpreting the Data

Figure shows an example of data acquired by a GPSDO unit over a period of time of more than 200 hours.

Figure 14 Captured data example



The red trace is EFC (crystal frequency control voltage). The crystal is aging (becoming faster in frequency over time). This requires the control voltage to be lowered to maintain precisely 10.0 MHz. A drift of ~2 mV is visible over 200 hours. On the left side of the screen, the EFC range over this 200 hour plot is displayed vertically as 0.00193 V. This means the drift of the EFC voltage due to aging is ~88 mV per year. The EFC sensitivity of the crystal is about 8 Hz per volt, so the crystal ages at:

$$8\text{Hz/V} * 0.088\text{V/Year} = 0.704\text{Hz/Year drift.}$$

At 10 MHz:

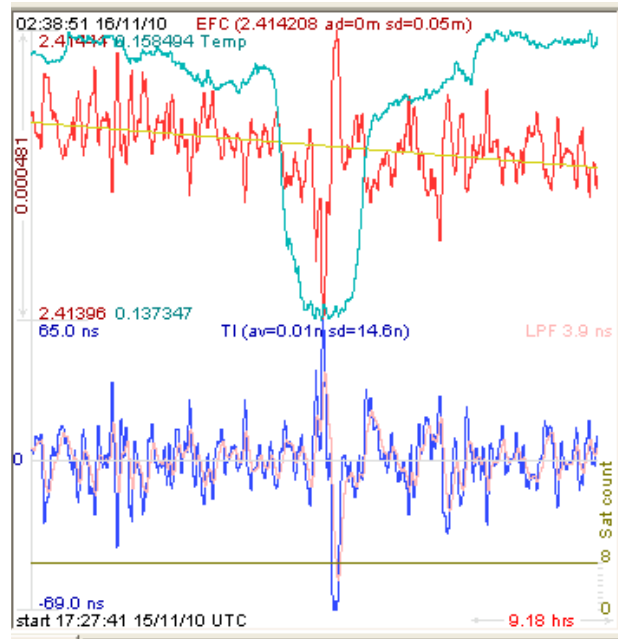
$$0.704\text{Hz} / 10\text{MHz} = 7.04\text{E-08 aging rate per year}$$

This is the same as 0.2ppb drift due to aging per day. This crystal aging is fully compensated by the firmware with and without GPS reception of course.

The board temperature is shown in tortoise. We can see it ranges from 0.135607A to 0.178146A. The OCXO current jumps lower every 24 hours because the unit is sitting next to a window, and the sun shines onto the OCXO in the evenings, heating it up, and thus making the unit use lower power during that event.

Figure 15 shows a zoom of the captured data.

Figure 15 Zoomed captured data example



The image shows a phase offset error of the internal OCXO to the UTC GPS reference. The maximum drift is -77 ns to +93 ns. The average is ($TI_{av} = -0.03$ ns). The standard deviation over the 200 hour plot is $sd = 11$ ns. This means the average error of the 10 MHz phase of this unit over 200 hours is only +/- -11 ns. Or, in other words, the average jitter (wander) over 200 hours is:

$$11\text{ns} / 200\text{Hrs} = 1.528\text{E-}014$$

The unit performs as well as a high quality Cesium Atomic reference clock over long periods of time. The unit disciplines its internal 10MHz reference to within less than +/- 80ns peak to peak of UTC at all times, which is less than one complete clock cycle at 10MHz.

SCPI Commands

The following topics are discussed in this chapter:

- [“Introduction” on page 24](#)
- [“General SCPI commands” on page 24](#)
- [“GPS Subsystem” on page 25](#)
- [“PTIME Subsystem” on page 70](#)
- [“SYNChronization Subsystem” on page 73](#)
- [“DIAGnostic Subsystem” on page 83](#)
- [“CSAC Subsystem” on page 84](#)
- [“RFOUTput Subsystem” on page 90](#)
- [“MEASURE Subsystem” on page 96](#)
- [“System subsystem” on page 97](#)
- [“SERVO Subsystem” on page 104](#)

Introduction

The Standard Commands for Programmable Instrumentation (SCPI) subsystem is accessed using a mini-USB or RS-232 serial cable and a terminal program. By default, the terminal settings are 115200, 8N1, no flow-control.

A number of commands that can be used are listed below. To get a listing of the available commands, send the `HELP?` query. This will return a list of all the available commands for the PNT-440 device. Additional information regarding the SCPI protocol syntax can be found here:

<http://www.ivifoundation.org/scpi/>

A basic familiarity with the SCPI protocol is recommended when reading this chapter.



NOTE

The symbols '<', '>', '[', ']', '|', and ',' in the parameter field listed in some commands in this chapter are used for ranging or separating ONLY. Do not include these symbols when sending the SCPI commands to avoid Command Error response.



NOTE

All lower-case letters in SCPI commands throughout this manual are optional. The abbreviated version of the SCPI commands such as `SYST:STAT?` and `SYNC?` will also work.

General SCPI commands

The following sections describe the general SCPI commands.

Quick Start commands

For a quick start, try the following SCPI serial port commands:

- `HELP?`
- `SYSTem:STATus?`
- `GPS?`
- `SYNChronization?`
- `MEASure?`
- `DIAGnostic?`

*IDN?

Outputs an identifying string. The response shows the following information:

`<company name>, <model number>, <firmware revision>`

For example:

```
VIAVI, PNT-440, Firmware Rev 0.71
```

HELP?

Returns a list of available commands.

GPS Subsystem



NOTE

The PNT-440 device displays antenna height in MSL Meters rather than in GPS Meters on all commands that return antenna height (unless otherwise specified). The NMEA position fixes are in the WGS84 coordinate system, while the X, Y, and Z positions and velocity vectors are given in the ECEF coordinate system.

The GPS subsystem regroups all the commands related to the control and status of the GNSS or STL receiver. The following commands are supported:

- GPS?
- GPS:RESET ONCE
- GPS:REfERENCE:ADELay <float> <s | ns > [-32767ns,32767ns]
- GPS:REfERENCE:PULse:SAWtooth?
- GPS:TMODe <ON|OFF|RSTSURV>
- GPS:SURVey ONCE
- GPS:SURVEY:STATUS?
- GPS:SURVey:DURation <sec>
- GPS:SURVey:VARIANCE <mm^2>
- GPS:HOLD:POSition <cm, cm, cm>
- GPS:DYNAMic:MODE <int> [0,8]
- GPS:DYNAMic:MODE?
- GPS:DYNAMic:MODE 8 (Automatic Dynamic Mode)
- GPS:DYNAMic:STATe?
- GPS:GPRMC <int> [0,255]
- GPS:GPGBA <int> [0,255]
- GPS:GGASat <int> [0,255]
- GPS:GPGLL <int> [0,255]
- GPS:PJLTS <int> [0,255]
- GPS:PJLTV <int> [0,255]
- GPS:XYZSPeed <int> [0,255]
- GPS:GPZDA <int> [0,255]

- GPS:PASHR <int> [0,255]
- GPS:GPGSA <int> [0,255]
- GPS:GPGSV <int> [0,255]
- GPS:SATellite:TRAcking:COUNT?
- GPS:SATellite:VISible:COUNT?
- GPS:JAMlevel?
- GPS:JAMI?
- GPS:FWver?
- GPS:INITial:DATE <yyyy,mm,dd>
- GPS:INITial:TIME <hour,min,sec>
- GPS:SYSTem:SElect [GPS | SBAS | QZSS | GAL | BD ^ GLO]
- GPS:SYSTem:LBAND [L1 | L2 | L3 | L5 | ALL]
- GPS:SYSTem:LBAND:AGCgain?
- GPS:SYSTem:LBAND:AGCPnorm?
- GPS:SYSTem:LBAND:SVID
- GPS:SYSTem:LBAND:CNO
- GPS:SYSTem:BANDTRAck [L1 | L2 | L3 | L5 | ALL]
- GPS:SYSTem:TIMESOURce <GPS|GAL|BD|AUTO>
- GPS:SYSTem:TIMESOURce:STATe?
- GPS:SYSem:ETA <ON|OFF>
- GPS:SYSTEM
- GPS:HEALth:IGNOre <ON|OFF>
- GPS:RTC:DATE <yyyy,mm,dd>
- GPS:RTC:TIME <hour,min,sec>
- GPS:HEIGHT?
- GPS:HEIGHT:MSL?
- GPS:HEIGHT:GPS?
- GPS:POLarity <LOW|HIGH>
- GPS:POSition?
- GPS:POSition:ECEF?
- GPS:CNOthres <int> [1,40]
- GPS:FOV <int> [3,85]
- GPS:STATus?
- GPS:STATus:STRing?
- GPS:PASSthru <GNSS|RSRSCPI|STL|TOD|CSAC|RSRGPS|MCE|RSRDEBUG>
- GPS:PASSthru:TIMEout <ON|OFF>
- GPS:PASSthru:COUNT <int> <s> [1,86400]
- GPS:TYPE?
- GPS:TYPE:MODE <AUTO|UBLOX|SEPTentrio|STL>
- GPS:SOURce:SElect <AUTO|STL|GNSS>

- GPS:FILTer:SElect <STL|UBX>
- GPS:FILTer:NMEA <int> [0,255]
- GPS:GNSS:MODE <ON|OFF>
- GPS:GNSS:RESET ONCE
- GPS:GNSS:POWer <ON|OFF>
- GPS:GNSS:FWver?
- GPS:GNSS:TYPE?
- GPS:STL:RESET <ONCE | WARMUP|HOLD|OFF>
- GPS:STL:RESET:AUTO <int> [0,604800]
- GPS:STL:SCPI <ASCII> <command>
- GPS:STL:ISP ON
- GPS:STL:BPMProcessed?
- GPS:STL:BPMReceived?
- GPS:STL:PFOM?
- GPS:STL:TFOM?
- GPS:STL:EXTCLOCK <ON|OFF>
- GPS:STL:AGC <int> [500,2700]
- GPS:STL:CNO?
- GPS:STL:CNO:MINThReshold <float> [1.0,100.0]
- GPS:STL:DOPpler:MINimum <double> [-36000.000000,0.0]
- GPS:STL:DOPpler:MAXimum <double> [0.0,36000.000000]
- GPS:STL:CLOCKmodel <double> [1E-20,1.0]
- GPS:STL:CLOCKmodel:FAST <double> [1E-20,1.0]
- GPS:STL:CLOCKmodel:MEDium <double> [1E-20,1.0]
- GPS:STL:SUBScription?
- GPS:STL:SUBScription:AUTHKEY <string>
- GPS:STL:SERVo:LOOP <ON|OFF>
- GPS:POWTLV <ms> [200,1000]
- GPS:PJLTV <int> [0,255]
- GPS:STL:SERVo:INDOOR <OFF|ON>
- GPS:ANTPower <ON|OFF>
-

GPS:SATellite

This group of commands describe the satellite constellation.

GPS:SATellite:TRAcking:COUNT?

This query returns the number of satellites being tracked for GNSS receiver only.

GPS:SATellite:VISible:COUNT?

This query returns the number of satellites (PRN) that the almanac predicts should be visible, given date, time, and position for GNSS receiver only.

NMEA Support

The following NMEA commands allow the PNT-440 device GNSDO to be used as an industry standard navigation GNSS receiver. The GGA, GLL, RMC, GSA, GSV, ZDA and PASHR NMEA commands comprise all necessary information about the antenna position, height, velocity, direction, satellite info, fix info, time, date and other information that can be used by standard navigation applications via the PNT-440 device serial interface. Not all standard NMEA messages such as GSA and GSV will be supported in STL receiver mode (`GPS:TYPE:MODE STL`).

The instrument will send out the enabled NMEA information on the serial transmit pins automatically every N seconds. The PNT-440 device will not send out NMEA messages if the 1PPS output is not yet enabled. See [“SYNChronization:OUTput:1PPS:RESET <ON|OFF>” on page 73](#) for descriptions of 1PPS generation and instructions to enable 1PPS output upon power on. All incoming serial commands are still recognized by PNT-440 device since the serial interface transmit and receive lines are completely independent and orthogonal of one another.

The GSV and GSA messages include the satellites being tracked or included in the GNSS solution and both messages use a modified satellite numbering scheme as detailed in [Table 10](#) that makes all satellite numbers unique. Also, to indicate the GNSS systems enabled and used to generate the NMEA data, the PNT-440 device GNSDO’s NMEA output includes a two character talker ID before the GGA, GLL, RMC, GSA, GSV, and ZDA sentence headers. [Table 10](#) shows the talker IDs for the supported GNSS systems.

If more than one GNSS system is enabled and GNSS receiver mode is selected, the talker ID of NMEA output is GN except for GSV sentence which outputs multiple sets of sentences for each main talker ID. The talker ID of NMEA output in STL receiver mode is GP. Refer to [“GPS:GPGSV <int> \[0,255\]” on page 39](#) for example GSV output format. Also, the GSV output uses a modified satellite numbering scheme as detailed in [Table 10](#) to allow all satellites in different GNSS systems to be differentiated.

Table 6 PRN numbering scheme for GNSS systems

GNSS Type	SV Range	GPGSV PRN vehicle numbering	Talker ID
GPS	G1-G32	1-32	GP
SBAS	S120-S158	33-64, 152-159	GP

Table 6 PRN numbering scheme for GNSS systems

GNSS Type	SV Range	GPSSV PRN vehicle numbering	Talker ID
Galileo	E1-E36	301-336	GA
BeiDou	B1-B37	401-437	GB
IMES	I1-I10	173-182	GP
QZSS	Q1-Q5	139-137	GP
GLOSNASS	R1-R32, R?	65-96, 0	GL

In addition to standard NMEA sentences, the PNT-440 device GNSDO also supports PJLTS and PJLTV VIAVI proprietary NMEA sentences, PASHR and a proprietary version of the GGA sentence (GGASat) described in [“GPS:GGASat <int> \[0,255\]” on page 38](#).

**NOTE**

The position, direction, and speed data is delayed by one second from when the GNSS receiver internally reported these to the PNT-440 device microprocessor, so the position is valid for the 1PPS signal previous to the last 1PPS signal at the time the data is sent (one second delay.)

The time and date are properly output with correct UTC synchronization to the 1PPS signal immediately prior to the data being sent.

Once set, the following command settings will be stored in NV memory, and generate selected NMEA sentence output information even after power to the unit has been cycled.

GPS:GPGGA <int> [0,255]

This command instructs the PNT-440 device to send the NMEA standard GGA message every N seconds, with N in the interval [0,255]. The command is disabled until the GNSS receiver achieves a first fix.

This command has the following format:

```
GPS:GPGGA <int> [0,255]
```

This command will query the state of this command:

```
GPS:GPGGA?
```

The GPGGA string has the following data format:

```
$GPGGA,hhmmss.00,1111.1111,S/N,yyyyy.yyyy,W/  
E,f,ss,hh.h,aa.a,M,gg.g,M,,*[checksum]
```

The GGA output message header includes the talker ID for the currently enabled GNSS system(s) or STL receiver mode as described in [“NMEA Support” on page 36](#). The

GGA message shows height in MSL Meters for GNSS receiver mode (GPS Meters for STL receiver mode), this is different from the STL receiver mode or traditional GPS receivers that display height in GPS Meters. The difference between MSL and GPS height can be significant, 35m or more are common.

GPS:GGATat <int> [0,255]

This command instructs the PNT-440 device to send a modified version of the NMEA standard GGA message every N seconds, with N in the interval [0,255]. The command is disabled until the GNSS receiver achieves a first fix.

This command has the following format:

```
GPS:GGATat <int> [0,255]
```

This command will query the state of this command:

```
GPS:GGATat?
```

The GGATat string has the following data format:

```
$GPGGA,hhmmss.00,llll.llll,S/N,yyyyy.yyyy,W/  
E,l,ss,hh.h,aa.a,M,gg.g,M,,*[checksum]
```

This command replaces the regular NMEA GGA validity flag with a decimal number indicating the lock-state of the unit. See [“SERVo:TRACe <int> \[0,255\]” on page 104](#) for a detailed description of the lock state variable. The command allows capture of the position and other information available in the GGA command, as well as tracking the lock state and health of the unit’s OCXO performance simultaneously.

The GGATat output message header includes the talker ID for the currently enabled GNSS system(s) or STL receiver mode as described in [“GPS:GGATat <int> \[0,255\]” on page 38](#). GGATat output shows height in MSL Meters for GNSS receiver mode (GPS Meters for STL receiver mode), this is different from the STL receiver mode or traditional GPS receivers that display height in GPS Meters. The difference between MSL and GPS height can be significant, 35m or more are common.

GPS:GPGLL <int> [0,255]

This command instructs the PNT-440 device to send the NMEA standard GLL message every N seconds, with N in the interval [0,255]. The command is disabled until the GNSS receiver achieves a first fix.

This command has the following format:

```
GPS:GPGLL <int> [0,255]
```

This command will query the state of this command:

```
GPS:GPGLL?
```

The GPGLL string has the following data format:

```
$GPGLL,1111.1111,S/N,yyyyy.yyyy,W/  
E,hhmmss.00,s,m*[checksum]
```

The GLL output message header includes the talker ID for the currently enabled GNSS system(s) or STL receiver mode as described in [“NMEA Support” on page 36](#).

GPS:GPRMC <int> [0,255]

This command instructs the PNT-440 device to send the NMEA standard RMC message every N seconds, with N in the interval [0,255]. The command is disabled until the GNSS receiver achieves a first fix.

This command has the following format:

GPS:GPRMC <int> [0,255]

This command will query the state of this command:

GPS : GPRMC?

The GPRMC string has the following data format:

```
$GPRMC, hhmmss.00, s, 1111.1111, S/N, yyyyyy.yyyy, W/  
E, k.k, d.d, ddmmyy, , , m*[checksum]
```

The RMC output message header includes the talker ID for the currently enabled GNSS system(s) or STL receiver mode as described in “NMEA Support” on page 36.

GPS:GPGSA <int> [0,255]

This command instructs the PNT-440 device to send the NMEA standard GSA message every N seconds, with N in the interval [0,255]. The command is disabled until the GNSS receiver achieves a first fix.

The command has the following format:

GPS:GPGSA <int>[0,255]

This command queries the state of this command:

GPS : GP GSA?

The GPGSA string has the following data format:

```
$GPGSA,a,a,x,x,x,x,x,x,x,x,x,x,x,x,x,x,x.x,x.x,x.x*[checksum]
```

The GSA output message header includes the talker ID for the currently enabled GNSS system(s) or STL receiver mode as described in [“NMEA Support” on page 36](#). This message is not currently supported for STL receiver mode.

GPS:GPGSV <int> [0,255]

This command instructs the PNT-440 device to send the NMEA standard GSV message every N seconds, with N in the interval [0,255]. The command is disabled until the GNSS receiver achieves a first fix.

This command has the following format:

```
GPS:GPGSV <int> [0,255]
```

This command will query the state of this command:

```
GPS:GPGSV?
```

The GPGSV string has the following data format:

```
$GPGSV,x,x,ss,nn,ee,aaa,ss,...*[checksum]  
$GAGSV,x,x,ss,nnn,ee,aaa,ss,...*[checksum]
```

A separate set of GSV messages are output for each talker ID for the currently enabled GNSS system(s) or STL receiver mode as described in [“NMEA Support” on page 36](#). This message is not currently supported for STL receiver mode.



NOTE

Due to the large number of GNSS satellites that can be tracked by the unit, more than the customary four GSV messages can be sent once per second.

With multiple GNSS systems enabled, a typical sky view may generate up to six GSV messages per second per talker ID.

GPS:GXGGA <int> [0,255]

This command enables broadcasting of the extNMEA input content for debugging purposes. This command should not be enabled during normal use.

The command has the following format:

```
GPS:GXGGA <int> [0, 255]
```

The options are:

- 0: Disabled
- 1: Enabled

GPS:GXRMC <int> [0,255]

This command enables broadcasting of the extNMEA input content for debugging purpose. This should not be enabled during normal use.

The command has the following format:

```
GPS:GXRMCM <int> [0,255]
```

The options are:

- 0:Disabled
- 1:Enabled

GPS:POWGPS <int> [0,255]

This command allows to enable the POWGPS NMEA message.

The command has the following format:

```
GPS:POWGPS <int> [0,255]
```

The options are:

- 0: Disabled
- 1: Enabled

GPS:POWTLV <int> [0,255]

This command allows the enabling of the POWTLV NMEA message.

The command has the following format:

```
GPS:POWTLV <int> [0,255]
```

The options are:

- 0:disabled
- 1:enabled

GPS:SENDNmea <NMEA-0183 GGA Message String|RMC Message String >

This command enables the input of external NMEA GGA or RMC message from an external system. The content of the message is the standard GGA and RMC NMEA messages.

GPS:SOURce:EXTNMEA <AUTO|ON|OFF>

This command allows to the transcoded RF signal to be derived from the incoming external NMEA input. In AUTO mode the EXTNMEA has the lowest priority and the transcoder output will only select this source in case that GPS or STL sources are unavailable. Forcing the input to ON will disable the other sources to be selected thus the RF output will only be derived from the EXTNMEA input source.

The command has the following format:

GPS:SOURce:EXTNMEA <AUTO|ON|OFF>

The options are:

AUTO — The EXTNMEA has the lowest priority and the transcoder output will only select this source if the GPS or STL sources are unavailable.

ON — Disables the other sources to be selected, thus the RF output will only be derived from the EXTNMEA input source.

OFF — Disables all sources.

GPS:SOURce:STATE?

This command allows to query which source is currently selected as source for the position input to the transcoder.

GPS:SOURce:STATE:value?

This command allows to query which source is currently selected as source for the position input to the transcoder. (numeric value)

GPS:STL:DYNAMICmodel?

This command queries the dynamic model used.

GPS:STL:FWver:SDPM?

This command allows to check the SDPM version of the STL module.

GPS:STL:FWver?

This command allows to read the FW version of the currently installed SW.

GPS:STL:GEOLOCation <STATIC|DYNAMIC>

This command allows the user to configure geo-location model being applied to the Kalman Filter of the STL receiver. The default and supported geo-location model is STATIC which assumes stationary Iridium antenna location and the STL solution will not compute updated position output when the antenna is moving.

Please contact your STL subscription/authentication key provider on potential support of other geo-location models of the STL receiver.

This command has the following format:

```
GPS:STL:GEOLOCation <STATIC>
```

GPS:STL:GEOLOCation?

This command queries the settings of the geo-location model.

GPS:STL:GUESSPOSition <BOOTSTRAP|TRUEPOS|MANUAL>

This command will configure the source of initial guess position for STL solution. By providing an accurate initial guess position, the Time to First Fix is reduced. Note that an incorrect guess position may further increase the time to first fix. The default guess position source is BOOTSTRAP mode where the STL receiver will compute the initial position without the information of guess position. The MANUAL or EEPROM settings require the user input of initial guess position with the GPS:STL:GUESSPOSition:MANUAL command.

The TRUEPOS setting allows the STL solution to configure the user input position described in “GPS:STL:TRUEPOSition <double> <aa.aaaaaa,bbb.bbbbbb,h.hh>” as the true antenna location. The TRUEPOS mode will stop computing new position in the solution attempting to improve timing performance in stationary application. The GNSS settings is currently not supported with the future plan of providing initial guess position from the on-board GNSS receiver. The settings changed in this command will be effective only after the next power cycle or when the SYSTem:CPURESET command is sent.

This command has the following format:

```
GPS:STL:GUESSPOSition <MANUAL|EEPROM|GNSS|BOOTSTRAP>
```

GPS:STL:GUESSPOSition?

This command queries the settings of the GPS:STL:GUESSPOSition command.

GPS:STL:SN?

This command allows to query the STL board serial number.

GPS:STL:PJLTGGA <int> [0,255]

This command will configure the NMEA output message for the currently selected serial port receiving this command. This command instructs the device to send the NMEA standard string \$PJLTGGA every N seconds, with N in the interval [0,255]. The output message configured by this command is invalid until the receiver achieves a first fix.

This command has the following format:

```
GPS:STL:PJLTGGA <int> [0,255]
```

GPS:STL:PJLTPOS <int> [0,255]

This command will configure the NMEA output message for the currently selected serial port receiving this command. This command instructs the device to send the NMEA standard string \$PJLTPOS every N seconds, with N in the interval [0,255]. The output message configured by this command is invalid until the receiver achieves a first fix.

This command has the following format:

```
GPS:PJLTPOS <int> [0,255]
```

The following command will query the state of PJLTPOS output for the serial port receiving this command:

```
GPS:PJLTPOS?
```

The PJLTPOS string has the following data format:

```
$PJLTPOS,v,ssssssssss,f,ll.lllllllll,yyy.yyyyyyyy,aaa.aa,xxx  
xxxxxx,yyyyyyyyyy,zzzzzzzzz,uu.uuuu,uu.uuuu,uu.uuuu,uu.uuuu,  
uu.uuuu,uu.uuuu,b.b,s.s,p,g*[checksum]
```

where:

- **v** is the message version
- **ssssssssss** is the current time with "format of time" below
- **f** is the format of time:
 - 0 = GPS time and
 - 1 = Unix time
- **ll.lllllllll** is the latitude in signed decimal degree
- **yyy.yyyyyyyy** is the longitude in signed decimal degree
- **aaa.aa** is the antenna altitude above/below GPS height in meters when receiving STL signals and MSL height when operating from the GNSS receiver
- **xxxxxxx** is the position in ECEF-X axis, in meter
- **yyyyyyyyyy** is the position in ECEF-Y axis, in meter
- **zzzzzzzzz** is the position in ECEF-Z axis, in meter
- **uu.uuuu** are the uncertainty represented as local level covariance for EE, NN, UU, EN, EU, and NU, respectively
- **b.b** is the age of position in seconds since last burst

- *s.s* is the age of position in seconds since last sensor update
- *p* is the position figure of merit (PFOM) of current solution
- *g* is the current geo-location model configured in “GPS:STL:GEOLOCation <STATIC|DYNAMIC>” on page 34 with 1 = STATIC.

GPS:STL:PJLTPPS <int> [0,255]

This command will configure the NMEA output message for the currently selected serial port receiving this command. This command instructs the device to send the NMEA standard string \$PJLTPPS every N seconds, with N in the interval [0,255]. The output message configured by this command is invalid until the receiver achieves a first fix.

This command has the following format:

```
RECEiver:PJLTPPS <int> [0,255]
```

The following command will query the state of PJLTPPS output for the serial port receiving this command:

```
RECEiver:PJLTPPS?
```

The PJLTPPS string has the following data format:

```
$PJLTPPS,v,ssssssssss,b,t,a.aaaaaa,r,ll,p,q*[checksum]
```

where

- *v* is the message version
- *ssssssssss* is the GPS time (number of seconds since GPS epoch in 1980)
- *b* is the number of seconds since last burst
- *t* is the number of seconds since last PPS timing update
- *a.aaaaaa* is the estimation of PPS accuracy in microseconds
- *r* is the PPS signal output ready status (1 = ready, 0 = not ready)
- *ll* is the leap second difference between GPS time and UTC time (in seconds)
- *p* is the pending leap second event (+# = positive leap second, 0 = no pending leap second -# = negative leap second)
- *q* is the current Time Figure of Merit (TFOM) value in the STL solution.

GPS:STL:PJLTRMC <int> [0,255]

This command will configure the NMEA output message for the currently selected serial port receiving this command. This command instructs the device to send the NMEA standard string \$PJLTRMC every N seconds, with N in the interval [0,255]. The output message configured by this command is invalid until the receiver achieves a first fix.

This command has the following format:

```
GPS:STL:PJLTPPS <int> [0,255]
```

GPS:STL:PJLTSTAT <int> [0,255]

This command will configure the NMEA output message for the currently selected serial port receiving this command. This command instructs the receiver to send the proprietary NMEA messages \$PJLTSTAT every N seconds, with N in the interval [0,255]. The GPS time in the \$PJLTSTAT output message is invalid until the receiver achieves a first fix.

This command has the following format:

```
GPS:STL:PJLTSTAT <int> [0,255]
```

The following command will query the state of PJLTSTAT output for the serial port receiving this command:

```
GPS:STL:PJLTSTAT?
```

The PJLTSTAT string has the following data format:

```
$PJLTSTAT,v,ssssssssss,b,s*[checksum]
```

where

- **v** is the message version
- **ssssssssss** is the GPS time (number of seconds since GPS epoch in 1980)
- **b** is the number of received bursts in last 60 seconds (bursts per minute, or BPM)
- **s** is the number of received strong (processed) bursts in last 60 seconds

GPS:STL:PJLTSVI <ON | OFF>

This command enables the passing of STL NMEA messages through the device.

This command has the following format:

```
GPS:PJLTSVI <ON | OFF>
```

The following command will query the state of this command for the serial port receiving this command:

```
GPS:PJLTSVI?
```

The PJLTSVI string has the following data format:

```
$PJLTSVI,v,s.sssssssss,wwww,i,cc.c,aa.a,ee.e,pp.p,pp.p,pp.p  
,vv.v,vv.v,vv.v,dd.d,rr.r,,,t.ttttttttt,0xVV*[checksum]
```

where

- **v** is the message version,
- **s.sssssssss** is the GPS time of week representing presumed time of validity (number of seconds since the beginning of the current GPS week),
- **wwww** is the GPS week (number of weeks since GPS epoch in 1980),
- **i** is the space vehicle ID number,

- `cc.c` is the CN0 (Carrier to Noise Ratio) of the burst associated with the SVI update, in dB-Hz,
- `aa.a` is the azimuth, in degrees
- `ee.e` is the elevation, in degrees
- `pp.p,pp.p,pp.p` are the SV position ECEF coordinates, in meters,
- `vv.v,vv.v,vv.v` are the SV velocity ECEF coordinates, in m/s,
- `dd.d` is the doppler, in Hz (approaching SV doppler is positive),
- `rrrrrrrrrrrrrr.r` is the satellite pseudo-range estimate, in meters,
- `t.tttttttttt` is the time of arrival of the burst associated with the SVI update, in seconds,
- `vv` is a hexadecimal value representing multiple validity flags for data contained within the current message, as described in [Table 7](#).

Table 7 SVI validity status

If...	Status
Pseudo-range is valid	Validity = 0x1;
SV Position is valid	Validity = 0x2;
SV Velocity is valid	Validity = 0x4;
Azimuth & Elevation are valid	Validity = 0x8;
Time of Validity is valid	Validity = 0x10;
SV ID is valid	Validity = 0x20;
Time of Arrival is valid	Validity = 0x40;

GPS:STL:PJLTVEL <int> [0,255]

This command will configure the NMEA output message for the currently selected serial port receiving this command. This command instructs the device to send the NMEA standard string \$PJLTVEL every N seconds, with N in the interval [0,255]. The output message configured by this command is invalid until the receiver achieves a first fix.

This command has the following format:

```
GPS:NMEA:SCPI:PJLTVEL <int> [0,255]
```

The following command will query the state of this command for the TTL SCPI serial port:

```
GPS:NMEA:SCPI:PJLTVEL?
```

The PJLTVEL string has the following data format:

```
$PJLTVEL,v,iiiiiiiiii,e.eeeeeee,n.nnnnnn,u.uuuuuu,sssssssss  
*[checksum]
```

where:

- `v` is the message version,

- `iiiiiiiiiii` is the serial number for the receiver described in “GPS:STL:SN?” on page 41
- `e.eeeeee` is the velocity in east direction, in m/s,
- `n.nnnnnnn` is the velocity in north direction, in m/s,
- `u.uuuuuuu` is the velocity in up direction, in m/s,
- `sssssssssss` is the GPS time (number of seconds since GPS epoch in 1980).

This query will return the current latitude and longitude in signed decimal degree, and altitude above/below GPS height in meters being reported from the STL or GNSS receiver. The GNSS receiver may also report the MSL height in meters.

GPS:STL:PPSFilter:PFOM <int> [1,9]

This command allows the user to configure the Position Figure of Merit (PFOM) threshold before the STL 1PPS output is enabled. The PFOM threshold is another way to set the Estimated Position Error (EPE) threshold. The default threshold value is 2 in which the default EPE threshold is < 50 meters before the STL 1PPS output is enabled after STL fix in static geo-location mode. PFOM values range from 1 to 9 and correspond to these industry-standard error estimates:

- 1: EPE < 25m
- 2: EPE < 50m
- 3: EPE < 75m
- 4: EPE < 100m
- 5: EPE < 200m
- 6: EPE < 500m
- 7: EPE < 1,000m
- 8: EPE < 5,000m
- 9: EPE >= 5,000m

This command has the following format:

```
GPS:STL:PPSFilter:PFOM [1,9]
```

GPS:STL:PPSFilter:TFOM <int> [1,9]

This command allows the user to configure the Time Figure of Merit (TFOM) threshold before the STL 1PPS output is enabled. The TFOM threshold is another way to set the Estimated Time Error (ETE) threshold. The default threshold value is 4 in which the default ETE threshold is < 1 microsecond before the STL 1PPS output is enabled after STL fix in static geolocation mode.

TFOM values range from 1 to 9 and correspond to these industry-standard error estimates:

- 1: EPE < 1ns
- 2: EPE < 10ns

- 3: EPE < 100ns
- 4: EPE < 1us
- 5: EPE < 10us
- 6: EPE < 100us
- 7: EPE < 1ms
- 8: EPE < 10ms
- 9: EPE >= 10ms

This command has the following format:

```
GPS:STL:PPSFilter:TFOM [1,9]
```

The following command will query the settings of this command:

```
GPS:STL:PPSFilter:TFOM?
```

GPS:STL:SN?

This command queries the STL board serial number.

GPS:STL:TRUEPOSition <double> <ll.llllllll,lll.llllllll,h.hh>

This command sets the true GPS position of the device.

The command has the following format:

```
GPS:STL:TRUEPOSition <double>  
<ll.llllllll,lll.llllllll,h.hh>
```

where:

- ll.llllllll,lll.llllllll is the latitude and longitude
- h.hh is the altitude.

GPS:STL:SUBScriptioN:ENDdate?

This command queries the end date of the STL subscription.

GPS:STL:SUBScriptioN:FEATure?

This query will return the current STL subscription features that are enabled. Subscription features include: timing, location, or combination of both features. Also refer to the activated features bit-field described in Table 5 for \$PJLTSUB message.

GPS:STL:SUBScription:STARTdate?

This command queries the STL subscription start date.

GPS:STL:SUBScription:STATus?

This command queries the status of the STL subscription.

GPS:STL:SUBScription:VALID?

This command queries the validity of the STL subscription.

GPS:XYZSPeed <int> [0,255]

This command is a 3D velocity vector output command. Enabling this command will output a 3 dimensional velocity vector indicating the unit's speed in centimeters per second in the ECEF coordinate system.

X, Y, and Z speed are individually given, and are independent of each other. An accuracy estimate in centimeters per second is also given. The velocity data is time-stamped using the time-of-week with a resolution of milliseconds.

Additionally, the number of accrued Leapseconds and GPS time of week is indicated in this message, which allows proper calculation of GPS time from UTC time as indicated by other messages, as well as proper handling of Leapsecond events.

Use the following format to generate the velocity vector every N seconds, with N in the interval [0,255]:

```
GPS:XYZSPeed <int> [0,255]
```

This command will query the state of this command:

```
GPS:XYZSPeed?
```

The XYZSP string has the following data format:

```
$XYZSP VX VY VZ ... Leapseconds: aaaaaa bbbbbb ccccc ddd  
eeeeeeee ff
```

where

- aaaaa is the velocity in the x direction (cm/s)
- bbbbbb is the velocity in the y direction (cm/s)
- ccccc is the velocity in the z direction (cm/s)
- ddd is the speed accuracy estimate (cm/s)
- eeeeeeeee is the GPS TOW in seconds
- ff is the current UTC leap second offset in seconds

GPS:GPZDA <int> [0,255]

This command instructs the PNT-440 device to send the NMEA standard ZDA message every N seconds, with N in the interval [0,255]. The command is disabled until the GNSS receiver achieves a first fix.

This command has the following format:

```
GPS:GPZDA <int> [0,255]
```

This command will query the state of this command:

```
GPS:GPZDA?
```

The GPZDA string has the following data format:

```
$GPZDA,hhmmss.00,dd,mm,yyyy,+00,00*[checksum]
```

The ZDA output message header includes the talker ID for the currently enabled GNSS system or systems as described in [“NMEA Support” on page 36](#).

GPS:PASHR <int> [0,255]

The NMEA string \$PASHR,POS has been added for compatibility to legacy GPS hardware. The PASHR command alongside the GPZDA command will give all relevant parameters such as time, date, position, velocity, direction, altitude, quality of fix, and more. As an example, the String has the following data format:

```
$PASHR,POS,0,12,191512.00,3610.11157,N,11518.89941,W,00887.70,????,000.00,000.0 1,-00.00,00.0,00.8,00.0,00.0,0.46*21
```



NOTE

The length of the string is fixed at 115 characters, plus the two binary 0x0d, 0x0a termination characters.

```
$PASHR,POS,0,aa,bbbbbb.00,cccc.ccccc,d,eeee.eeeee,f,ggggg.gg,hhhh,iii.ii,jjj.jj,kkkk.kk,ll.l,mm.m,nn.n,00.0,p.pp*[checksum]
```

Where:

- aa: Number of Sats
- bbbbbbb.00: Time of Day UTC
- cccc.ccccc, d: Latitude,S/N
- eeee.eeeee, f: Longitude,W/E
- ggggg.gg: Antenna Height in meters
- hhhh: Four fixed '?' symbols
- iii.ii: Course Over Ground
- jjj.jj: Speed in Knots
- kkkk.k: Vertical Velocity in meters/s

- 11.1: PDOP
- mm.m: HDOP
- nn.n: VDOP
- 00.0: Static number
- p.pp: Firmware Version

This command instructs the PNT-440 device to send the NMEA standard string \$PASHR every N seconds, with N in the interval [0,255]. The command is disabled until the GNSS receiver achieves a first fix.

This command has the following format:

```
GPS:PASHR <int> [0,255]
```

This command will query the state of this command:

```
GPS:PASHR?
```

GPS:PJLTS <int> [0,255]

This command instructs the PNT-440 device to send the proprietary NMEA string \$PJLTS every N seconds, with N in the interval [0,255]. This proprietary command includes information on the GNSSDO lock, oscillator status, and other telemetry status. The following is an example of the PJLTS sentence:

```
$PJLTS,3.39,3.62,21341,6,2.4627123,82.0904,1.3E-  
12,0,12,0x0*52
```

This command has the following format:

```
GPS:PJLTS <int> [0,255]
```

This command will query the state of this command:

```
GPS:PJLTS?
```

The format of the \$PJLTS command is:

```
$PJLTS,aaa.aa,bbb.bb,ccc,d,e.aaaaaaa,ff.ffff,g.gEhh,iii,jj,  
kkkk*[checksum]0x0d 0x0a
```

where

- aaa.aa: the filtered UTC offset in ns,
- bbb.bb: the raw UTC offset in ns from the time interval counter,
- ccc: the number of captured input 1PPS signals
- d is the lock status
- e.aaaaaaa is the EFC voltage,
- ff.ffff is the EFC percentage (0% to 100%),
- g.gEhh is the estimated frequency accuracy (similar to 100s ADEV) in scientific notation,
- iii is the seconds in holdover,

- *jj* is the number of satellites tracked, and
- *kkkk* is the health status explained in the SYNC:HEALTH? query shown in “SYNChronization:HEAlth?” on page 74.

GPS:DYNAMIC:MODE <int> [0,8]

This command allows the user to select the dynamic motion model being applied to the Kalman filters in the GNSS receiver ONLY. This allows for larger amounts of filtering for lower velocity applications, effectively reducing noise and multipath interference. Applications with high acceleration can now be used with fast filter settings to allow for the most accurate GPS coordinates to be provided in high-dynamic applications such as Jet aircraft. Doppler tracking is enabled in all airborne modes, as Carrier Phase tracking is very difficult to achieve in high velocity applications. The GNSS receiver will perform Carrier Phase tracking for non-airborne modes.

The command has the following syntax:

```
GPS:DYNAMIC:MODE <int> [0,8]
```

Sending the following command to the PNT-440 device will select a stationary GNSS dynamic model, for example:

```
GPS:DYNAM:MODE 1
```

[Table 11](#) lists all available dynamic modes.

Table 8 Supported Dynamic GNSS operating modes

Value	Model	Application
0	Portable	Recommended as a default setting
1	Stationary	Used in stationary applications
2	Pedestrian	Used in man-pack, pedestrian settings
3	Automotive	Vehicular velocity applications
4	Sea	Used on ships, where altitude is expected to be constant
5	Airborne <1g	Airborne applications with less than 1g acceleration
6	Airborne <2g	Airborne applications with less than 2g acceleration
7	Airborne <4g	Airborne applications with less than 4g acceleration
8	Automatic Mode	Selects one of the above states (0-7) based on the actual velocity of the vehicle

The current dynamic state being applied to the GNSS receiver can be queried with the command `GPS:DYNAMIC:STATE?`.



NOTE
This command syntax has changed from previous products such as the FireFly-IIA GPSDO units, which did not support Position Hold Auto Survey modes.

GPS:DYNAMIC:MODE 8 (Automatic Dynamic Mode)

Automatic Dynamic Mode allows the PNT-440 device GNSDO firmware to automatically configure the GNSS receiver Kalman filter parameters based on actual mission velocities and motion profiles. The unit will try to set the GNSS receiver to the optimal setting for any given velocity. The unit is able to set 7 different modes, as shown in “`GPS:DYNAMIC:MODE <int> [0,8]`” on page 43.

Table 12 shows the Dynamic mode the unit will program into the GNSS receiver when Automatic Mode is selected (Dynamic Mode 8).

Table 9 Auto Dynamic Mode switching rules

Velocity Threshold	Selected Dynamic Model	Fallback to lower setting
0-2 knots	Stationary	none
>2 knots	Pedestrian	<1 knots
>10 knots	Automotive	<8 knots
>60 knots and >400 Feet/min climb/descent	Airborne 1g	<50 knots
>150 knots	Airborne 2g	<130 knots
>240 knots	Airborne 4g	<210 knots

In this Automatic mode, the unit will configure the GNSS receiver based on the actual vehicle-velocity.



NOTE
In order to switch from the Automotive mode into the first Airborne (1g) mode, both a vehicle velocity greater than 60 knots as well as a climb/descent rate greater than 400 feet per minute are required. Alternatively, a vehicle velocity of greater than 100 Knots will also initiate a switch into airborne-1g mode.

Without an appropriate climb/descent, the unit will remain in Automotive mode unless 100 knots of velocity are exceeded.

The following command returns the setting of the GNSS dynamic model:

```
GPS:DYNAMIC:MODE?
```

The actual state chosen by the firmware for the GNSS receiver based on vehicle velocity can be obtained with the command:

```
GPS:DYNAMIC:STATE?
```

A value between 0 and 7 is then returned depending on vehicle dynamics. Settings will be applied immediately to the GNSS receiver, and are stored in Non-Volatile memory.

GPS:DYNAMIC:MODE?

This command returns the setting of the GNSS dynamic mode.

GPS:DYNAMIC:STATE?

This query returns the actual state of the dynamic model, chosen by the firmware to be applied to the GNSS receiver depending on vehicle velocity. It returns a value between 0 and 7, which correspond to one of the dynamic models defined in [Table 11 on page 43](#).

This state can be different from the user-selected Dynamic model mode for two reasons:

- If the dynamic mode is set to 8 (Automatic mode), the state will reflect the dynamic model being applied to the GNSS receiver depending on actual vehicle dynamics
- If the GPS Timing Mode is set to ON or to RSTSURV, the dynamic state will always be set to 1 (Stationary)

GPS:REFERENCE:ADELAY <float> <s | ns> [-32767ns,32767ns]

The ADElay command allows bi-directional shifting of the 1PPS output in relation to the UTC (USNO) 1PPS reference in one nanosecond steps for GNSS receiver ONLY. This allows antenna cable delay compensation, as well as retarding or advancing the 1PPS signal arbitrarily. Typical antenna delays for a 30 foot antenna cable with 1.5ns per foot propagation delay would be compensated with the following command:

```
GPS:REF:ADEL 45ns
```

This command can be used to fine-tune different units to have co-incident 1PPS signal outputs.



NOTE

During normal operation, the 1PPS signal may wander around the UTC 1PPS signal while the unit is tracking the GNSS signals. This present offset between the 1PPS output and the UTC 1PPS signal can be queried with the command `SYNC:TINT?`. This offset should be taken into account when calibrating the two units' 1PPS outputs to each other, as the lock algorithms will try to steer the OCXO for a 0.0ns offset to the UTC 1PPS time-pulse.

Change in this command will be effective after the next power cycle or after the `GPS:RESET ONCE` command is sent.

This command has the following format:

```
GPS:REFeRence:ADELay [-32767,32767]
```

This command will query the state of this command:

```
GPS:REFeRence:ADELay?
```

GPS:REFeRence:PULse:SAWtooth?

This command returns the momentary sawtooth correction factor that the GNSS receiver indicated.

GPS:RESET ONCE

This command issues a reset to the internal GNSS receiver. This can be helpful when changing the antenna for example, since the GNSS receiver measures the antenna system's C/No right after reset, and adjusts its internal antenna amplifier gains accordingly. It takes approximately 1 minute for locking to commence after a GNSS receiver reset, as indicated by the red blinking LED.

GPS:TMODe <ON|OFF|RSTSURV>

This command selects the Timing Mode of the SEPTENTRIO receiver when SEPTENTRIO type is selected in `GPS:TYPE?` settings. The Timing Mode is not currently supported for other receivers in PNT-440 device.

If the Timing Mode is OFF, the receiver will act as a regular GNSS receiver in 3D mobile mode. This mode has to be chosen if the unit is used with a moving antenna.

If the Timing Mode in ON, the timing features of the GNSS receiver are enabled. At power-up, the Hold position stored in NVRAM will be sent to the GNSS receiver and will be used as the reference. In order to use this mode, the receiver position must be

known as exactly as possible. Errors in the Hold position will translate into time errors depending on the satellite constellation.

The Hold position can be set manually by the user or can be the result of a position Auto Survey executed by the GNSS receiver.

If the Timing Mode is set to RSTSURV, the GNSS receiver will start an Auto Survey every time the unit is powered-on and following the Survey sequence, the GNSS receiver will run with the timing features enabled. Once in Position Hold mode, the antenna location should be held completely stationary.

This command has the following format:

```
GPS:TMODE <ON|OFF|RSTSURV>
```

The following command will query the settings of this command:

```
GPS:TMODE?
```

GPS:SURVey ONCE

The Timing Mode is not currently supported in PNT-440 device. This command starts a Survey. At the end of the Survey, the calculated Hold position will be stored in NVRAM. The Survey parameters can be set with the `GPS:SURVey:DURation <sec>` and `GPS:SURVey:VARIANCE <mm^2>` commands.

The GNSS receiver will stop the Survey when the minimal duration has been reached and the variance of the average position is under the specified minimum variance.

GPS:SURVey:DURation <sec>

This command sets the Survey minimal duration. This minimum value is used as a threshold under which the GNSS receiver can stop the Survey.

The GNSS receiver will stop the Survey when the minimal duration has been reached and the variance of the average position is under the specified minimum variance.

This command has the following format:

```
GPS:SURVey:DURation < int in sec >
```

This command will query the state of this command:

```
GPS:SURVey:DURation?
```

GPS:SURVey:VARIANCE <mm^2>

This command specifies the minimum variance of the average position computed during the Survey. This minimum value is used as a threshold under which the GNSS receiver can stop the Survey.

The GNSS receiver will stop the Survey when the minimal duration has been reached and the variance of the average position is under the specified minimum variance.

This command has the following format:

```
GPS:SURVey:VARiance <int>
```

This command will query the state of this command:

```
GPS:SURVey:VARiance?
```

GPS:HOLD:POSition <int, int, int>

This command allows the user to specify manually the exact position of the unit. This command will overwrite the Hold position in NVRAM. Subsequent Survey also will overwrite the Hold position. The Hold position is stored in ECEF coordinates.

This command has the following format:

```
GPS:HOLD:POSition <int,int,int>
```

This command will query the state of this command:

```
GPS:HOLD:POSition?
```

GPS:SURVEY:STATUS?

This query displays the current status of the survey. The status of the survey is in one of the 3 states:

- **ACTIVE:** a survey is in progress
- **VALID:** a survey has been achieved successfully and the GNSS receiver is now using this Hold position as reference.
- **INVALID:** no survey are in progress or have been achieved since the last power cycle

When in ACTIVE or VALID state, this query will also display the duration, the Hold position in ECEF coordinates and the position variance.

GPS:INITial:DATE <yyyy,mm,dd>

This command allows setting the internal RTC DATE manually when operating the unit in GNSS denied environments. This command is compatible to the `PTIME:OUTput ON` command available in CSAC based products to allow automatic time and date synchronization of two units to each other. The internal RTC is driven by the highly stable ovenized oscillator 10MHz signal, and thus has very high time accuracy.

If the internal RTC DATE has set itself to 1/1/2010 and the internal RTC TIME has set itself to 00:00:00 upon power-up, this indicates the clock is outdated and the user must wait for valid UTC data from the receiver or modify the time to a valid one.

This command has the following format:

```
GPS:INITial:DATE <yyyy,mm,dd>
```

This command will query the state of this command:

```
GPS:INITial:DATE?
```

GPS:INITial:TIME <hour,min,sec>

This command allows setting the internal RTC TIME manually when operating the unit in GPS denied environments. This command is compatible to the `PTIME:OUTput ON` command available in CSAC based products to allow automatic time and date synchronization of two units to each other. The internal RTC is driven by the highly stable ovenized oscillator 10MHz signal, and thus has very high time accuracy.

If the internal RTC DATE has set itself to 1/1/2010 and the internal RTC TIME has set itself to 00:00:00 upon power-up, this indicates the clock is outdated and the user must wait for valid UTC data from the receiver or modify the time to a valid one.

This command has the following format:

```
GPS:INITial:TIME <hour,min,sec>
```

This command will query the state of this command:

```
GPS:INITial:TIME?
```

GPS:RTC:DATE <yyyy,mm,dd>

This command allows setting the hardware RTC DATE manually when operating the unit in GNSS denied environments. The on-board super-capacitor C63 will keep the hardware RTC clock running for additional 1+ days after removing the PNT-440 device main power supply. The PNT-440 device will verify the hardware RTC date and time and assign initial internal RTC DATE if valid.

This command has the following format:

```
GPS:RTC:DATE <yyyy,mm,dd>
```

This command will query the state of this command:

```
GPS:RTC:DATE?
```

GPS:RTC:TIME <hour,min,sec>

This command allows setting the hardware RTC TIME manually when operating the unit in GNSS denied environments. The on-board super-capacitor C63 will keep the hardware RTC clock running for additional 1+ days after removing the PNT-440 device main power supply. The PNT-440 device will verify the hardware RTC date and time and assign initial internal RTC DATE if valid.

This command has the following format:

```
GPS:RTC:TIME <hour,min,sec>
```

This command will query the state of this command:

```
GPS:RTC:TIME?
```

GPS:SYSTem:SElect [GPS | SBAS | QZSS | GAL | BD ^ GLO]

This command selects the GNSS systems that are enabled in the GNSS receiver and are used to generate the timing and positioning information for the NMEA data, and to generate the 1PPS reference for the GNSDO. The command is followed by a list of the shortened names of the GNSS systems to enable. The shortened names of supported GNSS system include GPS, SBAS, QZSS, BD (BeiDou), GAL (Galileo) and GLO (GLONASS). Invalid combinations of GNSS systems will result in a Command Error response and no change to the configuration.

The following example command will enable GPS, SBAS, Galileo and GLONASS all concurrently:

```
GPS:SYST:SEL GPS SBAS GAL GLO
```

The following command will query the currently enabled GNSS systems:

```
GPS:SYST:SEL?
```

The PNT-440 device will respond to the query with the list of enabled GNSS systems such as:

```
GPS SBAS GAL GLO
```

GPS:SYSTem:LBAND [L1 | L2 | L3 | L5 | ALL]

This command configures the L band signal usage in PVT solution of SEPTENTRIO receiver when SEPTENTRIO type is selected in `GPS:TYPE?` settings. Configurable L band signals include L1, L2, and L5 for GPS, L1 and L5 for Galileo, BeiDou, SBAS, and QZSS constellations, L1, L2, and L3 for GLONASS.

The SEPTENTRIO receiver will generate PVT solution with PVT data from enabled L band signals for current enabled constellations in “[GPS:SYSTem:SElect \[GPS | SBAS | QZSS | GAL | BD ^ GLO\]](#)” on page 50. Invalid combinations of L band configuration and `GPS:SYSTem:SElect [GPS | SBAS | QZSS | GAL | BD ^ GLO]` settings will result in a Warning response and no change to the configuration.

The following example command will enable all available L band signals for PVT solution:

```
GPS:SYST:LBAND ALL
```

The following example command will enable signals from L1 and L5 bands for PVT solution:

```
GPS:SYST:LBAND L1 L5
```

The following command will query the currently enabled L band signals for PVT solution:

```
GPS:SYST:LBAND?
```

GPS:SYSTem:LBAND:AGCgain?

This command queries the ETA system and returns AGC Gain in Db.

GPS:SYSTem:LBAND:AGCPnorm?

This command queries the ETA system and returns the Euclidean norm of the AGC gain for all L-band trackers.

GPS:SYSTem:LBAND:SVID?

This command returns a list of Space Vehicle Identifiers (SVIDs) for each L-band tracker.

GPS:SYSTem:LBAND:CNO?

This command returns a list of signal power to measurement noise power density ratios (C/No) for each L-band tracker.



NOTE

Health Status will show 0x10000 when L-band beams used for ETA have C/No of less than 26.00 dB Hz.

GPS:SYSTem:BANDTRACK [L1 | L2 | L3 | L5 | ALL]

This command configures the L band signals to be tracked by the SEPTENTRIO receiver when SEPTENTRIO type is selected in `GPS:TYPE?` settings. Configurable L band signals include L1, L2, and L5 for GPS, L1 and L5 for Galileo, BeiDou, SBAS, and QZSS constellations, L1, L2, and L3 for GLONASS.

The SEPTENTRIO receiver will track enabled L band signals for satellites from current enabled constellations in `GPS:SYSTem:SElect [GPS | SBAS | QZSS | GAL | BD ^ GLO]` on page 50. Invalid combinations of L band configuration and

GPS:SYSTem:SElect [GPS | SBAS | QZSS | GAL | BD ^ GLO] settings will result in a Warning response and no change to the configuration.



NOTE

Tracking of GLONASS L3 band requires enabling of GLONASS L1 band as well.

The following example command will enable all available L band signals for satellite tracking:

```
GPS:SYST:BANDTRAC ALL
```

The following example command will enable satellite tracking of L5 band signals only if current enabled constellations in “GPS:SYSTem:SElect [GPS | SBAS | QZSS | GAL | BD ^ GLO]” on page 50 support L5 band:

```
GPS:SYST:BANDTRAC L5
```

The following command will query the currently enabled L band signals for satellite tracking:

```
GPS:SYST:BANDTRAC?
```

GPS:SYSTem:TIMESOURce <GPS|GAL|BD|AUTO>

This command selects the timing source for computation of receiver clock bias comparing to SEPTENTRIO receiver time when SEPTENTRIO type is selected in GPS:TYPE? settings. This command also determines the primary GNSS constellation in PVT solution.



NOTE

No PVT solution will be computed if insufficient satellites of selected timing resources are visible and tracked by the receiver.

The default timing source is GPS. Under GPS-denied challenged environment, changing to GAL for Galileo system time or BD for Beidou system time is recommended for potential alternative PVT solution.

The GPS:SYST:TIMESOUR AUTO settings will switch the timing source automatically based on the best available source.

The following example command will enable Galileo system time for primary system in PVT solution:

```
GPS:SYST:TIMESOUR GAL
```

The following command will query the currently selected timing source in PVT solution:

```
GPS:SYST:TIMESOUR?
```

GPS:SYSTem:TIMESOURce:STATe?

This command queries the current GNSS time source. The response will indicate the GNSS time source currently selected. The response will match the `GPS:SYSTem:TIMESOURce` setting except when configured in AUTO time source.

GPS:SYSTem:ETA <ON|OFF>

This command enables and disables Enhanced Timing and Authentication (ETA.)

GPS:SYSTem:ETA?

This command queries the ETA service and returns ETA status, including timing system used and status of satellites (can indicate that satellites and constellations are lost due to spoofing.)

GPS:SYSTem:UPDATERATe <Hz> [1,5,10,20]

This command allows to configure the systems update rate for GNSS. The default for release 1.6 is 10 Hz and it is recommended not to change it to allow maximum performance of the system.

GPS:SPOOF?

This command queries the ETA system and returns the current spoofing status. A value of:

- 0 indicates no spoofing has been detected
- 1 indicates spoofing was detected with the standard algorithm
- 2 indicates spoofing was detected by the ETA algorithm
- 3 indicates spoofing was detected by both the standard and ETA algorithms

GPS:HEIGHT?

This command returns the output from the following queries:

`GPS:HEIGHT:MSL?`

`GPS:HEIGHT:GPS?`

GPS:HEIGHT:MSL?

This query returns the Mean Sea Level height in meters which differs from the GPS ellipsoid height by up to +/-100 meters. This feature is not supported for STL receiver mode. This difference varies depending upon the actual antenna location. The MSL to GPS height differences are calculated by the GNSS receiver.

GPS:HEIGHT:GPS?

This query returns the height above the GPS ellipsoid in meters reported by the GNSS or STL receiver.

GPS:POLarity <LOW|HIGH>

This command will set the Septentrio GNSS receiver's starting polarity level for GNSS 1PPS output. This command is not supported for the uBlox GNSS receiver option or the STL receiver mode.

The default setting is LOW where the GNSS 1PPS output will trigger from low to high level to UTC time. A low to high polarity indicates a rising edge, and a high to low polarity indicates a falling edge. The selected rising or falling edge/polarity will be aligned to UTC time.

This command has the following format:

```
GPS:POLarity <LOW|HIGH>
```

This command will query the state of this command:

```
GPS:POLarity?
```

GPS:POSition?

This query will return the `GPS:GPRMC` latitude, longitude, the speed over ground in knots, and the course made good check. The course made good check indicates the direction a vehicle is traveling independent of the direction the vehicle is pointing. This query will also return the `GPS:GPGGA` antenna height in MSL Meters and the antenna height in GPS Meters for GNSS receiver mode. This query will return the GPS Meters ONLY for STL receiver mode.

GPS:POSition:ECEF?

This query will return the earth-centered, earth-fixed X,Y, and Z coordinates as well as the position accuracy of the GNSS receiver. This query is not currently supported for STL receiver mode.

GPS:CNOthres <int> [1,40]

This command allows the user to set a carrier to noise threshold for GNSS receiver ONLY. This sets up a navigation solution that will only be attempted if there exists the given number of SVs with signals that are, at minimum, as strong as the given threshold.

This command has the following format:

```
GPS:CNOthres [1,40]
```

This command will query the state of this command:

```
GPS:CNOthres?
```

GPS:FOV <int> [3,85]

This command allows the user to set the minimum elevation limit of a satellite above the horizon in order to be used in the GNSS navigation solution ONLY. Low elevation satellites may provide degraded accuracy, due to the long signal path through the atmosphere.

This command has the following format:

```
GPS:FOV [3,85]
```

This command will query the state of this command:

```
GPS:FOV?
```

GPS:STATus?

This query returns the status of the GNSS or STL receiver. For GNSS receiver mode, a status of 1 means a fix is not available. A status of 2 indicates a 2D fix has been achieved and a status of 3 indicates a 3D fix has been achieved. For STL receiver mode, a status of 1 or higher indicates a 3D fix.

GPS:STATus:ETA?

This query returns details of spoofing. Potentially spoofed satellites appear in “active satellites” but do not appear as “authentic” satellites.

GPS:STATus:STRing?

This query returns the status of the GNSS and STL receiver as a string. The receiver may have a status of “No Fix”, “2D Fix”, or “3D Fix”.

GPS:PASSthru

<GNSS|RSRSCPI|STL|TOD|CSAC|RSRGPS|MCE|RSRDEB UG>

This command allows the user to select a source for pass-through mode for debugging ONLY. Pass-through mode will allow one of these sources to pass data through their respective COM ports. The GNSS pass-through command will allow either uBlox or Septentrio data to pass through depending on the GNSS receiver connected.

This command has the following format:

```
GPS:PASSthru <GNSS|RSRSCPI|STL|TOD|CSAC|RSRGPS|MCE|RSRDEB UG>
```

This command will query the state of this command:

```
GPS:PASSthru?
```

GPS:PASSthru:TIMEout <ON|OFF>

This command will enable or disable the time-out feature of the pass-through mode. Setting this command to "ON" will cause the pass-through mode to time-out after the period specified in ["GPS:PASSthru:COUNT <int> <s> \[1,86400\]" on page 56](#). Setting this command to "OFF" will cause the pass-through mode to disable the time-out feature.

The default settings is ON which enables the time-out feature. See ["GPS:SATellite:VISible:COUNT?" on page 36](#) for the command to configure the length of time-out mode.

This command has the following format:

```
GPS:PASSthru:TIMEout <ON|OFF>
```

This command will query the state of this command:

```
GPS:PASSthru:TIMEout?
```

GPS:PASSthru:COUNT <int> <s> [1,86400]

This command will set length of time the device remains in pass-through time-out mode described in ["GPS:PASSthru:TIMEout <ON|OFF>" on page 56](#). The time entered should be in seconds and the default period is 60 seconds.

This command has the following format:

```
GPS:PASSthru:COUNT [1,86400]
```

This command will query the state of this command:

```
GPS: PASSthru:COUNT?
```

GPS:TYPE?

This query will return the current detected or selected receiver source. The possible receiver types are:

- AUTO DETECTING
- UBLOX
- SEPTENTRIO
- STL

GPS:TYPE:MODE <AUTO|UBLOX|SEPTentrio|STL>

This command will override the current receiver mode settings to the user selected mode. The default receiver mode is STL for order option with mounted STL-2600™ board.

The query command GPS? will display the current receiver mode and status for the current receiver mode. Some NMEA outputs and SCPI query commands will also reflect on the current receiver mode selected.

This command has the following format:

```
GPS:TYPE:MODE <AUTO|UBLOX|SEPTentrio|STL>
```

This command will query the state of this command:

```
GPS:TYPE:MODE?
```

GPS:SOURce:SElect <AUTO|STL|GNSS>

This command is not currently supported in PNT-440 device.

GPS:FILTer:SElect <STL|UBX>

This command is not currently supported in PNT-440 device.

GPS:FILTer:NMEA <int> [0,255]

This command is not currently supported in PNT-440 device.

GPS:GNSS:MODE <ON|OFF>

This command is not currently supported in PNT-440 device.

GPS:GNSS:RESET ONCE

This command will issue a restart sequence for the uBlox/Mosaic GNSS receiver. The uBlox/Mosaic GNSS receiver will be initialized to its default settings after the reset.

GPS:GNSS:POWer <ON|OFF>

This command will power on/power down the uBlox/Mosaic GNSS receiver. When the uBlox/Mosaic is powered on from off settings, it will conduct a cold start and initialize the uBlox/Mosaic. Use OFF settings to power down the GNSS receiver.



NOTE

When the Mosaic receiver is powered on there is a chance the receiver will enter its software upgrade mode and will wait for the user to upload its firmware. Alternatively, the Mosaic receiver will exit this upgrade mode after 200 seconds if there is no valid response from the user. The user may expect a 200 second delay before the receiver status in GPS? changes from Reserved to No fix/3D Fix.

This command has the following format:

```
GPS:GNSS:POWer <ON|OFF>
```

This command will query the state of this command:

```
GPS:GNSS:POWer?
```

GPS:GNSS:FWver?

This query prints the firmware version of the GNSS receiver.

GPS:GNSS:TYPE?

This command is not supported on the PNT-440 device.

GPS:STL:RESET <ONCE | WARMUP|HOLD|OFF>

This command will configure the reset state of STL-2600™ board if mounted.

The ONCE settings will issue a reset sequence for the STL-2600™ board once. The HOLD settings will keep and hold the STL-2600™ board on reset.

The following command will query the state of this command:

```
GPS:STL:RESET?
```

GPS:STL:RESET:AUTO <int> [0,604800]

This command will configure the reset state of STL-2600™ board if mounted.

This command adjusts the length of time during which the FASTLOCK feature is active, please see [“SERVo:FASTlock <int> \[1,20\]” on page 98](#).

The length can be set from 100 seconds to 20000 seconds. The Dynamic FASTLOCK gain is slowly reduced until it reaches a gain of 1.0 after the FALEngth period of seconds. During this time the PLL loop gain is increased by the amount specified in the `SERVo:FASTlock` parameter, which will result in a faster initial phase lock to UTC after power-on, while giving the lowest possible noise floor (best short term stability) during normal operation.

This command has the following format:

```
GPS:STL:RESET:AUTO <int> [0,604800]
```

This command will query the state of this command:

```
GPS:STL:RESET?
```

GPS:STL:SCPI <ASCII> <command>

This command will allow the user to send any ASCII command to the integrated STL-2600™ module, if present on the PNT-440 device. Note that the PNT-440 device may overwrite the settings on next connection for settings changed with this command.

This command has the following format:

```
GPS:STL:SCPI <ASCII> <command>
```

GPS:STL:ISP ON

This command will send the STL-2600™ board into ISP mode for reprogramming. Refer to Chapter 4 of the STL-2600™ User Manual for programming instructions.

GPS:STL:BPMProcessed?

This query will return the number of processed STL bursts in last 60 seconds on the STL-2600™ board if mounted. Typical values range from 0 to 400 depending on the quality of STL signal reception at the time of query.

GPS:STL:BPMReceived?

This query will return the number of received STL bursts in last 60 seconds on the STL-2600™ board if mounted. Typical values range from 0 to 400 depending on the quality of STL signal reception at the time of query.

GPS:STL:PFOM?

This query will return the current Position Figure of Merit (PFOM) value in range of 1 to 9 for STL-2600™ solution.

GPS:STL:TFOM?

This query will return the current Time Figure of Merit (TFOM) value in range of 1 to 9 for STL-2600™ solution.

GPS:STL:EXTCLOCK <ON|OFF>

This command will toggle the state of CSAC_LOCK pin (8 on the STL-2600™ 20-pin connector) and enable/disable the use of the external clock (10 MHz) input to the STL-2600™ board if mounted. The state (on/off) of LED D18 on the STL-2600™ board indicates if the external clock is enabled/disabled, respectively.

For testing purpose, this command can be configured to OFF to disable the external clock reference into the STL-2600™ board. The settings in `GPS:STL:SERVO:LOOP <ON|OFF>` command can also be ON to enable the internal servo loop on the STL-2600™ board for disciplining.

This command has the following format:

```
GPS:STL:EXTCLOCK <ON|OFF>
```

The following command will query the settings of this command:

```
GPS:STL:EXTCLOCK?
```

GPS:STL:AGC <int> [500,2700]

This command will configure the manual RF Analog Gain Control (AGC) for the STL receiver.

Sending minimum value of 500 mV in this command results in maximum analog gain. The maximum value of 2700 mV results in minimum analog gain, with a nearly linear change of gain for intermediate values. Use this command to compensate for excessive cable losses or indoor reception with low signal levels, or non-amplified antennae.

Observe the number of processed bursts per minute (BPM) with `GPS:STL:BPMPROCESSED?` command after changing the analog gain, and slowly adjust the value to achieve maximum number of processed BPM. A typical value when using the Tallysman 2346A amplified antenna with short antenna cabling is 1350 mV.

This command has the following format:

```
GPS:STL:AGC <int> [500,2700]
```

The following command will query the settings of this command:

GPS:STL:AGC?

GPS:STL:CNO?

This query command responds with the maximum STL CNO value observed since power on.

GPS:STL:CNO:MINTHReshold <float> [1.0,100.0]

This command allows the user to configure the minimum carrier to noise (C/No) threshold for any burst to be further processed in the STL solution. The default minimum C/No threshold value is 40.0 dB.

All bursts used in STL solution will need to have C/No value above the value configured in this command.

This command has the following format:

```
GPS:STL:CNO:MINTHReshold <float> [1.0,100.0]
```

The following command will query the settings of this command:

```
GPS:STL:CNO:MINTHReshold?
```

GPS:STL:DOPpler:MINimum <double> [-36000.000000,0.0]

This command allows the user to configure the minimum doppler value for any burst to be further processed in the STL solution. The value must be between -36,000.000000 Hz and 0.0 Hz.

All bursts used in STL solution will need to have doppler value inside the range of minimumthreshold configured in this command and maximum threshold configured in GPS:STL:DOPpler:MAXimum <double> [0.0,36000.000000] command.

This command has the following format:

```
GPS:STL:DOPpler:MINimum <double> [-36000.000000,0.0]
```

The following command will query the settings of this command:

```
GPS:STL:DOPpler:MINimum?
```

GPS:STL:DOPpler:MAXimum <double> [0.0,36000.000000]

This command allows the user to configure the maximum Doppler value for any burst to be further processed in the STL solution. The value must be between 0.0 Hz and 36,000.000000 Hz.

All bursts used in STL solution will need to have Doppler value inside the range of minimum threshold configured in `GPS:STL:DOPpler:MINimum <double> [-36000.000000,0.0]` command and maximum threshold configured in this command.

This command has the following format:

```
GPS:STL:DOPpler:MAXimum <double> [0.0,36000.000000]
```

The following command will query the settings of this command:

```
GPS:STL:DOPpler:MAXimum?
```

GPS:STL:CLOCKmodel <double> [1E-20,1.0]

This command allows the user to choose a clock model for the integrated STL-2600™ module. A smaller numeric value would instruct the STL receiver to expect higher stability in reference clock.



NOTE

The consequence to stability of configuring a somewhat optimistic, i.e. thinking the reference clock can maintain higher stability than its actual specification, clock model is significantly worse than if configuring the clock model somewhat pessimistic.

This command has the following format:

```
GPS:STL:CLOCKmodel <double> [1E-20,1.0]
```

The following command will query the settings of this command:

```
GPS:STL:CLOCKmodel?
```

GPS:STL:CLOCKmodel:FAST <double> [1E-20,1.0]

This command allows the user to choose a clock model for the integrated STL-2600™ module. A smaller numeric value would instruct the STL receiver to expect higher stability in reference clock.



NOTE

The consequence to stability of configuring a somewhat optimistic, i.e. thinking the reference clock can maintain higher stability than its actual specification, clock model is significantly worse than if configuring the clock model somewhat pessimistic.

This command has the following format:

```
GPS:STL:CLOCKmodel:FAST <double> [1E-20,1.0]
```

The following command will query the settings of this command:

GPS:STL:CLOCKmodel:FAST?

GPS:STL:CLOCKmodel:MEDium <double> [1E-20,1.0]

This command allows the user to choose a clock model for the integrated STL-2600™ module. A smaller numeric value would instruct the STL receiver to expect higher stability in reference clock.



NOTE

The consequence to stability of configuring a somewhat optimistic, i.e. thinking the reference clock can maintain higher stability than its actual specification, clock model is significantly worse than if configuring the clock model somewhat pessimistic.

This command has the following format:

```
GPS:STL:CLOCKmodel:MEDium <double> [1E-20,1.0]
```

The following command will query the settings of this command:

```
GPS:STL:CLOCKmodel:MEDium?
```

GPS:STL:SUBScription?

This query returns the subscription information for the integrated STL-2600™ module.

GPS:STL:SUBScription:AUTHKEY <string>

This command allows the user to configure an authorization key for the STL receiver. The Authorization key has different formats for Releases 4.x and below (SDPM1) and Releases 5.x and above (SDPM3).

Release 4.x



NOTE

The string entered should be 28 characters long unless otherwise specified.

This command allows the user to configure an authentication key for the STL receiver. The string entered should be 28 characters long unless otherwise specified.



ALERT

This command will overwrite the existing key. System reset is required for new key activation.



NOTE

The over-the-air (OTA) subscription update may be available upon request without sending this command. Contact your STL subscription/authentication key provider for more information.



ALERT

If a new over-the-air key is received, the existing key will be overwritten. The Subscription state may have changed and affected receiver operation; this is expected behavior in response to an over-the-air subscription update.

Once new key is accepted, an encrypted copy of previous existing key will be stored to the EEPROM for backup. The GPS:STL:AUTHKEY:REStore ONCE command can be used to retrieve the previous key, if available.

This command has the following format:

```
GPS:STL:AUTHKEY <string>
```

The following command will query the state of the key:

```
GPS:STL:AUTHKEY?
```

Release 5.x



NOTE

Updates to the Authorization Keys take effect upon device reboot.

The Authorization Key represents the 24-character license authorization key provided by Iridium per authorization serial number. Manufacturing requests authorization keys per serial number and license type (Commercial, Locust, ANS) from Iridium. The Authorization key is programmed during manufacturing after Iridium has provided the key.

In the event of an expired time-limited license, a new license (obtained from Iridium) can be entered using the GPS:STL:AUTHKEY command.

The format of this command is:

```
GPS:STL:AUTHKEY <string>
```

For example:

```
GPS:STL:AUTHKEY 0123456789ABCDEF01234567
```

The following command will query the state of the key:

```
GPS:STL:AUTHKEY?
```

GPS:STL:SERVo:LOOP <ON|OFF>

This command will enable or disable the internal servo loop on the integrated STL-2600™ module.

The default settings is OFF to disable the internal servo loop to avoid confusion on the STL-2600™ board.

For testing purpose, this command can be configured to ON to enable the internal servo loop on the STL-2600™ board for disciplining. The settings in GPS:STL:EXTCLOCK <ON|OFF> command should also be OFF to disable the external clock reference into the STL-2600™ board.

This command has the following format:

```
GPS:STL:SERVo:LOOP <ON|OFF>
```

This command will query the state of this command:

```
GPS:STL:SERVo:LOOP?
```

GPS:POWTLV <ms> [200,1000]

This command instructs the device to send the proprietary NMEA string \$POWTLV every M milliseconds (ms), with M in the multiple of 200 ms up to 5Hz rate. This proprietary command includes information about GPS time, GPS leap second, position and 3D velocity.

This command supports settings of M with values 200 for 5Hz, 250 for 4 Hz, 500 for 2Hz, and 1000 for 1Hz. Sending GPS:POWTLV 0 command will disable \$POWTLV output and configure position update rate to default settings of 1Hz in the "GPS:SYSTem:UPDATERATE <int> [1, 5]" command.

The following are example outputs of the POWTLV sentence at 5Hz rate (GPS:POWTLV 200) within same second:

```
$POWTLV,1,2312,323033000000,1,18,0,3610.11161,N,11518.90195
,W,864.900,891.600,0.011,0.000,0.000*1B
$POWTLV,1,2312,323033200000,1,18,0,3610.11161,N,11518.90195
,W,864.900,891.600,0.009,0.000,0.000*10
$POWTLV,1,2312,323033400001,1,18,0,3610.11161,N,11518.90196
,W,864.900,891.600,0.008,0.000,0.000*15
$POWTLV,1,2312,323033599998,1,18,0,3610.11161,N,11518.90196
,W,864.900,891.600,0.002,0.000,0.000*17
$POWTLV,1,2312,323033799999,1,18,0,3610.11161,N,11518.90196
,W,864.900,891.600,0.004,0.000,0.000*12
```

This command has the following format:

```
GPS:POWTLV <ms> [200, 1000]
```

The format of the \$POWTLV string is:

```
$POWTLV,a,bbbb,cccccccccccc,d,ee,f,gggg.ggggg,h,iiii.iiii,  
j,k.kkk,l.lll,m.mmm,n.nnn,o.ooo*[checksum]0xd 0xa
```

where

- **a** is the GPS Time Quality indicator with value 0 indicates Not valid and value 1 indicates Valid
- **bbbb** is the GPS week number
- **cccccccccccc** is the seconds in the GPS week in microseconds
- **d** is the Leap Second validity indicator with value 0 indicates Not valid and value 1 indicates Valid
- **ee** is the current Leap Second value in seconds
- **f** is the holdover flag with value 0 indicates No holdover and value 1 indicates GNSDO is in holdover
- **gggg.ggggg** is Latitude specified in WGS84 datum

GPS:PJLTV <int> [0,255]

This command instructs the device to send the proprietary NMEA string \$PJLTV every N seconds, with N in the interval [0,255]. This sentence includes information about the 3D velocity, GPS time and week, and UTC leap second offset. The following is an example output of the PJLTV sentence:

```
$PJLTV,-1,-3,3,26,515247,1986,18*42
```

This command has the following format:

```
GPS:PJLTV <int> [0,255]
```

The format of the \$PJLTV command is:

```
$PJLTV,aaaaa,bbbbbb,cccc,ddd,eeeeeeee,ffff,gg*[checksum]0x  
0d 0xa
```

where:

- **aaaaa** is the velocity in the x direction (cm/s)
- **bbbbbb** is the velocity in the y direction (cm/s)
- **cccc** is the velocity in the z direction (cm/s)
- **ddd** is the speed accuracy estimate (cm/s)
- **eeeeeeee** is the GPS TOW in seconds
- **ffff** is the GPS week number
- **gg** is the current UTC leap second offset in seconds

GPS:STL:SERVo:INDOOR <OFF|ON>

This command configures the device for indoor or outdoor use.

The command has the following format:

```
GPS:STL:SERVo:INDOOR <OFF|ON>
```

Setting the command to **ON** will configure the device with optimal parameters for an indoor testing environment with an indoor antenna.

Setting the command to **OFF** will configure the device with optimal parameters for an outdoor testing environment with an outdoor antenna.

GPS:FWver?

This command queries and returns the Firmware version for the current selected receiver mode.

GPS:JAMlevel?

This command queries and returns the numerical jamming value. The options are:

```
0: No jamming detected
1: Jamming detected
```

GPS:JAMI?

This command queries and returns a list of jammed frequencies.

```
GPS:SYSTem:LBAND:AVGPower?
```

This command displays the average power of the Lband.

GPS:SYSTem:LBAND:MODE <ON|OFF>

This commands allow to enable or disable the lband.

GPS:SYSTem:LBAND:MODE?

These commands allows to query the lband mode.

GPS:SYSTem:NOTCHFilter <id>[0-3],<mode>[0-2],<ctFrqkHz>[1100000-1700000],<bwkHz>[30-1600]

This command allows to configure notch filters in the MosaicT.

GPS:SYSTem:UPDATERATe <Hz> [1,5,10,20]

This command allow the user to configure the update rate for NMEA messages in the system. For release 1.65 this needs To be 10 hz to properly inter-operate with transcoder version 1.5.

GPS:ANTPower <ON|OFF>

This command allows the device to turn the GNSS antenna input power on or off.

GPS?

This query displays the configuration, position, speed, height and other relevant data of the GNSS or STL receiver in one convenient location.

PTIME Subsystem

The PTIME subsystem regroups all the commands related to the management of the time. The following commands are supported:

- PTIME:DATE?
- PTIME:TIME?
- PTIME:TIME:STRing?
- PTIME:TINTerval?
- PTIME:OUTput <ON|OFF>
- PTIME:LEAPsecond?
- PTIME:LEAPsecond:PENDING?
- PTIME:LEAPsecond:ACCumulated?
- PTIME:LEAPsecond:DATE?
- PTIME:LEAPsecond:DURation?

PTIME:DATE?

Returns the current calender date. The local calender date is referenced to UTC time. The year, month, and day are returned.

PTIME:TIME?

Returns the current 24-hour time. The local time is referenced to UTC time. The hour, minute, and second is returned.

PTIME:TIME:STRing?

This query returns the current 24-hour time suitable for display (for example, 13:24:56).

PTIME:TINterval?

This query is equivalent to the command SYNChronization:TINterval?

PTIME:OUTput <ON|OFF>

This command adds support for auto-initialization of time and date between two VIAVI GPSDO units. This allows connecting two units together through the serial port with a null-modem cable, and having the master unit send time and date information to the slave unit. The slave unit's 1PPS reference input can also be driven by the master unit's 1PPS output signal, by setting the slave unit to external 1PPS sync mode using the SYNC:SOUR:MODE EXT command. This allows time-synchronization at the nanosecond level between two units which can be useful when operating in GPS denied environments. The following command returns the PTIME output setting stored in NV memory:

```
PTIME:OUTput?
```

Sending the following command will cause the unit to automatically generate GPS:INIT:DATE and GPS:INIT:TIME sentences described in “GPS:INITial:DATE <yyyy,mm,dd>” on page 48 and “GPS:INITial:TIME <hour,min,sec>” on page 49 on the serial port once per second:

```
PTIME:OUTput ON
```

PTIME:LEAPsecond?

This command returns the results of the four following queries:

- PTIME:LEAPsecond:PENDING?
- PTIME:LEAPsecond:ACCumulated?
- PTIME:LEAPsecond:DATE?
- PTIME:LEAPsecond:DURation?

An example of a pending leapsecond event follows:

```
LEAPSECOND PENDING: 1
```

```
LEAPSECOND ACCUMULATED: 16
```

```
LEAPSECOND DATE: 2015,6,30
```

LEAPSECOND DURATION: 61



NOTE

The unit stores pending leap second events in NV memory and applies them correctly, even if the antenna is removed prior to the leap second, as long as the pending information has been stored in NV memory and the unit has a properly set RTC time and date.

PTIME:LEAPsecond:PENDING?

This command returns 1 if the GPS Almanac data contains a future pending leap second data and 0 if no future leap second is pending or Almanac data is not available. The GNSS receiver must have the GPS system enabled in “GPS:SYSTEM:SELECT [GPS | SBAS | QZSS | GAL | BD ^ GLO]” on page 50 for the GPS Almanac to be available in GNSS receiver mode.

PTIME:LEAPsecond:ACCumulated?

This command returns the internally applied leap second offset between GPS time and UTC time as stored in the EEPROM (GPS Almanac not received yet) or as indicated by the GNSS receiver (GPS Almanac is available).

PTIME:LEAPsecond:DATE?

This command returns the date of the pending leap second, if available for current receiver mode.

PTIME:LEAPsecond:DURATION?

This command returns the duration of the last minute of the day during a leap second event, if available for current receiver mode. The returned value is 59, 60 or 61 if GPS Almanac data is available, and 0 otherwise. A response of 60 indicates that no leap second is pending.

PTIME?

The PTIME? command is unrelated to OCXO disciplining. For more information on OCXO disciplining, please refer to “SYNChronization Subsystem” on page 67.

This query returns the results of the following queries at once:

- PTIME:DATE?
- PTIME:TIME?

- PTIME:TINTerval?
- PTIME:OUTput?
- PTIME:LEAPsecond:ACCumulated?

SYNChronization Subsystem

This subsystem regroups the commands related to the synchronization of the PNT-440 device with the GNSS, STL, or external timing source. The list of the commands supported for this subsystem is the following:

- SYNChronization:HOLDOver:DURATION?
- SYNChronization:HOLDOver:INITiate
- SYNChronization:HOLDOver:RECOvery:INITiate
- SYNChronization:HOLDOver:TUNCertainty:PREDicted?
- SYNChronization:SOURce:MODE
<STL|GPS|EXTernal|REF10Mhz|PRIority>
- SYNChronization:SOURce:PRIority <STL|GPS|EXTernal>
- SYNChronization:SOURce:PRIority:RESET DEFAULT
- SYNChronization:SOURce:PRIority:TOP <GPS|STL|EXTernal>
- SYNChronization:SOURce:MODE:EDGE <NEGative|POSitive>
- SYNChronization:SOURce:STATE?
- SYNChronization:SOURce:STATE:VALue?
- SYNChronization:HOLDOver:STATE?
- SYNChronization:TINTerval?
- SYNChronization:TINTerval:THReshold <int> [50,2000]
- SYNChronization:IMMEdiate
- SYNChronization:FEEstimate?
- SYNChronization:LOCKed?
- SYNChronization:OUTput:1PPS:RESET <ON|OFF>
- SYNChronization:OUTput:1PPS:WIDTH <int> <ms | us> [100us, 500ms]
- SYNChronization:HEALTH?
- SYNChronization:HEALTH:VERBoSe?
- SYNChronization:HEALTH:HISTory?
- SYNChronization:HEALTH:HISTory:RESet
- SYNChronization:HEALTH:HISTory:VERBoSe?
- SYNChronization:HEALTH:ALARMDelay:HOLDOver <int> [0,604800]
- SYNChronization?

SYNChronization:HOLDOver:DURation?

This query returns the duration of the present or most recent period of operation in the holdover and holdover processes. This is the length of time the reference oscillator was not locked to GNSS receiver 1PPS, and thus “coasting”. The time units are seconds.

The first number in the response is the holdover duration. The duration units are seconds, and the resolution is 1 second. If the Receiver is in holdover, the response quantifies the current holdover duration. If the Receiver is not in holdover, the response quantifies the previous holdover.

The second number in the response identifies the holdover state. A value of 0 indicates the Receiver is not in holdover; a value of 1 indicates the Receiver is in holdover.

SYNChronization:HOLDOver:INITiate

The SYNC:HOLD:INIT and SYNC:HOLD:REC:INIT commands allow the user to manually enter and exit the holdover state, even while GNSS signals are still being properly received. This forced-holdover allows the unit to effectively disable GNSS locking, while still keeping track of the state of the 1PPS output in relation to the UTC 1PPS signal as generated by the GNSS receiver.

When the unit is placed into forced-holdover with this command, the unit will indicate the time interval difference between the 1PPS output and the GNSS or STL 1PPS by using the SYNC:TINT? command. This allows the user to see the OCXO drift when not locked to GNSS signals for testing purposes, or to prevent the GNSS receiver from being spoofed and affecting the OCXO frequency accuracy. The measurement range is limited to +/-2,000ns phase offset. All other frequency-disciplining functions of the unit will behave as if the GNSS antenna is disconnected from the unit while in this forced-holdover state.

SYNChronization:HOLDOver:RECOvery:INITiate

This command will disable the forced holdover state (see the SYNC:HOLD:INIT command). The unit will resume normal GNSS or STL locking operation after this command has been sent.

SYNChronization:HOLDOver:TUNCertainty:PREDicted?

This query will print the holdover prediction for one day. Holdover is the length of time the reference oscillator was not locked to GNSS or STL 1PPS, and thus “coasting”. The time units are seconds with a resolution of 100 ns. The first number printed in response to this query is the holdover duration. The duration unit is 1 to represent one day. The second number in the response identifies the holdover state. A value of 0 indicates the receiver is not in holdover; a value of 1 indicates the Receiver is in holdover.

SYNChronization:SOURce:MODE <GPS|EXTernal|STL|PRiority>

The unit may be hard-coded to only use STL or the external PPS source by setting STL or EXT, or it may be auto-switched by the firmware with PRiority settings to switch to alternative source like STL or USB external PPS signal if the internal GNSS receiver does not generate 1PPS signals for longer than 15 seconds due to a weak signal or GPS failure.

By default, the unit is set to GPS priority source mode.

PRiority mode selects the source that is listed as first priority as default source and only switch to another stable reference signal source if the current reference signal is removed or unstable. The new signal chosen will be based in order of priority. The default order of priority is:

- 1 GPS
- 2 STL
- 3 EXTernal

For example, if the GPS signal is used as first priority and later removed from the device or is no longer stable, the unit will use the STL signal next if it is stable. If the STL signal is removed or unstable afterwards the device will then use the external 1pps signal if it is stable.

The priority order can be change via the command:

```
Sync:source:priority
```

The user also can force a selection of the 1 pps source by setting the source mode to a specific source instead of using priority mode. For example:

```
sync:source:mode STL
```

Setting the unit to:

- **STL** mode enables the STL as an external PPS source and disable GNSS and 1 ext 1pps source.
- **EXT** will cause the unit to disable the GNSS and STL receiver. This setting may also be used to initiate a manual forced-holdover if the GNSS signal is suspect, or unreliable.



NOTE

If one source is forced the device will no longer switch to another source if the selected source becomes unavailable or unstable, and goes directly into holdover until the source becomes stable again.

SYNChronization:SOURce:PRiority <STL|GPS|EXTernal>

This command will set or query the current priority order used with the SYNC:SOURce:MODE:PRI setting. This command has the following format:

```
SYNChronization:SOURce:PRIority <STL|GPS|EXTernal>
```

where multiple sources are listed separate by spaces. For example, the following command enables STL as the highest priority source followed by GPS as the second priority source:

```
SYNChronization:SOURce:PRIority STL GPS
```

The following command returns the current priority list setting stored in NV memory:

```
SYNChronization:SOURce:PRIority?
```

SYNChronization:SOURce:PRIority:RESET DEFAULT

This command is not currently supported in PNT-440 device. This command will reset the current priority levels to the default levels. The default levels are:

- 1 GPS
- 2 STL
- 3 EXT

SYNCHronization:SOURce:PRIority:TOP <STL|GPS|EXTernal>

This command is not currently supported in PNT-440 device. This command allows the priority list to be customized. The user can select a stable 1PPS source as the top or highest priority source with this command.

As an example, say this is the current priority list:

- 1 GPS
- 2 STL
- 3 EXT

If the STL is later determined by the user to be the highest priority source, the command should be entered like so:

```
SYNCHronization:SOURce:PRIority:TOP STL
```

The new priority list will be as follows:

- 1 STL
- 2 GPS
- 3 EXT

This query will return the priority list's current top source:

```
SYNCHronization:SOURce:PRIority:TOP?
```

SYNChronization:SOURce:MODE:EDGE <NEGative|POSitive>

This command is not currently supported in PNT-440 device.

SYNChronization:SOURce:STATe?

This command returns the state of the `SYNC:SOUR:MODE` command, and may return GPS, STL, or EXTERNAL.

SYNChronization:SOURce:STATe:VALue?

This command returns the state of the `SYNC:SOUR:MODE` command in numeric value, and may return 1 for STL, 2 for GPS, 3 for EXTERNAL.

SYNChronization:HOLDoVer:STATe?

This query returns the current holdover state with possible responses being NONE, MANUAL, or ON. NONE indicates the holdover state is off. MANUAL indicates the holdover is manually enabled with the `SYNChronization:HOLDoVer:INITiate` command. ON indicates that holdover is enabled due to lack of external PPS signal or valid GNSS/STL fix from GNSS/STL receiver.

SYNChronization:TINTerval?

This query returns the difference or timing shift between the PNT-440 device 1PPS output and the reference 1PPS signal indicated in `SYNChronization:SOURce:STATe?` query. The resolution is 1E-10 seconds.

SYNChronization:TINTerval:THReshold <int> [50,2000]

This command selects the oscillator driven 1PPS phase-offset threshold as compared to the reference 1PPS at which point the unit will initiate a counter-reset (jam-sync) aligning the oscillator driven 1PPS with the reference 1PPS phase. The oscillator phase is slowly and continuously adjusted toward 0ns offset to the reference 1PPS while the phase difference is less than the THReshold phase limit. The oscillator driven 1PPS phase is allowed to drift up to this threshold before a jam-sync is initiated.

The default setting is 300ns (1000ns for STL 1PPS), allowing a drift of up to +/- 300ns (1000ns for STL 1PPS). Reaching this selected threshold will cause a jam-sync phase-normalization to be initiated, which will also cause the `SYNC:HEALTH? Status` to indicate 0x200, and the lock status to be unlocked as indicated by the Green LED turning off for several minutes. Setting the threshold to larger values (i.e. 1000ns or higher) will prevent nuisance jam-sync events to happen to the 1PPS output phase, at the expense of potentially larger phase errors to accumulate on the 1PPS output

before the loop algorithms can pull the phase error toward 0ns.

The following command queries the state of
`SYNChronization:TINTerval:THReshold:`

SYNChronization:TINTerval:THReshold?

SYNChronization:IMMEdiate

This command initiates a near-instantaneous alignment of the GNSS/STL/EXT 1PPS and Receiver output 1 PPS. To be effective, this command has to be issued while not in holdover.

SYNChronization:FEEstimate?

This query returns the Frequency Error Estimate, similar to the Allan Variance using a 1000s measurement interval and comparing the GNSS/STL/EXT 1PPS to GNSS 1PPS offset.

Values less than 1E-012 are below the noise floor, and are not significant.

SYNChronization:LOCKed?

This query returns the lock state (0=OFF, 1=ON) of the PLL controlling the OCXO.

SYNChronization:OUTput:1PPS:RESET <ON|OFF>

This command allows the generation of the 1PPS signal upon power-on without an GNSS/Iridium antenna being connected to the unit or valid 1PPS input signal. By default the unit does not generate a 1PPS signal until the GNSS or STL receiver has locked onto the Satellites.

With the command `SYNChronization:OUTput:1PPS:RESET ON` the unit can be configured to generate an asynchronous 1PPS output after power-on even if a GNSS/Iridium antenna is not connected to the unit. Once valid 1PPS input is received, the output 1PPS signal will align itself to UTC time by stepping in 10 equally spaced steps toward UTC alignment. The default setting is OFF which means the 1PPS output signal is disabled until proper GNSS/STL lock is achieved.

This command has the following format:

```
SYNChronization:OUTput:1PPS:RESET <ON|OFF>
```

This command will query the state of this command:

```
SYNChronization:OUTput:1PPS:RESET?
```

SYNChronization:OUTput:1PPS:WIDTH <int> <ms | us> [100us, 500ms]

This command allows configuration of the 1PPS active high pulse width. The default pulse width on the standard PNT-440 device is 200 milliseconds and the pulse can be

configured from 100 microseconds to 500 milliseconds. The integer parameter is followed by optional us (microsecond) or ms (millisecond) units. The default unit is milliseconds. The output 1PPS signal is synchronized to UTC (USNO) on its rising edge if 3D fix is available.

This command has the following format:

```
SYNChronization:OUTput:1PPS:WIDTH <int> <ms | us >
[100us,500ms]
```

The following command returns the current pulse width setting stored in NV memory:

```
SYNChronization:OUTput:1PPS:WIDTH?
```

The unit of parameter for this command can be either us or ms. For example, command

`SYNC:OUT:1PPS:WIDTH 200000us` and command `SYNC:OUT:1PPS:WIDTH 200ms` will both configure the pulse width to 200ms. If the 1PPS output signal is already available before sending this command, a 1PPS pulse with the specified pulse width will be generated on the next second and every second thereafter.

SYNChronization:HEAlth?

The `SYNChronization:HEAlth?` query returns a hexadecimal number indicating the system's health-status. Error flags are encoded in a binary fashion so that each flag occupies one single bit of the binary equivalent of the hexadecimal health-status flag.

The following system parameters are monitored and indicated through the health-status indicator. Individual parameters are 'Ored' together which results in a single hexadecimal value encoding the following system status information:

If the OCXO coarse-DAC is maxed-out at 255	HEALTH STATUS = 0x1;
If the OCXO coarse-DAC is mined-out at 0	HEALTH STATUS = 0x2;
If the phase offset to UTC is >250ns	HEALTH STATUS = 0x4;
If the run-time is < 300 seconds	HEALTH STATUS = 0x8;
If the GNSS receiver is in holdover > 60s	HEALTH STATUS = 0x10;
If the Frequency Estimate is out of bounds	HEALTH STATUS = 0x20;
If the power supply or OCXO voltage is too high	HEALTH STATUS = 0x40;
If the power supply or OCXO voltage is too low	HEALTH STATUS = 0x80;

If the short-term-drift (ADEV @ 100s) > 100ns	HEALTH STATUS = 0x100;
For the first 3 minutes after a phase-reset, or a coarsedac change:	HEALTH STATUS = 0x200;
If the GNSS receiver indicates a strong jamming signal of >=50 (range is 0 to 255) and is in holdover	HEALTH STATUS = 0x800;
If spoofing is detected	HEALTH STATUS = 0x4000
If jamming is detected	HEALTH STATUS = 0x8000
If I-band loss is detected	HEALTH STATUS = 10000

For example, if the unit is in GNSS/STL receiver holdover, the OCXO voltage is too high, and the UTC phase offset is > 250ns, then the following errors would be indicated:

- 1 UTC phase > 250ns: 0x4
- 2 OCXO voltage too high: 0x40
- 3 GNSS/STL receiver in holdover: 0x10

'Oring' these values together results in:

$$0x40 \mid 0x10 \mid 0x4 = 0x54$$

The unit would thus indicate: HEALTH STATUS: 0x54

A health status of 0x0 indicates a properly locked, and warmed-up unit that is completely healthy and does not have any pending events.

SYNChronization:HEAlth:VERBose?

The SYNChronization:HEAlth:VERBose? query returns a human readable status for each health-status flag of the health indicator.

SYNChronization:HEAlth:HISTory?

The SYNChronization:HEAlth:HISTory? query returns a hexadecimal number indicating the system's health-status flags that have been set since power up or since last SYNChronization:HEAlth:HISTory:RESet command. The history status permits persistent tracking of any status flag that is set during runtime.

SYNChronization:HEAlth:HISTory:RESet

The `SYNChronization:HEAlth:HISTory:RESet` command clears the `SYNChronization:HEAlth:HISTory` health-status flags.

SYNChronization:HEAlth:HISTory:VERBose?

The `SYNChronization:HEAlth:HISTory:VERBose?` query returns a human readable status for each health-status flag of the heath history indicator.

SYNChronization:HEAlth:ALARMDelay:HOLDoVer <int> [0,604800]

Configures a delay in seconds before indicating a health alarm to avoid unnecessary alarms from short holdover periods. A setting of 0 disables the delay and the maximum setting is 604,800 seconds or one week.

This command has the following format:

```
SYNChronization:HEAlth:ALARMDelay:HOLDoVer <int> [0,604800]
```

This command will query the state of this command:

```
SYNChronization:HEAlth:ALARMDelay:HOLDoVer?
```

SYSTem:COMMunicate:DLOAD:TIMEout <int>[1,3600]

This command allow the configuration of the timeout before the unit switches back to normal mode after entering the dload mode. This is intended for Viavi internal use only.

SYNChronization:TINterval:RAW?

This command allows to configure the raw Time Interval Value.

SYSTem:FWConfig?

This command allows to retrieve a summary of the current firmware configuration

SYSTem:GETCFGCurrent?

This command allows to retrieve a summary of the current key parameters configured by the user

SYSTem:GETCFGDefault?

This command allows to retrieve a summary of default values of key parameters configured in manufacturing.

SYSTem:RESETCFGtodefault

This command allows the user to reset the key parameters to the default values configured in manufacturing.

SYSTem:ID:MODELname?

This command allows to retrieve the model name of the product

SYSTem:ID:SN:BOX?

This command allows to read the serial number of the box assembly.

SYSTem:ID:SN?

This command allows to read the serial number of the main board.

SYSTem:ID:UID?

This command returns the processor ID of the main systems processor

SYSTem:ID?

This command returns the model, PCBA serial number and box serial number as well as processor ID of the device.

SYSTem:STATus:GPSDO?

This command allows to retrieve the status of the GPS Disciplined Oscillator.

SYNChronization?

This query returns the results of the following queries:

- SYNChronization:SOURce:MODE?
- SYNChronization:SOURce:STATe?
- SYNChronization:OUTput:1PPS:RESET?
- SYNChronization:OUTput:1PPS:WIDTH?
- SYNChronization:LOCKed?
- SYNChronization:HOLDoVer:STATe?
- SYNChronization:HOLDoVer:DURation?
- SYNChronization:FEEstimate?
- SYNChronization:TINTerval?
- SYNChronization:TINTerval:THReshold?
- SYNChronization:HEALth?

DIAGnostic Subsystem

This subsystem regroups the queries related to the diagnostic of the OCXO. The list of the commands supported for this subsystem is as follows:

- DIAGnostic?
- DIAGnostic:ROSCillator:EFControl:RELative <float> [-100.0, 100.0]
- DIAGnostic:ROSCillator:EFControl:ABSolute <float> [0.0, 5.0]
- DIAGnostic:ROSCillator:EFControl:PPT <float> [-800000000.0, 800000000.0]
- DIAGnostic:LIFetime:COUNt?
- DIAGnostic:LIFetime:SECond?

DIAGnostic:ROSCillator:EFControl:RELative <float> [-100.0, 100.0]

This command sets the Electronic Frequency Control (EFC) output value of the internal reference oscillator. The parameter value should be a percentage between -100.0 to +100.0.

This command has the following format:

```
DIAGnostic:ROSCillator:EFControl:RELative [-100.0, 100.0]
```

This command will query the state of this command:

```
DIAGnostic:ROSCillator:EFControl:RELative?
```

DIAGnostic:ROSCillator:EFControl:ABSolute <float> [0.0, 5.0]

This command sets the Electronic Frequency Control (EFC) output value of the internal reference oscillator. The parameter value should be normalized to volts.

This command has the following format:

```
DIAGnostic:ROSCillator:EFControl:ABSolute [0.0, 5.0]
```

This command will query the state of this command:

```
DIAGnostic:ROSCillator:EFControl:ABSolute?
```

DIAGnostic:LIFetime:COUNT?

This command returns the number of hours the unit has been powered-on.

DIAGnostic:LIFetime:SECond?

This command returns the number of seconds the unit has been powered-on.

DIAGnostic?

Sending the command DIAG? returns the result of the three following queries:

- `DIAGnostic:ROSCillator:EFControl:RELative?` : Relative oscillator setting
- `DIAGnostic:ROSCillator:EFControl:ABSolute?` : Absolute oscillator setting
- `DIAGnostic:LIFetime:COUNT?` : Time since power-on in hours (lifetime)

An example of the syntax:

```
scpi > diag?  
EFControl Relative: 0.025000%  
EFControl Absolute: 5  
Lifetime : +871
```

CSAC Subsystem

The following commands are used to query the status of the CSAC or Rubidium oscillator. The following commands will return Command Error if the atomic oscillator is not detected or present.

CSAC:RS232?

This query returns the state (OK or FAIL) of the serial communication between the main CPU and the oscillator of the attached unit's internal microcontroller. When the state is

FAIL, there is a communication breakdown, and the unit should be power cycled to clear the communication error.

CSAC:STeer?

This query returns the current Frequency Adjustment in units of parts-per-trillion (1E-012). The Frequency Adjustment is relative to the latched frequency stored in the CSAC or MAC.

CSAC:TYPE?

This query returns the type/model of the connected atomic oscillator.

CSAC:LOCKed?

This query returns the lock status of the atomic oscillator. A value of 1 is returned if the CSAC or Rubidium is in atomic lock.

CSAC:STATus?

This query returns the status value of the CSAC or Rubidium clock.

If the Microsemi Rubidium is attached to the PNT-440 device, this query will return one of two values [0,1] as shown below. If the unit does not indicate STATUS 0 after 10 minutes from power-on, the Microsemi Rubidium may not have locked properly and may need to be power-cycled. This can happen if power supplies with insufficient current capability to handle the Microsemi Rubidium warm-up currents are used.

- 0 = The Microsemi Rubidium is locked and healthy
- 1 = The Microsemi Rubidium is not locked

If the CSAC is attached, one of the status values shown in [Table 13](#) will be displayed.

Table 10 CSAC Status definitions

Status	Definition
0	Locked
1	Microwave Frequency Steering
2	Microwave Frequency Stabilization
3	Microwave Frequency Acquisition
4	Laser Power Acquisition
5	Laser Current Acquisition
6	Microwave Power Acquisition

Table 10 CSAC Status definitions

Status	Definition
7	Heater Equalibration
8	Initial Warm-up
9	Asleep (ULP mode only)

CSAC:ALarm?

Returns the Alarm value of CSAC oscillator, as shown in [Table 14](#).

Table 11 Alarm values

Alarm	Definition
0x0001	Signal Contrast Low
0x0002	Synthesizer Tuning at Limit
0x0010	DC Light Level Low
0x0020	DC Light Level High
0x0040	Heater Power Low
0x0080	Heater Power High
0x0100	uW Power Control Low
0x0200	uW Power Control High
0x0400	TCXO Control Voltage Low
0x0800	TCXO Control Voltage High
0x1000	Laser Current Low
0x2000	Laser Current High
0x4000	Stack Overflow (Firmware Error)

CSAC:MODE?

This query returns the mode value of CSAC or Rubidium clock.

If the Microsemi Rubidium is attached to the PNT-440 device, this query will return one of two values:

- 0x0000 = Analog tuning is disabled
- 0x0001 = Analog tuning is enabled

If the CSAC is attached to the PTN-62xx, this query returns its mode as shown in [Table 15](#).

Table 12 Mode values

Mode	Definition
0x0001	Analog Tuning Enable
0x0002	Reserved
0x0004	Reserved
0x0008	1 PPS Auto-Sync Enable
0x0010	Discipline Enable
0x0020	Ultra-Low Power Mode Enable
0x0040	Reserved
0x0080	Reserved

CSAC:CONTrast?

This query returns an indication of signal level in CSAC oscillator, typically ~4000 when locked, and ~0 when unlocked.

CSAC:LASer?

This query returns the current (in mA) driving the laser in CSAC oscillator.

CSAC:TCXO?

This query returns the TCXO Tuning Voltage, 0-2.5 VDC tuning range ~ +/- 10 ppm.

CSAC:SIGnal?

If the Microsemi Rubidium is attached to the PNT-440 device, this query will return an indication of the Microsemi Rubidium Vapor Cell DC interrogation signal level in mV. If the CSAC is attached, this query will simply return the indication of signal level.

CSAC:HEATpackage?

If the Microsemi Rubidium is attached to the PNT-440 device, it will return the Physics package heater power in mW. If the CSAC is attached to the PNT-440 device, this query will return the Physics package heater power, typically 15mW under NOC will be returned.

CSAC:TEMP?

If the CSAC is attached to the PNT-440 device, this query will return the temperature measured by the unit in °C. Absolute accuracy is +/- 2°C. If a Microsemi Rubidium is attached to the PNT-440 device, the temperature measured by the Microsemi Rubidium oscillator unit is in °C.

CSAC:SN?

This query returns the Serial Number in the form `YYMMCSNNNNN` where `YYMM` is the year and month of production and `NNNNN` is the serialized unit of that month.

CSAC:FWver?

This query returns the firmware version of the CSAC or Rubidium unit.

CSAC:LIFETIME?

This query returns the accumulated number of hours the CSAC or Rubidium clock has been powered on since the last factory reset of the respective board. The value is stored in the external NVRAM and updated every hour when the unit is powered on.

CSAC:DDScenter?

This query is compatible with the Microsemi Rubidium. It will return the DDS frequency center current in Hertz.

CSAC:TECcontrol?

This query is compatible with the Microsemi Rubidium. It will return the TEC Control value in °C.

CSAC:STeer:LATch ONCE

This command stores the momentary steering offset into the respective unit's internal NVRAM. This is done automatically by the firmware once every 24 hours, so as not to damage the NVRAM which has a limited number of write cycles. The user may force this value to be stored by issuing the `CSAC:STeer:LATch ONCE` command.

CSAC:POWer <ON|OFF>

This command is compatible with the CSAC. By default the normal power mode is ON and sending `CSAC:POWer OFF` command will put the CSAC in ULP mode.

The power will always be shown as ON with the Rubidium attached to the PNT-440 device.

The CSAC may lose atomic lock while in ULP mode. It is recommended to keep the `CSAC:POWer ON` setting for normal operation and optimal performance on PNT-440 device.

This command has the following format:

```
CSAC:POWer <ON | OFF>
```

The following command returns the current power setting where ON means normal power mode:

```
CSAC:POWer?
```

CSAC?

This command returns the result of the following queries:

- `CSAC:RS232?`
- `CSAC:STeer?`
- `CSAC:TYPE?`
- `CSAC:LOCKed?`
- `CSAC:STATus?`
- `CSAC:ALarm?`
- `CSAC:MODE?`
- `CSAC:CONTrast?`
- `CSAC:LASer?`
- `CSAC:TCXO?`
- `CSAC:SIGnal?`
- `CSAC:HEATpackage?`
- `CSAC:TEMP?`
- `CSAC:SN?`
- `CSAC:FWver?`
- `CSAC:LIFEtime?`
- `CSAC:DDScenter?`
- `CSAC:TECcontrol?`
- `CSAC:POWer?`

RFOUTput Subsystem

This subsystem regroups the commands related to the functionality of the on-board Micro-Transcoder order option. The list of the commands supported for this subsystem is as follows:

- RFOUTput:MODE <ON|OFF>
- RFOUTput:POWer <float> [-200.0, 200.0]
- RFOUTput:DCBlock <ON|OFF>
- RFOUTput:RESET ONCE
- RFOUTput:ISP ON
- RFOUTput:SCPI <ASCII> <command>
- RFOUTput:SIM:RESTART ONCE
- RFOUTput:SIM:HOLDOVer <ON|OFF|LIMIT>
- RFOUTput:SIM:HOLDOVer:LIMIT <int> [5, 86400]
- RFOUTput:SIM:LNAV:SElect <AUTO|USER|LIVE>
- RFOUTput:STATE?
- RFOUTput:TINT?
- RFOUTput:FEE?
- RFOUTput:INTGPS:PPSERRor?
- RFOUTput:INTGPS:POSERRor?
- RFOUTput:INTGPS:VPOSERRor?
- RFOUTput:EFC?
- RFOUTput:VIN?
- RFOUTput:EXTGPS:PPSERRor?
- RFOUTput:EXTGPS:POSERRor?
- RFOUTput:EXTGPS:VPOSERRor?
- RFOUTput:FGPAdiag?
- RFOUTput:LOCKcode?
- RFOUTput:HEAlth?
- RFOUTput:HEAlth:VALue?
- RFOUTput:FWver?

RFOUTput:MODE <ON|OFF>

This command will enable or disable the transcoding RF output signal on the Micro-Transcoder. Transcoding is enabled by default. If the Transcoder is not transcoding properly, try sending the RFOUT ON command once.

This command has the following format:

```
RFOUTput:MODE <ON|OFF>
```

This command will query the state of this command:

RFOUtput:MODE?

RFOUtput:POWer <float> [-200.0, 200.0]

This command sets the power level at the RF output jack on an individual satellite basis.

The RFOUtput:POWer command sets the simulator's RF output power from the perspective of a single satellite, not the entire constellation. When N satellites are transmitting, the total RF output power will be approximately $10 \cdot \log_{10} N$ dB greater than specified.

The nominal output power of a live GPS satellite signal at the surface of the earth is typically about -125 dBm, but is often a few dB higher to allow for a reduction in power as the satellite ages. Additionally, target GPS receivers often expect to be used with an amplified antenna, so the output power is typically set higher than -125 dBm for optimum performance of the target GPS receiver.

This command has the following format:

```
RFOUtput:POWer <float> [-200.0, 200.0]
```

This command will query the state of this command:

```
RFOUtput:POWer?
```

RFOUtput:DCBlock <ON|OFF>

This command will enable or disable the transcoding RF output signal on the Micro-Transcoder. Transcoding is enabled by default. If the Transcoder is not transcoding properly, try sending the RFOU ON command once.

This command has the following format:

```
RFOUtput:DCBlock <ON|OFF>
```

This command will query the state of this command:

RFOUtput:RESET ONCE

This command will issue a reset sequence for the Micro Transcoder™ board once.

RFOUtput:ISP ON

This command will send the Micro Transcoder™ board into ISP mode for reprogramming. Refer to Chapter 4 of the Micro Transcoder™ User Manual for programming instructions.

RFOUtput:SCPI <ASCII> <command>

This command sends a SCPI command to the Transcoder and can be used for changing settings not configurable through the PNT-440 device's SCPI interface.

RFOUtput:SIM:RESTART ONCE

This command will stop the current simulation once and restart transcoding shortly for RF output of Micro Transcoder™ board.

RFOUtput:SIM:HOLDover <ON|OFF|LIMIT>

This command controls the behavior when the external PVT reference indicates that position information is invalid while operating in Transcoding simulation mode.

This command has the following format:

```
RFOUtput:SIM:HOLDover <ON|OFF|LIMIT>
```

When the LIMIT mode is selected, the limit time is specified with the SIM:HOLD:LIMIT command, and the RF output is disabled when the unit reaches the specified holdover time limit described in [“RFOUtput:SIM:HOLDover:LIMIT <int> \[5, 86400\]” on page 86](#).

This command will query the state of this command:

```
RFOUtput:SIM:HOLDover?
```

RFOUtput:SIM:HOLDover:LIMIT <int> [5, 86400]

This command specifies the limit, in seconds, that the holdover state should remain active in the Limit simulation holdover mode. The maximum limit is 1 day or 86400 seconds. Once this holdover limit is reached, the RF output is disabled. For longer holdover periods, use the ON simulation holdover mode.

This command has the following format:

```
RFOUtput:SIM:HOLDover:LIMIT <int> [5, 86400]
```

This command will query the state of this command:

```
RFOUtput:SIM:HOLDover:LIMIT?
```

RFOUtput:SIM:LNAV:SElect <AUTO|USER|LIVE>

This command is not supported on the PNT-440 device.

RFOUtput:STATE?

This query responds with the current simulation state of the Transcoder. Possible responses include:

- DETECTING GPS
- WAITING GPS FIX
- WAITING PPS
- TRANSCODING

RFOUtput:TINT?

This query returns the timing offset in nanoseconds between the PNT-440's 1PPS output and simulation timing. This measurement is filtered by the Transcoder's clock disciplining algorithm.

RFOUtput:FEE?

This query returns the Frequency Error Estimate (FEE) between the PNT-440's 1PPS output and the simulation timing. This estimate is similar to the Allan Variance using a 1000s measurement interval.

RFOUtput:INTGPS:PPSERRor?

This query returns the difference in seconds between the Transcoder's internal GPS receiver's 1PPS rising edge and the current simulation timing.

RFOUtput:INTGPS:POSERRor?

This query returns the horizontal difference in meters between the position reported by the Transcoder's internal GPS receiver and the simulated position.

RFOUtput:INTGPS:VPOSERRor?

This query returns the difference in meters between the altitude reported by the Transcoder's internal GPS receiver and the simulated altitude.

RFOUtput:EFC?

This query returns the Electronic Frequency Control (EFC) output value of the Transcoder's internal reference oscillator. It returns a value in volts.

RFOUtput:VIN?

This query returns the Transcoder's internally regulated 2.5V supply voltage.

RFOUtput:EXTGPS:PPSERRor?

This query returns the difference in nanoseconds between the PNT-440's 1PPS output and the simulation timing. This measurement is a raw unfiltered phase difference measurement.

RFOUtput:EXTGPS:POSERRor?

This query returns the horizontal difference in meters between the position reported by the PNT440's position source and the simulated position.

RFOUtput:EXTGPS:VPOSERRor?

This query returns the vertical difference in meters between the altitude reported by the PNT-440's position source and the simulated altitude.

RFOUtput:FGPAdiag?

This query returns the Transcoder's internal FPGA diagnostics code in hexadecimal format.

RFOUtput:LOCKcode?

This query returns the Transcoder's bitwise lock flags in hexadecimal format. The lock flags contain the following status bits:

External 10 MHz reference present	FLAGS = 0x1
1PPS source present	FLAGS = 0x2
1PPS servo lock in progress	FLAGS = 0x4
1PPS servo lock complete	FLAGS = 0x8
Valid GPS signal has been transmitted for >100 seconds	FLAGS = 0x40
Valid 3D fix reported by internal GNSS receiver	FLAGS = 0x80

RFOUtput:LOCKcode:VALue?

This query returns the Transcoder's bitwise lock flags in decimal format. This query is useful for logging the lock flags in GPSCon.

RFOUtput:HEALTH?

This query returns the Transcoder's bitwise health flags in hexadecimal format. The health flags contain the following status bits:

If the TCXO coarse-DAC is maxed-out at 255	HEALTH STATUS = 0x1
If the TCXO coarse-DAC is mined-out at 0	HEALTH STATUS = 0x2;
If the phase offset to UTC is >250ns	HEALTH STATUS = 0x4
If the run-time is < 300 seconds	HEALTH STATUS = 0x8
If the GNSS receiver is in holdover > 60s	HEALTH STATUS = 0x10
If the Frequency Estimate is out of bounds	HEALTH STATUS = 0x20
If the supply voltage is too high	HEALTH STATUS = 0x40
If the supply voltage is too low	HEALTH STATUS = 0x80
If the short-term-drift (ADEV @ 100s) > 100ns	HEALTH STATUS = 0x100
For the first 3 minutes after a phase reset, or a coarse DAC change	HEALTH STATUS = 0x200
If the GNSS receiver indicates a strong jamming signal of >=50 (range is 0 to 255) (supported only on uBlox external receivers)	HEALTH STATUS = 0x800
If FPGA error occurs	HEALTH STATUS = 0x2000
If simulation update timeouts occur	HEALTH STATUS = 0x4000
If internal GPS receiver C/No values out of range (>2dB error)	HEALTH STATUS = 0x8000
If internal GPS receiver position out of range (>10 meters)	HEALTH STATUS = 0x10000
If internal GPS receiver 1PPS timing out of range (>100 ns)	HEALTH STATUS = 0x20000
If internal GPS receiver has not achieved lock after 100 seconds	HEALTH STATUS = 0x40000

RFOUTput:HEAlth:VALue?

This query returns the Transcoder's health flags in decimal format. This query is useful for logging the health flags in GPSCon.

RFOUTput:FWver?

This query returns the Transcoder's firmware version number.

RFOUTput:ERRORcode:VALue?

This command allows to read the value of the transcoder error code.

RFOUTput:ERRORcode?

This command allows to read the error code of the transcoder.

RFOUTput:SIM:POSition <OFF | lat, lon, alt>

This command allows the user to enter a manual position.

MEASURE Subsystem

This subsystem regroups the queries related of some parameters that are measured on-board on the PNT-440 device. The list of the commands supported for this subsystem is the following:

- MEASure:APOWersupply?
- MEASure:BPOWersupply?
- MEASure:SUPPLY5Volt?
- MEASure?

MEASure:APOWersupply?

This query returns the voltage supplied by voltage supply A.

MEASure:BPOWersupply?

This query returns the voltage supplied by voltage supply B.

MEASure:SUPPLY5Volt?

This query returns the voltage supplied by the 5V voltage source.

MEASure:POWersupply:V33?

This command allows to read the power of the V33 for debug purpose.

MEASure:POWersupply:VOscillator?

This command allows to read the power of the VOscillator for debug purpose.

MEASure:POWersupply?

This command allows to read the power of the power supply.

MEASure?

This query returns the result of the three following queries:

- MEASure:APOWersupply?
- MEASure:BPOWersupply?
- MEASure:SUPPLY5Volt?

System subsystem

This subsystem regroups the commands related to the general configuration of the PNT-440 device. The list of the commands supported for this subsystem follows:

- SYSTem:COMMunicate:SERial:ECHO <ON|OFF>
- SYSTem:COMMunicate:SERial:PROmpt <ON|OFF>
- SYSTem:COMMunicate:SERial:BAUD
<9600|19200|38400|57600|115200>
- SYSTem:COMMunicate:SERial:FAST <ON|OFF>
- SYSTem:COMMunicate:GNSS:BAUD <9600 | 19200 | 38400 | 57600 |
115200>
- SYSTem:COMMunicate:GNSS:MODE <ON|OFF>
- SYSTem:COMMunicate:RSRSCPI:BAUD <9600 | 19200 | 38400 |
57600 | 115200>
- SYSTem:COMMunicate:RSRSCPI:MODE <ON|OFF>

- `SYSTem:COMMunicate:STL:BAUD <9600 | 19200 | 38400 | 57600 | 115200>`
- `SYSTem:COMMunicate:STL:MODE <ON|OFF>`
- `SYSTem:COMMunicate:TOD:BAUD <9600 | 19200 | 38400 | 57600 | 115200>`
- `SYSTem:COMMunicate:TOD:MODE <ON|OFF>`
- `SYSTem:COMMunicate:CSAC:BAUD <9600 | 19200 | 38400 | 57600 | 115200>`
- `SYSTem:COMMunicate:CSAC:MODE <ON|OFF>`
- `SYSTem:COMMunicate:RSRGPS:BAUD <9600 | 19200 | 38400 | 57600 | 115200>`
- `SYSTem:COMMunicate:RSRGPS:MODE <ON|OFF>`
- `SYSTem:COMMunicate:MCE:BAUD <9600 | 19200 | 38400 | 57600 | 115200>`
- `SYSTem:COMMunicate:MCE:MODE <ON|OFF>`
- `SYSTem:POWer <SLEEP|DSLEEP|PDOWN|DPDOWN>`
- `SYSTem:COMMunicate:DLOAD <STL|GNSS|TRANscoder>`
- `SYSTem:STATus?`
- `SYSTem:FACToryReset ONCE`
- `SYSTem:CPURESET`
- `SYSTem:ISP`

SYSTem:COMMunicate:SERial:ECHO <ON|OFF>

This command enables/disables echo on the SCPI serial port. This command has the following format:

```
SYSTem:COMMunicate:SERial:ECHO <ON|OFF>
```

SYSTem:COMMunicate:SERial:PROmpt <ON|OFF>

This command enables/disables the prompt “`scpi>`” on the SCPI command lines. The prompt must be enabled when used with the GPSCon utility discussed in [Chapter 3 “GPSCon Utility”](#). This command has the following format:

```
SYSTem:COMMunicate:SERial:PROmpt <ON|OFF>
```

SYSTem:COMMunicate:SERial:BAUD <9600|19200|38400|57600|115200>

This command sets the serial speed for SCPI serial port. The serial configuration is always 8 bit, 1 stop bit, no parity, no HW flow control. Upon Factory reset, the speed is set at 115200 bauds. Higher baud rates such as 115200 are suggested if several NMEA

commands are enabled, or high rates of serial communications are expected to avoid serial port overflow. This command has the following format:

```
SYSTem:COMMunicate:SERial:BAUD <9600|19200|38400|57600|115200>
```

SYSTem:COMMunicate:SERial:FAST <ON|OFF>

This command turns the serial port's fast mode on or off.

This command has the following format:

```
SYSTem:COMMunicate:SERial:FAST <ON|OFF>
```

This command will query the state of this command:

```
SYSTem:COMMunicate:SERial:FAST?
```

SYSTem:COMMunicate:GNSS:BAUD <9600 | 19200 | 38400 | 57600 | 115200>

This command sets the GNSS receiver's serial speed. This command will work for both the uBlox and Septentrio GNSS receivers. The serial configuration is always 8 bit, 1 stop bit, no parity, no flow control. The factory default setting is 115200.

This command has the following format:

```
SYSTem:COMMunicate:GNSS:BAUD <9600|19200|38400|57600|115200>
```

This command will query the state of this command:

```
SYSTem:COMMunicate:GNSS:BAUD?
```

SYSTem:COMMunicate:GNSS:MODE <ON|OFF>

This command will enable or disable the GNSS port for communication. This command will work for both the uBlox and Septentrio GNSS receivers.

This command has the following format:

```
SYSTem:COMMunicate:GNSS:MODE <ON|OFF>
```

This command will query the state of this command:

```
SYSTem:COMMunicate:GNSS:MODE?
```

SYSTem:COMMunicate:RSRSCPI:BAUD <9600 | 19200 | 38400 | 57600 | 115200>

This command sets the serial speed for SCPI port of the Micro-Transcoder. The serial configuration is always 8 bit, 1 stop bit, no parity, no flow control. The factory default setting is 115200 baud.

This command has the following format:

```
SYSTem:COMMunicate:RSRSCPI:BAUD <9600 | 19200 | 38400 | 57600 | 115200>
```

This command will query the state of this command:

```
SYSTem:COMMunicate:RSRSCPI:BAUD?
```

SYSTem:COMMunicate:RSRSCPI:MODE <ON|OFF>

This command will enable or disable the communication for SCPI port of the Micro-Transcoder port.

This command has the following format:

```
SYSTem:COMMunicate:RSRSCPI:MODE <ON|OFF>
```

This command will query the state of this command:

```
SYSTem:COMMunicate:RSRSCPI:MODE?
```

SYSTem:COMMunicate:STL:BAUD <9600 | 19200 | 38400 | 57600 | 115200>

This command sets the STL serial speed. The serial configuration is always 8 bit, 1 stop bit, no parity, no flow control. The factory default setting is 115200 baud.

This command has the following format:

```
SYSTem:COMMunicate:STL:BAUD <9600 | 19200 | 38400 | 57600 | 115200>
```

This command will query the state of this command:

```
SYSTem:COMMunicate:STL:BAUD?
```

SYSTem:COMMunicate:STL:MODE <ON|OFF>

This command will enable or disable the STL port for communication.

This command has the following format:

```
SYSTem:COMMunicate:STL:MODE <ON|OFF>
```

This command will query the state of this command:

```
SYSTem:COMMunicate:STL:MODE?
```

SYSTem:COMMunicate:TOD:BAUD <9600 | 19200 | 38400 | 57600 | 115200>

This command sets the serial speed for the TOD serial port. The serial configuration is always 8 bit, 1 stop bit, no parity, no flow control. The factory default setting is 115200 baud.

This command has the following format:

```
SYSTem:COMMunicate:TOD:BAUD <9600 | 19200 | 38400 | 57600 | 115200>
```

This command will query the state of this command:

```
SYSTem:COMMunicate:TOD:BAUD?
```

SYSTem:COMMunicate:TOD:MODE <ON|OFF>

This command will enable or disable the communication for the TOD serial port.

This command has the following format:

```
SYSTem:COMMunicate:TOD:MODE <ON|OFF>
```

This command will query the state of this command:

```
SYSTem:COMMunicate:TOD:MODE?
```

SYSTem:COMMunicate:CSAC:BAUD <9600 | 19200 | 38400 | 57600 | 115200>

This command sets the CSAC serial speed. The serial configuration is always 8 bit, 1 stop bit, no parity, no flow control. The factory default setting is 115200 baud.

This command has the following format:

```
SYSTem:COMMunicate:CSAC:BAUD <9600 | 19200 | 38400 | 57600 | 115200>
```

This command will query the state of this command:

```
SYSTem:COMMunicate:CSAC:BAUD?
```

SYSTem:COMMunicate:CSAC:MODE <ON|OFF>

This command will enable or disable the CSAC port for communication.

This command has the following format:

```
SYSTem:COMMunicate:CSAC:MODE <ON|OFF>
```

This command will query the state of this command:

```
SYSTem:COMMunicate:CSAC:MODE?
```

SYSTem:COMMunicate:RSRGPS:BAUD <9600 | 19200 | 38400 | 57600 | 115200>

This command sets the serial speed for GPS port of the Micro-Transcoder port. The serial configuration is always 8 bit, 1 stop bit, no parity, no flow control. The factory default setting is 115200 baud.

This command has the following format:

```
SYSTem:COMMunicate:RSRGPS:BAUD <9600 | 19200 | 38400 | 57600  
| 115200>
```

This command will query the state of this command:

```
SYSTem:COMMunicate:RSRGPS:BAUD?
```

SYSTem:COMMunicate:RSRGPS:MODE <ON|OFF>

This command will enable or disable the communication for GPS port of the Micro-Transcoder port.

This command has the following format:

```
SYSTem:COMMunicate:RSRGPS:MODE <ON|OFF>
```

This command will query the state of this command:

```
SYSTem:COMMunicate:RSRGPS:MODE?
```

SYSTem:COMMunicate:MCE:BAUD <9600 | 19200 | 38400 | 57600 | 115200>

This command sets the serial speed for MCE serial port. The serial configuration is always 8 bit, 1 stop bit, no parity, no flow control. The factory default setting is 115200 baud.

This command has the following format:

```
SYSTem:COMMunicate:MCE:BAUD <9600 | 19200 | 38400 | 57600 |  
115200>
```

This command will query the state of this command:

```
SYSTem:COMMunicate:MCE:BAUD?
```

SYSTem:COMMunicate:MCE:MODE <ON|OFF>

This command will enable or disable the communication for MCE serial port.

This command has the following format:

```
SYSTem:COMMunicate:MCE:MODE <ON|OFF>
```

This command will query the state of this command:

```
SYSTem:COMMunicate:MCE:MODE?
```

SYSTem:POWer <SLEEP|DSLEEP|PDOWN|DPDOWN>

This command configures the power-down state of the microprocessor. The possible states include: sleep, deep sleep, power down mode, or deep power down mode.

The SLEEP command puts the device into a power saving state where all actions are stopped. Normal operations and full power can be resumed within a few seconds. Use the deep sleep command, DSLEEP, to turn the clock generator off. The processor does not need to keep its cache coherent, but maintains other states. The processor will take a bit longer to wake up. The PDOWN command will power down the device and clear the state of RAM. Use the DPDOWN command to put the device into a deep power down state for minimal power consumption.

SYSTem:COMMunicate:DLOAD <STL|GNSS|TRANScoder>

This command will enable the PNT-440 device to enter download/programming mode for the specified serial port. The ports that can be used for downloading are: STL, GNSS, or TRANScoder SCPI ports. The unit will require manual reset after the download is completed.

This command has the following format:

```
SYSTem:COMMunicate:DLOAD <STL|GNSS|TRANScoder>
```

This command will query the state of this command:

```
SYSTem:COMMunicate:DLOAD?
```

SYSTem:STATus?

This query returns a full page of GNSS status in ASCII format. The output is compatible with the GPSCon Windows program discussed in [Chapter 3 “GPSCon Utility”](#). The `SYST:STAT?` command is one of the most useful single commands to query as it contains a wealth of information.

This command is used by the GPSCon application.

SYSTem:FACToryReset ONCE

This command applies the Factory Reset setting to the EEPROM. All aging, tempco, and user parameters are overwritten with factory default values.

SYSTem:CPURESET

This command causes the PNT-440 device processor to reset.

SYSTem:ISP

This command causes the PNT-440 device processor to reset into In System Programming (ISP) mode for firmware upgrades. Please see [“Firmware Upgrade” on page 113](#) for details on updating the firmware in ISP mode.

Once the system is placed into ISP mode it will need to be power-cycled or an ISP-Reset command will need to be sent through the NXP ISP programming interface commands to establish a normal operating mode.

SERVO Subsystem

This subsystem regroups all the commands related to the adjustment of the servo loop:

- `SERVo?`
- `SERVo:MODE <SLOW|MEDium|FAST|AUTO>`
- `SERVo:MODE:FASTDURation <int> [60, 604800]`
- `SERVo:MODE:MEDDURation <int> [60, 604800]`
- `SERVo:DACGain <float> [0.001, 10000]`
- `SERVo:EFCScale <float> [0.0, 500.0]`
- `SERVo:EFCScale:MEDium <float> [0.0, 500.0]`
- `SERVo:EFCScale:FAST <float> [0.0, 500.0]`
- `SERVo:EFCDamping <int> [2, 4000]`
- `SERVo:EFCDamping:MEDium <int> [2, 4000]`
- `SERVo:EFCDamping:FAST <int> [2, 4000]`

- `SERVo:SLOPe <NEG|POS>`
- `SERVo:TEMPCompensation <float> [-4000.0, 4000.0]`
- `SERVo:AGINGcompensation <float> [-10.0,10.0]`
- `SERVo:PHASECOrrrection <float> [-500.0,500.0]`
- `SERVo:PHASECOrrrection:MEDium <float> [-500.0,500.0]`
- `SERVo:PHASECOrrrection:FAST <float> [-500.0,500.0]`
- `SERVo:COARSedac <int> [0,255]`
- `SERVo:1PPSoffset <int> <ns> [-5000000,5000000]`
- `SERVo:TRACe <int> [0,255]`
- `SERVo:FASTlock <int> [1,20]`
- `SERVo:FALEnGth <int> [100,20000]`

SERVo:FASTlock <int> [1,20]

The FASTlock command enables the FASTLOCK mode, and sets its gain parameter. Fastlock works by momentarily multiplying the EFCScale gain to a value determined by this `SERVo:FASTlock` parameter. Gain values of 1x to 20x can be set, with a gain of 1x effectively disabling the FASTLOCK feature.

By selecting gain values of > 1, the PLL loop parameter Proportional gain (`SERVo:EFCscale`) will be increased after power on, thus increasing loop aggressiveness and improving lock PLL time. It is not desirable to maintain a high loop gain for longer than necessary to lock the PLL since high loop gains come at the expense of increased phase noise (reduced short term stability). The FASTLOCK mechanism will automatically reduce the FASTLOCK gain over a period of time specified by the `SERVo:FALEnGth` command, during which time the FASTLOCK gain is slowly decreased from its initial value to 1.0x.

Setting the FASTLOCK gain to 2 for example will result in the Proportional gain value stored in the `SERVo:EFCscale` parameter to be multiplied by 2x initially after power on.

This dynamic gain is slowly reduced until the gain is back to 1.0x, the value stored in the `SERVo:EFCscale` parameter. For example if we set `SERVo:FASTlock` to 2, and `SERVo:FALEnGth` to 3600, and `SERVo:EFCscale` is set to 0.7, initially the unit will multiply the `EFCscale` by 2x, and an effective `EFCscale` value of 1.4 is applied to the PLL loop.

This increased gain value difference will be reduced every second by 1/3600, so that the gain after two seconds would be: 1.3998, until after 3600 seconds the gain has been reduced back to its long term value of 0.70 as stored in the `SERVo:EFCscale` parameter.

Disabling the FASTLOCK mode is accomplished by setting the `SERVo:FASTlock` to 1. This will set the dynamic gain to 1.0, effectively disabling the fastlock feature.

This command has the following format:

```
SERVo:FASTlock <int> [1,20]
```

This command will query the state of this command:

```
SERVo:FASTlock?
```

SERVo:FALEngth <int> [100,20000]

This command adjusts the length of time during which the FASTLOCK feature is active. For more information, see the “[SERVo:FASTlock <int> \[1,20\]](#)” on page 98.

The length can be set from 100 seconds to 20000 seconds. The Dynamic FASTLOCK gain is slowly reduced until it reaches a gain of 1.0 after the FALEngth period of seconds. During this time the PLL loop gain is increased by the amount specified in the `SERVo:FASTlock` parameter, which will result in a faster initial phase lock to UTC after power-on, while giving the lowest possible noise floor (best short term stability) during normal operation.

This command has the following format:

```
SERVo:FALEngth <int> [100,20000]
```

This command will query the state of this command:

```
SERVo:FALEngth?
```

SERVo:COARSedac <int> [0,255]

This command sets the coarse DAC that controls the EFC. The PNT-440 device control loop automatically adjusts this setting. The user should not have to change this value, however it is very useful to measure loop stability when experimenting with alternate SERVO parameters by inserting a large frequency step into the OCXO, then monitoring the behavior of the PI loop as it corrects for the synthetic frequency error.

This command has the following format:

```
SERVo:COARSedac <int> [0,255]
```

This command will query the state of this command:

```
SERVo:COARSedac?
```

SERVo:MODE <SLOW|MEDIum|FAST|AUTO>

This command is useful in setting the filter loop time constants for different mission profiles to accommodate different usage scenarios of the CSAC GPSDO. The loop time constant of the CSAC can be chosen with these settings.

The AUTO mode lets the firmware determine by itself which loop time constant is used, thus optimizing the units' performance dynamically depending on the environmental effects on the CSAC. The AUTO setting will quickly react to phase perturbations and will try to set `NORMal` settings (long time constant) whenever the unit is in stable conditions to minimize residual noise, and to improve the ADEV performance of the

filtered output as much as possible for a particular environment. NORMa1 settings are useful in stationary applications, whereas FAST settings are preferred for mobile applications such as in vehicles, man-packs, or aircraft.

The following command returns the current mode setting:

```
SERVo:MODE?
```

SERVo:MODE:FASTDURation <int> [60, 604800]

This command sets the minimum duration for remaining in the fast filter state when the `SERVo:MODE AUTO` setting is enabled. This setting is useful for preventing cycling between filter states with an unstable reference.

The following command returns the current setting:

```
SERVo:MODE:FASTDURation?
```

SERVo:MODE:MEDDURation <int> [60, 604800]

This command sets the minimum duration for remaining in the medium filter state when the `SERVo:MODE AUTO` setting is enabled. This setting is useful for preventing cycling between filter states with an unstable reference.

The following command returns the current setting:

```
SERVo:MODE:MEDDURation?
```

SERVo:STATe?

This query responds with the current filter state as configured by the `SERVo:MODE` command. When the `SERVo:MODE` setting is SLOW, MEDIUM or FAST, the `SERVo:STATe?` response matches the mode setting. When the `SERVo:MODE` setting is AUTO, the `SERVo:STATe?` responds with the automatically selected state--SLOW, MEDIUM or FAST.

SERVo:DACGain <float> [0.001, 10000]

This command is used for factory setup ONLY.

This command will query the state of this command:

```
SERVo:DACGain?
```

SERVo:EFCScale <float> [0.0,500.0]

Controls the Proportional part of the PID loop. Typical values are 0.7 (double oven OCXO) to 6.0 (simple single oven OCXO). Larger values increase the loop control at the expense of increased noise while locked. Setting this value too high can cause loop instabilities.

This command has the following format:

```
SERVo:EFCScale <float> [0.0,500.0]
```

This command will query the state of this command:

```
SERVo:EFCScale?
```

SERVo:EFCScale:MEDium <float> [0.0,500.0]

This command controls the SERVo:EFCScale setting when the servo state as reported by the SERVo:STATe? query is MEDIUM. The servo state and this setting only apply for the STL synchronization source.

SERVo:EFCScale:FAST <float> [0.0,500.0]

This command controls the SERVo:EFCScale setting when the servo state as reported by the SERVo:STATe? query is FAST. The servo state and this setting only apply for the STL synchronization source.

SERVo:EFCDamping <int> [2,4000]

This command sets the Low Pass filter effectiveness of the DAC. Values from 2 to 50 are typically used. Larger values result in less noise at the expense of phase delay.

This command has the following format:

```
SERVo:EFCDamping <int> [2,4000]
```

This command will query the state of this command:

```
SERVo:EFCDamping?
```

SERVo:EFCDamping:MEDium <int> [2,4000]

This command controls the SERVo:EFCDamping setting when the servo state as reported by the SERVo:STATe? query is MEDIUM. The servo state and this setting only apply for the STL synchronization source.

SERVo:EFCDamping:FAST <int> [2,4000]

This command controls the `SERVo:EFCDamping` setting when the servo state as reported by the `SERVo:STATE?` query is `FAST`. The servo state and this setting only apply for the STL synchronization source.

SERVo:SLOPe <NEG|POS>

The parameter determines the sign of the slope between the EFC and the frequency variation of the OCXO. This parameter should be set to match the OCXO's EFC frequency slope.

This command has the following format:

```
SERVo:SLOPe <NEG|POS>
```

This command will query the state of this command:

```
SERVo:SLOPe?
```

SERVo:TEMPCompensation <float> [-4000.0, 4000.0]

This command is not currently supported in PNT-440 device. This parameter represents the value needed to compensate the temperature of the TCXO in holdover mode.

This command has the following format:

```
SERVo:TEMPCompensation [-4000.0, 4000.0]
```

This command will query the state of this command:

```
SERVo:TEMPCompensation?
```

SERVo:AGINGcompensation <float> [-10.0,10.0]

This parameter is a coefficient that represents the drift of the EFC needed to compensate the natural drift in frequency of the OCXO due to aging. This coefficient is automatically computed and adjusted over time by the firmware. This command has the following format:

```
SERVo:AGINGcompensation <float> [-10.0,10.0]
```

This command will query the state of this command:

```
SERVo:AGINGcompensation?
```

SERVo:PHASECOrrrection <float> [-500.0,500.0]

This parameter sets the Integral part of the PID loop. Loop instability will result if the parameter is set too high. Typical values are 10.0 to 30.0. This command has the following format:

```
SERVo:PHASECOrrrection <float> [-500.0,500.0]
```

This command will query the state of this command:

```
SERVo:PHASECOrrrection?
```

SERVo:PHASECOrrrection:MEDium <float> [-500.0,500.0]

This command controls the `SERVo:PHASECOrrrection` setting when the servo state as reported by the `SERVo:STATe?` query is MEDIUM. The servo state and this setting only apply for the STL synchronization source.

SERVo:PHASECOrrrection:FAST <float> [-500.0,500.0]

This command controls the `SERVo:PHASECOrrrection` setting when the servo state as reported by the `SERVo:STATe?` query is FAST. The servo state and this setting only apply for the STL synchronization source.

SERVo:1PPSoffset <int> <ns> [-5000000,5000000]

This command sets the PNT-440 device 1PPS signal's offset to UTC in 5.55ns steps. The maximum configurable 1PPS offset is +/- 5ms.

Using the `SERVo:1PPS` command results in immediate phase change of the 1PPS output signal.

This command has the following format:

```
SERVo:1PPSoffset <int> <ns> [-5000000,5000000]
```

This command will query the state of this command:

```
SERVo:1PPSoffset?
```

SERVo:SLOPe <NEG|POS>

The parameter determines the sign of the slope between the EFC and the frequency variation of the OCXO. This parameter should be set to match the OCXO's EFC frequency slope and should not be changed from the factory setting.

This command has the following format:

`SERVo:SLOPe <NEG|POS>`

This command will query the state of this command:

`SERVo:SLOPe?`

SERVo:QUIet <ON|OFF>

This command enables/disables readings done to the ADC value in continuous mode.

This command has the following format:

`SERVo:QUIet <ON|OFF>`

This command will query the state of this command:

`SERVo:QUIet?`

SERVo:DBGMode <int> [0, 255]

This command sets the debug level that enables/disables debug messages to print for debugging ONLY. A level of 0 indicates no debug messages will print and a level greater than 0 will begin printing debug messages.

This command has the following format:

`SERVo:DBGMode [0, 255]`

This command will query the state of this command:

`SERVo:DBGMode?`

SERVo:TRACe <int> [0,255]

This command sets the period in seconds for the debug trace. Debug trace data can be used with Ulrich Bangert's "Plotter" utility or Excel etc. to show UTC tracking versus time etc.

This command has the following format:

`SERVo:TRACe <int > [0,255]`

This command will query the state of this command:

`SERVo:TRACe?`

An example output is described here:

```
08-07-31 373815 60685 -32.08 -2.22E-11 14 10 6 0x54 [date][1PPS
Count][Fine DAC][UTC offset ns][Frequency Error Estimate][Sats
Visible][Sats Tracked][Lock State][Health Status]
```

See “[SYNChronization:HEALTH?](#)” on page 74 for detailed information on how to decode the health status indicator values. The Lock State variable indicates one of the states in [Table 16](#).

Table 13 Lock state variables

Value	State
0	OCXO warmup
1	Holdover
2	Locking (OCXO training)
4	[Value not defined]
5	Holdover, but still phase locked (stays in this state for about 100s after GNSS lock is lost)
6	Locked, and GNSS active.

SERVo?

This command returns the result of the following queries:

- `SERVo:MODE?`
- `SERVo:STATe?`
- `SERVo:MODE:FASTDURation?`
- `SERVo:MODE:MEDDURation?`
- `SERVo:DACGain?`
- `SERVo:EFCScale?`
- `SERVo:EFCScale:MEDium?`
- `SERVo:EFCScale:FAST`
- `SERVo:PHASECOrrrection?`
- `SERVo:PHASECOrrrection:MEDium?`
- `SERVo:PHASECOrrrection:FAST?`
- `SERVo:EFCDamping?`
- `SERVo:SLOPe?`
- `SERVo:COARSedac?`
- `SERVo:TEMPCompensation?`
- `SERVo:AGINGcompensation?`
- `SERVo:1PPSoffset?`
- `SERVo:TRACe?`
- `SERVo:FASTlock?`
- `SERVo:FALEngth?`

Firmware Upgrade

The following topics are discussed in this chapter:

- [“Introduction” on page 114](#)
- [“Putting the device into In-Circuit Programming \(ISP\) mode” on page 114](#)
- [“Downloading the firmware” on page 114](#)
- [“Using the JLTterm programming terminal” on page 114](#)
- [“Verifying the firmware update” on page 116](#)

Introduction

The following sections provide instructions on how to upgrade the PNT-440 device firmware. Please follow the instructions in order to prevent accidentally corrupting the PNT-440 device Flash.

Putting the device into In-Circuit Programming (ISP) mode

Issuing the SCPI command `SYST:ISP` discussed in “[SYSTem:ISP](#)” on page 104 in a terminal program such as JLTerm causes the board to reset into ISP mode from normal operation. See “[Using the JLTerm programming terminal](#)” on page 114 for more information about the JLTerm programming terminal.

Downloading the firmware

Download the latest version of PNT-440 device firmware from the VIAVI support page and store it in a place that will be remembered. The firmware file should be in .hex format.

The unit needs to be connected to the computer’s USB serial port prior to firmware download.

Using the JLTerm programming terminal

- 1 Download the JLTerm application from <https://www.viavisolutions.com/en-us/software-download/jlterm-software>
- 2 Install and open the JLTerm application.
- 3 Select the COM port in JLTerm as needed on your PC.

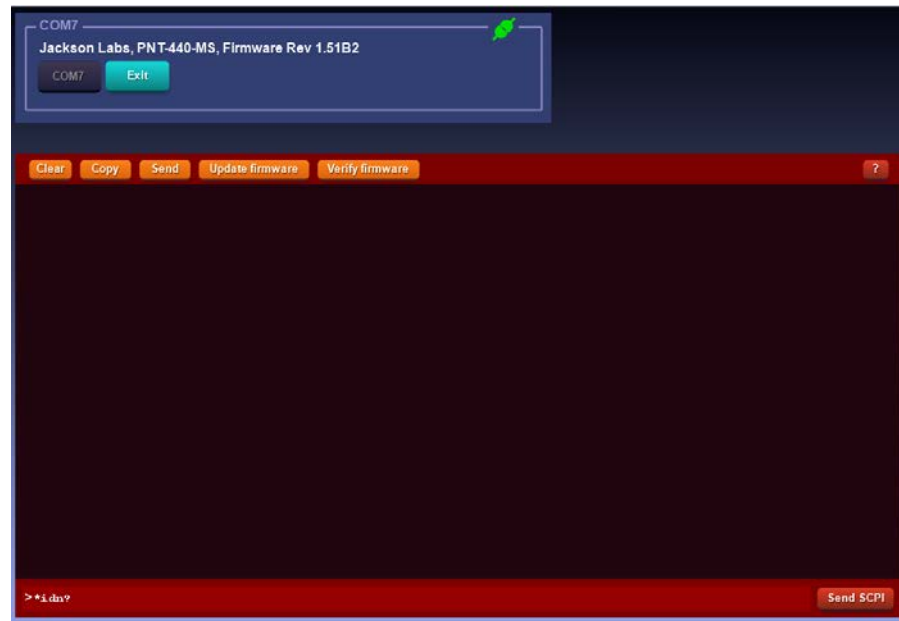
Once a successful connection is established, the connection icon becomes green, as shown in [Figure 16](#). The PNT-440 must be in normal operation with working SCPI communication prior to JLTerm connection.



NOTE

If there is no valid response from the COM port, check for valid driver and port number for the COM port in the Device Manager on your computer. The COM poer number may be conflicting with another COM port device. Each device should have a different COM port. Ensure GPSCon is not running in the background and using the same COM port.

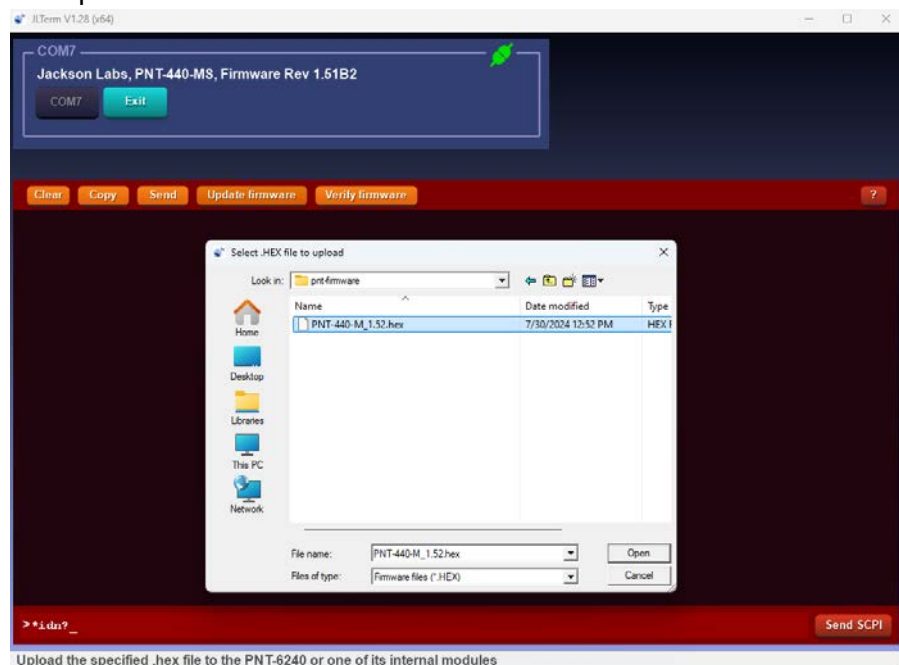
Figure 16 Successful connection



If necessary, change the COM port number, install or update the driver, and unplug and reconnect the serial cable, then try to establish a connection again.

- 4 Once the device is connected in JLTerm, click the orange **Update device firmware...** button.
- 5 Choose the correct hex file to program the device and click **Open** in the pop-up, as shown in [Figure 17](#). The firmware automatically downloads and the board is reset.

Figure 17 Open Hex file and download firmware



If an error occurs during the programming process, check the USB cable connection.

[Figure 18](#) shows an error in programming.

Figure 18 Error in programming



Verifying the firmware update

During power on, the unit sends an ID string out of the serial port at 115200 Baud by default. The firmware version can also be queried by sending the *IDN? command. Verify that the firmware version is the version that was downloaded.



22183212
R001, September 2025
English

Viavi Solutions

North America:	1.844.GO VIAVI / 1.844.468.4284
Latin America	+52 55 5543 6644
EMEA	+49 7121 862273
APAC	+1 512 201 6534
All Other Regions:	viavisolutions.com/contacts
email	TAC@viavisolutions.com