Micro-JLT GNSS Disciplined Oscillator

From Jackson Labs Technologies, acquired by VIAVI Solutions

About this guide

This guide provides general information about the Micro-JLT GPSDO, including.

Technical Assistance Center and Knowledge Base

To find the Technical Assistance Center phone number and email in your region, or to search the VIAVI Solutions Knowledge Base, visit the VIAVI Solutions Technical & Product Support site at support.viavisolutions.com.

Preface

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

- "Purpose and scope" on page vi
- "Assumptions" on page vi
- "Related Information" on page vi
- "Conventions" on page vi
- "Safety and compliance information" on page viii
- "Technical assistance" on page ix

Purpose and scope

This manual is intended to help you use the capabilities of the PNT-62xx Assured Position, Navigation, and Timing Reference.

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).	`$NANGT_DATA_DIR/results` (directory) – `test_products/users/defaultUser.xml` (file name) – `All results okay.` (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: **$BASEDIR/startup/npiu_init restart** Type: **a:\set.exe** in the dialog box.
References to guides, books, and other publications appear in *this typeface*.	Refer to *Newton's Telecom Dictionary.*
Command line option separators.	platform [a|b|e]
Optional arguments (text variables in code).	login [platform name]
Required arguments (text variables in code).	<password>

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, *will* 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, *could* 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-62xx.

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.

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 an 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 guide iii

Technical Assistance Center and Knowledge Base . . . iii

Preface v

Purpose and scope . . . vi
Assumptions . . . vi
Related Information . . . vi
Conventions . . . vi
Safety and compliance information . . . viii
  California Proposition 65 . . . viii
  Federal Communications Commission (FCC) . . . ix
  Product Environmental Compliance . . . ix
  EU REACH . . . ix
  Additional standards compliance . . . ix
Technical assistance . . . ix

Table of contents xi

Chapter 1 Introduction 1

Overview . . . 2
General Safety Precautions . . . 3
  Antenna lightning protector . . . 3
  Grounding . . . 3
  Power Connections . . . 3
  Environmental Conditions . . . 3

Chapter 2 Quick-Start Instructions 7

Powering Up the Unit . . . 8
  PCB photos . . . 9
  Mechanical drawings . . . 10
  Notes on Signal Interfacing and J1 Connector . . . 13
  Coaxial connectors . . . 13
Power . . . 14

Concurrent-GNSS Capabilities 14
Connecting to the GNSS antenna 15
Remote serial control 15
Loop parameter adjustment 16

Chapter 3 SCPI-Control Quick Start Instructions 19

Introduction 20
General SCPI commands 20
  *IDN? 21
  HELP? 21
GPS Subsystem 21
  GPS:SATellite 22
  GPS:SATellite:TRAcking:COUNt? 22
  GPS:SATellite:VISible:COUNt? 22
  NMEA Support 22
  GPS:GPGGA <int> [0,255] 24
  GPS:GGASTat <int> [0,255] 24
  GPS:GPGLL <int> [0,255] 25
  GPS:GPRMC <int> [0,255] 25
  GPS:GPGSA <int> [0,255] 25
  GPS:GPGSV <int> [0,255] 26
  GPS:XYZSPeed <int> [0,255] 26
  GPS:GPVTG <int> [0,255] 27
  GPS:GPZDA <int> [0,255] 27
  GPS:PASHR <int> [0,255] 27
  GPS:PJLTV <int> [0,255] 29
  GPS:GYRO 30
  GPS:GYRO:CAL 30
  GPS:DYNAMic:MODE <int> [0,7] 30
  GPS:DYNAMic:MODE 8 (Automatic Dynamic Mode) 31
  GPS:DYNAMic:STATe? 32
  GPS:REFerence:ADELay <float> <s | ns > [-32767ns,32767ns] 33
  GPS:REFerence:PULse:SAWtooth? 33
  GPS:RESET ONCE 33
  GPS:TMODe <ON|OFF|RSTSURV> 34
  GPS:SURVey ONCE 34
  GPS:SURVey:DURation <sec> 34
  GPS:SURVey:VARIANCE <mm^2> 34
  GPS:HOLD:POSition <cm, cm, cm> 35
  GPS:SURVEY:STATUS? 35
  GPS:INITial:DATE <yyyy,mm,dd> 35
  GPS:INITial:TIME <hour,min,sec> 35
  GPS:SYSTem:SELect [GPS | SBAS | QZSS | GAL | BD ^ GLO] 35
  GPS:JAMlevel? 36
  GPS:FWver? 36
  GPS? 36
  GYRO SUBSYSTEM 36
  GYRO:MODE <ON|OFF> 36

GYRO:TRACE <int> [0,255] . . . 37
GYRO:CAL <float,float,float,float,float,float> . . . 37
GYRO:CAL:COMPUTE . . . 37
GYRO:SENS, GYRO:EFC, and GPS:CAL:RESET . . . 37
GYRO:GLOAD? . . . 37
PTIME Subsystem . . . 37
PTIMe:DATE? . . . 38
PTIMe:TIME? . . . 38
PTIMe:TIME:STRing? . . . 38
PTIMe:TINTerval? . . . 38
PTIMe:OUTput <ON|OFF> . . . 38
PTIMe:LEAPsecond? . . . 39
PTIMe:LEAPsecond:PENDing? . . . 39
PTIMe:LEAPsecond:ACCumulated? . . . 39
PTIMe:LEAPsecond:DATE? . . . 40
PTIMe:LEAPsecond:DURation? . . . 40
PTIMe:STAMp Subsystem . . . 40
PTIMe:STAMp:MODE <OFF|ASYnc|MEMory|OUTput|DIFFer> . . . 42
PTIMe:STAMp:SOURce <GPS|EXTernal|ALL> . . . 43
PTIMe:STAMp:DIFFer <ALL|EXT|GPS> . . . 44
PTIMe:STAMp:FILTer [OFF | REL | DUR | PPS] . . . 44
PTIMe:STAMp:FILTer:PPSecond <int> [1,100] . . . 46
PTIMe:STAMp:FILTer:1PPeriod <int> [1,604800] . . . 47
PTIMe:STAMp:LAST? . . . 48
PTIMe:STAMp:MEMory? . . . 48
PTIMe:STAMp:RESET . . . 49
PTIMe:STAMp? . . . 49
PTIMe? . . . 49
SYNChronization Subsystem . . . 50
SYNChronization:HOLDover:DURation? . . . 50
SYNChronization:HOLDover:INITiate . . . 51
SYNChronization:HOLDover:RECovery:INITiate . . . 51
SYNChronization:SOURce:MODE <GPS|EXTernal|AUTO> . . . 51
SYNChronization:SOURce:MODE:EDGE <NEGative|POSitive> . . . 52
SYNChronization:SOURce:STATE? . . . 52
SYNChronization:HOLDover:STATE? . . . 53
SYNChronization:TINTerval? . . . 53
SYNChronization:TINTerval:THReshold <int> [50,2000] . . . 53
SYNChronization:IMMEdiate . . . 54
SYNChronization:FEEstimate? . . . 54
SYNChronization:LOCKed? . . . 54
SYNChronization:OUTput:1PPS:RESET <ON|OFF> . . . 54
SYNChronization:OUTput:1PPS:RESET? . . . 54
SYNChronization:OUTput:1PPS:WIDTH <int> <ms | us> [100us, 500ms] . . . 54
SYNChronization:HEAlth? . . . 55
SYNChronization? . . . 56
DIAGnostic Subsystem . . . 56
DIAGnostic:ROSCillator:EFControl:RELative? . . . 56
DIAGnostic:ROSCillator:EFControl:ABSolute? . . . 56

DIAGnostic:LIFetime:COUNt? . . . 57
DIAGnostic? . . . 57
MEASURE Subsystem . . . 57
MEASure:VOLTage? . . . 57
MEASure:POWersupply? . . . 57
MEASure:POWersupply:VOSCillator? . . . 57
MEASure? . . . 58
SYSTEM Subsystem . . . 58
SYSTem:COMMunicate:SERial:ECHO <ON|OFF> . . . 58
SYSTem:COMMunicate:SERial:PROmpt <ON|OFF> . . . 58
SYSTem:COMMunicate:SERial:BAUD <9600|19200|38400|57600|115200> . . . 58
SYSTem:STATus? . . . 59
SYSTem:FACToryReset ONCE . . . 59
SYSTem:CPURESET . . . 59
SYSTem:ISP . . . 59
SERVO Subsystem . . . 59
SERVo:FASTlock <int> [1,20] . . . 60
SERVo:FALEngth <int> [100,20000] . . . 61
SERVo:COARSedac <int> [0,255] . . . 61
SERVo:DACGain . . . 61
SERVo:EFCScale <float> [0.0,500.0] . . . 61
SERVo:EFCDamping <int> [2,4000] . . . 62
SERVo:SLOPe <NEG|POS> . . . 62
SERVo:TEMPCOmpensation . . . 62
SERVo:AGINGcompensation <float> [-10.0,10.0] . . . 62
SERVo:PHASECOrrection <float> [-500.0,500.0] . . . 63
SERVo:1PPSoffset <int> <ns> [-5000000,5000000] . . . 63
SERVo:TRACe <int> [0,255] . . . 63
SERVo? . . . 64

Chapter 4 Firmware Upgrade Instructions 65

Introduction . . . 66
ISP Flash Loader Utility installation . . . 66
Putting the PCB into In-Circuit Programming (ISP) mode . . . 66
Downloading the firmware . . . 67
4.4.1 Using the JLTerm Programming Terminal . . . 67
Using the Flash Magic programming utility . . . 70
Using the Flash Magic Classic version . . . 72
Verifying firmware update . . . 74

Chapter 5 GPSCon Utility 77

Description . . . 78
Installation . . . 78
Using GSPCon . . . 78
Setting options . . . 78
Sending manual commands to the receiver . . . 81
Using the mouse in the Graph display . . . 82

Exporting graphics . . . 83
Interpreting the Data . . . 84

Chapter 6 Certification and Warranty 87

Certification . . . 88
Warranty . . . 88
Limitation of Warranty . . . 88
Exclusive Remedies . . . 88

1

Introduction

The following information is discussed in this chapter:

- "Overview" on page 2
- "General Safety Precautions" on page 3

Overview

The Micro-JLT GNSS^{TM} board supports three concurrent multi-GNSS tracking by using a latest generation concurrent GPS/Glonass/Galileo/BeiDou/QZSS/ SBAS capable fixed-position GNSS timing receiver, as well as optimizes phase noise performance, power supply consumption, cost, and lead-time, while adding differential interfacing for robustness in adverse environmental conditions. The Micro-JLT GNSS^{TM} is designed for fast warm-up with optimum stability with excellent ADEV and Phase Noise performance for high-volume, low-cost applications that can encounter the harshest environments.

Figure shows the GNSS DOCXO GNSDO.

Figure 1 GNSS SOCXO GNSDO

The Micro-JLT GNSS^{TM} DNSDO uses a multi-GNSS receiver that can run in stationary (Position Hold) mode, as well as 3D mobile mode to discipline either a single oven or optional double oven OCXO local oscillator to better than 0.1 ppb frequency accuracy. It supplies one 10 MHz output with ultra-low phase-noise, exceptinoal ADEC performance, and very low spurs and harmonics. The board includes an RS-422/TTL control port for NMEA and SCPI communication, and an LVDS 1PPS timing output, driven by the on-board ultra-low-jitter holdover oscillator. The board operates from a single +5.5V nominal supply with 5.2V to 6.0V range (a 5.0V +/-0.2V operating range can be optionally ordered), and has connector pins to indicate GNSS and oscillator LOCK and ALARM events. The board includes a high-performance GNSS receiver that can acquire and track up to 72 GNSS signals down to a state-of-the-art -167dBm (GPS and GLOSNASS), a low-power 32-bit processor that runs a Real Time OS, one 10 MHz +8.5dBm Sine Wave output, 1PPS UTC synchronized LVDS output, RS-422/TTL serial control interface, and precision voltage references and DACs. The board also includes the latest generation multi-level power supply management and conditioning for low power-consumption and extremely high power-supply noise and stability rejection ratios.

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 wth specific warnings elsewhere in this manual violates the safety standards of design manufacture, and intended use of the instrument. Jackson Labs Technologies, Inc. assumes no liability for the customer's failure to comply with these requirements.

Antenna lightning protector

Always use a UL approved and properly installed GNSS Antenna Lightning protector on the coaxial GNSS antenna feed to prevent damage, injury, and/or death in case of a lightning strike.

Grounding

To avoid damaging the sensitive electronic components in the Micro-JLT GNSS™ GNSDO 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 Up 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.

2

Quick-Start Instructions

The following information is discussed in this chapter:

- "Powering Up the Unit" on page 8
- "Power" on page 14
- "Concurrent-GNSS Capabilities" on page 14

Powering Up the Unit

Perform the following steps to operate the unit:

1 Connect a 3.3V-compatible GNSS antenna to SMA connector J2.

The GNSS board supplies a current-limited 3.3V power to the antenna.

> **NOTE**
>
> Use an appropriate lightning surge suppressor on the antenna cable.

2 Connect +5.5V (-0.3V/+5.0V) DC power to J1 pin 10 and pun 12 on the unit, and ground to pin 2 and pin 9 of J1.

The unit consumes up to 8Q of power during oscillator warm-up.

The unit will now lock to GNSS signals (Red LED D2 blinks to indicate that the GNSS receiver has a proper fix) and indicates proper lock and no events pending when the Green LED D3 goes on, with the Red LED D2 internittenly blinking as well. Once the green LED D3 is on, the unit iptups 10MHz with significantly better than 1ppb frequency accuracy.

> **NOTE**
>
> By default the unit will be set to Position Hold mode, and an antenna position Auto Survey process is started after power-on that may take up to 3 hours to finish. The antenna should never be moved when the unit is set to run in Position Hold mode as this would result in loss of frequency, timing accuracy, and incorrect GNSS fixes. The unit must be configured to mobile 3D mode using the GPS:TMODe OFF command when used in an application where the antenna position can move during normal operation.
>
> Using the Auto Survey process and Position Hold mode allows the unit to over-determine the timing solution. It also allows TRAIM operation which will result in a higher timing stability and more robust GNSS reception in challenged environmental conditions such as under foliage, in urban canyons, etc.
>
> **NOTE**
>
> The GNSS receiver establishes the internal antenna gain right after power-on. For proper operation, the GNSS antenna should always be connected prior to turning on the +5.5V prime mover.

Connect a terminal program (TeraTerm is recommended) to the unit via an RS-422 or TTL TTL, (the choice depends on your order option), to a USB serial connector to the appropriate pins on J1 with a setting of 115.2KB 8N1 and no flow-control. You may also download the GPSCon program discussed in “GPSCon Utility” on page 77 from the JLT support website page for free.

> **WARNING**
>
> Do not connect RS-232 serial levels to connector J1. The unit may become damaged from RS-232 serial levels on connector J1 as the default configuration is RS-422 differential or TTL levels.

Try some of the following SCPI commands:

- HELP?
- SYSTem:STATus?
- GPS?
- SYNChronization?
- MEASure?
- DIAGnostic?
- *IDN?

See "SCPI-Control Quick Start Instructions" on page 19 for a full list of SCPI commands.

PCB photos

Figure shows the Micro-JLT GNSS™ PCB connectors and indicators.

Figure 2 Micro-JLT GNSS™ Single Oven PCB

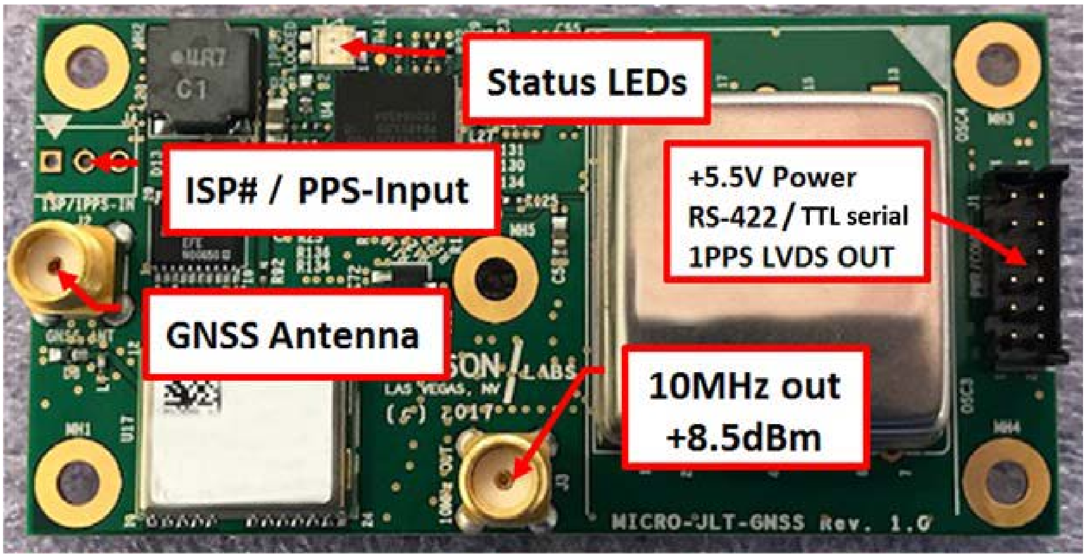


Figure shows a Micro-JLT GNSS™ unit with a double oven OCXO.

Figure 3 Micro-JLT GNSS^{TM} Double Oven PCB

Mechanical drawings

Figure 4 shows the Micro-JLT GNSS^{TM} mechanical dimensions.

Figure 4 Micro-JLT GNSS^{TM} mechanical dimensions

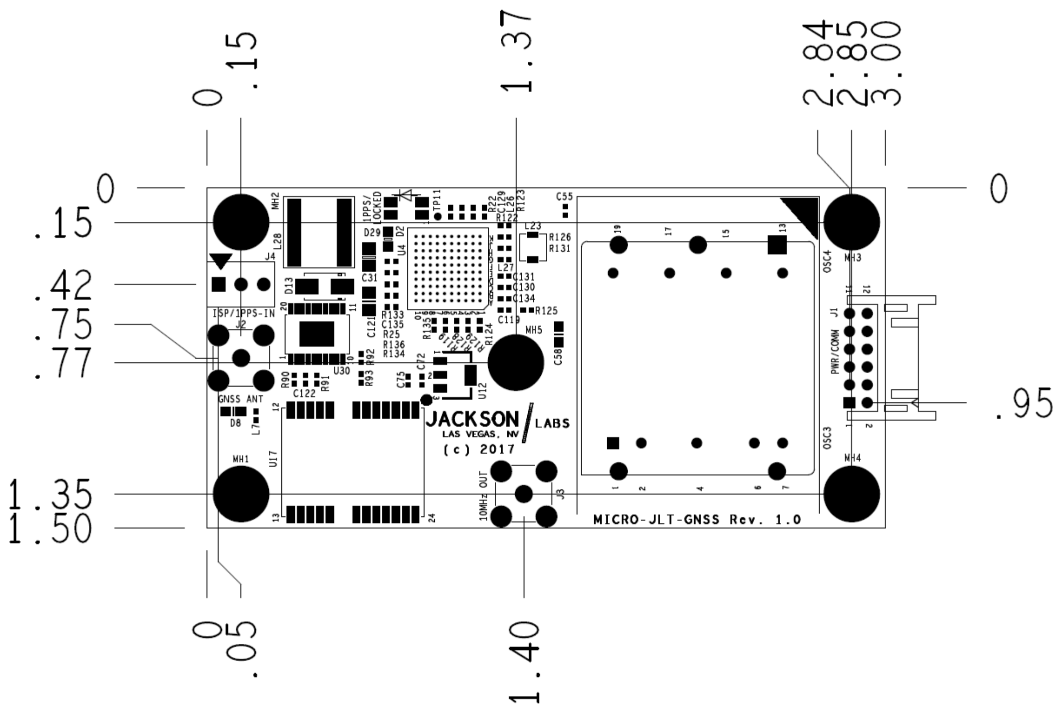


Table 1 describes the Micro-JLT GNSS^{TM} hardware pins.

Table 1 Hardware pins

Ref	Name	Function	Specification	Description
J1 Pin 1	ISP#	Enables In System Programming mode (ISP)	0V to 3.3V, pulled-high by 4.7K resistor	Pull to Ground during power-on to set processor into ISP mode. Leave floating during normal operation
J1 Pin 2	Ground	Ground	Ground	Ground
J1 Pin 3	1PPS OUT-P	LVDS 1PPS positive signal output	LVDS level, positive signal	Flywheel OCXO-generated 1PPS output in LVDS level
J1 Pin 4	1PPS OUT-N	LVDS 1PPS negative signal output	LVDS level, negative signal	Flywheel OCXO-generated 1PPS output in LVDS level
J1 Pin 5	TX-Serial-P	RS-422 positive signal output/TXD_TTL	RS-422 level, positive signal/ TTL level	RS-422 positive signal/ TTL NMEA and SCPI communications port output from unit
J1 Pin 6	TX-Serial-N	RS-422 negative signal output	RS-422 level, negative signal	RS-422 NMEA and SCPI communications port output from unit, negative signal
J1 Pin 7	RX-Serial-P	RS-422 positive signal input/ RXD_TTL	RS-422 level, positive signal/ TTL level	RS-422 positive signal/ TTL SCPI communications port input to unit
J1 Pin 8	RX-Serial-N	RS-422 negative signal input	RS-422 level, negative signal	RS-422 SCPI communications port input to unit, negative signal
J1 Pin 9	Ground	Ground	Ground	Ground
J1 Pin 10	+5.5V Power	Prime Power Input	+5.5V, -0.3V +0.5V max	Prime Power input. 2.6W typical steady-state, up to 8W during warmup. Connect to J1 pin 12

Table 1 Hardware pins

Ref	Name	Function	Specification	Description
J1 Pin 11	LOCK-OK/ EVENT#	Indicates Event, Alarm, and LOCK-OK conditions	3.3V == unit locked and healthy 0V == Event or Alarm exist	Output will turn on (3.3V) when unit is warmed up, locked to GNSS, and healthy. Output will turn off (0V) when an event is pending, and alarm happened, or no GNSS signal is present. Query the SYNC:HEALTH? indicator for more details on the event/alarm state.
J1 Pin 12	+5.5V Power	Prime Power Input	+5.5V, -0.3V +0.5V max	Prime Power input. 2.6W typical steady-state, up to 8W during warmup. Connect to J1 pin 10.
J2	GNSS Antenna	Antenna Input	L1, +3.3V output, <=40mA	Supplies 3.3V to an external active GNSS antenna
J3	10MHz output	10MHz Sine Wave Output	+8.5dBm, +/-0.5 dBm	10MHz output driven by on-board OCXO. Does not require a load if unused.
J4 Pin 1	ISP#	Enables ISP mode. Wired in paralell to J1 Pin 1	0V to 3.3V, pulled-high by 4.7K resistor	Pull to Ground during power-on to set processor into ISP mode. Leave floating during normal operation. May be momentarily shorted during power-on to Pin 2 of J4 using tweezers etc.
J4 Pin 2	Ground	Ground	Ground	Ground

Table 1 Hardware pins

Ref	Name	Function	Specification	Description
J4 Pin 3	External PPS reference Input	Optional PPS External Reference input	-0.5V to 5.0V with a recommended <10ns rise/fall time on the signal. Terminated by 5K pull-down resistor.	Optional external PPS reference input in CMOS TTL format.

Notes on Signal Interfacing and J1 Connector

Connector J1 is a straight-up, 12-pin dual row 2mm spacing Hirose connector part number: **DF11-12DP-2DSA(01)**. A right angle type board-to-board connector may also be optionally stuffed at the factory for high-volume custom orders. Pin out descriptions are listed in Table 1.

The optional LOCK_OK_OUT signal on J1 pin 11 is a 3.3V LVCMOS signal, and thus require a series resistor of typically 390 to 470 Ohms when used to drive an external signalling LED. The serial port RX and TX lines cannot be directly connected to a DB9 connector as these are driven with RS-422/TTL signal levels. Connecting these pins to an RS-232 serial interface may damage the board. External RS-422/TTL to RS-232 converters may be used (RS-485 converters will typically not work as they are bi-directional and power-down when no transmission is active). The output of the RS-422/TTL to RS-232 converter may then be connected to standard RS-232 USB connector adaptors. Signalling on the RS-422 serial port should be done in differential mode, although the RS-422 TX positive output (TXD_TTL) on pin 5 of connector J1 is equivalent to a standard 3.3V TTL serial output, and may be used as such when used without termination.

If the Micro-JLT GNSS™ is ordered with the TTL option, pin 5 of connector J1 (TXD_TTL), pin 7 of connector J1 (RXD_TTL), and a ground (pin 2 or 9 of connector J1) pin should be connected to the TTL-to-RS-232 / TTL-to-USB converter.

Connect an active GNSS GPS/Glonass antenna that is compatible to 3.3V antenna power with typically between 10dB to 40dB gain and less than 1.5dB NF, and connect this antenna prior to turning-on the board. Use a lightning arrestor on the antenna feed for safety and to prevent damage, injury, or death in case of a lightning strike.

Coaxial connectors

The GNSS antenna connector and the 10MHz connector on the Micro-JLT GNSS^{TM} board are generic SMA female types. Optionally the female MCX connector type can be ordered.

Power

The unit is powered from a +5.2V to +6V max. DC source, with +5.5V nominal voltage. The current is typically less than 0.6A at 5.5V. Connect a clean +5.5V power supply to pins 10 and 12 of J1, and ground to pin 2 and 9 of J1. The unit may also be optionally ordered with 5.0V +/-10% supply range for large volume custom orders.

Concurrent-GNSS Capabilities

The Micro-JLT GNSS™ is capable of simultaneously receiving up to three concurrent GNSS systems at one time. Concurrent GNSS operation aids performance by allowing reception of up to 72 GNSS satellites in challenged reception areas such as in urban canyons, under foliage, indoors, or close to the earth's poles, etc. Using multiple GNSS systems also increases robustness by not relying on a single GNSS system. Several of the systems operate at different carrier frequencies, so using multiple GNSS systems can increase immunity against jamming which often occurs at only one carrier frequency.

The multiple GNSS systems each have their own reference time and representation of UTC. For example, GPS uses the GPS time standard and provides a method to convert GPS time to UTC as defined by the US Naval Observatory (USNO). GLONASS, Galileo and BeiDou have similar reference times and UTC definitions. The Micro-JLT GNSS™ automatically adjusts for offsets between the different GNSS time reference standards and synchronizes the 1PPS output to UTC (USNO) when GPS is being received. If GPS is not being received, the 1PPS output is synchronized to the best representation of UTC (USNO) available.

The selection of GNSS systems is made with the GPS:SYST:SEL command as detailed in "GPS:SYSTem:SELect [GPS | SBAS | QZSS | GAL | BD ^ GLO]" on page 35. As these systems operate at different carrier frequencies with different bandwidths, it is necessary to use a GNSS antenna that is designed to receive all the required GNSS systems.

NOTE

The new and emerging Galileo system is now functional, and uses the same carrier frequency as GPS L1, albeit with a wider bandwidth. Galileo sats can be received with good C/No carrier to noise figures (>40dB) with standard legacy GPS antennae and distribution amplifiers/splitters. JLT thus recommends enabling at a minimum GPS and Galileo concurrently with the command: GPS:SYST:SEL GPS GAL. Using other GNSS systems such as Glonass will require an antenna system designed to support Glonass signals.

NOTE

As of November 2019, JLT received multiple reports on SBAS satellites causing timing issue on the GPSDO units around the world. Until further notice, JLT recommends disabling the SBAS in GPS:SYST:SEL command to avoid potential timing issue or monitoring the timing stability of Micro-JLT GNSS™ unit if SBAS is desired.

The internal GNSS receiver can track up to three different GNSS systems concurrently, such as GPS, Galileo, Glonass, and SBAS at the same time, however only two different carrier frequencies may be received at any given time, so BeiDou and Glonass cannot be enabled concurrently when either GPS, Galileo, or QZSS are simultaneously enabled as that would require tracking three different carrier frequencies.

Attempting to configure an invalid combination of GNSS systems with the GPS:SYST:SEL command will result in a Command Error response with no change to the configuration.

Connecting to the GNSS antenna

Connect the GNSS antenna to the SMA connector J2. Caution: use a Lightning Arrestor on your Antenna setup. Use an amplified GNSS antenna that is 3.3V LNA compatible. The Micro-JLT GNSS™ GNSDO GNSS receiver is a 72 channel high-sensitivity multi-GNSS receiver with very fast lock time.

> **NOTE**
>
> The Micro-JLT GNSS™ unit is factory set to use Position Hold mode, and will initiate an Auto Survey process after power-on to establish its new position. The Auto Survey process may last up to three hours. The unit antenna should not be moved when set to Position Hold mode. Set the unit to 3D Mobile Mode using the GPS:TMODe OFF command as described in “GPS:TMODe <ON|OFF|RSTSURV>” on page 34 when the unit is to be used in a mobile environment where the antenna position can change more than 1 foot during operation.

The Micro-JLT GNSS™ is capable of generating standard navigation messages (see GPS:GPGGA, GPS:GPZDA, GPS:GPGSV, GPS:PASHR, and GPS:GPRMC RS-422/TTL serial commands) that are compatible with most GPS-based navigation software.

The GNSS receiver generates a 1PPS time signal that is phase synchronized to UTC(USNO). This 1PPS signal is used to frequency-lock the 10MHz output of the Micro-JLT GNSS™ GNSDO to UTC, thus disciplining the unit’s 10MHz frequency output to the US Naval master clock for very high frequency accuracy (typically better than 10 digits of frequency accuracy when locked to GNSS signals).

Remote serial control

The unit is controlled via the RS-422/TTL serial port at 115200 baud, 8N1. Other baud rates can be set via the SCPI command found in “SYSTem:COMMunicate:SERial:BAUD <9600|19200|38400|57600|115200>” on page 58.

Attach the Micro-JLT GNSS™ unit to your PC’s terminal program, the cost-free Jackson Labs GPSCon software package available on the JLT website support page (see “GPSCon Utility” on page 77), or a third-party free-ware Windows-based application program called Z38xx also available on the JLT website support page. The GPSCon and Z38xx programs can be used to control and track the performance of the Micro-JLT GNSS™.

Loop parameter adjustment

All loop parameters can be controlled via the RS-422/TTL serial port.

Loop parameters are optimized for the OCXO on the board, and changing the factory settings may result in the unit's performance to deteriorate. By default the settings are optimized for quick frequency error correction and slow phase offset error correction, favoring the highest possible frequency accuracy and ADEV performance over fast phase error corrections.

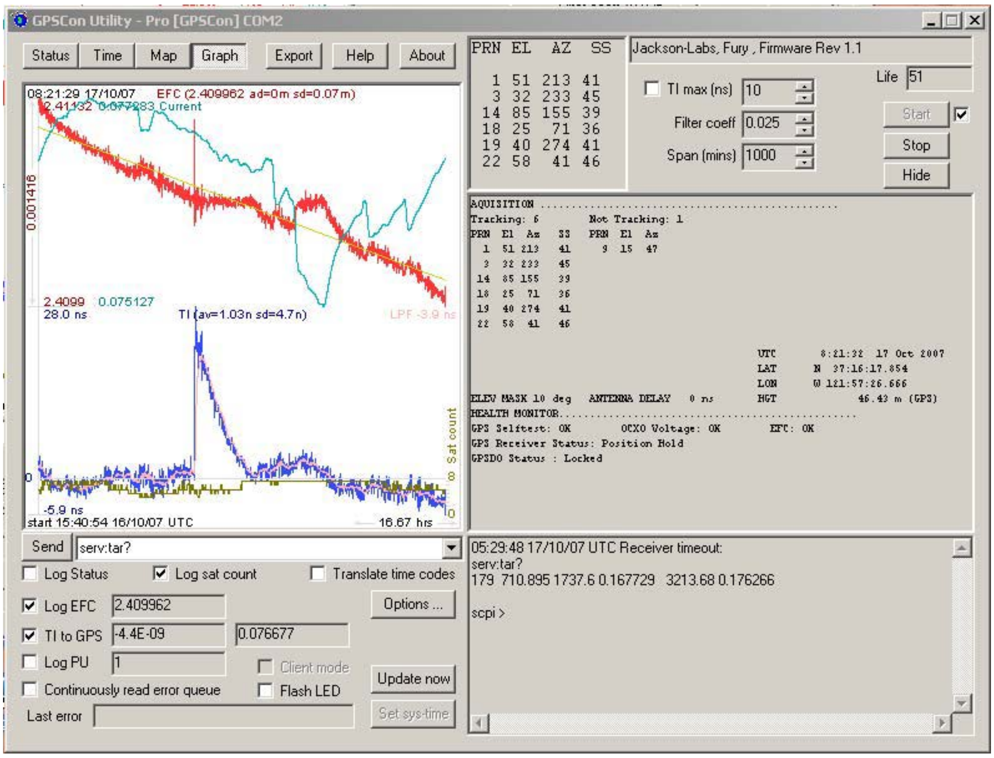
The commands to control the loop parameters are part of the servo? command. See also the SERVO Subsystem section below.

The individual commands are:

- **EFC Scale**: this is the proportional gain of the PID loop. Higher values will give quicker convergence, and faster locking of the GPS time (lower loop time constant), lower values give less noise. Values between 0.7 (good double oven OCXO) and 6.0 (simple single-oven OCXO) are typical.
- **EFC Damping**: overall IIR filter time constant. higher values increase loop time-constant. Jackson Labs Technologies, Inc. typically uses values between 10 to 50. Setting this value too high may cause loop instability.
- **Phase compensation**: this is the Integral part of the PID loop. This corrects phase offsets between the Micro-JLT GNSS™ 1PPS signal and the UTC 1PPS signal as generated by the GNSS receiver. Set higher values for tighter phase-following at the expense of frequency stability. Typical values range from 4 - 30. Setting this value too high may cause loop instability.

A well-compensated unit will show performance similar to the plot shown in Figure 5 when experiencing small perturbations:

Figure 5 Micro-JLT GNSS™ phase compensation plot



3

SCPI-Control Quick Start Instructions

The following information is discussed in this chapter:

- "Introduction" on page 20
- "General SCPI commands" on page 20
- "GPS Subsystem" on page 21
- "PTIME Subsystem" on page 37
- "SYNChronization Subsystem" on page 50
- "DIAGnostic Subsystem" on page 56
- "MEASURE Subsystem" on page 57
- "SYSTEM Subsystem" on page 58
- "SERVO Subsystem" on page 59

Introduction

The SCPI (Standard Commands for Programmable Instrumentation) subsystem is accessed via the RS-422/TTL serial interface and a terminal program. By default the terminal settings are:

- 115200
- 8N1
- no flow-control.

There are a number of commands that can be used as listed below. Most of these are identical or similar to Symmetricom 58503A commands. To get a listing of the available commands, send the `HELP?` query. This will return a list of all the available commands for the Micro-JLT GNSS™ GNSDO.

Additional information regarding the SCPI protocol syntax can be found on the following web site:

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 of this chapter are used for ranging or separating ONLY. Do not include these symbols when sending the SCPI commands to avoid Command Error response.

General SCPI commands

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

NOTE

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

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

*IDN?

This query outputs an identifying string. The response will show the following information:

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

The serial number field represents the unique identifier of the 32-bit processor.

An example response of this command is provided below:

```
Jackson Labs, Micro-JLT, 0D02001C AAB22D23 57FD8E0F F50007C6,
0.46
```

HELP?

This query returns a list of the commands available for the Micro-JLT GNSS™ GNSDO.

GPS Subsystem

> **NOTE**
> The Micro-JLT GNSS™ displays antenna height in MSL Meters rather than in GPS Meters on all commands that return antenna height. The NMEA position fixes are in the WGS84 coordinate system, while the X,Y, and Z 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 receiver. The list of the commands supported is the following:

- `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> [0,7]`
- `GPS:DYNAMic:MODE?`
- `GPS:DYNAMic:MODE 8 (Automatic Dynamic Mode)`
- `GPS:DYNAMic:STATe?`
- `GPS:GPRMC <int> [0,255]`

- `GPS:GPGGA <int> [0,255]`
- `GPS:GGASTat <int> [0,255]`
- `GPS:GPGLL <int> [0,255]`
- `GPS:PJLTS <int> [0,255]`
- `GPS:PJLTV <int> [0,255]`
- `GPS:XYZSPeed <int> [0,255]`
- `GPS:GPVTG <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:FWver?`
- `GPS:INITial:DATE <yyyy,mm,dd>`
- `GPS:INITial:TIME <hour,min,sec>`
- `GPS:SYSTem:SELect [GPS | SBAS | QZSS | GAL | BD ^ GLO]`

GPS:SATellite

This group of commands describe the satellite constellation.

GPS:SATellite:TRAcking:COUNt?

This query returns the number of satellites being tracked.

GPS:SATellite:VISible:COUNt?

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

NMEA Support

The following NMEA commands allow the Micro-JLT GNSS™ GNSDO to be used as an industry standard navigation GNSS receiver. The GGA, VTG, 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 Micro-JLT GNSS™ serial interface.

Micro-JLT GNSS™ will send out the enabled NMEA information on the RS-422/TTL serial transmit pins automatically every N seconds. The Micro-JLT will not send out NMEA messages if the 1PPS output is not yet enabled. See "SYNChronization:OUTput:1PPS:RESET <ON|OFF>" on page 54 for descriptions of 1PPS generation and instructions to enable 1PPS output upon power on. All incoming serial commands are still recognized by Micro-JLT GNSS™ 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 solution and both messages use a modified satellite numbering scheme as detailed in the table below that makes all satellite numbers unique. Also, to indicate the GNSS systems enabled and used to generate the NMEA data, the Micro-JLT GNSS™ GNSDO's NMEA output includes a two character talker ID before the GGA, VTG, GLL, RMC, GSA, GSV, and ZDA sentence headers. Table 2 shows the talker IDs for the supported GNSS systems. If more than one GNSS system is enabled, the talker ID of NMEA output is GN except for GSV sentence which outputs multiple sets of sentences for each main talker ID. Refer to "GPS:GPZDA <int> [0,255]" on page 27 for example GSV output format. Also, the GSV output uses a modified satellite numbering scheme as detailed in Table 2 to allow all satellites in different GNSS systems to be differentiated.

Table 2 PRN numbering scheme for GNSS systems

GNSS Type	SV Range	GPGSC PRN vehicle numbering	Talker ID
GPS	G1-G32	1-32	GP
SBAS	S120-S158	33-64, 152-158	GP
Galileo	E1-E36	301-336	GA
BeiDou	B1-B37	401-437	GB
IMES	I1-I10	173-182	GP
QZSS	Q1-Q5	193-197	GP
GLONASS	R1-R32, R?	65-96, 0	GL

In addition to standard NMEA sentences, the Micro-JLT GNSS™ GNSDO also supports PJLTS and PJLTV Jackson Labs proprietary NMEA sentences, PASHR and a proprietary version of the GGA sentence (GGASTat) described in "GPS:GGASTat <int> [0,255]" on page 24.

NOTE

The position, direction, and speed data is delayed by one second from when the GNSS receiver internally reported these to the Micro-JLT GNSS™ Microprocessor, so the position is valid for the 1PPS signal previous to the last 1PPS signal at the time the dat is sent (one second delay.) The time and date are properly output with the 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 Micro-JLT GNSS™ 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]
```

The GPGGA string has the following data format:

```
$GNGGA,hhmmss.00,llll.llll,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 or systems as described in “NMEA Support” on page 22 . The GGA message shows height in MSL Meters, this is different from 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:GGASTat <int> [0,255]

This command instructs the Micro-JLT GNSS™ 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:GGASTat <int> [0,255]

The GGASTat string has the following data format:

$GNGGA,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:PHASECOrrection <float> [-500.0,500.0]” on page 63 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 GGASTat output message header includes the talker ID for the currently enabled GNSS system or systems as described in “NMEA Support” on page 22. GGASTat output shows height in MSL Meters, this is different from 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 Micro-JLT GNSS™ 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]

The GPGLL string has the following data format:

$GNGLL,llll.llll,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 or systems as described in “NMEA Support” on page 22.

GPS:GPRMC <int> [0,255]

This command instructs the Micro-JLT GNSS™ 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]
```

The GPRMC string has the following data format:

```
$GNRMC,hhmmss.00,s,llll.llll,S/N,yyyyy.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 or systems as described in “NMEA Support” on page 22.

GPS:GPGSA <int> [0,255]

This command instructs the Micro-JLT GNSS™ 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.

This command has the following format:

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

The GPGSA string has the following data format:

```
$GNGSA,a,a,x,x,x,x,x,x,x,x,x,x,x,x,x,x,x.x,x.x,x.x*[checksu
m]
```

The GSA output message header includes the talker ID for the currently enabled GNSS system or systems as described in “NMEA Support” on page 22.

GPS:GPGSV <int> [0,255]

This command instructs the Micro-JLT GNSS™ 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]
```

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 systems as described in “NMEA Support” on page 22 .

> **NOTE**
>
> Due to the large number of GNSS satellites that cab be tracked in this 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: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]
```

The XYZSP string has the following data format:

```
$XYZSP VX VY VZ ... Leapseconds: aaaaa bbbbb ccccc ddd
eeeeeeeee ff
```

where:

aaaaa is the velocity in the x direction (cm/s),

bbbbb 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:GPVTG <int> [0,255]

This command instructs the Micro-JLT GNSS™ to send the NMEA standard VTG 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:GPVTG <int> [0,255]

The GPVTG string has the following data format:

$GNVTG,x.x,T,x.x,M,x.x,N,x.x,K,m*[checksum]

The VTG output message header includes the talker ID for the currently enabled GNSS system or systems as described in “NMEA Support” on page 22.

GPS:GPZDA <int> [0,255]

This command instructs the Micro-JLT GNSS™ 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]
```

The GPZDA string has the following data format:

```
$GNZDA,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 22.

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,????,000.00,000.01,-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,eeeee.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
- bbbbbb.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
- ll.l: PDOP
- mm.m HDOP
- nn.n VDOP
- 00.0: Static number
- p.pp: Firmware Version

This command instructs the Micro-JLT GNSS™ 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]
GPS:PJLTS <int> [0,255]
```

This command instructs the Micro-JLT GNSS™ to send the proprietary NMEA string $PJLTS every N seconds, with N in the interval [0,255]. This proprietary command includes information on the GNSDO lock, oscillator status, and other telemetry status. The following is example output 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]
```

The format of the $PJTLS command is:

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

where:

- aaa.aa is the filtered UTC offset in ns,
- bbb.bb is the raw UTC offset in ns from the time interval counter,
- ccc is the number of captured input 1PPS signals,
- d is the lock status,
- e.eeeeeee 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 55.

GPS:PJLTV <int> [0,255]

This command instructs the Micro-JLT GNSS™ 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 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 $PJTLV command is:

$PJLTV,aaaaa,bbbbb,ccccc,ddd,eeeeeeeee,ffff,gg*[checksum]0x0d 0x0a

where:

- aaaaa is the velocity in the x direction (cm/s)
- bbbbb 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
- ffff is the GPS week number
- gg is the current UTC leap second offset in seconds

GPS:GYRO

This command is not supported in the Micro-JLT GNSS™.

GPS:GYRO:CAL

This command is not supported in the Micro-JLT GNSS™.

GPS:DYNAMic:MODE <int> [0,7]

This command allows the user to select the dynamic motion model being applied to the Kalman filters in the GNSS receiver. 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 Micro-JLT GNSS™ will select a stationary GNSS dynamic model, for example:

GPS:DYNAM:MODE 1

Table lists all available dynamic modes.

Table 3 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.

Table 3 Supported dynamic GNSS operating modes

Value	Model	Application
8	Automatic Mode	Select one of the above states (0 – 7) based on the actual velocity of the vehicle.

The Micro-JLT GNSS™ GNSDO uses a GNSS receiver that is capable of running in a stationary mode with Position Auto Survey called Position Hold Mode. This mode increases timing stability by storing the position into memory, and solving the GNSS signal only for time as the position is not expected to change. Two modes can be selected for Auto Survey operation (see “GPS:TMODe <ON|OFF|RSTSURV>” on page 34 for a description of the GPS:TMODe command):

1. Manually setting Timing Mode to ON with a hard-coded position in NV memory
2. Enabling Auto Survey to start automatically after power-on by setting Timing Mode to RSTSURV

If either one of the above two GPS:TMODe Auto Survey/Position Hold modes is selected, the GPS:DYNAMIC:MODE command is disabled internally and its setting is ignored as the unit does not expect any motion on the antenna. In this case, the dynamic state as programmed into the GNSS receiver is set to STATIONARY independent of the user selection for GPS:DYNAMIC:MODE.

The current dynamic state being applied to the GNSS receiver can be queried with the following 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 Micro-JLT GNSS™ 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> [0,7]” on page 30.

Table 3.3 shows the Dynamic mode the unit will program into the GNSS receiver when Automatic Mode is selected (Dynamic Mode 8).

Table 4 Auto Dynamic Mode switching rules

Velocity Threshold	Selected Dynamic Model	Fallback to lower setting
0 – 2 knots	Stationary	none

Table 4 Auto Dynamic Mode switching rules (Continued)

Velocity Threshold	Selected Dynamic Model	Fallback to lower setting
>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

Please note that 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 cab be obtained with the command:

```
GPS:DYNAMic:STATe?
```

A value between 0 and 7 is then returned depending on vehicle dynamics.

The dynamic state is always set to STATIONARY if one of the Position Hold Auto Survey stationary modes is selected using the command GPS:TMODe, as the Position Hold mode setting overrides any dynamic state user setting.

Settings will be applied immediately to the GNSS receiver, and are stored in Non-volatile memory.

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 "GPS:DYNAMic:MODE 8 (Automatic Dynamic Mode)" on page 31.

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. This allows antenna cable delay compensation, aswell 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. The present offset between the 1PPS output and the UTC 1PPS signal can be queried with the command SYNC:TINT? and this offset should be taken into account when calibrating two unit’s 1PPS outputs to each other, since 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.

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 GNSS receiver.

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.

GPS:SURVey ONCE

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.

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.

GPS:HOLD:POSition <cm, cm, cm>

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.

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 JLT 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.

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 JLT 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.

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). Please see Section 2.3 for restrictions on the concurrent GNSS systems that can be enabled. 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 Micro-JLT GNSS™ will respond to the query with the list of enabled GNSS systems such as:

```
GPS SBAS GAL GLO
```

GPS:JAMlevel?

This command provides GNSS signal jamming-indicator. The GNSS receiver will detect, and flag jamming interference with levels ranging from 0 (no jamming) to 255 (strong jamming). Any level exceeding 50 and accompanied by a loss of the 1PPS signal from the GNSS receiver will cause a SYNC:HEALTH 0x800 event to be flagged, and the unit to disable the LOCK_OK signal on connector J1.

GPS:FWver?

This command queries and returns the Firmware version used inside the GNSS receiver itself.

GPS?

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

GYRO SUBSYSTEM

This command is not supported in the Micro-JLT GNSS™.

GYRO:MODE <ON|OFF>

This command is not supported in the Micro-JLT GNSS™.

GYRO:TRACE <int> [0,255]

This command is not supported in the Micro-JLT GNSS™.

GYRO:CAL <float,float,float,float,float,float>

This command is not supported in the Micro-JLT GNSS™.

GYRO:CAL:COMPUTE

This command is not supported in the Micro-JLT GNSS™.

GYRO:SENS, GYRO:EFC, and GPS:CAL:RESET

This command is not supported in the Micro-JLT GNSS™.

GYRO:GLOAD?

This command is not supported in the Micro-JLT GNSS™.

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:STAMp:MODE <OFF|ASYnc|MEMory|OUTput|DIFFer>
- PTIMe:STAMp:SOURce <GPS|EXTernal|ALL>
- PTIMe:STAMp:DIFFer <ALL|EXT|GPS>
- PTIMe:STAMp:FILTer [OFF | REL | DUR | PPS]

- PTIMe:STAMp:FILTer:PPSecond <int> [1,100]
- PTIMe:STAMp:FILTer:1PPeriod <int> [1,604800]
- PTIMe:STAMp:LAST?
- PTIMe:STAMp:MEMory?
- PTIMe:STAMp:RESET
- PTIMe:STAMp?
- PTIMe?

PTIMe:DATE?

This query returns the current calendar date. The local calendar date is referenced to UTC time. The year, month, and day are returned.

PTIMe:TIME?

This query 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 Jackson Labs Technologies, Inc. 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 35 and "GPS:REFerence:ADELay <float> <s | ns > [-32767ns,32767ns]" on page 33 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
```

Please note that the unit stores pending leapsecond events in NV memory, and applies them correctly even if the antenna is removed prior to the leapsecond as long as the pending information has been received and stored in NV memory, and the unit has a properly set RTC time and date. This feature sets JLT units apart from competitive products that only apply a leapsecond correctly if the GNSS antenna is connected to the unit during the actual leapsecond event, and may thus miss the leapsecond event entirely.

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 35 for the GPS Almanac to be available.

PTIMe:LEAPsecond:ACCumulated?

This command returns the internally applied leapsecond 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 any.

PTIMe:LEAPsecond:DURation?

This command returns the duration of the last minute of the day during a leap second event. 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.

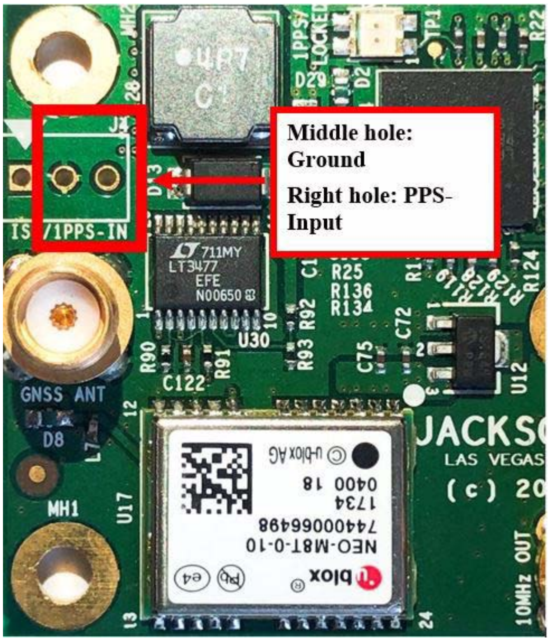
PTIMe:STAMp Subsystem

This subsystem is unrelated to OCXO disciplining. See "SYNChronization Subsystem" on page 50 for more information on OCXO disciplining.

The following commands are implemented for time-stamping management of GPS and external PPS sources. The PPS input pins are shown in Figure 6 on page 41.

- `PTIMe:STAMp:MODE <OFF|ASYnc|MEMory|OUTput|DIFFer>`
- `PTIMe:STAMp:SOURce <GPS|EXTernal|ALL>`
- `PTIMe:STAMp:DIFFer <ALL|EXT|GPS>`
- `PTIMe:STAMp:FILTer [OFF | REL | DUR | PPS]`
- `PTIMe:STAMp:FILTer:PPSecond <int> [1,100]`
- `PTIMe:STAMp:FILTer:1PPeriod <int> [1,604800]`
- `PTIMe:STAMp:LAST?`
- `PTIMe:STAMp:MEMory?`
- `PTIMe:STAMp:RESET`
- `PTIMe:STAMp?`

Figure 6 PPS input pins



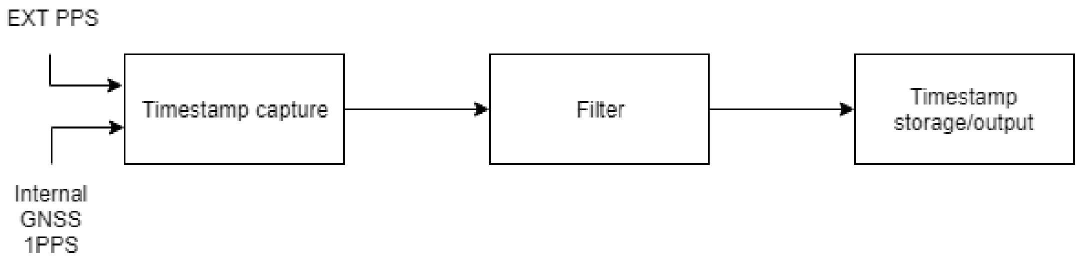
The Micro-JLT GNSS™ provides a time-stamping ability for an externally-supplied signal with TTL or CMOS 3.3V/5V input level compatibility. The external signal to be time-stamped will need to be applied to pin-3 of J4 with pin-2 of J4 being ground, as shown in Figure 3.1 above. Time-stamping can either be done on the rising or the falling edge of the externally-provided reference signal, as described in “SYNChronization:SOURce:MODE:EDGE <NEGative|POSitive>” on page 52.

A suitable 3-pin, 100-mil pin-spacing connector will need to be soldered onto the board to enable this optional feature. Please ensure that the externally supplied signal is in the range of -0.5V to 5.0V maximum at all times including under and overshoot, as lower or higher voltages can damage the PCB. The time-stamping input pin-3 of connector J4 is pulled low by an internal 5K Ohms resistor. The rise/fall times of the externally supplied signal to be time-stamped is recommended to be <10ns.

See Table 1 for pin descriptions of connector J4.

A block diagram is presented in “PTIMe:STAMp block diagram” on page 42 to demonstrate the PTIME subsytem time-stamping capability. The process begins with EXT PPS and internal GNSS 1PPS signals as sources for time-stamping. These time-stamp sources are captured and alined to UTC time by using the GNSS 1PPS signal as a reference. Next, the time-stamp results may be filtered before output depending on the user’s preference. Please refer to “PTIMe:STAMp:FILTer [OFF | REL | DUR | PPS]” on page 44 for more information on filters. The time-stamps can be outputted to the RS-422/TTL serial interface. See “PTIMe:STAMp:MODE <OFF|ASYnc|MEMory|OUTput|DIFFer>” on page 42.

Figure 7 PTIMe:STAMp block diagram



PTIMe:STAMp:MODE <OFF|ASYnc|MEMory|OUTput|DIFFer>

The PTIMe:STAMp:MODE command is unrelated to OCXO disciplining. For more information on OCXO disciplining, please refer to “SYNChronization Subsystem” on page 50.

The board may be configured to record and store time-stamps of a pulse source into memory for later retrieval if any one of ASY, MEM, OUT, or DIFF modes is selected. The unit will store up to 100 time-stamps of the most-recent pulses, including the first and last pulses that arrived within a one-second interval, and discard all extra time-stamps if there are more than 100 time-stamps.

The unit can process a maximum of 10,000 pulses (10 KHz signal) during the one-second interval. Connecting a signal with higher frequency such as a 10 MHz reference to PPS input will result in truncated time-stamp recording. All stored time-stamps are in GPS time-of-week format, as in the example shown below, which contains the week number and week second with nanosecond precision synchronized to the best representation of UTC (USNO) available.

NOTE

The leapsecond offset provided in PTIMe:LEAPsecond:ACCumulated? response is applied to all stored time-stamps.

By default, PTIMe:STAMp:MODE is in OFF mode which disables the time-stamping of the pulses. When changed to the ASYnc mode, the Micro-JLT GNSS™ will output the time-stamp of each accepted pulse instantaneously after the pulse(s) arrive. Configuring the unit to the OUTput mode will print 100 of the most-recent time-stamps at once every second. The 100th element is the time-stamp of the most-recent pulse. The MEMory mode can be configured to process and store the time-stamps of pulses in background without outputting the time-stamps. PTIMe:STAMp:MEMory? command in can be used to retrieve the stored time-stamps once.

If the mode is switched to the DIFFer, the unit can display the time-stamp offset between GPS 1PPS and other PPS signal sources. The source(s) displayed in DIFF mode can be chosen using the PTIMe:STAMp:DIFFer command in "PTIMe:STAMp:DIFFer <ALL|EXT|GPS>" on page 44.

> **NOTE**
> The PTIMe:STAMp:SOURce ALL command will be configured automatically when enabling the DIFFer mode.

The following command will query the state of the PTIMe:STAMp:MODE command:

```
PTIMe:STAMp:MODE?
```

All time-stamps have the following format:

```
wwww, ssssss.sssssssss
```

Where:

- wwww = week number
- ssssss.sssssssss = week seconds with nanosecond precision

Below is an example of the results displayed for the PTIMe:STAMp:MODE ASYnc command:

```
GPS time-stamp: 2059, 520524.000000016
```

Below is few samples of the results displayed when using the PTIMe:STAMp:MODE OUTput or PTIMe:STAMp:MEMory? commands:

```
01: 2060, 167182.000000061; (first time-stamp)
. . .
100: 2060, 167236.000000306; (last time-stamp)
```

Below is an example of the results displayed for the PTIMe:STAMp:MODE DIFFer command:

```
EXT Offset: 266 ns
GPS Offset: 5 ns
```

PTIMe:STAMp:SOURce <GPS|EXTernal|ALL>

The PTIMe:STAMp:SOURce command is unrelated to OCXO disciplining. To select a source for OCXO disciplining, please refer to "SYNChronization:SOURce:MODE <GPS|EXTernal|AUTO>" on page 51.

This command configures the source of PPS signals for time-stamping ONLY. The time-stamping will only apply to PPS signals from configured source(s) in this command. By default, the time-stamp source is set to GPS ONLY and external PPS signals are discarded for time-stamping.

The following command will configure the unit to process PPS signals of both GPS and external PPS sources:

```
PTIMe:STAMp:SOURce ALL
```

> **NOTE**
> The the PTIMe:STAMp:SOURce ALL command will be configured automatically when enabling the `PTIMe:STAMp:MODE DIFFer` mode as described “PTIMe:STAMp:MODE <OFF|ASYnc|MEMory|OUTput|DIFFer>” on page 42.

The time-stamp source may also be hard-coded to external PPS source with the following command and the GPS 1PPS signal will be discarded for time-stamping:

```
PTIMe:STAMp:SOURce EXT
```

The following command will query the state of the PTIMe:STAMp:SOURce command:

```
PTIMe:STAMp:SOURce?
```

PTIMe:STAMp:DIFFer <ALL|EXT|GPS>

The PTIMe:STAMp:DIFFer command is unrelated to OCXO disciplining. For more information on OCXO disciplining, please refer to “SYNChronization Subsystem” on page 50.

This command selects the PPS offset source to print when the PTIMe:STAMp:MODE DIFF command from “PTIMe:STAMp:MODE <OFF|ASYnc|MEMory|OUTput|DIFFer>” on page 42 is selected. The default offset source is set to EXT. When EXT is the offset source, only the external time-stamp offset will be printed. If the GPS setting is selected, only the GPS time-stamp offset will be printed. If the ALL setting is selected, both GPS and EXT time-stamp offsets will be printed.

The following command will query the state of the PTIMe:STAMp:DIFFer command:

```
PTIMe:STAMp:DIFFer?
```

Below is an example of the results displayed for the PTIMe:STAMp:DIFF ALL command:

```
EXT Offset: 266 ns
GPS Offset: 5 ns
```

PTIMe:STAMp:FILTer [OFF | REL | DUR | PPS]

The PTIMe:STAMp:FILTer command is unrelated to OCXO disciplining. For more information on OCXO disciplining, please refer to “SYNChronization Subsystem” on page 50.

This command enables or disables the time-stamping filter. If any filter is enabled, the board will filter out matching time-stamps based on the type of filter. By default, the filter is in OFF mode which the filter is disabled allowing all time-stamps to be processed.

Multiple filters can be enabled at once, except the OFF mode such as the PTIMe:STAMp:FILTer REL DUR PPS command.

With the REL filter, the unit will compare the time-stamp of the incoming pulse against the average time-stamp of GPS 1PPS. If the incoming pulse arrives outside of the GPS 1PPS time window of +/- 500 ns, the incoming pulse will be ignored without storing. When enabling the REL filter, the unit will automatically enable the GPS 1PPS source for time-stamping. This filter is useful to filter out spurious pulses in an external PPS source while keeping the UTC/GPS time aligned 1PPS signal.

Below is an example of the results displayed for the PTIMe:STAMp:FILT REL command. A timing diagram for this example can be found in Figure 7 on page 42.

`GPS Average time-stamp: 2060, 168899.000000012` (the base comparison value)

`GPS time-stamp: 2060, 168899.000000038` (this incoming pulse is kept)

`EXT time-stamp: 2060, 168899.000000600` (this incoming pulse is filtered out)

If the DUR filter is enabled, the unit will compare the time-stamp of the incoming pulse to the time-stamp of the last arrived pulse of the same source. If the elapsed time between the incoming pulse and the last received pulse is less than 500ns of the same second, the incoming pulse will be ignored.

Below is an example of the results displayed for the PTIMe:STAMp:FILT DUR command. A timing diagram for this example can be found in Figure 9 on page 46.

`EXT time-stamp: 2059, 518862.999999977` (last received pulse)

`EXT time-stamp: 2059, 518863.000000227` (an incoming pulse that will be filtered)

When the PPS filter is enabled, the unit will only store M pulses every N seconds. The variable M is the stored pulses per second limit specified with the PTIMe:STAMp:FILTer:PPSecond command found in “PTIMe:STAMp:FILTer:PPSecond <int> [1,100]” on page 46. Variable N is the stored pulse period limit specified with the PTIMe:STAMp:FILTer:1PPeriod command found in “PTIMe:STAMp:FILTer:1PPeriod <int> [1,604800]” on page 47. By default, variable M is set to 1 and variable N is set to 1 so only the first pulse will be stored every one second (1PPS) if the PPS filter is enabled. Please refer to “PTIMe:STAMp:FILTer:PPSecond <int> [1,100]” on page 46 and “PTIMe:STAMp:FILTer:1PPeriod <int> [1,604800]” on page 47 for timing examples.

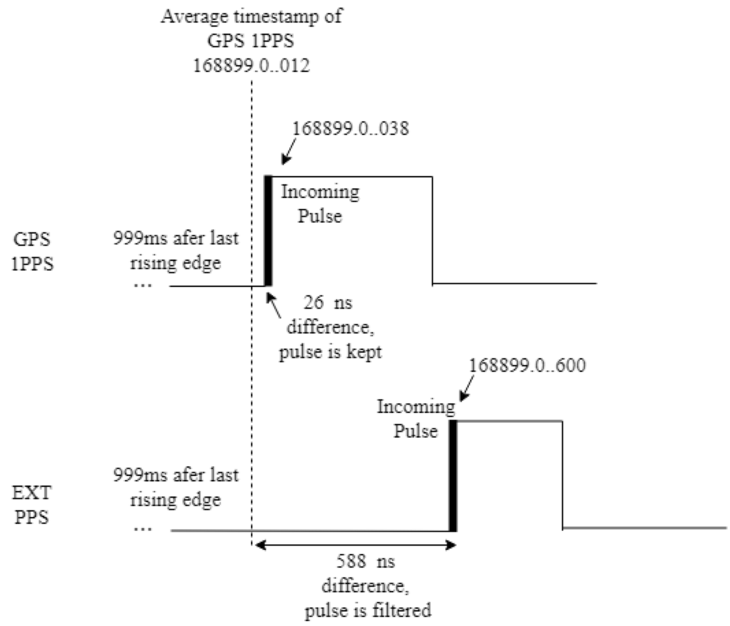
The following command will query the state of the PTIMe:STAMp:FILTer command:

```
PTIMe:STAMp:FILTer?
```

The following example command will enable REL, DUR, PPS filters all concurrently:

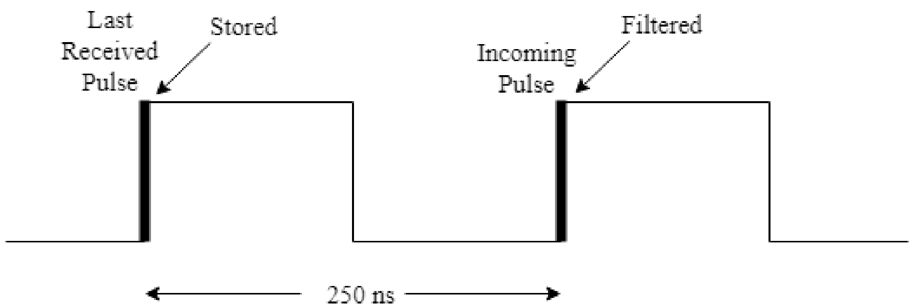
```
PTIMe:STAMp:FILTer REL DUR PPS
```

Figure 8 PTIMe:STAMP:FILT REL timing example



The EXT incoming pulse will be filtered out because it arrived outside of the GPS 1PPS 500 ns window.

Figure 9 PTIMe:STAMp:FILT DUR timing example



The incoming pulse will be filtered out because it arrived within a 500 ns window of the last arrived pulse.

PTIMe:STAMp:FILTer:PPSecond <int> [1,100]

The PTIMe:STAMp:FILTer:PPSecond command is unrelated to OCXO disciplining. For more information on OCXO disciplining, please refer to “SYNChronization Subsystem” on page 50.

This command specifies the maximum number of time-stamps that can be stored over any one second interval. However, this command will not be effective if the PTIMe:STAMp:FILT PPS filter described in “PTIMe:STAMp:FILTer [OFF | REL | DUR | PPS]” on page 44 is disabled. If this command is sent, then the pulse storage period specified in “PTIMe:STAMp:FILTer:1PPeriod <int> [1,604800]” on page 47 will be reset to one. The maximum number of pulses stored per second is 100 pulses.

The following command will query the state of the PTIMe:STAMp:FILTer:PPSecond command:

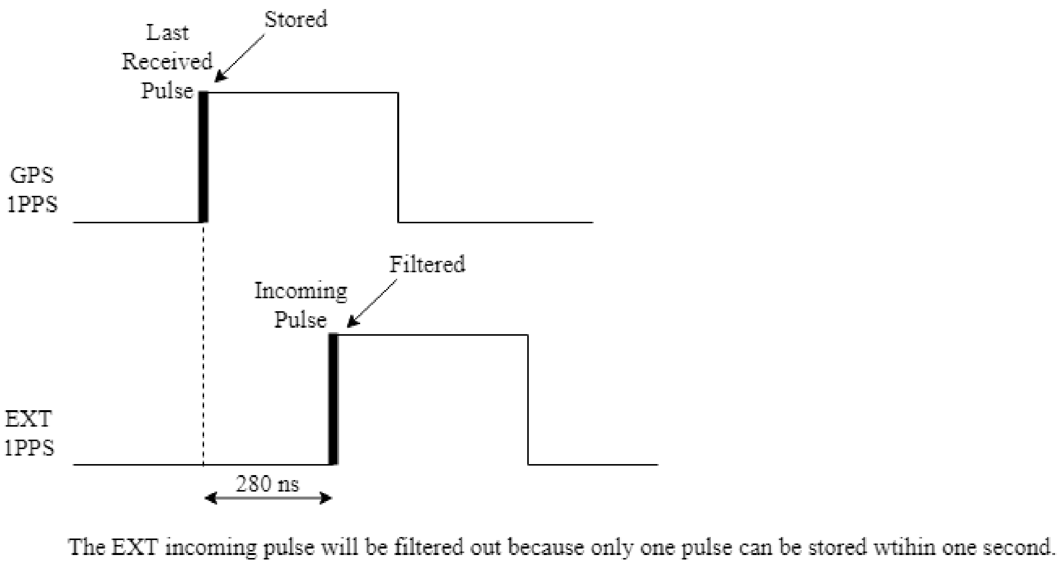
PTIMe:STAMp:FILTer:PPSecond?

Below is an example of the results displayed for the PTIMe:STAMp:FILTer:PPSecond 1 command. A timing diagram for this example can be found in Figure 10 on page 47.

`GPS time-stamp: 2060, 174275.000000025` (This time-stamp is stored)

`EXT time-stamp: 2060, 174275.000000560` (This time-stamp is filtered out with the PTIMe:STAMp:FILTer:PPSecond 1 command)

Figure 10 PTIMe:STAMp:FILT:PPSecond 1 timing example



PTIMe:STAMp:FILTer:1PPeriod <int> [1,604800]

The PTIMe:STAMp:FILTer:1PPeriod command is unrelated to OCXO disciplining. For more information on OCXO disciplining, please refer to "SYNChronization Subsystem" on page 50.

This command specifies the maximum number of time-stamp pulses stored within one pulse period. Only the first pulse time-stamp is stored and all other pulses will be ignored until the next pulse period. This command will not be effective if the PPS filter setting described in "PTIMe:STAMp:FILTer [OFF | REL | DUR | PPS]" on page 44 is disabled.

If this command is sent, then the limit of stored time-stamps specified in the PTIMe:STAMp:FILTer:PPSecond command in "PTIMe:STAMp:FILTer:PPSecond <int> [1,100]" on page 46 will be reset to one. The maximum number of pulse period is one pulse every 604800 seconds (one pulse per week).

The following command will query the state of the PTIMe:STAMp:FILTer:1PPeriod command:

```
PTIMe:STAMp:FILTer:1PPeriod?
```

Below is an example of the results displayed for the PTIMe:STAMp:FILTer:1PPeriod 2 command. A timing diagram for this example can be found in Figure 11.

(Since a period of 2 was sent in the command, only 1 pulse is stored every 2 seconds)

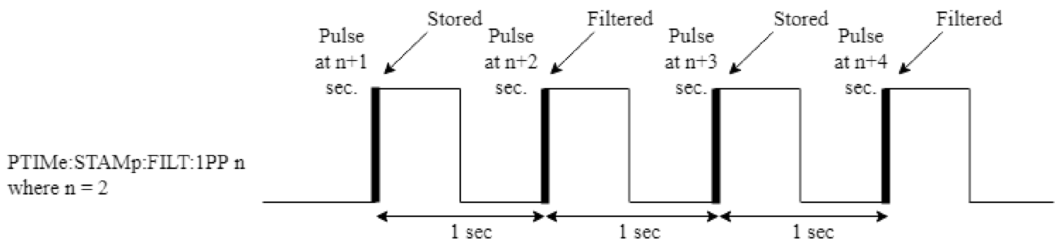
`GPS time-stamp: 2060, 242924.000000014`

`GPS time-stamp: 2060, 242925.000000025` (This time-stamp will be filtered)

`GPS time-stamp: 2060, 242926.000000020`

`GPS time-stamp: 2060, 242927.000000022` (This time-stamp will be filtered)

Figure 11 PTIME:STAMP:FILT:1PP 2 timing example



PTIMe:STAMp:LAST?

The PTIMe:STAMp:LAST? command is unrelated to OCXO disciplining. For more information on OCXO disciplining, please refer to "SYNChronization Subsystem" on page 50.

This query returns the most-recent time-stamp of a PPS signal.This time-stamp is in UTC time format with year, month, day, hour, minute, and second with nanosecond precision.

Below is an example of the results displayed for the PTIMe:STAMp:LAST? command:

```
EXT, 2019,7,1 23:02:27.000000172
```

PTIMe:STAMp:MEMory?

The PTIMe:STAMp:MEMory? command is unrelated to OCXO disciplining. For more information on OCXO disciplining, please refer to "SYNChronization Subsystem" on page 50.

This query returns all 100 most-recent time-stamps at once. These time-stamps are in GPS time format with week number and week second with nanosecond precision.

Below are samples of the results displayed when sending the PTIMe:STAMp:MEMory? command:

```
01: 2060, 167182.000000061; (first time-stamp)
. . .
100: 2060, 169347.000000172; (last time-stamp)
```

PTIMe:STAMp:RESET

The PTIMe:STAMp:RESET command is unrelated to OCXO disciplining. For more information on OCXO disciplining, please refer to “SYNChronization Subsystem” on page 50.

This command will restart the time-stamping and reset all stored time-stamps. All fields in PTIMe:STAMp:LAST? and PTIMe:STAMp:MEMory? query commands will be reset to 0’s.

PTIMe:STAMp?

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

```
PTIMe:STAMp:MODE?
PTIMe:STAMp:SOURce?
PTIMe:STAMp:DIFFer?
PTIMe:STAMp:FILTer?
PTIMe:STAMp:FILTer:PPSecond?
PTIMe:STAMp:FILTer:1PPeriod?
PTIMe:STAMp:LAST?
```

PTIMe?

The PTIMe? command is unrelated to OCXO disciplining. For more information on OCXO disciplining, please refer to “SYNChronization Subsystem” on page 50.

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

```
PTIMe:DATE?
PTIMe:TIME?
PTIMe:STAMp:LAST?
PTIMe:TINTerval?
PTIMe:OUTput?
PTIMe:LEAPsecond:ACCumulated?
```

SYNChronization Subsystem

The SYNChronization subsytem is unrelated to the Micro-JLT GNSS™ time-stamping subsystem. For more information on time-stamping, please refer to "PTIMe:STAMp Subsystem" on page 40.

This subsystem regroups the commands related to the synchronization of the Micro-JLT GNSS™ with the GNSS receiver. The following commands are supported for this subsystem:

- SYNChronization:HOLDover:DURation?
- SYNChronization:HOLDover:INITiate
- SYNChronization:HOLDover:RECovery:INITiate
- SYNChronization:SOURce:MODE <GPS|EXTernal|AUTO>
- SYNChronization:SOURce:MODE:EDGE <NEGative|POSitive>
- SYNChronization:SOURce:STATE?
- 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:RESET?
- SYNChronization:OUTput:1PPS:WIDTH <int> <ms | us> [100us, 500ms]
- SYNChronization:HEAlth?
- SYNChronization?

SYNChronization:HOLDover:DURation?

The SYNChronization:HOLDover:DURation? command is unrelated to the Micro-JLT GNSS™ time-stamping subsystem. For more information on time-stamping, please refer to "PTIMe:STAMp Subsystem" on page 40.

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 SYNChronization:HOLDover:INITiate command is unrelated to the Micro-JLT GNSS™ time-stamping subsystem. For more information on time-stamping, please refer to “PTIMe:STAMp Subsystem” on page 40.

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 receiver 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

The SYNChronization:HOLDover:RECovery:INITiate command is unrelated to the Micro-JLT GNSS™ time-stamping subsystem. For more information on time-stamping, please refer to “PTIMe:STAMp Subsystem” on page 40.

This command will disable the forced holdover state (see “SYNChronization:HOLDover:INITiate” on page 51). The unit will resume normal GPS locking operation after this command has been sent.

SYNChronization:SOURce:MODE <GPS|EXTernal|AUTO>

The SYNChronization:SOURce:MODE command is unrelated to the Micro-JLT GNSS™ time-stamping subsystem. To select a source for time-stamping, please refer to the PTIMe:STAMp:SOURce command in “PTIMe:STAMp:SOURce <GPS|EXTernal|ALL>” on page 43.

The board may be configured lock to an external PPS source rather than the internal GNSS receiver. An optional TTL or CMOS 3.3V/5V PPS signal may be applied to Pin 3 of connector J4, and this signal may be used to feed an external CMOS rising-edge PPS signal with -0.5V < x < 5.0V signal level, and >= 100us minimum pulse width into the unit. See Table 1 on page 11 for pin descriptions of connector J4.

By default the unit is set to GNSS disciplining. The unit may be hard-coded to only use the external PPS source by setting EXT, or it may be auto-switched by the firmware to the 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. When set to the AUTO setting, the unit will switch back to the internal GNSS receiver once 1PPS signals are generated internally again. Setting the unit to the EXT setting will cause the unit to disable the GNSS receiver, and this setting may also be used to initiate a manual forced-holdover if the GNSS signal is suspect, or unreliable.

The command has the following format:

```
SYNChronization:SOURce:MODE <GPS | EXTernal | AUTO>
```

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

```
SYNChronization:SOURce:MODE?
```

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

This command configures the triggering edge of the external PPS input source (Pin-3 of J4 connector) shown in Table 6 on page 41 for both OCXO disciplining discussed in "SYNChronization Subsystem" on page 50 and time-stamping subsystem in "PTIMe:STAMp Subsystem" on page 40.

By default the external PPS input capturing is triggered at the rising (or "POSITIVE") edge. The triggering edge can be changed to falling (or "NEGATIVE") edge with the SYNC:SOUR:MODE:EDGE NEG command.

The NEGATIVE edge setting allows the Micro-JLT GNSS™ to synchronize the rising edge of 1PPS output to the falling edge of external PPS input if SYNC:SOUR:MODE EXT command is configured in "SYNChronization:SOURce:MODE <GPS|EXTernal|AUTO>" on page 51. Alternatively the NEGATIVE setting allows the Micro-JLT GNSS™ to compute the time-stamp offset between the GPS 1PPS and the falling edge of external PPS input if PTIMe:STAMp:MODE DIFFer command is configured in "PTIMe:STAMp:MODE <OFF|ASYnc|MEMory|OUTput|DIFFer>" on page 42.

To avoid undesired OCXO disciplining or time-stamping, JLT recommends confirming the characteristics of the external PPS input signal, proper settings in SYNChronization:SOURce:MODE? and PTIMe:STAMp:MODE? commands before changing to NEGATIVE setting.

The following command returns the current configured edge for external PPS input:

```
SYNChronization:SOURce:MODE:EDGE?
```

SYNChronization:SOURce:STATE?

The SYNChronization:SOURce:STATE? command is unrelated to the Micro-JLT GNSS™ time-stamping subsystem. For more information on time-stamping, please refer to "PTIMe:STAMp Subsystem" on page 40.

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

SYNChronization:HOLDover:STATE?

The SYNChronization:HOLDover:STATE? command is unrelated to the Micro-JLT GNSS™ time-stamping subsystem. For more information on time-stamping, please refer to "PTIMe:STAMp Subsystem" on page 40.

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 fix from internal receiver.

SYNChronization:TINTerval?

The SYNChronization:TINTerval? command is unrelated to the Micro-JLT GNSS™ time-stamping subsystem. For more information on time-stamping, please refer to "PTIMe:STAMp Subsystem" on page 40.

This query returns the difference or timing shift between the Micro-JLT GNSS™ 1PPS output and the GNSS 1PPS signals. The resolution is 1E-10 seconds.

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

The SYNChronization:TINTerval:THReshold command is unrelated to the Micro-JLT GNSS™ time-stamping subsystem. For more information on time-stamping, please refer to "PTIMe:STAMp Subsystem" on page 40.

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 220ns, allowing a drift of up to +/-220ns. 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

The SYNChronization:IMMEdiate command is unrelated to the Micro-JLT GNSS™ time-stamping subsystem. For more information on time-stamping, please refer to “PTIMe:STAMp Subsystem” on page 40.

This command initiates a near-instantaneous alignment of the GNSS 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 internal 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 external GNSS antenna being connected to the unit. By default the unit does not generate a 1PPS signal until the GNSS receiver has locked onto the Satellites.

With the command SYNC:OUT:1PPS:RESET ON the unit can be configured to generate an asynchronous 1PPS output after power-on even if a GNSS antenna is not connected to the unit. Once the GNSS receiver locks, the 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 lock is achieved.

SYNChronization:OUTput:1PPS:RESET?

This query returns the state of the SYNC:OUT:1PPS:RESET command, and may return ON or OFF.

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 Micro-JLT GNSS™ 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

As an example, if the unit is in GNSS receiver holdover, and 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 receiver in holdover: 0x10

'Oring' these values together results in:

0x40 | 0x10 | 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?

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:ROSCillator:EFControl:RELative?`
- `DIAGnostic:ROSCillator:EFControl:ABSolute?`
- `DIAGnostic:LIFetime:COUNt?`
- `DIAGnostic?`

DIAGnostic:ROSCillator:EFControl:RELative?

This query returns the Electronic Frequency Control (EFC) output value of the internal reference oscillator. It returns a percentage value between -100% to +100%.

DIAGnostic:ROSCillator:EFControl:ABSolute?

This query returns the Electronic Frequency Control (EFC) output value of the internal reference oscillator. It returns a value in volts between 0 and 5 V.

DIAGnostic:LIFetime:COUNt?

This command returns the number of hours 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
```

MEASURE Subsystem

This subsystem regroups the queries related of some parameters that are measured on-board on the Micro-JLT GNSS™. The list of the commands supported for this subsystem is the following:

- `MEASure:VOLTage?`
- `MEASure:POWersupply?`
- `MEASure:POWersupply:VOSCillator?`
- `MEASure?`

MEASure:VOLTage?

This query returns the EFC voltage applied to the OCXO.

MEASure:POWersupply?

This query returns the power supply voltage applied to the Micro-JLT GNSS™ board.

MEASure:POWersupply:VOSCillator?

This query returns the power supply voltage applied to the OCXO which is around 10.45V nominal.

MEASure?

This query returns the result of the three following queries:

- `MEASure:VOLTage?`
- `MEASure:POWersupply?`
- `MEASure:POWersupply:VOSCillator?`

SYSTEM Subsystem

This subsystem regroups the commands related to the general configuration of the Micro-JLT GNSS™. 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:STATus?`
- `SYSTem:FACToryReset ONCE`
- `SYSTem:CPURESET`
- `SYSTem:ISP`

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

This command enables/disables echo on RS-422/TTL 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 5. 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 RS-422/TTL 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: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 5. 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 Micro-JLT GNSS™ processor to reset.

SYSTem:ISP

This command causes the Micro-JLT GNSS™ processor to reset into In System Programming (ISP) mode for firmware upgrades. Please see Chapter 4 for details on updating the firmware in ISP mode.

Initiating this command is equivalent to pulling the ISP# pin 1 on J1 or pin 1 on J4 low during power-on. Both LEDs on D2 should dim indicating the unit is properly placed into 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:DACGain`
- `SERVo:EFCScale <float> [0.0,500.0]`
- `SERVo:EFCDamping <int> [2,4000]`

- SERVo:SLOPe <NEG|POS>
- SERVo:TEMPCOmpensation
- SERVo:AGINGcompensation <float> [-10.0,10.0]
- SERVo:PHASECOrrection <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 (SERV: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 Then 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, please see the command "SERVo:FASTlock <int> [1,20]" on page 60.

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 Micro-JLT GNSS™ 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:DACGain

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: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: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

This command is not supported on the Micro-JLT GNSS™ board.

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 Jackson-Labs 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:PHASECOrrection <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:PHASECOrrection <float> [-500.0,500.0]
```

This command will query the state of this command:

```
SERVo:PHASECOrrection?
```

SERVo:1PPSoffset <int> <ns> [-5000000,5000000]

This command sets the Micro-JLT GNSS™ 1PPS signal's offset to UTC in 5.55ns steps. The maximum configurable 1PPS offset is +/- 5ms.

Using the SERV: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: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 the “SYNChronization:HEAlth?” on page 55 command for detailed information on how to decode the health status indicator values. The Lock State variable indicates one of the states in Table 5.

Table 5 Lock State

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:DACGain?`
- `SERVo:EFCScale?`
- `SERVo:PHASECOrrection?`
- `SERVo:EFCDamping?`
- `SERVo:SLOPe?`
- `SERVo:COARSedac?`
- `SERVo:AGINGcompensation?`
- `SERVo:1PPSoffset?`
- `SERVo:TRACe?`
- `SERVo:FASTlock?`
- `SERVo:FALEngth?`

4

Firmware Upgrade Instructions

This chapter discusses the following topics:

- “Introduction” on page 66
- “ISP Flash Loader Utility installation” on page 66
- “Putting the PCB into In-Circuit Programming (ISP) mode” on page 66
- “Downloading the firmware” on page 67
- “Verifying firmware update” on page 74

Introduction

The following is a short tutorial on how to upgrade the Micro-JLT GNSS™ GNSDO firmware. Please follow the instructions in-order to prevent accidentally corrupting the Micro-JLT GNSS™ Flash, which may require re-flashing at the factory.

ISP Flash Loader Utility installation

Jackson Labs Technologies, Inc. recommends using the JLTerm application to upgrade the contents of Flash memory on the Micro-JLT GNSS™. The JLTerm application can be downloaded for free from our support page with the link provided below. Alternatively, you can also use the Flash magic utility to perform the same upgrade.

The JLTerm application is available for download at:

http://www.jackson-labs.com/index.php/support

The Flash Magic utility is available for download on the Flash Magic website:

http://www.flashmagictool.com/

Follow the directions given on the website for installing the utility on your computer.

Putting the PCB into In-Circuit Programming (ISP) mode

Two methods are supported for enabling the Micro-JLT GNSS™ in System Programming (ISP) mode:

- Issuing the SCPI command SYST:ISP discussed in Section 3.9.7 in JLTerm or another terminal program causes the board to reset into ISP mode from normal operation. Both LEDs on D2 should dim indicating the unit is properly placed into ISP mode.

  If Flash Magic is used instead of the JLTerm, a power cycle is required after programming has finished to restart the board with the updated firmware. Although the SYST:ISP command is recommended, the second method will be useful when the Micro-JLT GNSS™ stopped communicating.
- The board can also be put into ISP mode by shorting-out the ISP# pin 1of header J4 to pin 2 (Ground) of header J4 of the Micro-JLT GNSS™ board shown in Figure 2.1 during power-on. Pin 1 of connector J1 is also wired in parallel to this ISP# signal trace and may be alternatively used.

  It may be useful to connect a push-button to pin 1 and pin 2 of J4 to allow in-field firmware upgrades on the Micro-JLT GNSS™ PCB. Both LEDs should dim after power-cycling the board, indicating the unit is properly placed into ISP mode. If the red or green LED lights up after power-on, the unit is not in ISP mode.

Downloading the firmware

Download the latest version of Micro-JLT GNSS™ firmware from the Jackson Labs Technologies, Inc. support website 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 RS-422/TTL serial port prior to firmware download. The Micro-JLT GNSS™ is RS-422-level/TTL compatible. An RS-422/TTL to RS-232 converter must be used when connecting to the computer's RS-232 serial port or to industry-standard RS-232 to USB adapters.

4.4.1 Using the JLTerm Programming Terminal

1 Download the JLTerm application from http://www.jackson-labs.com/index.php/support.

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 12. The Micro-JLT GNSS™ 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.

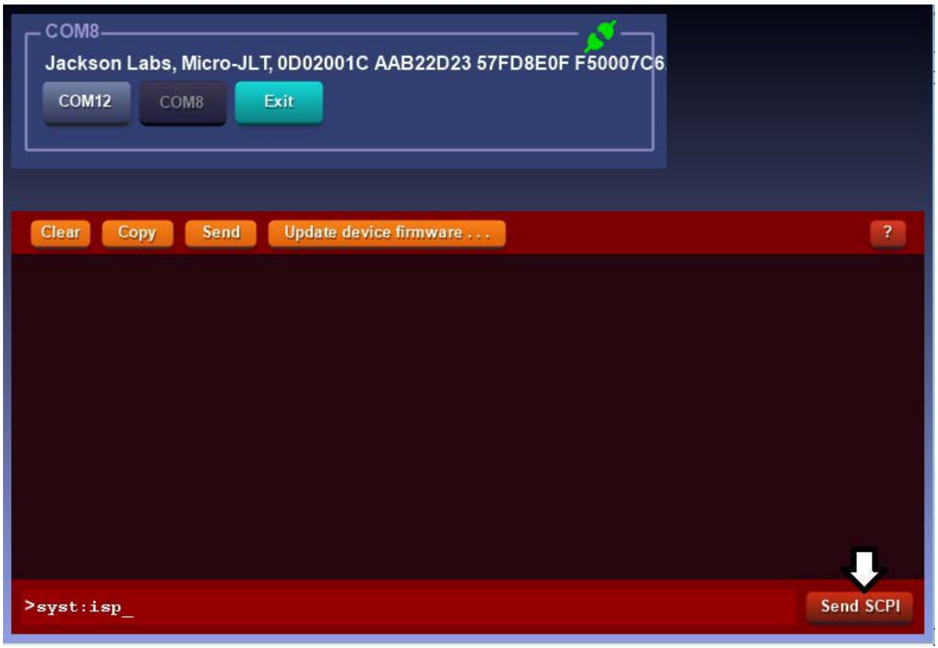
Figure 12 JLTerm COM port connection success

If there is no valid response from the COM port, check for valid driver and port number for the COM port in Device Manager application on your computer. The COM port number may be conflicting with another COM port device and each device should have different port number.

4 If necessary, change the COM port number, install or update driver, unplug and reconnect the serial cable, then try to establish a connection again. In the case of an unsuccessful connection to JLTerm after several attempts, follow instructions in “Putting the PCB into In-Circuit Programming (ISP) mode” on page 66 and “Using the Flash Magic programming utility” on page 70 for an alternative firmware upgrade method.

5 In the red command line, type the command SYST:ISP and click **Send SCPI**, as shown in Figure 13. The command must be sent to ensure that the unit is in ISP mode before programming begins.

Figure 13 Properly program the unit in ISP mode

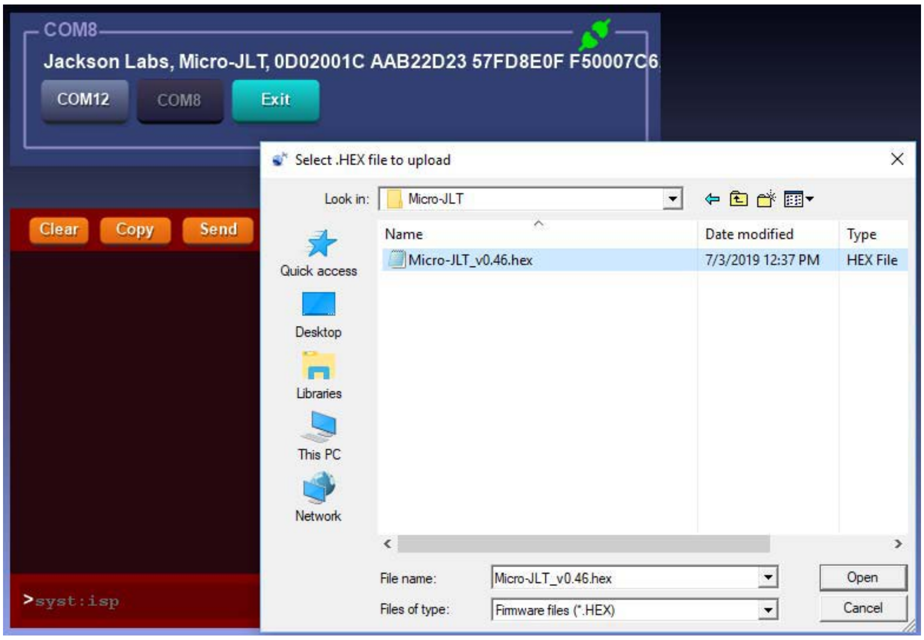


It is important to put the device in ISP mode prior to conducting a firmware upgrade. Failure to send the SYST:ISP command may result in programming errors such as the one shown in Figure 14. If the error in Figure 14 occurs, power-cycle the unit and send the SYST:ISP command again in JLTerm, as shown in Figure 13.

Figure 14 Error if device is not set to ISP Mode

6 Once the device is in ISP mode, select the orange **Update device firmware...** button, then choose the correct hex file to program the device with and select **Open** in the pop-up menu, as shown in Figure 15. The firmware will begin downloading and automatically reset the board. If there is a failure or error, proceed to the next step.

Figure 15 Open hex file and download firmware



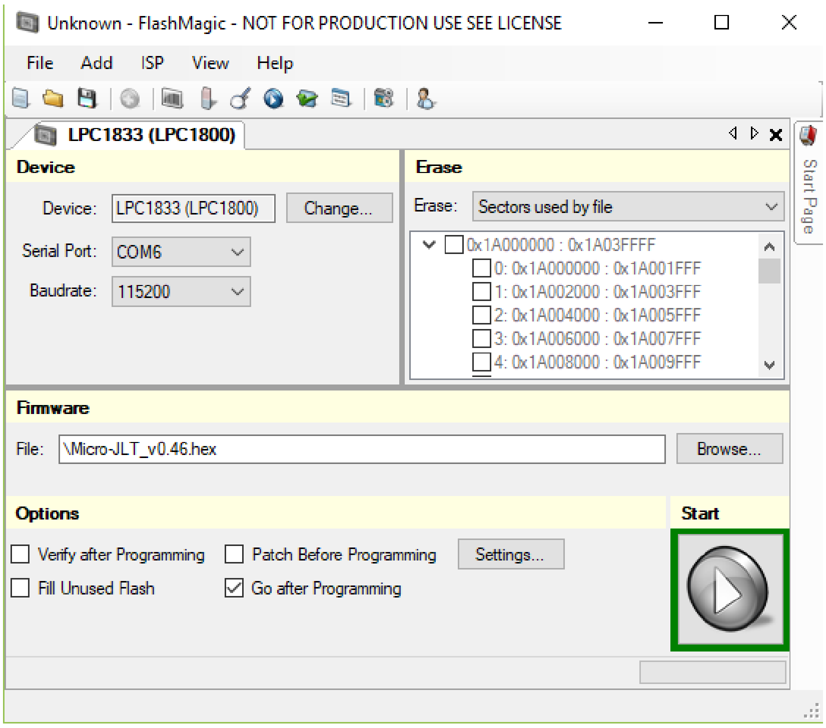
7 If an error occurs during the programming process, as shown in FIGURE, check the USB cable connection. While the unit is still in ISP mode after the error occurred, follow the instructions in "Using the Flash Magic programming utility" on page 70 and attempt to finish the firmware upgrade using Flash Magic.

Figure 16 Error in programming

Using the Flash Magic programming utility

1 Open the Flash Magic utility. Select the correct Device LPC1833 by clicking "Change..." and select LPC1800->LPC1833. If Flash Magic version is 11.20 or older, perform the steps in "Using the Flash Magic Classic version" on page 72.

Figure 17 Flash Magic programming utility v12.xx

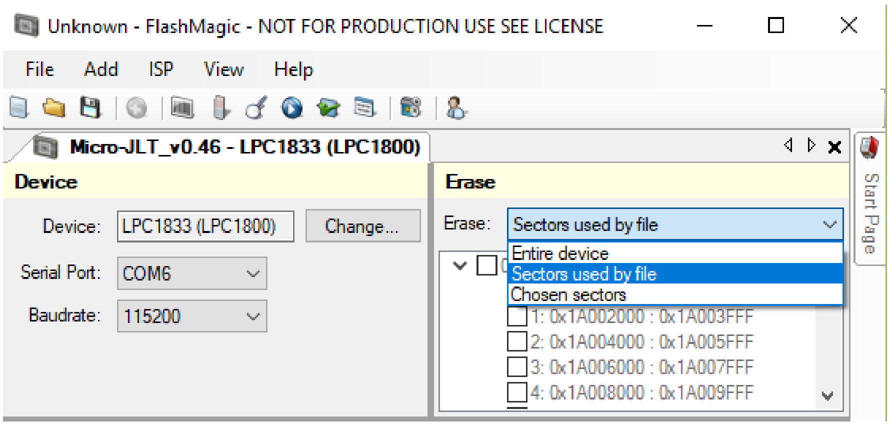


2 Select Erase “Sectors used by file” option as shown in Figure 18.

WARNING

Ensure the Erase option is set to “Sectors used by file” at all times. Leaving the Erase option as “Entire device” will erase factory calibration data and the unit will not operate, requiring the device to be returned to the factory. Programming Micro-JLT GNSS™ with the Erase “Entire device” option on the ISP utility will void the warranty.

Figure 18 Erase Sectors used by file



3 Change the COM port in the Flash Magic application as needed and select **Baud Rate** 115200.Slower baud rates will work, but will take longer to finish the programming cycle.

4 In the Firmware section, click **Browse...** to browse for the hex file downloaded in “Downloading the firmware” on page 67.

5 Ensure the **Erase** option is set to “Sectors used by file”, as shown in Figure 18.

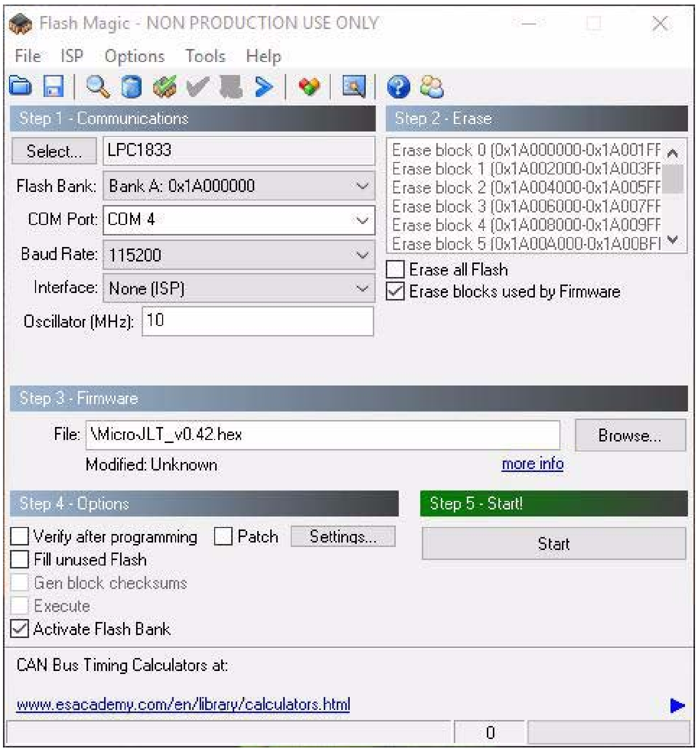
6 In the Start section, press the gray start button. The firmware will download to the processor.

7 Verify the firmware update as described in "Verifying firmware update" on page 74.

Using the Flash Magic Classic version

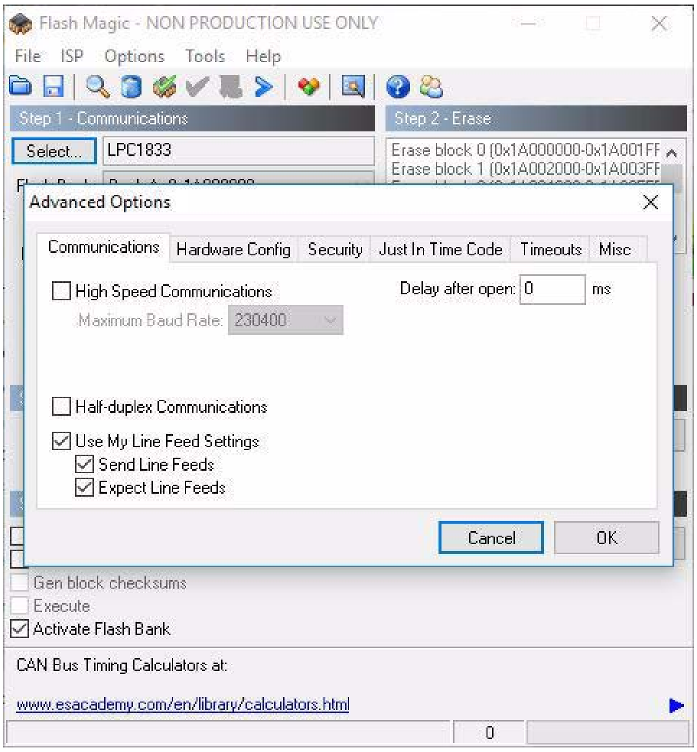
1 Set the COM port in the Flash Magic application as needed on your PC, as shown in Figure 19.

Figure 19 Flash Magic programming utility Classic



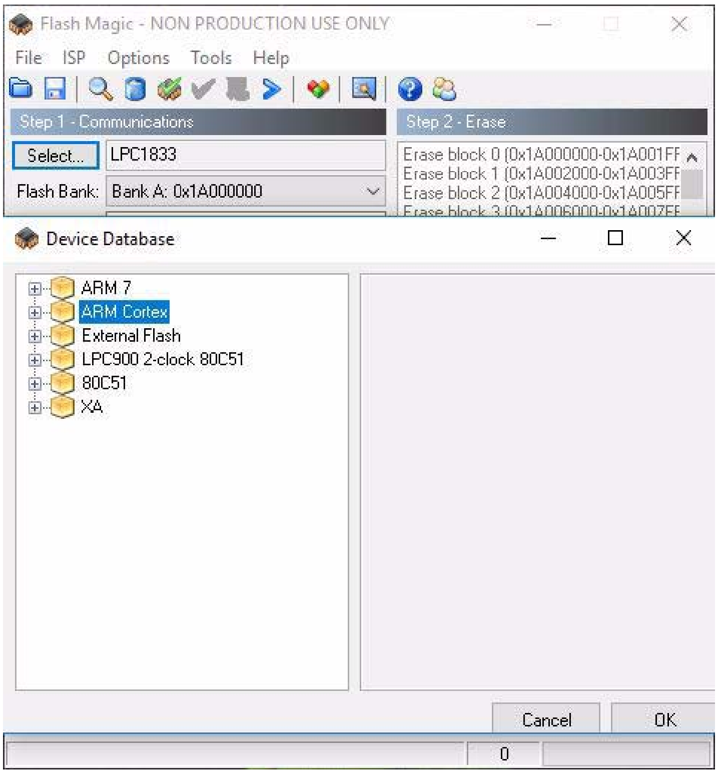
2 Open **Options > Advanced** options tabs on the application and ensure that the "Use My Line Feed Settings", "Send Line Feeds" and "Expect Line Feeds" options are all highlighted and selected as shown in Figure 20.

Figure 20 Required Flash Magic communications options



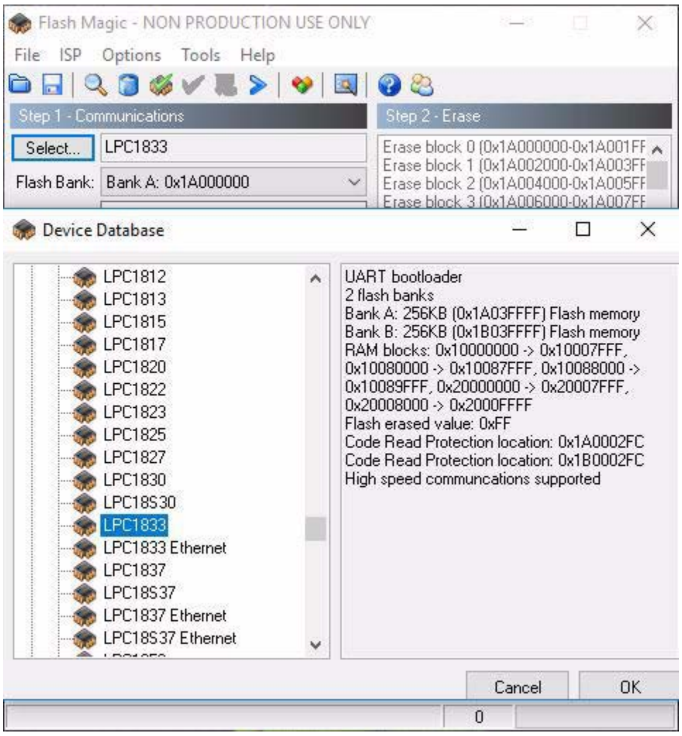
3 Press the **Select Device** button. The Device selection window appears, as shown in Figure 21.

Figure 21 Device selection window



4 Expand the **ARM CORTEX** folder and select the appropriate processor, in this case **LPC1833**.

Figure 22 Expanded device selection window



5. Set the **Baud Rate** to 115200, as shown in FIGURE. Slower baud rates will also work, but will take longer to finish the programming cycle.
6. Set the **Oscillator (MHz)** to 10.0, as shown in Figure.
7. Check the box marked **Erase blocks used by Hex File**.

> **WARNING**
>
> Ensure you do NOT check the box marked **Erase all Flash**. Checking this box will erase factory calibration data, and the unit will not operate and will have to be returned to the factory. Checking this box will void the warranty.

8. Under **Step 3 - Hex File**, click **Browse...** to browse for the hex file downloaded in "Downloading the firmware" on page 67.
9. Under **Step 4 - Options**, check **Activate Flash Bank**.
10. Ensure the Erase option is set to **Sectors used by file** as shown in Figure 18.
11. In the Start section, press the gray start button. The firmware will download to the processor.
12. Verify the firmware update as described in "Verifying firmware update" on page 74.

Verifying firmware update

Power cycle the unit with the pin 1of J4 and pin 1 of J1 (ISP#) left floating. Both LED's should blink.

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.

5

GPSCon Utility

The following topics are discussed in this chapter:

- "Description" on page 78
- "Installation" on page 78
- "Using GSPCon" on page 78
- "Interpreting the Data" on page 84

Description

GPSCon – Jackson Labs Edition is a free program for the monitoring and control of Jackson Labs Technologies, Inc, GPSDO, Simulator, and Receiver products. It communicates with the receiver using the SCPI command set. This free version of the GPSCon utility is only compatible with Jackson Labs products, and is available for download from the support section of the Jackson Labs website:

http://www.jackson-labs.com/index.php/support

Installation

Extract the contents of the ZIP file downloaded from the Jackson Labs website and execute the MSI installer. Follow the on-screen instructions to complete the installation of GPSCon.

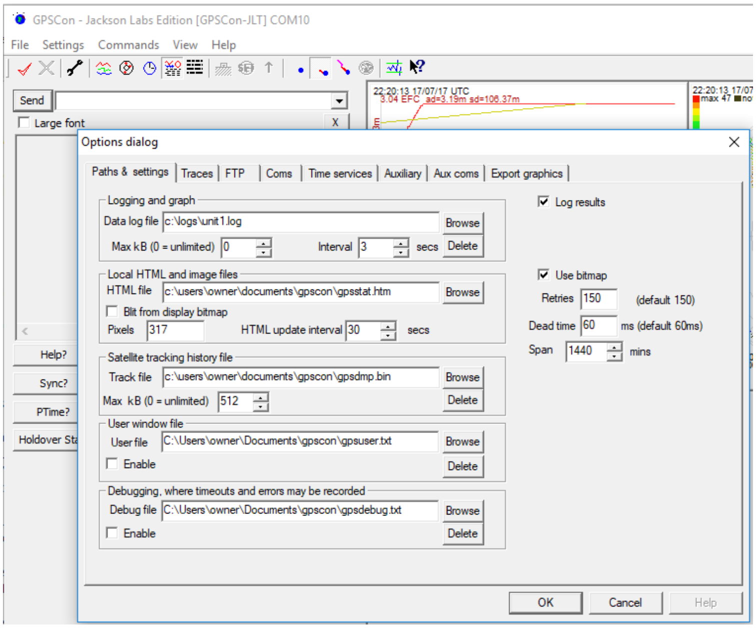
Using GSPCon

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 23 will appear. You can select from the tabs which options you wish to set.

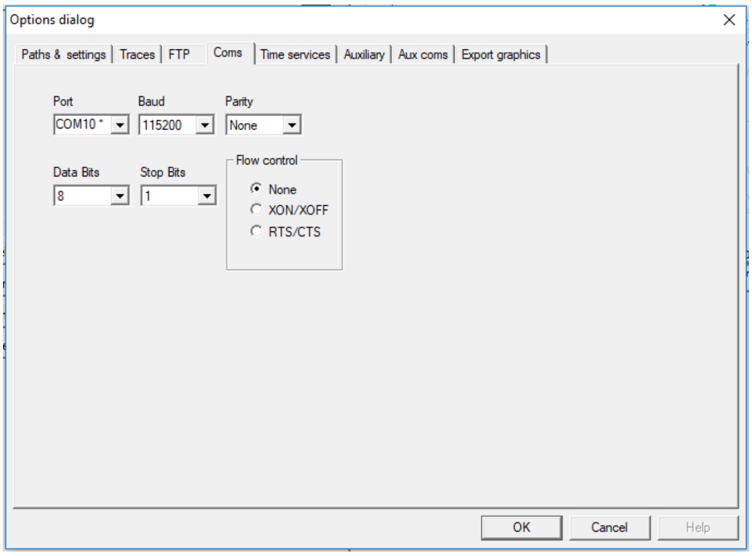
Figure 23 Options window



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 24.

Figure 24 Coms parameters tab

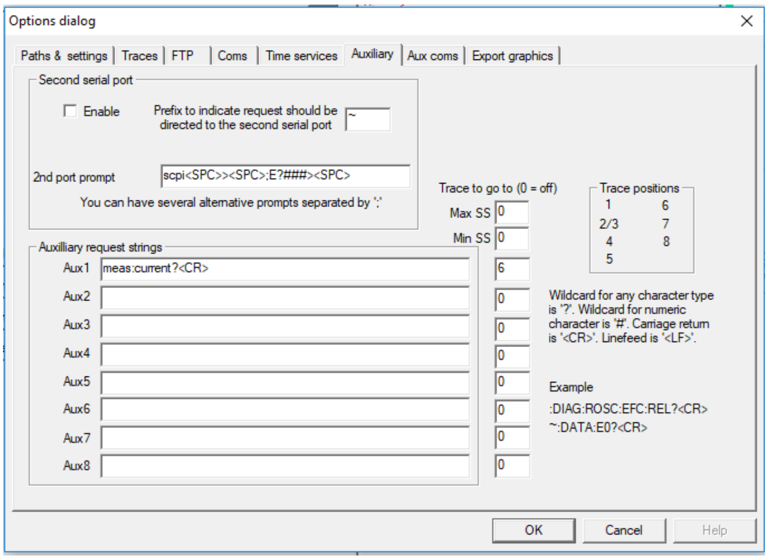


Available COM ports for your particular computer are indicated with a “*” symbol next to the COM port number in the drop-down menu. You can only select COM ports with this adjacent symbol. Select the correct COM port for your computer’s serial port connection and set the baud rate to 115200, parity to 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 25 shows an example of an auxiliary measurement.

Figure 25 Auxiliary parameters tab

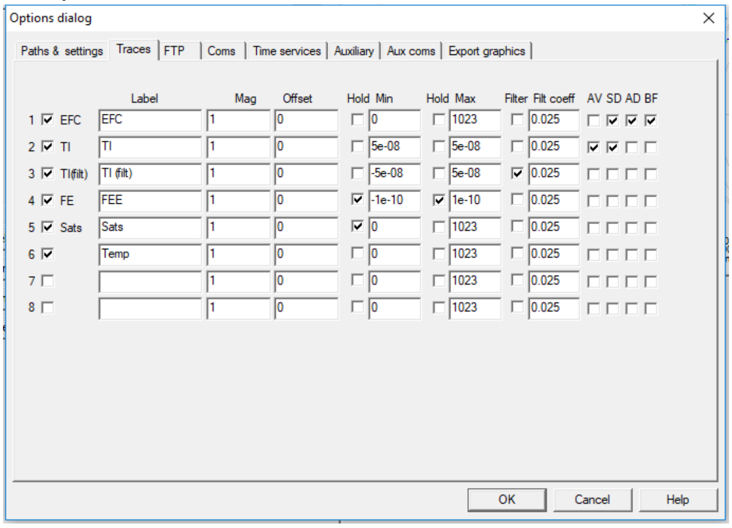


In the above example, the **Aux1** request string has been set to `meas:current?<CR>` and the **Trace to go to** is set to trace position 6. See the **Trace Position** diagram for the arrangement of the trace positions in the trace window. In this example the data obtained from the meas:current? query will be plotted in trace position 6. This command is not supported on the Micro-JLT GNSS™ and it is recommended to change this default setting to some other available parameter to be tracked.

Traces parameters

After pressing the wrench icon, select the “Traces” tab and configure the trace labels and vertical plot ranges. Figure 26 shows an example of an auxiliary measurement. 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 “Sending manual commands to the receiver” on page 81. Press the “Help” button for a full description of each option in the Traces tab.

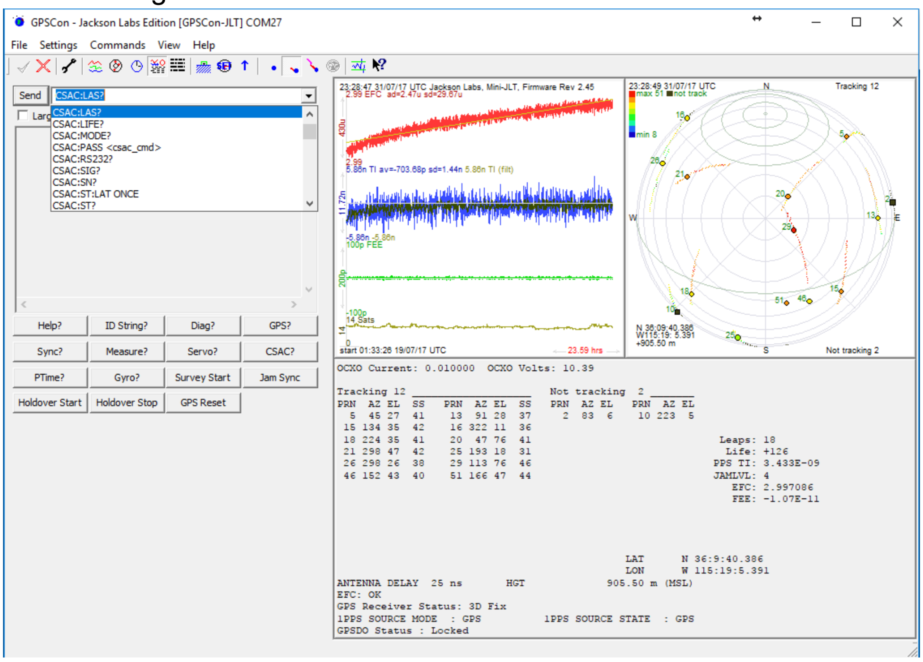
Figure 26 Traces parameters tab



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 27.

Figure 27 Sending manual commands



Once a command is selected, press **Send** to send it to the Micro-JLT GPSDO. 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.

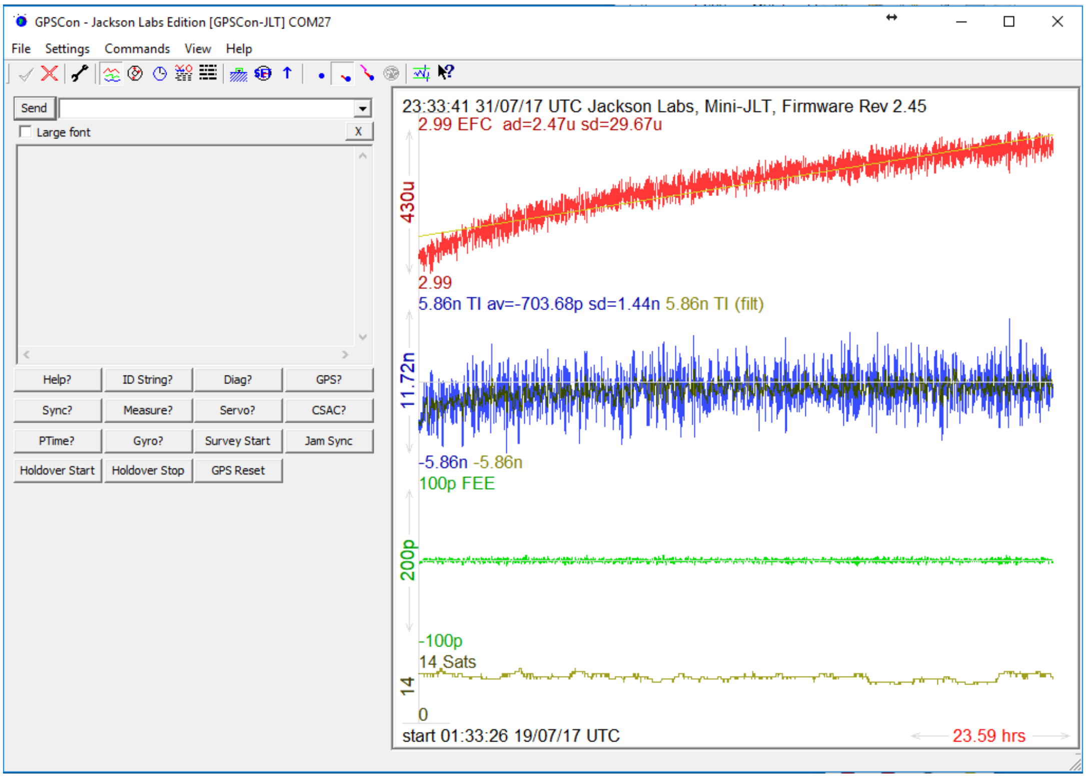
NOTE

Ensure the selected command is supported by the Micro-JLT GPSDO.

Using the mouse in the Graph display

Figure 28 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 28 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.

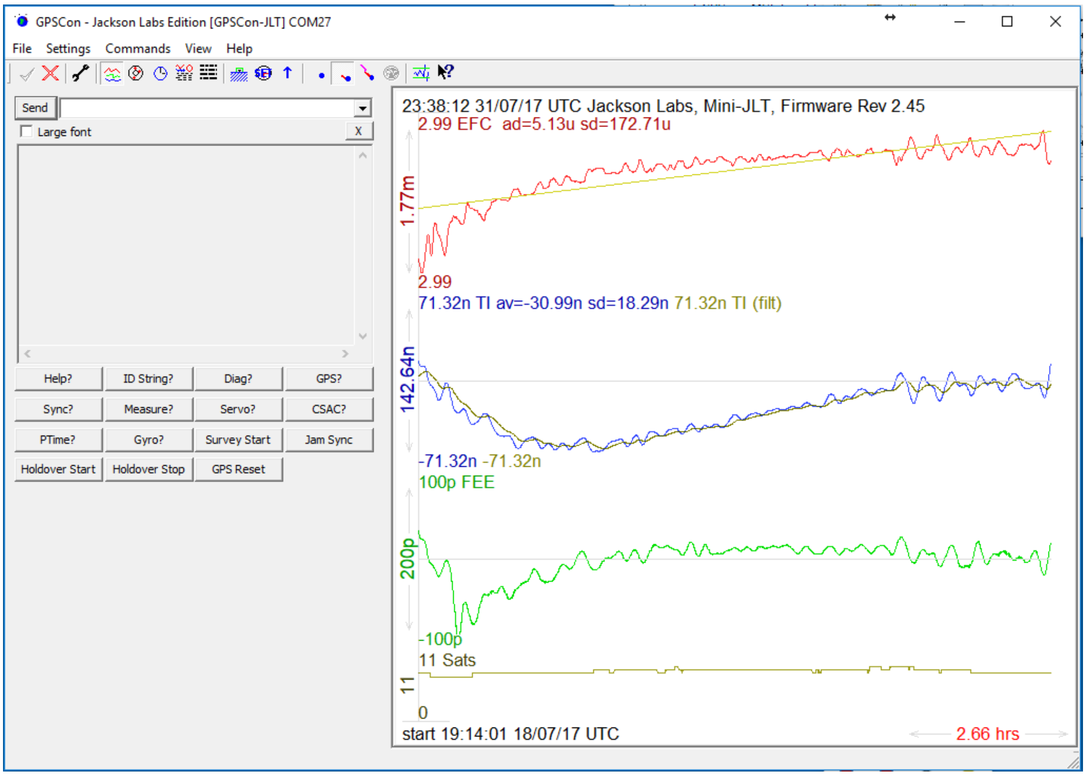
2 Set the stop time by right-clicking on the point that marks the right most portion of the magnified curve.

> **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 29 shows the extended graph display.

Figure 29 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.

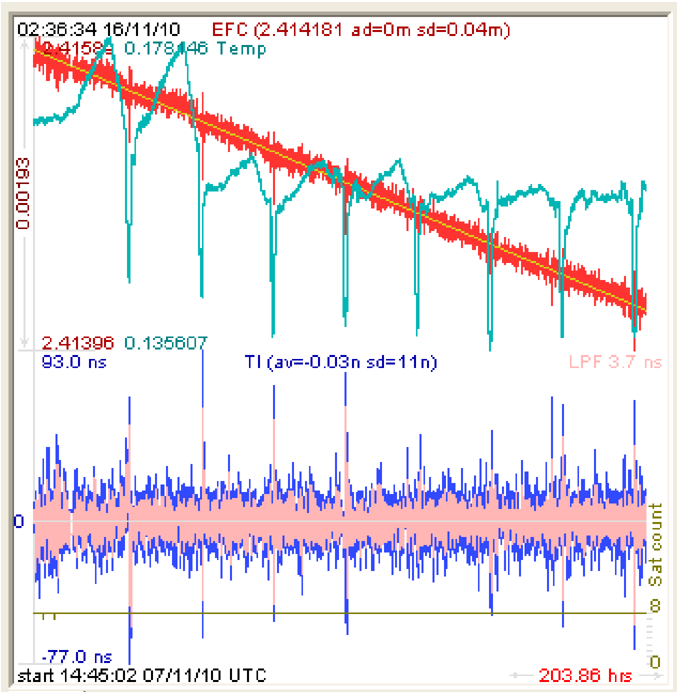
- To export automatically on a timed basis, enter a non-zero value in seconds to choose an export time interval.

- 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 30 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 teh 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:

8Hz/V * 0.088V/Year = 0.704Hz/Year drift.

At 10 MHz:

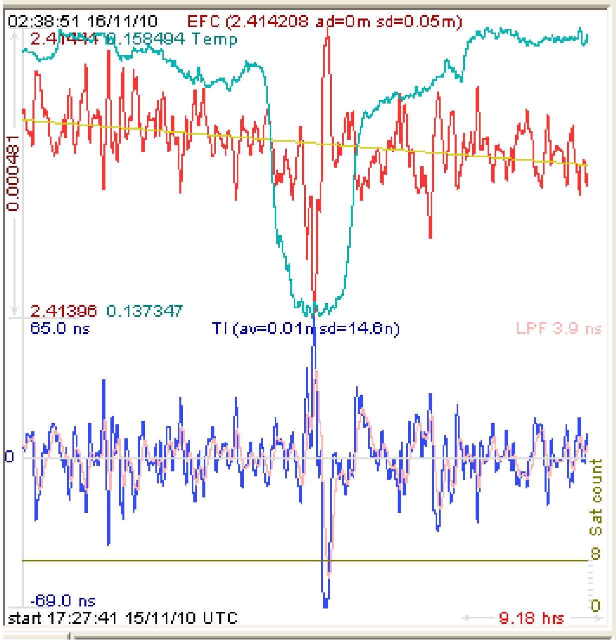
0.704Hz / 10MHz = 7.04E-08 aging rate per year

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 31 shows a zoom of the captured data.

Figure 31 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:

11ns / 200Hrs = 1.528E-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.

6

Certification and Warranty

This chapter discusses the following topics:

- "Certification" on page 88
- "Warranty" on page 88
- "Exclusive Remedies" on page 88

Certification

Jackson Labs Technologies, Inc. certifies that this product met its published specifications at time of shipment.

Warranty

This Jackson Labs Technologies, Inc. hardware product is warranted against defects in material and workmanship for a period of 1 (one) year from date of delivery. During the warranty period Jackson Labs Technologies, Inc. will, at its discretion, either repair or replace products that prove to be defective. Jackson Labs Technologies, Inc. does not warrant that the operation for the software, firmware, or hardware shall be uninterrupted or error free even if the product is operated within its specifications.

For warranty service, this product must be returned to Jackson Labs Technologies, Inc. or a service facility designated by Jackson Labs Technologies, Inc. Customer shall prepay shipping charges (and shall pay all duties and taxes) for products returned to Jackson Labs Technologies, Inc. for warranty service. Except for products returned to Customer from another country, Jackson Labs Technologies, Inc. shall pay for return of products to Customer. If Jackson Labs Technologies, Inc. is unable, within a reasonable time, to repair or replace any product to condition as warranted, the Customer shall be entitled to a refund of the purchase price upon return of the product to Jackson Labs Technologies, Inc.

Limitation of Warranty

The foregoing warranty shall not apply to defects resulting from improper or inadequate maintenance by the Customer, Customer-supplied software or interfacing, unauthorized modification or misuse, opening of the instruments enclosure or removal of the instruments panels, operation outside of the environmental or electrical specifications for the product, or improper site preparation and maintenance. **JACKSON LABS TECHNOLOGIES, INC. SPECIFICALLY DISCLAIMS THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.** No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted by this document. Jackson Labs Technologies, Inc. products are not intended for use in medical, life saving, or life sustaining applications.

Exclusive Remedies

THE REMEDIES PROVIDED HEREIN ARE THE CUSTOMER'S SOLE AND EXCLUSIVE REMEDIES. JACKSON LABS TECHNOLOGIES, INC. SHALL NOT BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, WHETHER BASED ON CONTRACT, TORT, OR ANY OTHER LEGAL THEORY.

Contact Us: +1 800 835 2352 | avcomm.sales@viavisolutions.com.

© 2025 VIAVI Solutions Inc. Product specifications and descriptions in this document are subject to change without notice. Patented as described at viavisolutions.com/patents

viavisolutions.com