
# SPAN-CPT™

# USER MANUAL

# SPAN-CPT Receiver User Manual

**Publication Number:** OM-20000122
**Revision Level:** Rev 3
**Revision Date:** 2009/08/10

This manual reflects SPAN-CPT Firmware Version CPT3.620.

# Proprietary Notice



Manufactured and protected under U.S. Patent:

**Narrow Correlator**
#5,101,416
#5,390,207
#5,495,499
#5,809,064
**PAC Correlator**
#6,243,409 B1
#5,414,729
**Dual Frequency GPS**
#5,736,961
**Anti-Jamming Technology**
#5,734,674
**Position for Velocity Kalman Filter**
#6,664,923 B1
#7,193,559 B2
**SPAN Technology**
#6,721,657 B2
#6,750,816 B1
#7,346,452
10/758,363 (pending)
10/932,497 (pending)

**KVH Technology**
#5,444,534
#5,481,358
#5,512,904
#5,552,887
#5,739,944
#5,768,462
#6,041,149
#6,134,356
#6,351,310 B1
#6,370,289 B1
#6,429,939
#6,441,779
#6,466,596
#6,542,651
#6,594,020
#6,718,097
#6,763,153
#6,836,334
#6,856,300
#6,864,347
#6,891,622
#7,120,323

# Table of Contents

# Figures

# Tables

# Software License

**BY INSTALLING, COPYING, OR OTHERWISE USING THE SOFTWARE PRODUCT, YOU AGREE TO BE BOUND BY THE TERMS OF THIS AGREEMENT. IF YOU DO NOT AGREE WITH THESE TERMS OF USE, DO NOT INSTALL, COPY OR USE THIS ELECTRONIC PRODUCT (SOFTWARE, FIRMWARE, SCRIPT FILES, OR OTHER ELECTRONIC PRODUCT WHETHER EMBEDDED IN THE HARDWARE, ON A CD OR AVAILABLE ON THE COMPANY WEB SITE)** (hereinafter referred to as "Software").

**1. License**: NovAtel Inc. ("NovAtel") grants you a non-exclusive, non-transferable license (not a sale) to, where the Software will be used on NovAtel supplied hardware or in conjunction with other NovAtel supplied software, use the Software with the product(s) as supplied by NovAtel. You agree not to use the Software for any purpose other than the due exercise of the rights and licences hereby agreed to be granted to you.

**2. Copyright**: NovAtel owns, or has the right to sublicense, all copyright, trade secret, patent and other proprietary rights in the Software and the Software is protected by national copyright laws, international treaty provisions and all other applicable national laws. You must treat the Software like any other copyrighted material except that you may make one copy of the Software solely for backup or archival purposes (one copy may be made for each piece of NovAtel hardware on which it is installed or where used in conjunction with other NovAtel supplied software), the media of said copy shall bear labels showing all trademark and copyright notices that appear on the original copy. You may not copy the product manual or written materials accompanying the Software. No right is conveyed by this Agreement for the use, directly, indirectly, by implication or otherwise by Licensee of the name of NovAtel, or of any trade names or nomenclature used by NovAtel, or any other words or combinations of words proprietary to NovAtel, in connection with this Agreement, without the prior written consent of NovAtel.

**3. Patent Infringement**: NovAtel shall not be liable to indemnify the Licensee against any loss sustained by it as the result of any claim made or action brought by any third party for infringement of any letters patent, registered design or like instrument of privilege by reason of the use or application of the Software by the Licensee or any other information supplied or to be supplied to the Licensee pursuant to the terms of this Agreement. NovAtel shall not be bound to take legal proceedings against any third party in respect of any infringement of letters patent, registered design or like instrument of privilege which may now or at any future time be owned by it. However, should NovAtel elect to take such legal proceedings, at NovAtel's request, Licensee shall co-operate reasonably with NovAtel in all legal actions concerning this license of the Software under this Agreement taken against any third party by NovAtel to protect its rights in the Software. NovAtel shall bear all reasonable costs and expenses incurred by Licensee in the course of co-operating with NovAtel in such legal action.

**4. Restrictions: You may not:**

(a) copy (other than as provided for in paragraph 2), distribute, transfer, rent, lease, lend, sell or sublicense all or any portion of the Software except in the case of sale of the hardware to a third party;

(b) modify or prepare derivative works of the Software;

(c) use the Software in connection with computer-based services business or publicly display visual output of the Software;

(d) transmit the Software over a network, by telephone or electronically using any means (except when downloading a purchased up[grade from the NovAtel web site); or

(e) reverse engineer, decompiler or disassemble the Software.

You agree to keep confidential and use your best efforts to prevent and protect the contents of the Software from unauthorized disclosure or use.

**5. <u>Term and Termination</u>**: This Agreement and the rights and licences hereby granted shall continue in force in perpetuity unless terminated by NovAtel or Licensee in accordance herewith. In the event that the Licensee shall at any time during the term of this Agreement: i) be in breach of its obligations hereunder where such breach is irremediable or if capable of remedy is not remedied within 30 days of notice from NovAtel requiring its remedy; then and in any event NovAtel may forthwith by notice in writing terminate this Agreement together with the rights and licences hereby granted by NovAtel. Licensee may terminate this Agreement by providing written notice to NovAtel. Upon termination, for any reasons, the Licensee shall promptly, on NovAtel's request, return to NovAtel or at the election of NovAtel destroy all copies of any documents and extracts comprising or containing the Software. The Licensee shall also erase any copies of the Software residing on Licensee's computer equipment. Termination shall be without prejudice to the accrued rights of either party, including payments due to NovAtel. This provision shall survive termination of this Agreement howsoever arising.

**6. <u>Warranty</u>**: NovAtel does not warrant the contents of the Software or that it will be error free. The Software is furnished "AS IS" and without warranty as to the performance or results you may obtain by using the Software. The entire risk as to the results and performance of the Software is assumed by you. See product enclosure, if any for any additional warranty.

**7. <u>Indemnification</u>**: NovAtel shall be under no obligation or liability of any kind (in contract, tort or otherwise and whether directly or indirectly or by way of indemnity contribution or otherwise howsoever) to the Licensee and the Licensee will indemnify and hold NovAtel harmless against all or any loss, damage, actions, costs, claims, demands and other liabilities or any kind whatsoever (direct, consequential, special or otherwise) arising directly or indirectly out of or by reason of the use by the Licensee of the Software whether the same shall arise in consequence of any such infringement, deficiency, inaccuracy, error or other defect therein and whether or not involving negligence on the part of any person.

**8. <u>Disclaimer and Limitation of Liability:</u>**

**(a)** **THE WARRANTIES IN THIS AGREEMENT REPLACE ALL OTHER WARRANTIES, EXPRESS OR IMPLIED, INCLUDING ANY WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. NovAtel DISCLAIMS AND EXCLUDES ALL OTHER WARRANTIES. IN NO EVENT WILL NovAtel's LIABILITY OF ANY KIND INCLUDE ANY SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES, INCLUDING LOST PROFITS, EVEN IF NovAtel HAS KNOWLEDGE OF THE POTENTIAL LOSS OR DAMAGE.**

(b) NovAtel will not be liable for any loss or damage caused by delay in furnishing the Software or any other performance under this Agreement.

(c) NovAtel's entire liability and your exclusive remedies for our liability of any kind (including liability for negligence) for the Software covered by this Agreement and all other performance or non-performance by NovAtel under or related to this Agreement are to the remedies specified by this Agreement.

**9. <u>Governing Law</u>**: This Agreement is governed by the laws of the Province of Alberta, Canada. Each of the parties hereto irrevocably attorns to the jurisdiction of the courts of the Province of Alberta.

**10. <u>Customer Support</u>**: For Software UPDATES and UPGRADES, and regular customer support, contact the NovAtel GPS Hotline at 1-800-NOVATEL (U.S. or Canada only), or 403-295-4900, Fax 403-295-4901, e-mail to support@novatel.ca,
website: http://www.novatel.com or write to:

NovAtel Inc.
Customer Service Dept.
1120 - 68 Avenue NE,
Calgary, Alberta, Canada T2E 8S5

# Standard Terms and Conditions of Sales

**1. PRICES:** All prices are Firm Fixed Price, FCA 1120 - 68th Avenue N.E., Calgary, Alberta. All prices include standard commercial packing for domestic shipment. All transportation, insurance, special packing costs and expenses, and all Federal, provincial and local excise, duties, sales, and other similar taxes are the responsibility of the Purchaser.

**2. PAYMENT:** Terms are prepayment unless otherwise agreed in writing. Interest shall be charged on overdue accounts at the rate of 18% per annum (1.5% per month) from due date. To expedite payment by wire transfer to NovAtel Inc.: Bank - HSBC Bank of Canada

| | | | |
|---|---|---|---|
| Bank: | HSBC Bank of Canada | US Account # | 788889-002 |
| | 407 - 8 Avenue S.W. | CDN Account # | 788889-001 |
| | Calgary, AB, Canada T2P 1E5 | EURO Account # | 788889-270 |
| | | Transit # | 10029-016 |
| | | Swift | HKBCCATTCAL |

**3. DELIVERY:** Purchaser shall supply shipping instructions with each order. (Ship to and bill to address, NovAtel Quotation #, Preferred carrier and account #, Custom broker/freight forwarder including name and contact #) In the absence of specific instructions, NovAtel may select a carrier and insure Products in transit and charge Purchaser accordingly. NovAtel shall not be responsible for any failure to perform due to unforeseen circumstances or causes beyond its ability to reasonably control. Risk of loss, damage or destruction shall pass to Purchaser upon delivery to carrier. Goods are provided solely for incorporation into the Purchaser’s end product and shall not be onward delivered except as incorporated in the Purchaser’s end product.

**4. COPYRIGHT AND CONFIDENTIALITY:** Copyright in any specification, drawing, computer software, technical description and other document supplied by NovAtel under or in connection with the Order and all intellectual property rights in the design of any part of the Equipment or provision of services, whether such design be registered or not, shall vest in NovAtel absolutely. The Buyer shall keep confidential any information expressed or confirmed by NovAtel in writing to be confidential and shall not disclose it without NovAtel's prior consent in writing to any third party or use it other than for the operation and maintenance of any Equipment provided.

**5. GENERAL PROVISIONS:** All Purchase Orders are subject to approval and acceptance by NovAtel. Any Purchase Order or other form from the Purchaser, which purports to expand, alter or amend these terms and conditions, is expressly rejected and is and shall not become a part of any agreement between NovAtel and the Purchaser. This agreement shall be interpreted under the laws of the Province of Alberta.

**6. LIMITED WARRANTY AND LIABILITY:** Warranty Period: Products - 1 year; Accessories - 90 days (in each case from the date of invoice). NovAtel warrants that during the Warranty Period that (a) the Product will be free from defects in material and workmanship and conform to NovAtel specifications; (b) the software will be free from error which materially affect performance; and (c) if applicable as defined in the User's Manual, be eligible for access to post contract support and software updates when available. **THESE WARRANTIES ARE EXPRESSLY IN LIEU OF ALL OTHER WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, WITHOUT LIMITATION, ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. NOVATEL SHALL IN NO EVENT BE LIABLE FOR SPECIAL, INDIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY KIND OR NATURE DUE TO ANY CAUSE.**

Purchaser's exclusive remedy for a claim under this warranty shall be limited to the repair or replacement at NovAtel's option and at NovAtel's facility, of defective or nonconforming materials, parts or components or in the case of software, provision of a software revision for implementation by the Buyer. All material returned under warranty shall be returned to NovAtel prepaid by the Buyer and returned to the Buyer, prepaid by NovAtel. The foregoing warranties do not extend to (i) nonconformities, defects or errors in the Products due to accident, abuse, misuse or negligent use of the Products or use in other than a normal and customary manner, environmental conditions not conforming to NovAtel's specifications, or failure to follow prescribed installation, operating and maintenance procedures, (ii) defects, errors or nonconformities in the Products due to modifications, alterations, additions or changes not made in accordance with NovAtel's specifications or authorized by NovAtel, (iii) normal wear and tear, (iv) damage caused by force of nature or act of any third person, (v) shipping damage, (vi) service or repair of Product by the Purchaser without prior written consent from NovAtel, (vii) Products designated by NovAtel as beta site test samples, experimental, developmental, preproduction, sample, incomplete or out of specification Products, (viii) returned Products if the original identification marks have been removed or altered or (ix) Services or research activities.

**7. EXCLUSION OF LIABILITY:** If a Party would, but for this paragraph (7), have concurrent claims in contract and tort (including negligence) such claims in tort (including negligence) shall to the extent permitted by law be wholly barred, unenforceable and excluded.

NovAtel shall not be liable to the Buyer by way of indemnity or by reason of any breach of the Order or of statutory duty or by reason of tort (including but not limited to negligence) for any loss of profit, loss of use, loss of production, loss of contracts or for any financing costs or for any indirect or consequential damage whatsoever that may be suffered by the Buyer.

In the event and to the extent that NovAtel shall have any liability to Buyer pursuant to the terms of the Order, NovAtel shall be liable to Buyer only for those damages which have been foreseen or might have reasonably been foreseen on the date of effectivity of the Order and which are solely an immediate and direct result of any act or omission of NovAtel in performing the work or any portion thereof under the Order and which are not in the aggregate in excess of ten (10%) percent of the total Order price.

# Warranty Policy

NovAtel Inc. warrants that its Global Positioning System (GPS) products are free from defects in materials and workmanship, subject to the conditions set forth below, for the following time periods:

| | |
|---|---|
| SPAN-CPT Unit | One (1) Year |
| GPSAntenna™ Series | One (1) Year |
| Cables and Accessories | Ninety (90) Days |
| Computer Discs | Ninety (90) Days |
| Software Warranty | One (1) Year |

Date of sale shall mean the date of the invoice to the original customer for the product. NovAtel's responsibility respecting this warranty is solely to product replacement or product repair at an authorized NovAtel location only.

Determination of replacement or repair will be made by NovAtel personnel or by technical personnel expressly authorized by NovAtel for this purpose (*continued on Page 14*).

***WARNING:*** **Only return an IMU to its manufacturer and not to NovAtel.**

NovAtel warrants that during the Warranty Period that (a) the Product will be free from defects in material and workmanship and conform to NovAtel specifications; and (b) the software will be free from error which materially affect performance. THESE WARRANTIES ARE EXPRESSLY IN LIEU OF ALL OTHER WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, WITHOUT LIMITATION, ALL IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. NOVATEL SHALL IN NO EVENT BE LIABLE FOR SPECIAL, INDIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES OF ANY KIND OR NATURE DUE TO ANY CAUSE.

Purchaser's exclusive remedy for a claim under this warranty shall be limited to the repair or replacement at NovAtel's option and at NovAtel's facility, of defective or nonconforming materials, parts or components or in the case of software, provision of a software revision for implementation by the Buyer. All material returned under warranty shall be returned to NovAtel prepaid by the Buyer and returned to the Buyer, prepaid by NovAtel.

**THE FOREGOING WARRANTIES DO NOT EXTEND TO (I) NONCONFORMITIES, DEFECTS OR ERRORS IN THE PRODUCTS DUE TO ACCIDENT, ABUSE, MISUSE OR NEGLIGENT USE OF THE PRODUCTS OR USE IN OTHER THAN A NORMAL AND CUSTOMARY MANNER, ENVIRONMENTAL CONDITIONS NOT CONFORMING TO NOVATEL'S SPECIFICATIONS, OR FAILURE TO FOLLOW PRESCRIBED INSTALLATION, OPERATING AND MAINTENANCE PROCEDURES, (II) DEFECTS, ERRORS OR NONCONFORMITIES IN THE PRODUCTS DUE TO MODIFICATIONS, ALTERATIONS, ADDITIONS OR CHANGES NOT MADE IN ACCORDANCE WITH NOVATEL'S SPECIFICATIONS OR AUTHORIZED BY NOVATEL, (III) NORMAL WEAR AND TEAR, (IV) DAMAGE CAUSED BY FORCE OF NATURE OR ACT OF ANY THIRD PERSON, (V) SHIPPING DAMAGE; OR (VI) SERVICE OR REPAIR OF PRODUCT BY THE DEALER WITHOUT PRIOR WRITTEN CONSENT FROM NOVATEL. IN ADDITION, THE FOREGOING WARRANTIES SHALL NOT APPLY TO PRODUCTS DESIGNATED BY NOVATEL AS BETA SITE TEST SAMPLES, EXPERIMENTAL, DEVELOPMENTAL, PREPRODUCTION, SAMPLE, INCOMPLETE OR OUT OF SPECIFICATION PRODUCTS OR TO RETURNED PRODUCTS IF THE ORIGINAL IDENTIFICATION MARKS HAVE BEEN REMOVED OR ALTERED. THE WARRANTIES AND REMEDIES ARE EXCLUSIVE AND ALL OTHER WARRANTIES, EXPRESS OR IMPLIED, WRITTEN OR ORAL, INCLUDING THE IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE ARE EXCLUDED. NOVATEL SHALL NOT BE LIABLE FOR ANY LOSS, DAMAGE, EXPENSE, OR INJURY ARISING DIRECTLY OR INDIRECTLY OUT OF THE PURCHASE, INSTALLATION, OPERATION, USE OR LICENSING OR PRODUCTS OR SERVICES. IN NO EVENT SHALL NOVATEL BE LIABLE FOR SPECIAL, INDIRECT, INCIDENTAL OR CONSEQUENTIAL DAMAGES OF ANY KIND OR NATURE DUE TO ANY CAUSE.**

There are no user serviceable parts in the GPS receiver and no maintenance is required. When the status code indicates that a unit is faulty, replace with another unit and return the faulty unit to NovAtel Inc.

Before shipping any material to NovAtel or Dealer, please obtain a Return Material Authorization (RMA) number from the point of purchase.

Once you have obtained an RMA number, you will be advised of proper shipping procedures to return any defective product. When returning any product to NovAtel, please return the defective product in the original packaging to avoid ESD and shipping damage.

# Customer Service

## Firmware Upgrades

Firmware upgrades are firmware releases, which increase basic functionality of the receiver from one model to a higher level model type. When available, upgrades may be purchased at a price, which is the difference between the two model types on the current NovAtel GPS Price List plus a nominal service charge.

Please refer to the *PC Software and Firmware* chapter in the *OEMV Installation and Operation User Manual.*

## Contact Information

Firmware upgrades are accomplished through NovAtel authorized dealers.

Contact your local NovAtel dealer first for more information. To locate a dealer in your area or if the problem is not resolved, contact NovAtel Inc. directly using one of the following methods:

Call the NovAtel GPS Hotline at 1-800-NOVATEL (North America), or 403-295-4900 (international)

Fax: 403-295-4901 E-mail: support@novatel.ca Website: http://www.novatel.com

Write: NovAtel Inc., Customer Service Dept., 1120 - 68 Avenue NE, Calgary, AB., Canada, T2E 8S5

---

✉ Before contacting NovAtel Customer Service regarding software concerns, please do the following:

1. Establish communication with the receiver

2 Log the following data to a file on your PC for 30 minutes:

| | |
|---|---|
| RXSTATUSB | once |
| RAWEPHEMB | onchanged |
| RANGEB | ontime 1 |
| BESTPOSB | ontime 1 |
| RXCONFIGA | once |
| VERSIONB | once |
| RAWIMUSB | onnew |
| INSPVASB | ontime 0.1 |
| INSCOVSB | onchanged |
| INSUPDATEB | onchanged |
| BESTGPSPOSB | ontime 1 |

3. Send the file containing the logs to NovAtel Customer Service using the support@novatel.com e-mail address.

---

# Notices

***CAUTION***

1. This device incorporates circuitry to absorb most static discharges. However, severe static shock may cause inaccurate operation of the unit. Use anti-static precautions where possible.
2. This device is a precision instrument. It performs best when handled with care.

# Foreword

## Congratulations!

Congratulations on purchasing your SPAN-CPT GPS/INS receiver. SPAN (Synchronized Position Attitude Navigation) Technology features tight integration of a NovAtel GPS receiver and an Inertial Measurement Unit (IMU). SPAN provides continuous navigation information, using an Inertial Navigation System (INS), to bridge short Global Position System (GPS) outages. Designed for dynamic applications, SPAN provides precise position, velocity and attitude information. SPAN CPT (Compact, Portable, and Tightly Coupled) combines the GPS and IMU hardware inside one enclosure for simple installation and operation. Commercial components have been chosen for integration into SPAN-CPT in order to offer the same benefits of other SPAN products but with fewer export restrictions.

By complementing GPS with inertial measurements, SPAN-CPT Technology provides robust positioning in challenging conditions where GPS alone is less reliable. During short periods of GPS outage, or when less than four satellites are received, SPAN-CPT Technology offers uninterrupted position and attitude output. The tight coupling of inertial technology with GPS also provides the benefits of faster satellite reacquisition and faster RTK initialization after outages.

NovAtel's OEMV-3 receiver is the processing engine of SPAN-CPT and the IMU components are manufactured by KVH Industries.

## Scope

This manual contains sufficient information on the installation and operation of the SPAN-CPT system. It is beyond the scope of this manual to provide details on service or repair. Contact your local NovAtel dealer for any customer-service related inquiries; see Customer Service on *page 15*.

After the addition of accessories, an antenna and a power supply, the SPAN-CPT system is ready to go.

The OEMV-3 in the receiver utilizes a comprehensive user-interface command structure, which requires communications through its communications (COM) ports. This manual describes the INS specific commands and logs. Other supplementary manuals are available to aid you in using the other commands and logs available with OEMV family products. It is recommended that these documents be kept together for easy reference.

SPAN system output is compatible with post-processing software from NovAtel's Waypoint Products Group. Visit our website at www.novatel.com for details.

# Prerequisites

The installation chapters of this document provide information concerning the installation requirements and considerations for the different parts of the SPAN-CPT system.

To run the SPAN-CPT system software, your personal computer must meet or exceed this minimum configuration:

- Microsoft Windows user interface (Windows 2000 or higher)
- Pentium Microprocessor recommended
- VGA Display
- Windows compatible mouse or pointing device

Although previous experience with Windows is not necessary to use the SPAN-CPT system software, familiarity with certain actions that are customary in Windows will assist in the usage of the program. This manual has been written with the expectation that you already have a basic familiarity with Windows.

# Chapter 1 Introduction


**Figure 1: SPAN-CPT System**

NovAtel's SPAN-CPT technology brings together two very different but complementary positioning and navigation systems namely GPS and an Inertial Navigation System (INS). By combining the best aspects of GPS and INS into one system, SPAN technology is able to offer a solution that is more accurate and reliable than either GPS or INS could provide alone. The combined GPS/INS solution has the advantage of the absolute accuracy available from GPS and the continuity of INS through traditionally difficult GPS conditions.

GPS positioning observes range measurements from orbiting Global Positioning System Satellites. From these observations, the receiver can compute position and velocity with high accuracy. NovAtel GPS positioning systems have been established as highly accurate positioning tools, however GPS in general has some significant restrictions, which limit its usefulness in some situations. GPS positioning requires line of site view to at least four satellites simultaneously. If these criteria are met, differential GPS positioning can be accurate to within a few centimetres. If however, some or all of the satellite signals are blocked, the accuracy of the position reported by GPS degrades substantially, or may not be available at all.

An INS uses forces and rotations measured by an IMU to calculate position, velocity and attitude. This capability is embedded in the firmware of the SPAN-CPT. Forces are measured by accelerometers in three perpendicular axes within the IMU and the gyros measure angular rotation rates around those axes. Over short periods of time, inertial navigation gives very accurate acceleration, velocity and attitude output. The INS must have prior knowledge of its initial position, initial velocity, initial attitude, Earth rotation rate and gravity field. Since the IMU measures changes in orientation and acceleration, the INS determines changes in position and attitude, but initial values for these parameters must be provided from an external source. Once these parameters are known, an INS is capable of providing an autonomous solution with no external inputs. However, because of errors in the IMU measurements that accumulate over time, an inertial-only solution degrades with time unless external updates such as position, velocity or attitude are supplied.

The SPAN-CPT system's combined GPS/INS solution integrates the raw inertial measurements with all available GPS information to provide the optimum solution possible in any situation. By using the high accuracy GPS solution, the IMU errors can be modeled and mitigated. Conversely, the continuity and relative accuracy of the INS solution enables faster GPS signal reacquisition and RTK solution convergence.

The advantages of using SPAN-CPT technology are its ability to:

- Provide a full attitude solution (roll, pitch and azimuth)
- Provide continuous solution output (in situations when a GPS-only solution is impossible)
- Provide faster signal reacquisition and RTK solution resolution (over stand-alone GPS because of the tightly integrated GPS and INS filters)
- Output high-rate (up to 100 Hz) position, velocity and attitude solutions for high-dynamic applications, see also *Logging Restriction Important Notice* on *page 72*.
- Use raw phase observation data (to constrain INS solution drift even when too few satellites are available for a full GPS solution)

## 1.1 System Components

The SPAN-CPT system consists of the following components:

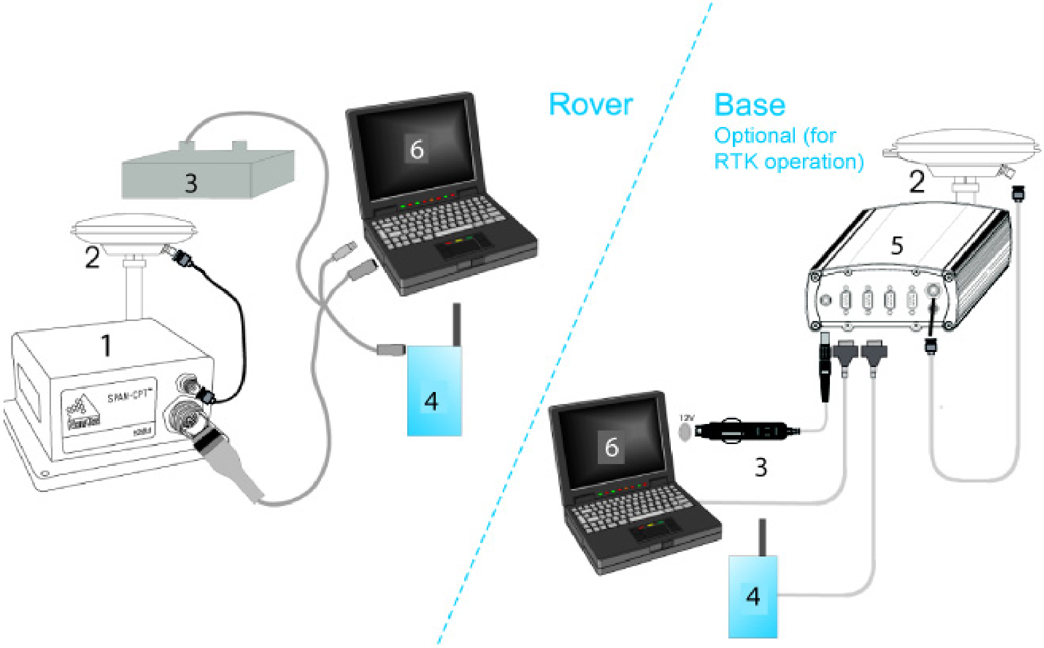
- **SPAN-CPT Integrated INS/GPS unit.** This unit has 3 accelerometers, 3 gyroscopes (gyros) and a NovAtel OEMV3 Receiver. Excellent acquisition and re-acquisition times allow this receiver to operate in environments where very high dynamics and frequent interruption of signals can be expected.
- **PC software**. Real-time data collection, status monitoring and receiver configuration is possible through NovAtel's Control and Display Unit (CDU) software utility, see *SPAN-CPT Configuration with CDU* on *page 30.*

# Chapter 2 SPAN-CPT Installation

## 2.1 Hardware Description

The hardware setup consists of a SPAN-CPT enclosure containing the GPS and IMU components (see *Figure 1* on *page 19*) a GPS antenna, power and a radio link (if your application requires real time differential operation).

*Figure 2* shows a typical hardware set-up.




| Reference | Description |
|---|---|
| 1 | A SPAN-CPT receiver connected to a laptop for data storage |
| 2 | User-supplied NovAtel GPS antenna |
| 3 | User-supplied power supply<br>+9 to +18 V DC |
| 4 | User-supplied radio device to COM2 |
| 5 | User-supplied base station OEMV Family receiver |
| 6 | User-supplied PC, for setting up and monitoring, to COM1 or USB |

**Figure 2: Basic Set-Up**

The sections that follows outline how to set up the system's parts and cables. See *Appendix A Technical Specifications s*tarting on *page 44*.

✉ Use a USB cable to log raw data. Serial communication is fine for configuring and monitoring the SPAN through Hyperterminal or **CDU**. USB is required if you have a post-processing application requiring 100 Hz IMU data. We also recommend you use **CDU** to collect the data.

### 2.1.1 SPAN-CPT Hardware

The SPAN-CPT receiver contains the OEMV-3 GPS receiver and an IMU containing 3 accelerometers and 3 gyroscopes. Communication is done using either the com ports or USB through the multi-I/O connector.


**Figure 3: SPAN-CPT Enclosure**

## 2.2 Hardware Set-Up

Review this section’s hardware set-up subsections and follow the numbered steps, in bold, to install your SPAN system.

### 2.2.1 Mount Antenna

For maximum positioning precision and accuracy, as well as to minimize the risk of damage, ensure that the antenna is securely mounted on a stable structure that will not sway or topple. Where possible, select a location with a clear view of the sky to the horizon so that each satellite above the horizon can be tracked without obstruction. The location should also be one that minimizes the effect of multipath interference.

### 2.2.2 Mount SPAN-CPT

Mount the SPAN-CPT in a fixed location where the distance from the SPAN-CPT to the GPS antenna phase center is constant. Ensure that the SPAN-CPT orientation with respect to the vehicle and antenna is also constant.

**For greatest ease of use, the SPAN-CPT should be mounted such that the positive Z-axis marked on the SPAN-CPT enclosure points up and the Y-axis points forward through the front of the vehicle, in the direction of track.**

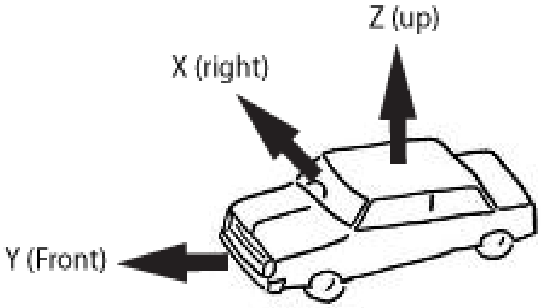



**Figure 4: SPAN-CPT Enclosure Mounting**

1. **Mount the SPAN-CPT enclosure and antenna securely to a vehicle**. Ensure they cannot move due to dynamics and that the distance and relative direction between them is fixed.

2. **Measure the lever arm offsets from the SPAN-CPT navigation centre to the antenna phase centre in the SPAN-CPT enclosure frame**.

Also, it is important to measure the distance from the SPAN-CPT to the antenna (the Antenna Lever Arm), each installation on the axis defined on the SPAN-CPT enclosure. See *Appendix A, Technical Specifications* .

---

✉ 1. The closer the antenna is to the SPAN-CPT, the more accurate the position solution. Also, your measurements when using the SETIMUTOANTOFFSET command must be as accurate as possible, or at least more accurate than the GPS positions being used. **For example, a 10 cm error in recording the antenna offset will result in at least a 10 cm error in the output. Millimeter accuracy is preferred.**

2. The offset from the SPAN-CPT to the antenna, and/or a user point device, must remain constant especially for RTK or DGPS data. Ensure the SPAN-CPT, antenna and user point device are bolted in one position perhaps by using a custom bracket.

---

### 2.2.3 Connect COM Cables

SPAN-CPT has one multi-purpose I/O connector that contains pins for the RS232 com ports, USB ports, pps signals and event input triggers. Refer to Appendix A for port pin definitions.

3. **Connect the USB port of the SPAN-CPT to a computer USB port.** Alternatively, connect the COM 1 port of the SPAN-CPT to the computer COM port via a null modem cable.

4. **Connect the antenna to the antenna port on the enclosure using an appropriate coaxial cable**.

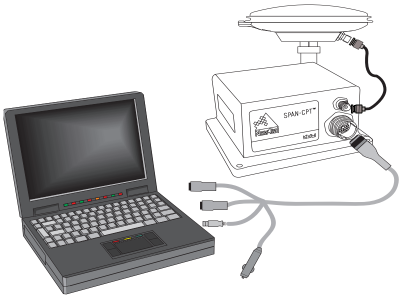


**Figure 5: Connect the antenna to the antenna port**

### 2.2.4 Connect Power

The SPAN-CPT system receiver requires an input supply voltage between +9 VD C and +18 VD C. The receiver has an internal power module that does the following:

- filters and regulates the supply voltage
- protects against over-voltage, over-current, and high-temperature conditions
- provides automatic reset circuit protection

---

☒ The power supply used to power the SPAN-CPT must be monotonic during power on to ensure internal logic blocks are initialized appropriately and proceed to valid operating states. If the power supply is not monotonic during power on, the accelometer status in the IMU status may show a failure and the accelerometer measurements in the RAWIMUS log (see the RAWIMUS log description starting on *page 122*) will be zero. Power cycling with a monotonic power up clears this error state.

---

Power input pins are located on the multi-purpose I/O connector. Be sure to connect the power with the correct polarity and ensure the power source is within specifications. See *Appendix A Technical Specifications* for power input requirements.

There is always a drop in voltage between the power source and the power port due to cable loss. Improper selection of wire gauge can lead to an unacceptable voltage drop at the SPAN-CPT system. A paired wire run represents a feed and return line. Therefore, a 2-m wire pair represents a total wire path of 4 m. For a SPAN-CPT system operating from a 12 V system, a power cable longer than 2.1 m (7 ft.) should not use a wire diameter smaller than 24 AWG.

5. **Apply power to the SPAN-CPT**. See *Figure 6* on the following page.

---

☒ It is recommended that you place a back-up battery between the SPAN-CPT and its voltage supply as a power buffer if installed in a vehicle. When a vehicle engine is started, power can dip to 9.6 V DC or cut-out to ancillary equipment.

---

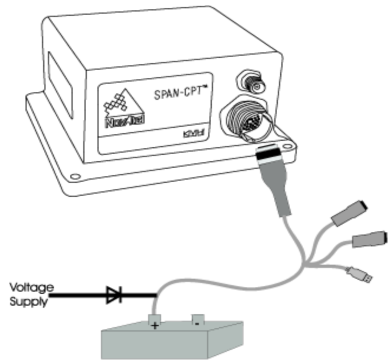



**Figure 6: Apply Power to the SPAN-CPT**

# Chapter 3 SPAN-CPT Operation

Before operating your SPAN-CPT system, ensure that you have followed the installation and setup instructions in *Chapter 2, SPAN-CPT Installation* starting on *page 21*.

You can use NovAtel's **CDU** software to configure receiver settings and to monitor data in real-time, between a rover SPAN-CPT system and base station.

SPAN-CPT system output is compatible with post-processing software from NovAtel's Waypoint Products Group. Visit our website at http://www.novatel.com/products/waypoint_pps.htm for details.

***WARNING:*** **Ensure the Control Panel's Power Settings on your PC are not set to go into Hibernate or Standby modes. Data will be lost if one of these modes occurs during a logging session.**

## 3.1 Definition of Reference Frames Within SPAN

The reference frames that are most frequently used throughout this manual are the following:

- The Local-Level Frame
- The SPAN Body Frame
- The Enclosure Frame
- The Vehicle Frame

### *3.1.1 The Local-Level Frame (ENU)*

The definition of the local level coordinate frame is as follows:

- z-axis– pointing up (aligned with gravity)
- y-axis– pointing north
- x-axis- pointing east

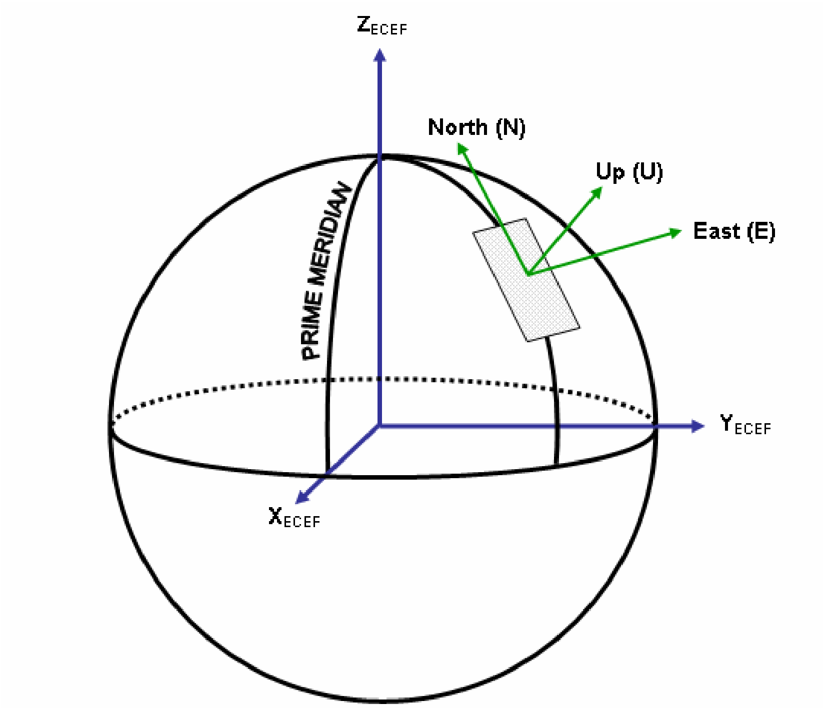



**Figure 7: Local-Level Frame (ENU)**

### *3.1.2 The SPAN Body Frame*

The definition of the SPAN body frame is as follows:

- z-axis– pointing up (aligned with gravity)
- y-axis– defined by how user has mounted the IMU
- x-axis – defined by how user has mounted the IMU

To determine your SPAN x-axis and y-axis, see *Table 8* on *page 59*. This frame is also known as the computation frame and is the frame where all the mechanization equations are computed.

### 3.1.3 The Enclosure Frame

The definition of the enclosure frame is defined on the IMU and represents how the sensors are mounted in the enclosure. If the IMU is mounted with the z-axis (as marked on the IMU enclosure) pointing up, the IMU enclosure frame is the same as the SPAN frame.

This origin of this frame is not the enclosure center, but the Center of Navigation (sensor center).

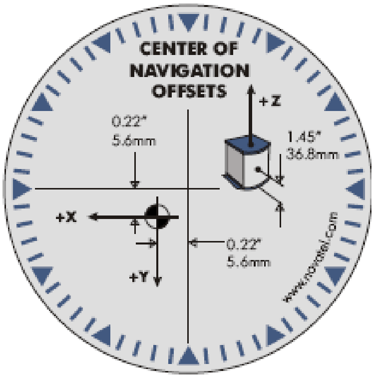



**Figure 8: The Enclosure Frame**

### 3.1.4 The Vehicle Frame

The definition of the vehicle frame is as follows:

- z-axis– points up through the roof of the vehicle perpendicular to the ground
- y-axis– points out the front of the vehicle in the direction of travel
- x-axis–completes the right-handed system (out the right-hand side of the vehicle when facing forward

See the VEHICLEBODYROTATION command on *page 91* for information on entering the rotation into the system and see the RVBCALIBRATE command on *page 76* for information on calculating this rotation.

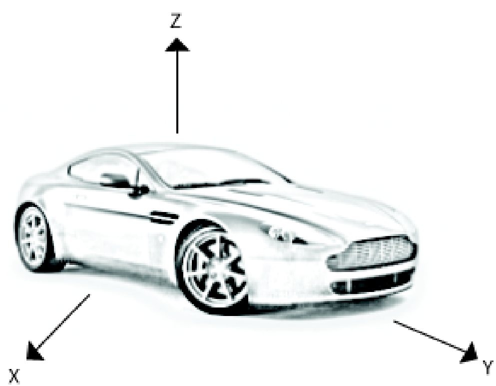


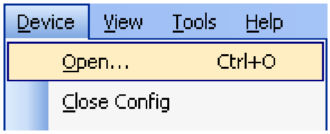

**Figure 9: Vehicle Frame**

## 3.2 Communicating with the SPAN-CPT System

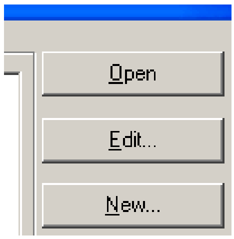
Once the receiver is connected to the PC, antenna, and power supply, install NovAtel's OEMV PC Utilities (**CDU** and *Convert*). You can find installation instructions in your receiver's *Quick Start Guide*. (Alternatively, you can use a terminal emulator program such as HyperTerminal to communicate with the receiver.) Refer also to the **CDU** Help file for more details on **CDU**. The Help file is accessed by choosing *Help* from the main menu in **CDU**.

Start **CDU** on your PC to enable communication:

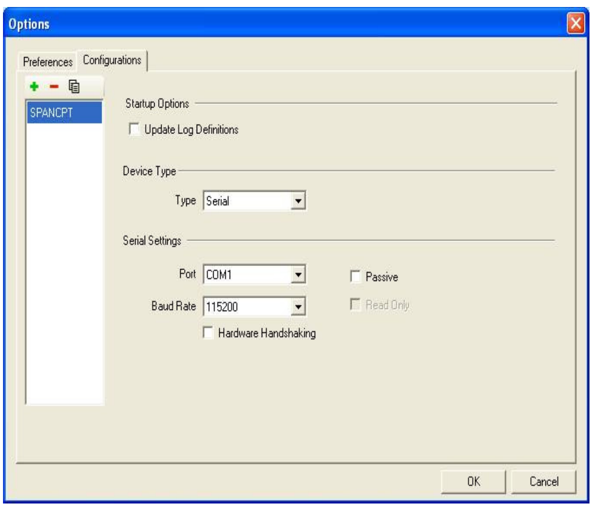
1. Launch **CDU** from the *Start* menu folder specified during the installation process. The default location is *Start* | *Programs* | *NovAtel OEMV* | *NovAtel CDU*.
2. Select *Open....* from the *Device* menu.




3. Select the *New...* button in the *Open* dialog box. The *Options* | *Configuration* dialog opens.




4. Use the ✚ button at the top of the configurations selection box to add a new configuration. To delete a configuration, select it from the list and click on the ▬ button. To duplicate an existing configuration, click on the button. You can select any name in the list and edit it to change it.



5. Select *Serial,* or *USB,* from the *Type* list and select the PC/laptop port, that the SPAN-CPT is connected to, from the *Port* list.
6. Select *115200* from the *Baud Rate* list.
7. Uncheck the *Hardware handshaking* checkbox.
8. Select *OK* to save the new device settings.
9. Select the new configuration from the *Available device configs* area of the *Open* dialog.
10. Select the *Open* button to open SPAN-CPT communications.


11. As **CDU** establishes the communication session with the receiver, a progress box is displayed.
12. Select *Tools | Logging Control Window* from the **CDU** main menu to control the receiver's logging to files and serial ports. Refer to **CDU**'s on-line Help for more information.
13. Use the *Console* window to enter commands. See also *Section 3.4, Data Collection* on *page 41*.

---

If you have to power down your receiver, ensure that all windows, other than the Console window, are closed in **CDU** and then use the SAVECONFIG command.

---

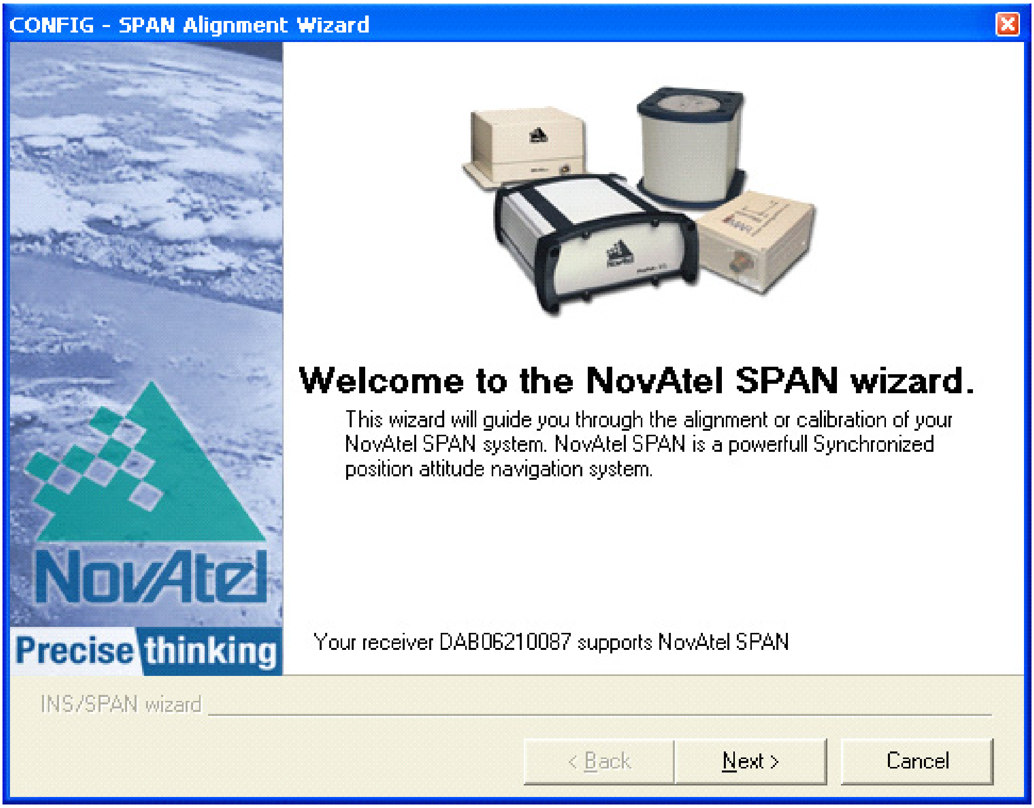
### 3.2.1 SPAN-CPT Configuration with CDU

Follow these steps to enable INS as part of the SPAN system using the NovAtel **CDU** software utility:

---

The **CDU** screen shots in this manual are from **CDU** Version 3.3.0.3 and may differ from your **CDU** version.

---




#### 3.2.1.1 SPAN-CPT basic configuration:

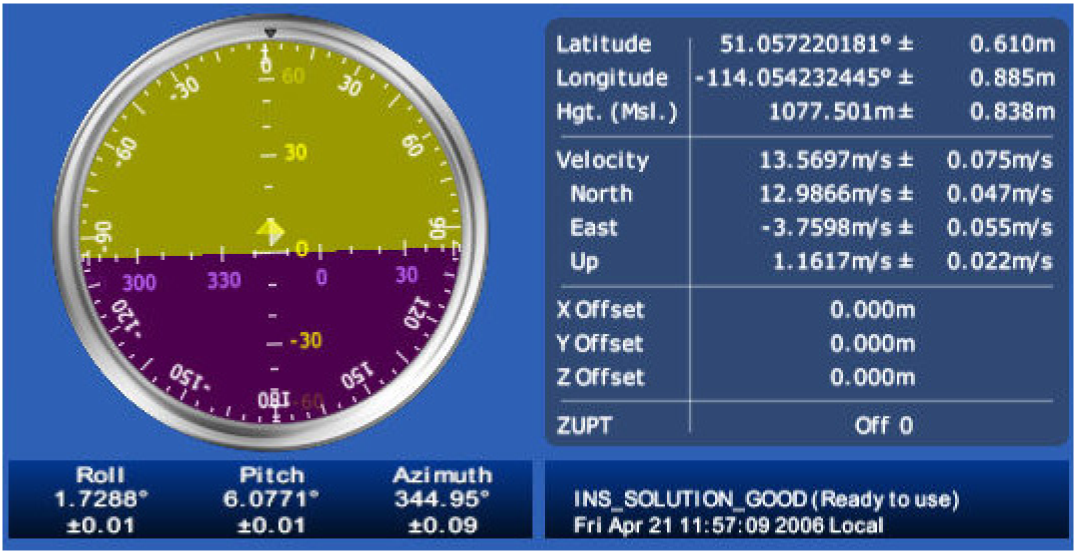
Select *Tools* | *SPAN-CPT Alignment Wizard* from the main menu of CDU. This wizard takes you through the steps to complete a coarse or fast alignment, select the type of IMU and configure the receiver port, connected to the IMU, to accept IMU data.

### 3.2.2 INS Window in CDU

**CDU** is a 32 bit Windows application. The application provides a graphical user interface to set-up and monitor the operation of the SPAN system by providing a series of windows.

The INS Window in **CDU** is described below. Please refer to the *OEMV Family Installation and Operation User Manual* for more details on **CDU** and other OEMV Family PC software programs.

•***INS Window:*** The Position, Velocity and Attitude (roll, pitch and azimuth) sections display data from the INSPVA log along with standard deviations calculated from the INSCOV log. Information in the ZUPT (Zero Velocity Update) section reflects the current INSZUPT command setting. The receiver uses the *X*,*Y* and *Z Offset* fields to specify an offset from the IMU, for the output position and velocity of the INS solution, as specified by the SETINSOFFSET command or **CDU**'s SPAN-CPT wizard. The *INS Configuration/Status* section displays the IMU type, IMU Status and local date/time information. The dial is a graphical display of the Roll, Pitch and Azimuth values indicated by an arrow on each axis.




### 3.2.3 SPAN-CPT Configuration using Command Line

#### 3.2.3.1 GPS Configuration

The GPS configuration can be set up for different accuracy levels such as single point, SBAS, DGPS and RTK (RTCA, RTCM, RTCM V3, CMR, and CMR Plus). ProPak-V3 receivers can also be set up for Omnistar HP, Omnistar XP, Omnistar VBS or CDGPS. Refer to the *OEMV User Manuals* for details on DGPS, RTK, L-band or SBAS setup and operation.

With no additional configuration, the system operates in single point mode.

#### 3.2.3.2 INS Configuration

Once communication has been established to the receiver, issue the SETIMUTOANTOFFSET command to enter the distance from the SPAN-CPT to the GPS antenna, see *page 60*. The offset between the antenna phase centre and the IMU navigation centre must remain constant and be known accurately (m). The X, Y and Z positive directions are clearly marked on the SPAN-CPT enclosure. The SETIMUTOANTOFFSET parameters are (where the standard deviation fields are optional and the distances are measured from the IMU navigation centre to the Antenna Phase Centre):

SETIMUTOANTOFFSET x_offset y_offset z_offset [x_stdev] [y_stdev] [z_stdev]

A typical RTK GPS solution is accurate to a few centimeters. For the SPAN-CPT system to have this level of accuracy, the offset must be measured to within a centimeter. Any offset error between the two systems shows up directly in the output position. For example, a 10 cm error recording this offset will result in at least a 10 cm error in the output.

CDU can also be used to configure the SPAN-CPT. See *Section 3.2.1, SPAN-CPT Configuration with CDU* on *page 30*.

## 3.3 Real-Time Operation

SPAN-CPT operates through the OEMV command and log interface. Commands and logs specifically related to SPAN-CPT operation are documented in *Appendices B* and *C* of this manual respectively.

Real-time operation notes:

- Inertial data does not start until FINESTEERING time status is reached, and therefore, the SPAN-CPT system does not function unless a GPS antenna is connected with a clear view of the sky.
- The Inertial solution is computed separately from the GPS solution. The GPS solution is available from the SPAN-CPT system through the GPS-specific logs even without SPAN running. The integrated INS/GPS solution is available through special INS logs documented in *Appendix C* of this manual.
- The IMU solution is available at the maximum rate of output of the SPAN-CPT (100 Hz). Because of this high data rate, a shorter header format was created. These shorter header logs are defined with an S (RAWIMUSB rather than RAWIMUB). We recommend you use these logs instead of the standard header logs to save throughput on the COM port.

Status of the inertial solution can be monitored using the inertial status field in the INS logs, *Table 1* below.

**Table 1: Inertial Solution Status**

| Binary | ASCII | Description |
|---|---|---|
| 0 | INS_INACTIVE | IMU logs are present, but the alignment routine has not started; INS is inactive. |
| 1 | INS_ALIGNING | INS is in alignment mode. When in this status, the user can move to initiate the kinematic alignment or send a SETINITAZIMUTH command. This status also shows if the IMU status is not valid. The IMU status is given in the RAWIMU and RAWIMUS logs. See *Section B.2.11* on *page 63.* |
| 2 | INS_SOLUTION_NOT_GOOD | The INS solution is still being computed but the azimuth solution uncertainty has exceed 2 degrees. The solution is still valid but you should monitor the solution uncertainty in the INSCOV log. You may encounter this state during times when the GPS, used to aid the INS, is absent. |
| 3 | INS_SOLUTION_GOOD | The INS filter is in navigation mode and the INS solution is good. |
| 6 | INS_BAD_GPS_AGREEMENT | The INS filter is in navigation mode, and the GPS solution is suspected to be in error.<br>This may be due to multipath or limited satellite visibility. The inertial filter has rejected the GPS position and is waiting for the solution quality to improve. |
| 7 | INS_ALIGNMENT_COMPLETE | The INS filter is in navigation mode, but not enough vehicle dynamics have been experienced for the system to be within specifications. |

### 3.3.1 System Start-Up and Alignment Techniques

The system requires an initial attitude estimate to start the navigation filter. This is called system alignment. On start-up the system has no position, velocity or attitude information. When the system is first powered up, the following sequence of events happens:

1. The first satellites are tracked and coarse time is solved
2. Enough satellites are tracked to compute a position
3. Receiver "fine time" is solved, meaning the time on board the receiver is accurate enough to begin timing IMU measurements. The time status in the log headers will indicate FINESTEERING when this happens
4. Raw IMU measurements begin to be timed by the receiver and are available to the INS filter. They are also available to you in the RAWIMUS log, see *page 96*. The INS Status field reports INS_INACTIVE.
5. The inertial alignment routine starts and the INS Status field reports INS_ALIGNING. At this point there are three options for completing the alignment. See the following sections for the options.
6. Alignment is complete and the INS Status field changes to INS_ALIGNMENT_COMPLETE. The system transitions to navigation mode. The GPS/INS solution is available at this point.
7. The solution is refined using updates from GPS. Once the system is operating within specifications, after some vehicle movement, the INS Status field changes to INS_SOLUTION_GOOD. This indicates that the estimated azimuth standard deviation is below 2°. If it increases above 2°, the status changes to INS_SOLUTON_NOTGOOD.

#### 3.3.1.1 Default Kinematic Alignment

The Fast or Kinematic alignment is the default alignment routine for SPAN-CPT.

If the system is mounted as recommended with the Z axis pointing up and the Y axis aligned with the forward direction of the vehicle, then no additional configuration is required to complete a moving alignment.

Once the INS status reaches "INS_ALIGNING" the moving alignment will happen once the vehicle velocity reaches 1.15m/s (~4km/h) and the INS status will change to "INS_ALIGNMENT_COMPLETE". The moving alignment transfers the GPS course over ground pitch and azimuth to the attitude of the IMU. This alignment routine is best suited for ground vehicles where the direction of travel is coincident with the forward axis of the vehicle, and the roll of the vehicle is close to zero. The fast alignment routine may not be suitable for some marine or airborne where the direction of travel may be different from the forward axis of the vehicle because of factors like a crab angle.

If SPAN-CPT is installed with the IMU axes NOT aligned with the vehicle, then additional configuration is needed to complete the moving alignment. These settings can be set graphically using the INS configuration wizard in the CDU interface program or through the command interface by issuing the following commands:

1. Specify which IMU axis is most closely aligned with gravity using the SETIMUORIENTATION command. See *page 57* for a description of this command, and table with the number

corresponding to each orientation. For example, if the Z-axis of your SPAN-CPT is pointing up, you would send this command:

SETIMUORIENTATION 5

2. Specify the angular offset from the vehicle frame to the SPAN body frame (known as the vehicle/body rotation or RVB) using the VEHICLEBODYROTATION command, see *page 68*. Following the example started above, if the IMU is installed rotated so that the Y axis points out the right hand side of the vehicle instead of forward, then you would then enter this command:

VEHICLEBODYROTATION 0 0 -90

---

☒ Note: these angular rotations are difficult to visualize, so if you have complex rotations or some other axis that +Z pointing up, use the CDU SPAN wizard for assistance with these settings.

---

The accuracy of the initial attitude will depend on the dynamics of the vehicle and the accuracy of the angles input in the VEHICLEBODYROTATION command. The alignment is only an estimate of the attitude of the vehicle and as the vehicle experiences dynamics, the accuracy of the attitude solution will improve. Once the attitude accuracy has converged the INS status will change to "INS_SOLUTION_GOOD".

#### 3.3.1.2 Coarse Alignment

A coarse alignment uses an average reading of the accelerometers and gyroscopes to estimate the attitude of the IMU. Coarse alignment begins once the INS status changes to "INS_ALIGNING" and requires the vehicle to remain stationary for approximately 60 seconds. The minimum amount of time the vehicle must remain stationary is 5 seconds, but the quality of a coarse alignment computed over a period this short will be very poor.

In order for SPAN-CPT to execute a coarse alignment, an initial estimate of the heading of the vehicle must be entered using the "SETINITAZIMUTH" command. If the IMU Y-axis is facing east to within 10 degrees standard deviation, then the command would be

SETINITAZIMUTH 90 10

The accuracy of the initial attitude will depend on the duration of the stationary period for roll and pitch, and the standard deviation entered in the SETINITAZIMUTH command for the azimuth. The input azimuth standard deviation must match the actual quality of the input azimuth for your SPAN-CPT to perform optimally. If you are unsure of the input azimuth quality, it is best to use a larger standard deviation rather than a smaller one.

The alignment is only an estimate of the attitude of the vehicle and as the vehicle experiences dynamics, the accuracy of the attitude solution will improve. Once the attitude accuracy has converged the INS status will change to "INS_SOLUTION_GOOD".

#### 3.3.1.3 Manual Alignment

If you know the attitude of your vehicle (roll, pitch, azimuth) you can manually enter the attitude information using the SETINITATTITUDE command. Details of this command start on *page 61*.

### 3.3.2 Navigation Mode

Once the alignment routine has successfully completed, SPAN-CPT enters navigation mode.

SPAN-CPT computes the solution by accumulating velocity and rotation increments from the IMU to generate position, velocity and attitude. SPAN-CPT models system errors by using a Kalman filter. The GPS solution, phase observations and automatic zero velocity updates (ZUPTs) provide updates to the Kalman filter. When a sensor is connected to the system, wheel displacement updates are also used in the filter.

Following the alignment the attitude is coarsely defined, especially in heading. Vehicle dynamics, specifically turns, stops and starts, allow the system to observe the heading error and allows the heading accuracy to converge. Three to five changes in heading should be sufficient to resolve the heading accuracy. The INS Status field changes to INS_SOLUTION_GOOD once convergence is complete. If the attitude accuracy decreases, the INS Status field changes to INS_SOLUTION_NOTGOOD. When the accuracy converges again, the INS status continues as INS_SOLUTION_GOOD.

### 3.3.3 Vehicle to SPAN-CPT Frame Angular Offsets Calibration Routine

Kinematic fast alignment requires that the angular offset between the vehicle and the SPAN-CPT frame is known approximately. If the angles are simple (that is, a simple rotation about one axis) the values can easily be entered manually through the VEHICLEBODYROTATION command, see *page 68*. If the angular offset is more complex (rotation is about 2 or 3 axis), then the calibration routine may provide a more accurate estimation of the values. The steps for the calibration routine are:

1. Apply power to the SPAN-CPT, see the *SPAN-CPT technical specifications* starting on *page 44*.
2. Configure the SPAN-CPT, see *SPAN-CPT Configuration with CDU* on *page 30*.
3. Ensure that an accurate lever arm has been entered into the system.
4. Allow the system to complete a coarse alignment using the SETINITAZIMUTH command. See Coarse Alignment on *page 36* for procedures.
5. Enable the vehicle to body calibration using the RVBCALIBRATE ENABLE command, see *page 56*.
6. Start to move the system under good GPS conditions. Movement of the system under good GPS conditions is required for the observation of the angular offsets.

Vehicle speed must be greater than 5 m/s (18 km/hour) for the calibration to complete. Drive straight on a level surface if possible.

7. When the solved angles are verified (after approximately 30 seconds), the calibration stops and the VEHICLEBODYROTATION log will provide the solved values, *see Page 68*. Log VEHICLEBODYROTATION using the ONNEW trigger to monitor the progress of the calibration.

The rotation parameters are saved in NVM for use on start-up in case a fast-alignment is required. Each time the SPAN-CPT is re-mounted this calibration should be performed again. See *Section 3.3.1.1,* on *page 35* and *Section 3.3.1.2,* on *page 36* for details on fast and coarse alignment.

***WARNING:*** **After the RVBCALIBRATE ENABLE command is entered, there are no vehicle-body rotation parameters present and a kinematic alignment is NOT possible. Therefore this command should only be entered after the system has performed either a static or kinematic alignment and has a valid INS solution.**

☒ The solved rotation values are used only for a rough estimate of the angular offsets between the SPAN-CPT and vehicle frames. The offsets are used when aligning the system while in motion (see *Section 3.3.1, System Start-Up and Alignment Techniques* starting on *page 35*). The angular offset values are not applied to the attitude output, unless the APPLYVEHICLEBODYROTATION command is enabled, see *page 50*.

## 3.3.4 SPAN-CPT Wheel Sensor

The SPAN-CPT system supports wheel sensor inputs, integrated via the SPAN-CPT. The SPAN-CPT accepts TTL- level input pulses from a wheel sensor through the multi-pin connector. See *Appendix A* on *page 44* for specifications on the wheel sensor interface.

### 3.3.4.1 Wheel Sensor Update Logic

The wheel sensor information is sent to the SPAN-CPT along with the raw IMU data. The Corrsys Datron wheel pulse transducer is used as an example, see *Section A.1.1,* on *page 45*.

The SPAN-CPT Kalman filter uses sequential TIMEDWHEELDATA logs to compute a distance traveled between update intervals (1Hz). This information can be used to constrain free-inertial drift during times of poor GPS visibility. The filter also contains a state for modeling the circumference of the wheel as it may change due to hardware changes or environmental conditions.

The modeled wheel circumference is available in the WHEELSIZE log, see *page 102*. Information on how the wheel sensor updates are being used is available in the INSUPDATE log, see *page 92*.

### 3.3.4.2 Odometer Requirements

SPAN-CPT is compatible with any wheel sensor meeting the following requirements:

- Input range less than or equal to 45 KHz
- Input duty cycle is symmetric 40%-60%
- Active input voltage is greater than or equal to 2.5 VDC with a max input voltage of 50 VDC

- Inactive voltage is less than or equal to 1 VDC
- Input current is approximately 3.5 mA at 5 VDC with a maximum of 5 mA at 50 VDC
- Ensure input current does not exceed 5 mA. There is a current limiting diode that can dissipate 800 mW on the input opto-isolator
- Quadrature, pulse and direction type odometers are compatible

An example of a SPAN-CPT compatible odometer is the WPT (Wheel Pulse Transducer) from Corrsys Datron. (www.corrsys-datron.com)

A transducer traditionally fits to the outside of a non-drive wheel. A pulse is then generated from the transducer which is fed directly to the ODO connector on the IMU cable.

**Figure 10: Corrsys Datron WPT**

The WPT mounts to the wheel lug nuts via adjustable mounting collets. The torsion protection rod, which maintains rotation around the wheel axis, affixes to the vehicle body with suction cups. Refer to the Corrsys Datron WPT user manual for mounting instructions.

SPAN-CPT will power the odometer. See *Appendix A* on *page 44* for the pin outs of the SPAN-CPT cable. Connect the appropriate pins to your chosen odometer. If you chose the Corrsys-Datron WPT, first modify the cable at the WPT end. The cable modification is shown in *Table 2* on *page 40*.

**Table 2: Cable Modification for Corrsys-Datron WPT**

| 8-pin M12 connector on the Corrsys-Datron cable [a, b] | | | Female DB9 connector |
|---|---|---|---|
| Pin 1 | GND | White | No change |
| Pin 2 | $+U_B$ (Input Power) | Brown | |
| Pin 3 | Signal A | Green | 6 |
| Pin 4 | Signal A inverted | Yellow | 7 |
| Pin 5 | Signal B | Grey | 3 |
| Pin 6 | Signal B inverted | Pink | 1 |
| Pin 7 | Reserved | | No change |
| Pin 8 | | | |

a. Pin 2 is wired to a red banana plug (Power in) and Pin 1 is wired to a black banana plug (Power return) so the WPT needs power to operate (+10 to +30 V). Solder the shield on the WPT cable to the female DB9 housing.

b. This modification is for the Corrsys Datron WPT 8-pin M12-plug cable number 14865.

## 3.4 Data Collection

The INS solution is available in the INS-specific logs with either a standard or short header. Other parameters are available in the logs shown in *Table 3* on *page 41*:

**Table 3: Solution Parameters**

| Parameter | Log |
|---|---|
| Position | INSPOS *or* INSPOSS<br>INSPVA *or* INSPVAS |
| Velocity | INSVEL *or* INSVELS<br>INSSPD *or* INSSPDS<br>INSPVA *or* INSPVAS |
| Attitude | INSATT *or* INSATTS<br>INSPVA *or* INSPVAS |
| Solution Uncertainty | INSCOV *or* INSCOVS |

Note that the position, velocity and attitude are available together in the INSPVA and INSPVAS logs.

The inertial solution is available up to the rate of 100 Hz. Data can be requested at a specific rate up to the maximum IMU output rate, or can be triggered by the mark input trigger at rates up to 20 Hz.

The GPS-only solution is still available through the GPS-only logs such as RTKPOS, PSRPOS and OMNIHPPOS. When running SPAN-CPT, rates of non-INS logs should be limited to a maximum rate of 5 Hz. Refer to the *OEMV Family Firmware Reference Manual* for more details on these logs. INS-only data logging and output can be at rates of up to the rate of the IMU data.

✉ **The highest rate that you should request GPS logs (RANGE, BESTPOS, RTKPOS, PSRPOS, and so on) while in INS operation is 5 Hz. If the receiver is not running INS, GPS logs can be requested at rates up to 20 Hz.**

***WARNING:*** Ensure that all windows, other than the Console, are closed in **CDU** and then use the SAVECONFIG command to save settings in NVM. Otherwise, unnecessary data logging occurs and may overload your system.

Specific logs need to be collected for post-processing. See *Section 3.4* on *page 41*.

To store data with a SPAN-CPT, connect a laptop computer to it. The laptop computer should be equipped with a data storage device such as a Compact Flash Card, CD or MP3 disc.

### Logging Restriction Important Notice

High-rate data logging is regulated in SPAN to prevent logging of unusable data or overloading the system. Please note these 3 rules when configuring your SPAN-CPT system:

1. Only one high-rate INS log can be configured for output at a time. Once a log is selected for output at a rate equal to 100 Hz, all other log requests are limited to a maximum rate of 50 Hz. Below are examples of acceptable logging requests:

| | |
|---|---|
| LOG RAWIMUSB ONNEW | (100 Hz) |
| LOG INSPVASB ONTIME 0.02 | (acceptable 50 Hz logging) |

The following is rejected because RAWIMU has already been requested at 100 Hz:

| | |
|---|---|
| LOG INSPOSSB ONTIME 0.01 | (100 Hz request) |

Below is another example set of acceptable logging requests:

| | |
|---|---|
| LOG INSPOSSB ONTIME 0.01 | (100 Hz request) |
| LOG INSVELSB ONTIME 0.02 | (50 Hz request) |

The following are rejected in this case because INSPOSSB has already been requested at a high rate.

| | |
|---|---|
| LOG RAWIMUSB ONNEW | (100 Hz request) |
| LOG INSATTSB ONTIME 0.01 | (100 Hz request) |

2. RAWIMU and RAWIMUS logs are only available with the ONNEW or ONCHANGED trigger. These logs are not valid with the ONTIME trigger. The raw IMU observations contained in these logs are sequential changes in velocity and rotation. As such, you can only use them for navigation if they are logged at their full rate. See details of these log starting on *page 96.*

3. In order to collect wheel sensor information, useful in post-processing, the TIMEDWHEELDATA log should only be used with the ONNEW trigger.

## 3.5 Data Collection for Post Processing

Some operations such as aerial measurement systems do not require real-time information from SPAN-CPT. These operations are able to generate the position, velocity or attitude solution post-mission in order to generate a more robust and accurate solution than is possible in real-time.

In order to generate a solution in post-processing, data must be simultaneously collected at a base station and each rover. The following logs must be collected in order to successfully post process data:

From a base:

- RANGECMPB ONTIME 1
- RAWEPHEMB ONCHANGED

From a rover:

- RANGECMPB ONTIME 1
- RAWEPHEMB ONCHANGED
- RAWIMUSB ONNEW
- BESTLEVERARMB ONNEW

Post processing is performed through the Waypoint Inertial Explorer software package available from from NovAtel's Waypoint Products Group. Visit our website at [http://www.novatel.com/products/waypoint_inertial.htm](http://www.novatel.com/products/waypoint_inertial.htm) for details.

---

✉ The highest rate that you should request GPS logs (RANGE, BESTPOS, RTKPOS, PSRPOS, and so on) while in INS operation is 5 Hz. If the receiver is not running INS, GPS logs can be requested at rates up to 20 Hz.

---

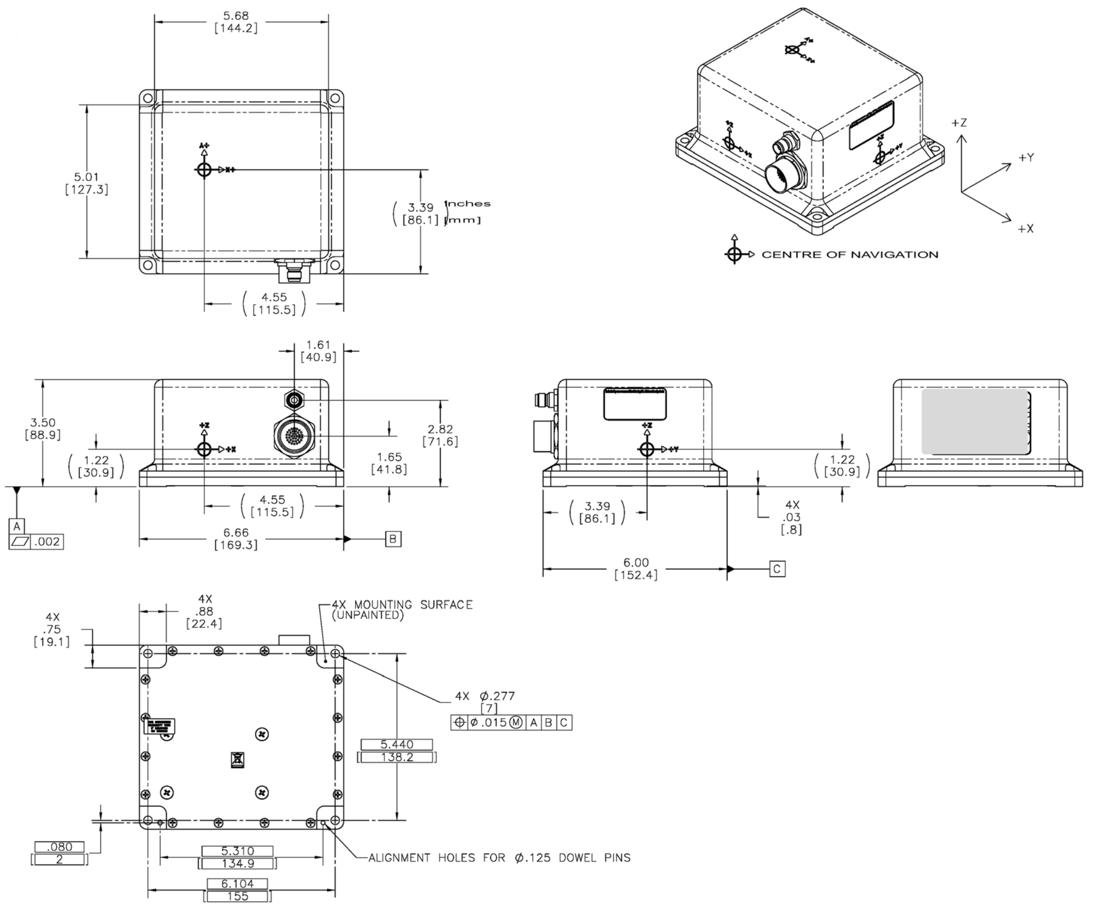
# Appendix A Technical Specifications

This appendix details the technical specifications of the SPAN-CPT.

## A.1 SPAN-CPT Technical Specifications

**Table 4: Technical HW Specs for SPAN-CPT**

| *PHYSICAL* | |
|---|---|
| SPAN-CPT Enclosure Size | 127.0 mm L X 105.41 mm W X 88.9 mm H |
| SPAN-CPT Weight | 1.81 kg |



## A.1.1 SPAN-CPT Cable

The NovAtel part numbers for the SPAN-CPT cable are:

- KVH Standard Unterminated Cable - 60723107
- KVH Development Terminated Cable - 60723108

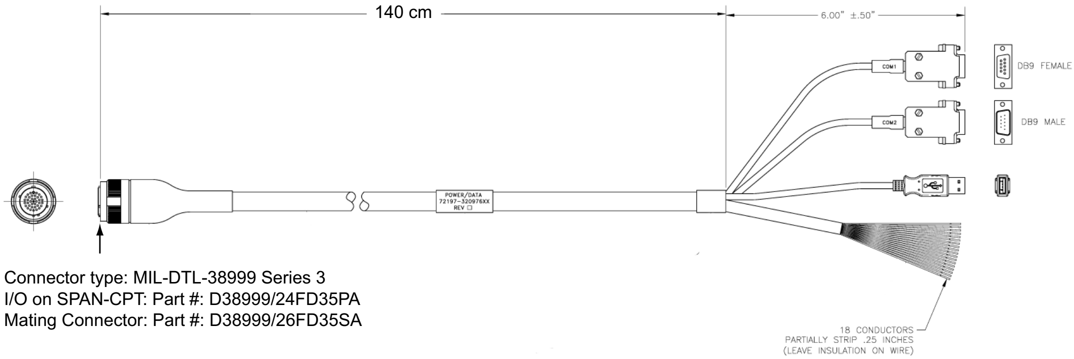



**Figure 12: SPAN-CPT-KVH Development Terminated Cable**

☒ The SPAN-CPT cable also has a green ground line that is not shown in this drawing. The green ground line is grounded to the SPAN-CPT connector and enclosure.

**Table 5: SPAN-CPT-KVH Connector Pin-Out Descriptions**

| Pin No. | Function | Bare Connectors | Female DB9 to COM1 | Male DB9 to COM2 | USB | Comments |
|---|---|---|---|---|---|---|
| 1 | Power Return | Black | | | | |
| 2 | 9-18 VDC Power Input | White | | | | |
| 3 | COM1 RS422 TX+/ RS232 TX From OEMV-3 Module | | 2 | | | |
| 4 | COM1 RS422 TX-/ RS232 RTS\ From OEMV-3 Module | | 8 | | | |
| 5 | COM1 RS422 RX+/ RS232 RX TO OEMV-3 Module | | 3 | | | |
| 6 | COM1 RS422 RX-/ RS232 CTS\ TO OEMV-3 Module | | 7 | | | |
| 7 | COM1 RS422 Select In | | | | | Tie together with pin number 8 to select RS422 |

*Continued on the following page.*

| Pin No. | Function | Bare Connectors | Female DB9 to COM1 | Male DB9 to COM2 | USB | Comments |
|---|---|---|---|---|---|---|
| 8 | COM1 RS422 Select Out | | | | | Tie together with pin number 7 to select RS422 |
| 9 | COM1 SIGNAL GND | | 5 | | | |
| 10 | OEMV-3 USB D+ | Black | | | 3 | |
| 11 | OEMV-3 USB D- | White | | | 2 | |
| 12 | OEMV-3 USB SIGNAL GND | | | | 4 | |
| 13 | Odometer Power | White | | | | |
| 14 | Odometer Power Return | Black | | | | |
| 15 | ODO SIGA | Black | | | | |
| 16 | ODO SIGA Inverted | White | | | | |
| 17 | ODO SIGB | Black | | | | |
| 18 | ODO SIGB Inverted | White | | | | |
| 19 | COM2 RS232 TX FROM OEMV-3 Module | | | 3 | | |
| 20 | COM2 RS232 RX TO OEMV-3 Module | | | 2 | | |
| 21 | COM2 RS232 RTS to OEMV-3 Module | | | 7 | | |
| 22 | COM2 RS232 CTS to OEMV-3 Module | | | 8 | | |
| 23 | COM2 RS232 DTR to OEMV-3 Module | | | 4 | | |
| 24 | COM2 RS232 DCD to OEMV-3 Module | | | 6 | | |
| 25 | COM2 Signal GND | | | 5 | | |
| 26 | Reserved | White | | | | |
| 27 | Reserved | White | | | | |
| 28 | Reserved | White | | | | |
| 29 | PPS from OEMV-3 | White | | | | |
| 30 | EVENT1 to OEMV-3 | Black | | | | |
| 31 | OEMV-3 Signal GND | White | | | | |

*Continued on the following page.*

| Pin No. | Function | Bare Connectors | Female DB9 to COM1 | Male DB9 to COM2 | USB | Comments |
|---|---|---|---|---|---|---|
| 32 | CAN2L OEMV-3 | White | | | | |
| 33 | CAN2H OEMV-3 | Black | | | | |
| 34 | CAN2 SIGNAL GND | Black | | | | |
| 35 | N/C | | | | | |
| 36 | N/C | | | | | |
| 37 | Chassis GND | White | | | | |
| - | Chassis GND | Green | | | | |

## A.1.2 KVH IMU Sensor Specs

| ***PERFORMANCE* - FIBER OPTIC GYROS** | | |
|---|---|---|
| Bias Offset | ± 20 | °/hr |
| Turn On To Turn On Bias Repeatability (Compensated) | ± 3 | °/hr |
| In Run Bias Variation, At Constant Temperature | 1 | °/hr @ 1σ |
| Scale Factor Error (Total) | 1500 | ppm, 1σ |
| Scale Factor Linearity | 1000 | ppm, 1σ |
| Temperature Dependent SF Variation | 500 | ppm, 1σ |
| Angular Random Walk | 0.0667 | °/√hr @ 1σ |
| Max Input | ± 375 | °/sec |

| ***PERFORMANCE* - ACCELEROMETERS** | | |
|---|---|---|
| Bias Offset | ± 50 | mg |
| Turn On To Turn On Bias Repeatability | ± 0.75 | mg |
| In Run Bias Variation, At Constant Temperature | 0.25 | mg @ 1σ |
| Temperature Dependent Bias Variation | 0.5 | mg/°C @ 1σ |
| Scale Factor Error (Total) | 4000 | ppm, 1σ |
| Temperature Dependent SF Variation | 1000 | ppm, 1σ |
| Accel Noise | 55 | μg/√Hz @ 1σ |
| Bandwidth | 50 | Hz |
| Max Input | ±10 | g |

## A.1.3 Electrical and Environmental

| ***ELECTRICAL*** | |
|---|---|
| Input Power | 9 - 18 VDC |
| Power consumption | 15 W (MAX) |
| Start-Up Time (Valid Data) | < 5 secs |
| ***ENVIRONMENTAL*** | |
| Temperature, operational | -40°C to +65°C |
| Temperature, non-operational | -50°C to +80°C |
| Vibration, operational | 6 g rms, 20 Hz - 2 KHz |
| Vibration, non-operational | 8 g rms, 20 Hz -2 KHz |
| Shock, operational | 7g 6-10 msec, 1/2 sine |
| Shock, non-operational | 60 g 6-10 msec, 1/2 sine |
| Altitude | -1000 to 50,000 ft. |
| Humidity | 95% at 35°C, 48 hrs |
| MTBF | ≥ 10,500 hours |

# Appendix B INS Commands

The INS-specific commands are described further in this chapter.

For information on other available commands, refer to the *OEMV Family Firmware Reference Manual*.

## B.1 Using a Command as a Log

All NovAtel commands may be used for data input, as normal, or used to request data output (a unique OEMV Family feature). INS-specific commands may be in Abbreviated ASCII, ASCII, or Binary format.

Consider the *lockout* command (refer to the *OEMV Family Firmware Reference Manual*) with the syntax:

lockout prn

You can put this command into the receiver to de-weight an undesirable satellite in the solution, or you can use the *lockout* command as a log to see if there is a satellite PRN that has already been locked out. In ASCII, this might be:

log com1 lockouta once

Notice the ‘a’ after *lockout* to signify you are looking for ASCII output.

---

☒ The highest rate that you should request GPS logs (RANGE, BESTPOS, RTKPOS, PSRPOS, and so on) while in INS operation is 5 Hz. If the receiver is not running INS, GPS logs can be requested at rates up to 20 Hz depending on the software model.

---

***WARNING:*** Ensure that all windows, other than the Console, are closed in **CDU** and then use the SAVECONFIG command to save settings in NVM. Otherwise, unnecessary data logging occurs and may overload your system.

---

## B.2 INS-Specific Commands

Please refer to the *OEMV Family Firmware Reference Manual* for a complete list of commands categorized by function and then detailed in alphabetical order.

## B.2.1 APPLYVEHICLEBODYROTATION Enable Vehicle to Body Rotation

This command allows you to apply the vehicle to body rotation to the output attitude (that was entered from the VEHICLOBODYROTATION command, see *page 68*). This rotates the SPAN body frame output in the INSPVA, INSPVAS and INSATT logs to the vehicle frame. APPLYVEHICLEBODYROTATION is disabled by default.

**Abbreviated ASCII Syntax:** **Message ID: 1071**

APPLYVEHICLEBODYROTATION [switch]

| Field | Field Type | ASCII Value | Binary Value | Description | Binary Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 1 | header | - | - | This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively. | - | H | 0 |
| 2 | switch | Disable | 0 | Enable/disable vehicle body rotation using values entered in the vehiclebodyrotation command. default = disable | Enum | 4 | H |
| | | Enable | 1 | | | | |

**Input Example:**

applyvehiclebodyrotation enable

## B.2.2 FRESET Factory Reset

This command clears data which is stored in non-volatile memory. Such data includes the almanac, ephemeris, and any user-specific configurations. The receiver is forced to hardware reset.

**Abbreviated ASCII Syntax:** **Message ID: 20**

FRESET [target]

| Field | Field Type | ASCII Value | Binary Value | Description | Binary Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 1 | header | - | - | This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively. | - | H | 0 |
| 2 | target | See *Table 6* | | What data is to be reset by the receiver. | Enum | 4 | H |

**Input Example:**

FRESET COMMAND

**Table 6: FRESET Target**

| Binary | ASCII | Description |
|---|---|---|
| 0 | STANDARD | Resets commands, ephemeris, and almanac (default). Also resets all OmniSTAR related data except for the subscription information. |
| 1 | COMMAND | Resets the stored commands (saved configuration) |
| 2 | GPSALMANAC | Resets the stored almanac |
| 3 | GPSEPHEM | Resets stored ephemeris |
| 5 | MODEL | Resets the currently selected model |
| 11 | CLKCALIBRATION | Resets the parameters entered using the CLOCKCALIBRATE command |
| 20 | SBASALMANAC | Resets the stored SBAS almanac |
| 21 | LAST_POSITION | Resets the position using the last stored position |
| 22 | VEHICLE_BODY_R | Resets stored vehicle to body rotations |
| 24 | INS_LEVER_ARM | Resets the GPS antenna to IMU lever arm |

## B.2.3 INSCOMMAND INS Control Command

This command allows you to enable or disable INS positioning. When INS positioning is disabled, no INS position, velocity or attitude is output. Also, INS aiding of RTK initialization and tracking reacquisition is disabled. If the command is used to disable INS and then re-enable it, the INS system has to go through its alignment procedure (equivalent to issuing a RESET command). See also *Section 3.3.1, System Start-Up and Alignment Techniques* starting on *page 35*

**Abbreviated ASCII Syntax:** **Message ID: 379**

INSCOMMAND action

| Field | Field Type | ASCII Value | Binary Value | Description | Binary Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 1 | header | - | - | This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively. | - | H | 0 |
| 2 | action | RESET | 0 | Resets the GPS/INS alignment and restarts the alignment initialization. | Enum | 4 | H |
| | | DISABLE | 1 | Disables INS positioning. | | | |
| | | ENABLE | 2 | Enables INS positioning where alignment initialization starts again. (default) | | | |

**Abbreviated ASCII Example:**

INSCOMMAND ENABLE

## B.2.4 INSPHASEUPDATE INS Phase Update Control

This command allows you to control the INS phase updates.

When enabled, raw GPS phase measurements are used to control errors in the inertial filter. In a typical INS/GPS integration, GPS positions are used to control inertial drifts. Some features of phase updates include:

- updates can be performed even when too few satellites are available to compute a GPS solution
- as few as 2 satellites must be in view to perform a precise update
- system performance is significantly improved in conditions challenging to GPS such as urban canyons and foliage.

**Abbreviated ASCII Syntax:** **Message ID: 639**

INSPHASEUPDATE switch

| Field | Field Type | ASCII Value | Binary Value | Description | Binary Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 1 | header | - | - | This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively. | - | H | 0 |
| 2 | switch | DISABLE | 0 | Disable INS delta-phase updates. | Enum | 4 | H |
| | | ENABLE | 1 | Enable INS delta-phase updates. (default) | | | |

**Abbreviated ASCII Example:**

INSPHASEUPDATE ENABLE

## B.2.5 INSZUPT Request Zero Velocity Update

This command allows you to manually perform a Zero Velocity Update (ZUPT), that is, to update the receiver when the system has stopped.

NovAtel's SPAN Technology System does ZUPTs automatically. It is not necessary to use this command under normal circumstances.

***WARNING:*** **This command should only be used by advanced users of GPS/INS.**

**Abbreviated ASCII Syntax:** **Message ID: 382**

INSZUPT

## B.2.6 NMEATALKER Set the NMEA talker ID V123

This command allows you to alter the behavior of the NMEA talker ID. The talker is the first 2 characters after the $ sign in the log header of the GPGLL, GPGST, GPRMB, GPRMC, and GPVTG log outputs. Other NMEA logs are not affected by the NMEATALKER command.

The default GPS NMEA message (`nmeatalker GP`) outputs GPas the talker ID regardless of the position type given in position logs such as BESTPOS. The `nmeatalker auto` command switches the talker ID between GP and IN according to the position type given in position logs.

**Abbreviated ASCII Syntax:** **Message ID: 861**

NMEATALKER [ID]

**Factory Default:**

nmeatalker gp

**ASCII Example:**

nmeatalker auto

---

✉ This command only affects NMEA logs that are capable of a GPS position output. For example, GPGSV is for information on GPS satellites and its output always uses the GP ID. *Table 7* shows the NMEA logs and whether they use GP or GP + IN IDs with `nmeatalker auto`.

---

**Table 7: NMEA Talkers**

| Log | GPGLL | GPGST | GPRMB | GPRMC | GPVTG |
|---|---|---|---|---|---|
| Talker IDs | GP/IN | GP/IN | GP/IN | GP/IN | GP/IN |

| Field | Field Type | ASCII Value | Binary Value | Description | Binary Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 1 | NMEA-TALKER header | - | - | This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively. | - | H | 0 |
| 2 | ID | GP | 0 | GPS (GP) only | Enum | 4 | H |
| | | AUTO | 1 | GPS and/or Inertial (IN) | | | |

## B.2.7 RVBCALIBRATE Vehicle to Body Rotation Control

The RVBCALIBRATE command is used to enable or disable the calculation of the vehicle to SPAN body angular offset. This command should be entered when the SPAN-CPT is re-mounted in the vehicle or if the rotation angles available are known to be incorrect.

*WARNING:* **After the RVBCALIBRATE ENABLE command is entered, there are no vehicle-body rotation parameters present and a kinematic alignment is NOT possible. Therefore this command should only be entered after the system has performed either a static or kinematic alignment and has a valid INS solution.**

A good INS solution and vehicle movement are required for the SPAN-CPT system to solve the vehicle to body SPAN offset. The solved vehicle-body rotation parameters are output in the VEHICLEBODYROTATION log when the calibration is complete. When the calibration is done, the rotation values are fixed until the calibration is re-run by entering the RVBCALIBRATE command again, or by entering the VEHICLEBODYROTATION command with known values.

☒ The solved rotation values are used only for a rough estimate of the angular offsets between the SPAN-CPT and vehicle frames. The offsets are used when aligning the system while in motion (see *Section 3.3.1, System Start-Up and Alignment Techniques* starting on *page 35*). The angular offset values are not applied to the attitude output, unless the APPLYVEHICLEBODYROTATION command is enabled.

**Abbreviated ASCII Syntax:** **Message ID: 641**

RVBCALIBRATE reset

| Field | Field Type | ASCII Value | Binary Value | Description | Binary Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 1 | Log Header | - | - | This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively. | - | H | 0 |
| 2 | Switch | RESET | 0 | Control the vehicle body rotation computation | ENUM | 4 | H |
| | | DISABLE | 1 | | | | |
| | | ENABLE | 2 | | | | |

**Abbreviated ASCII Example:**

RVBCALIBRATE reset

## B.2.8 SETIMUORIENTATION Set IMU Orientation

The SETIMUORIENTATION command is used to specify which of the SPAN-CPT axis is aligned with gravity. The SPAN-CPT orientation can be saved using the SAVECONFIG command so that on start-up, the SPAN-CPT system does not have to detect the orientation of the SPAN-CPT with respect to gravity. This is particularly useful for situations where the receiver is powered while in motion.

---

☒ 1. The default SPAN-CPT axis definitions are:

Y - forward

Z - up

X - out the right hand side.

It is strongly recommended that you mount your SPAN-CPT in this way with respect to the vehicle.

2. You only need to use this command if the system is to be aligned while in motion using the fast alignment routine, see *Section 3.3.1.1, Default Kinematic Alignment* on *page 35*.

---

***WARNING:*** Ensure that all windows, other than the Console, are closed in **CDU** and then use the SAVECONFIG command to save settings in NVM. Otherwise, unnecessary data logging occurs and may overload your system.

---

This orientation command serves to transform the incoming SPAN-CPT signals in such a way that a 5 mapping is achieved, see *Table 8* on *page 59*. For example, if the SPAN-CPT is mounted with the X-axis pointing UP and a mapping of 1 is specified then this transformation of the raw SPAN-CPT data is done:

X⇒Z, Y⇒X, Z⇒Y (where the default is X⇒X, Y⇒Y, Z⇒Z)

Notice that the X-axis observations are transformed into the Z axis, resulting in Z being aligned with gravity and a 5 mapping. The SPAN frame is defined so that Z is always pointing up along the gravity vector. If the IMU mapping is set to 1, the X axis of the IMU enclosure is mapped to the SPAN frame Z axis (pointing up), its Y axis to SPAN frame X and its Z axis to SPAN frame Y.

The X (pitch), Y (roll) and Z (azimuth) directions of the inertial enclosure frame are clearly marked on the SPAN-CPT, see the technical specifications starting on *page 44*.

---

☒ 1.Azimuth is positive in a clockwise direction while yaw is positive in a counter-clockwise direction when looking down the axis centre. Yaw follows the right-handed system convention where as azimuth follows the surveying convention.

2.The data in the RAWIMUS log is never mapped. The axes referenced in the RAWIMUS log description form the SPAN-CPT enclosure frame (as marked on the enclosure).

---

**Abbreviated ASCII Syntax:** **Message ID: 567**

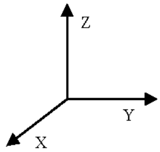
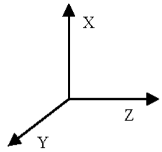
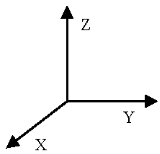
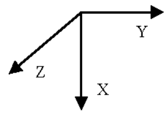
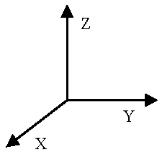
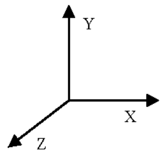
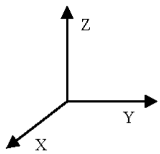
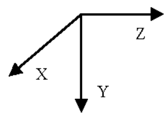
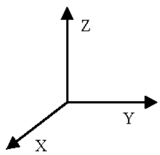
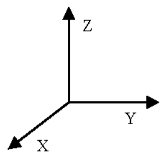
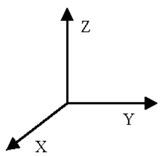
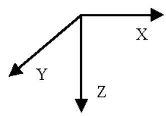
SETIMUORIENTATION switch

| Field | Field Type | ASCII Value | Binary Value | Description | Binary Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 1 | Log Header | - | - | This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively. | - | H | 0 |
| 2 | Switch | 0 | 0 | IMU determines axis orientation automatically during coarse alignment. (default) | ENUM | 4 | H |
| | | 1 | 1 | IMU **X** axis is pointing **UP** | | | |
| | | 2 | 2 | IMU **X** axis is pointing **DOWN** | | | |
| | | 3 | 3 | IMU **Y** axis is pointing **UP** | | | |
| | | 4 | 4 | IMU **Y** axis is pointing **DOWN** | | | |
| | | 5 | 5 | IMU **Z** axis is pointing **UP** | | | |
| | | 6 | 6 | IMU **Z** axis is pointing **DOWN** | | | |

**Abbreviated ASCII Example:**

SETIMUORIENTATION 1

## Table 8: Full Mapping Definitions

| Mapping | SPAN Frame Axes | SPAN Frame | IMU Enclosure Frame Axes | IMU Enclosure Frame |
|---|---|---|---|---|
| 1 | X |  | Y |  |
| | Y | | Z | |
| | Z | | X | |
| 2 | X |  | Z |  |
| | Y | | Y | |
| | Z | | -X | |
| 3 | X |  | Z |  |
| | Y | | X | |
| | Z | | Y | |
| 4 | X |  | X |  |
| | Y | | Z | |
| | Z | | -Y | |
| 5 (default) | X |  | X |  |
| | Y | | Y | |
| | Z | | Z | |
| 6 | X |  | Y |  |
| | Y | | X | |
| | Z | | -Z | |

## B.2.9 SETIMUTOANTOFFSET Set IMU to Antenna Offset

It is recommended that you mount the SPAN-CPT as close as possible to the GPS antenna, particularly in the horizontal plane. This command is used to enter the offset between the SPAN-CPT and the GPS antenna. The measurement should be done as accurately as possible, preferably to within millimeters especially for RTK operation. The x, y and z fields represent the vector from the SPAN-CPT to the antenna phase center in the IMU enclosure frame. The a, b and c fields allow you to enter any possible errors in your measurements. If you think that your 'x' offset measurement is out by a centimeter for example, enter 0.01 in the 'a' field.

The X (pitch), Y (roll) and Z (azimuth) directions of the inertial frame are clearly marked on the SPAN-CPT.

This command must be entered before the INS alignment mode (not after).

**Abbreviated ASCII Syntax:** **Message ID: 383**

SETIMUTOANTOFFSET x y z [a] [b] [c]

| Field | Field Type | ASCII Value | Binary Value | Description | Binary Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 1 | header | - | - | This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively. | - | H | 0 |
| 2 | x | ± 20 | | x offset (m) | Double | 8 | H |
| 3 | y | ± 20 | | y offset (m) | Double | 8 | H+8 |
| 4 | z | ± 20 | | z offset (m) | Double | 8 | H+16 |
| 5 | a | 0 to +1 | | Uncertainty in x (m) (Defaults to 10% of the x offset to a minimum of 0.01 m) | Double | 8 | H+24 |
| 6 | b | 0 to +1 | | Uncertainty in y (m) (Defaults to 10% of the y offset to a minimum of 0.01 m) | Double | 8 | H+32 |
| 7 | c | 0 to +1 | | Uncertainty in z (m) (Defaults to 10% of the z offset to a minimum of 0.01 m) | Double | 8 | H+40 |

**Abbreviated ASCII Example:**

SETIMUTOANTOFFSET 0.54 0.32 1.20 0.03 0.03 0.05

## B.2.10 SETINITATTITUDE Set Initial Attitude of SPAN in Degrees

This command allows you to input a known attitude to start SPAN operation, rather than the usual coarse alignment process. The caveats and special conditions of this command are listed below:

- This alignment is instantaneous based on the user input. This allows for faster system startup; however, the input values must be accurate or SPAN will not perform well.
- If you are uncertain about the standard deviation of the angles you are entering, lean on the side of a larger standard deviation.
- Sending SETINITATTITUDE resets the SPAN filter. The alignment is instantaneous, but some time and vehicle dynamics are required for the SPAN filter to converge. Bridging performance is poor before filter convergence.
- The roll (about the y-axis), pitch (about the x-axis), and azimuth (about the z-axis) are with respect to the SPAN frame. If the SPAN-CPT enclosure is mounted with the z axis pointing upwards, the SPAN frame is the same as the markings on the enclosure. If the SPAN-CPT is mounted in another way, SPAN transforms the SPAN frame axes such that z points up for SPAN computations. You must enter the angles in SETINITATTITUDE with respect to the transformed axis. See SETIMUORIENTATION for a description of the axes mapping that occurs when the IMU is mounted differently from z up.

---

1. Azimuth is positive in a clockwise direction when looking towards the z-axis origin.
2. You do not have to use the SETIMUORIENTATION command, see *page 57,* unless you have your SPAN-CPT mounted with the z axis not pointing up. Then use the tables in the SETIMURIENTATION command, on *Pages 58-59*, to determine the azimuth axis that SPAN is using.

---

**Abbreviated ASCII Syntax:** **Message ID: 862**

SETINITATTITUDE pitch roll azimuth pitchSTD rollSTD azSTD

| Field | Field Type | ASCII Value | Binary Value | Description | Binary Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 1 | header | - | - | This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively. | - | H | 0 |
| 2 | pitch | -360° to +360° | | Input pitch angle, about the x-axis, in degrees | Double | 8 | H |
| 3 | roll | -360° to +360° | | Input roll angle, about the y-axis, in degrees | Double | 8 | H+8 |
| 4 | azimuth | -360° to +360° | | Input azimuth angle, about the z-axis, in degrees | Double | 8 | H+16 |
| 5 | pitchSTD | 0.000278° to 180° default = 1 | | Input pitch standard deviation (STD) angle in degrees | Double | 8 | H+24 |
| 6 | rollSTD | | | Input roll STD angle in degrees | Double | 8 | H+32 |
| 7 | azSTD | | | Input azimuth STD angle in degrees | Double | 8 | H+40 |

**Abbreviated ASCII Example:**

SETINITATTITUDE 0 0 90 5 5 5

In this example, the initial roll and pitch has been set to zero degrees, with a standard deviation of 5 degrees for both. This means that the SPAN-CPT system is very close to level with respect to the local gravity field. The azimuth is 90 degrees (see the SETINITAZIMUTH example on *page 63*), also with a 5 degrees standard deviation.

## B.2.11 SETINITAZIMUTH Set Initial Azimuth and Standard Deviation

This command allows you to start SPAN operation with a previously known azimuth. Azimuth is the weakest component of a coarse alignment, and is also the easiest to know from an external source (i.e. like the azimuth of roadway). This command is needed to perform a coarse alignment. Roll and pitch will be determined using averaged gyro and accelerometer measurements.

- This command is needed to perform a coarse alignment.
- Input azimuth values must be accurate for good system performance.
- Sending SETINITAZIMUTH resets the SPAN filter. The alignment will take approximately 1 minute, but some time and vehicle dynamics are required for the SPAN filter to converge. Bridging performance will be poor before filter convergence.
- The azimuth angle is with respect to the SPAN frame. If the SPAN-CPT enclosure is mounted with the z axis pointing upwards, the SPAN frame is the same as what is marked on the enclosure. If the SPAN-CPT is mounted in another way, SPAN transforms the SPAN frame axes such that z points up for SPAN computations. You must enter the azimuth with respect to the transformed axis. See SETIMUORIENTATION on *page 57*, for a description of the axes mapping that occurs when the SPAN-CPT is mounted differently from z pointing up.

---

1. Azimuth is positive in a clockwise direction when looking towards the z-axis origin.
2. You do not have to use the SETIMUORIENTATION command, see *page 57,* unless you have your SPAN-CPT mounted with the z axis not pointing up. Then, use the tables in the SETIMURIENTATION command, on *pages 58-59*, to determine the azimuth axis that SPAN is using.

---

**Abbreviated ASCII Syntax:** **Message ID: 863**

SETINITAZIMUTH azimuth azSTD

| Field | Field Type | ASCII Value | Binary Value | Description | Binary Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 1 | header | - | - | This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively. | - | H | 0 |
| 2 | azimuth | -360° to +360° | | Input azimuth angle in degrees | Double | 8 | H |
| 3 | azSTD | 0.000278° τo +180° | | Input azimuth standard deviation angle in degrees | Double | 8 | H+8 |

**Abbreviated ASCII Example:**

SETINITAZIMUTH 90 5

In this example, the initial azimuth has been set to 90 degrees. This means that the SPAN system y axis is pointing due East, within a standard deviation of 5 degrees. Note that if you have mounted your SPAN system with the positive z axis (as marked on the enclosure) not pointing up, please refer to the SETIMUORIENTATION command to determine the SPAN frame axes mapping that SPAN automatically applies.

## B.2.12 SETINSOFFSET Set INS Offset

The SETINSOFFSET command is used to specify an offset from the SPAN-CPT for the output position and velocity of the INS solution. This command shifts the position and velocity in the INSPOS, INSPOSS, INSVEL, INSVELS, INSSPD, INSSPDS, INSPVA and INSPVAS logs by the amount specified in metres with respect to the SPAN-CPT enclosure frame axis.

**Abbreviated ASCII Syntax:** **Message ID: 676**

SETINSOFFSET xoffset yoffset zoffset

| Field | Field Type | ASCII Value | Binary Value | Description | Binary Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 1 | header | - | - | This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively. | - | H | 0 |
| 2 | X offset | ± 100 | | Offset along the IMU enclosure frame X axis (m) | Double | 8 | H |
| 3 | Y offset | ± 100 | | Offset along the IMU enclosure frame Y axis (m) | Double | 8 | H+8 |
| 4 | Z offset | ± 100 | | Offset along the IMU enclosure frame Z axis (m) | Double | 8 | H+16 |

**Abbreviated ASCII Example:**

SETINSOFFSET 0.15 0.15 0.25

## B.2.13 SETMARK1OFFSET Set Mark1 Offset

Set the offset to the Mark1 trigger event.

**Abbreviated ASCII Syntax:** **Message ID: 1069**

SETMARK1OFFSET xoffset yoffset zoffset αoffset βoffset γoffset

| Field | Field Type | ASCII Value | Binary Value | Description | Binary Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 1 | header | - | - | This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively. | - | H | 0 |
| 2 | x offset | ± 360 | | Offset along the IMU enclosure frame X axis (m) for Mark1 | Double | 8 | H |
| 3 | y offset | ± 360 | | Offset along the IMU enclosure frame Y axis (m) for Mark1 | Double | 8 | H+8 |
| 4 | z offset | ± 360 | | Offset along the IMU enclosure frame Z axis (m) for Mark1 | Double | 8 | H+16 |
| 5 | αoffset | ± 360 | | Roll offset for Mark1 (degrees) | Double | 8 | H+24 |
| 6 | βoffset | ± 360 | | Pitch offset for Mark1 (degrees) | Double | 8 | H+32 |
| 7 | γoffset | ± 360 | | Azimuth offset for Mark1 (degrees) | Double | 8 | H+40 |

**Abbreviated ASCII Example:**

SETMARK1OFFSET -0.324 0.106 1.325 0 0 0

## B.2.14 SETWHEELPARAMETERS Set Wheel Parameters

The SETWHEELPARAMETERS command can be used when wheel sensor data is available. It allows you to give the filter a good starting point for the wheel size scale factor. It also gives the SPAN filter an indication of the expected accuracy of the wheel data.

Usage of the SETWHEELPARAMETERS command depends on what wheel sensor you are using.

The SETWHEELPARAMETERS command allows you to set the number of ticks per revolution that is correct for your wheel installation (the default is 58).

**Abbreviated ASCII Syntax:** **Message ID: 847**

SETWHEELPARAMETERS ticks circ spacing

| Field | Field Type | ASCII Value | Binary Value | Description | Binary Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 1 | header | - | - | This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively. | - | H | 0 |
| 2 | ticks | 1-10 000 | | Number of ticks per revolution | Ushort | 4 [a] | H |
| 3 | circ | 0.1-100 | | Wheel circumference (m) (default = 1.96 m) | Double | 8 | H+4 |
| 4 | spacing | 0.001-1000 | | Spacing of ticks, or resolution of the wheel sensor (m) | Double | 8 | H+12 |

a. In the binary log case, an additional 2 bytes of padding are added to maintain 4-byte alignment.

**Abbreviated ASCII Example:**

SETWHEELPARAMETERS 58 1.96 0.025

☒ Fields 2, 3 and 4 do not have to 'add up'. Field 4 is used to weight the wheel sensor measurement. Fields 2 and 3 are used with the estimated scale factor to determine the distance travelled.

## B.2.15 VEHICLEBODYROTATION Vehicle to SPAN frame Rotation

Use the VEHICLEBODYROTATION command to set angular offsets between the vehicle frame (direction of travel) and the SPAN body frame (direction that the SPAN-CPT computational frame is pointing). If you estimate the angular offsets using the RVBCALIBRATE command, the VEHICLEBODYROTATION command values are used as the initial values. The uncertainty values are optional (defaults = 0.0). Please see *Section 3.3.3, Vehicle to SPAN-CPT Frame Angular Offsets Calibration Routine* starting on *page 37* for more details. RVBCALIBRATE command information is on *page 56*.

If you use the APPLYVEHICLEBODYROTATION command, the reported attitude in the INSPVA or INSATT logs are in the vehicle frame. Otherwise, the reported attitude is in the SPAN Frame.

The vehicle frame is as follows:

- Vehicle Z Axis - points up through the roof of the vehicle perpendicular to the ground
- Vehicle Y Axis - points out the front of the vehicle in the direction of travel
- Vehicle X Axis - completes the right-handed system (out the right-hand side of the vehicle when facing forward)

The rotation values are used during kinematic alignment. The rotation is used to transform the vehicle frame attitude estimates from GPS into the SPAN frame of the IMU during kinematic alignment.

The uncertainty values report the accuracy of the angular offsets.

---

⌧ If your SPAN-CPT is mounted with the Z-axis (as marked on the IMU enclosure) pointing up, the IMU enclosure frame is the same as the SPAN frame.

---

Follow these steps to measure the rotation angles in the order and direction required for input in the VEHICLEBODYROTATION command:

1. Start with SPAN-CPT enclosure in the vehicle frame as described above.
2. Rotate about the vehicle Z-axis. This angle is the gamma-angle in the command and follows the right-hand rule for sign correction.
3. Rotate about the new X-axis to complete the transformation into the SPAN frame. This angle is the alpha-angle in the command.
4. Finally, rotate about the new Y-axis to align the X-Y plane with the SPAN frame. This angle is the beta-angle in the command.

---

⌧ Enter rotation angles in degrees.

---

To apply the vehicle to body rotation angles, the APPLYVEHICLEBODYROTATION needs to be enabled. Please see *page* 50 for more information.

**Abbreviated ASCII Syntax:** **Message ID: 642**

VEHICLEBODYROTATION alpha beta gamma [σ alpha] [σ beta] [σ gamma]

| Field # | Field Type | Data Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | Log Header | Log header | - | H | 0 |
| 2 | X Angle | Right hand rotation about vehicle frame X axis, degrees | Double | 8 | H |
| 3 | Y Angle | Right hand rotation about vehicle frame Y axis, degrees | Double | 8 | H+8 |
| 4 | Z Angle | Right hand rotation about vehicle frame Z axis, degrees | Double | 8 | H+16 |
| 5 | X Uncertainty | Uncertainty of X rotation, degrees (default=0) | Double | 8 | H+24 |
| 6 | Y Uncertainty | Uncertainty of Yrotation, degrees (default=0) | Double | 8 | H+32 |
| 7 | Z Uncertainty | Uncertainty of Z rotation, degrees (default=0) | Double | 8 | H+40 |
| 8 | xxxx | 32-bit CRC | Hex | 4 | H+48 |
| 9 | [CR][LF] | Sentence terminator (ASCII only) | - | - | - |

Refer also to our application note on *Vehicle to Body Rotations*, NovAtel part number APN-037 (available on our website at http://www.novatel.com/support/applicationnotes.htm).

**Abbreviated ASCII Example:**

```
VEHICLEBODYROTATION 0 0 90 0 0 5
```

# Appendix C INS Logs

The INS-specific logs follow the same general logging scheme as normal OEMV Family logs. They are available in ASCII or binary formats and are defined as being either synchronous or asynchronous. All the logs in this chapter can be used only with the SPAN system.

For information on other available logs and output logging, please refer to the *OEMV Family Firmware Reference Manual*.

One difference from the standard OEMV Family logs is that there are two possible headers for the ASCII and binary versions of the logs. Which header is used for a given log is described in the log definitions in this chapter. The reason for having the alternate short headers is that the normal OEMV-3 binary header is quite long at 28 bytes. This is nearly as long as the data portion of many of the INS logs, and creates excess storage and baud rate requirements. Note that the INS-related logs contain a time tag within the data block in addition to the time tag in the header. The time tag in the data block should be considered the exact time of applicability of the data. All the described INS logs except the INSCOV, INSPOSSYNC and INSUPDATE logs can be obtained at rates up to 100 Hz, subject to the limits of the output baud rate. The covariance log is available once per second.

---

1. Each log ends with a hexadecimal number preceded by an asterisk and followed by a line termination using the carriage return and line feed characters, for example, `*1234ABCD[CR][LF]`. This value is a 32-bit CRC of all bytes in the log, excluding the '#' or '%' identifier and the asterisk preceding the four checksum digits. See also *Section C.1, Description of ASCII and Binary Logs with Short Headers* on *page 71*.
2. The highest rate that you should request GPS logs (RANGE, BESTPOS, RTKPOS, PSRPOS, and so on) while in INS operation is 5 Hz. If the receiver is not running INS (no IMU is attached), GPS logs can be requested at rates up to 20 Hz.

---

Please also refer to the *OEMV Family Firmware Reference Manual* for information on the supplied Convert4 program that lets you change binary to ASCII data, or short binary to short ASCII data, and vice versa. Convert4 is also capable of RINEX conversions to and from ASCII or binary.

*Table 1, Inertial Solution Status* on *page 34* shows the status values included in the INS position, velocity and attitude output logs. If you think you have an IMU unit hooked up properly and you are not getting a good status value, something is wrong and the hardware setup must be checked out. This situation can be recognized in the RAWIMU data by observing accelerometer and gyro values which are not changing with time.

# C.1 Description of ASCII and Binary Logs with Short Headers

These logs are set up in the same way normal ASCII or binary logs are, except that a normal ASCII or binary header is replaced with a short header (see *Tables 9 and 10*). For the message header structure of OEMV-3 regular Binary and ASCII logs, please refer to the *OEMV Family Firmware Reference Manual*.

**Table 9: Short ASCII Message Header Structure**

| Field # | Field Type | Field Type | Description |
|---|---|---|---|
| 1 | % | Char | % symbol |
| 2 | Message | Char | This is the name of the log |
| 3 | Week Number | Ushort | GPS week number |
| 4 | Milliseconds | Ulong | Milliseconds from the beginning of the GPS week |

**Table 10: Short Binary Message Header Structure**

| Field # | Field Type | Field Type | Description | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | Sync | Char | Hex 0xAA | 1 | 0 |
| 2 | Sync | Char | Hex 0x44 | 1 | 1 |
| 3 | Sync | Char | Hex 0x13 | 1 | 2 |
| 4 | Message Length | Uchar | Message length, not including header or CRC | 1 | 3 |
| 5 | Message ID | Ushort | Message ID number | 2 | 4 |
| 6 | Week Number | Ushort | GPS week number | 2 | 6 |
| 7 | Milliseconds | Ulong | Milliseconds from the beginning of the GPS week | 4 | 8 |

## C.2 INS-Specific Logs

The receivers are capable of generating many NovAtel-format output logs, in either Abbreviated ASCII, ASCII or binary format. Please refer to the *OEMV Family Firmware Reference Manual* for a complete list of logs categorized by function and then detailed in alphabetical order.

INS-specific commands and logs provide attitude data such as roll, pitch and azimuth.

### Logging Restriction Important Notice

High-rate data logging is regulated in SPAN to prevent logging of unusable data or overloading the system. Please note these 3 rules when configuring your SPAN system:

1. Only one high-rate INS log can be configured for output at a time. Once a log is selected for output at a rate of 100 Hz, all other log requests are limited to a maximum rate of 50 Hz. Below are examples of acceptable logging requests:

   LOG RAWIMUSB ONNEW (100 Hz)
   LOG INSPVASB ONTIME 0.02 (acceptable 50 Hz logging)

   The following is rejected because RAWIMU has already been requested at 100 Hz:

   LOG INSPOSSB ONTIME 0.01 (100 Hz request)

   Below is another example set of acceptable logging requests:

   LOG INSPOSSB ONTIME 0.01 (100 Hz request)
   LOG INSVELSB ONTIME 0.02 (50 Hz request)

   The following are rejected in this case because INSPOSSB has already been requested at a high rate.

   LOG RAWIMUSB ONNEW (100 Hz request)
   LOG INSATTSB ONTIME 0.01 (100 Hz request)

2. RAWIMUS logs are only available with the ONNEW or ONCHANGED trigger. These logs are not valid with the ONTIME trigger. The raw IMU observations contained in these logs are sequential changes in velocity and rotation. As such, you can only use them for navigation if they are logged at their full rate. See details of these log starting on *page 96*.

3. In order to collect wheel sensor information, useful in post-processing, the TIMEDWHEELDATA log should only be used with the ONNEW trigger. See also *page 100* for details on this log.

☒ The periods available when you use the ONTIME trigger are 0.01 (100 Hz), 0.02 (50 Hz), 0.05, 0.1, 0.2, 0.25, 0.5, 1, 2, 3, 5, 10, 15, 20, 30 or 60 seconds.

The highest rate that you should request GPS logs (RANGE, BESTPOS, RTKPOS, PSRPOS, and so on) while in INS operation is 5 Hz. If the receiver is not running INS, GPS logs can be requested at rates up to 20 Hz depending on the software model.

## C.2.1 BESTGPSPOS Best GPS Position

This log contains the best available GPS position (without INS) computed by the receiver. In addition, it reports several status indicators, including differential age, which is useful in predicting anomalous behavior brought about by outages in differential corrections. A differential age of 0 indicates that no differential correction was used.

With the system operating in an RTK mode, this log reflects the latest low-latency solution for up to 60 seconds after reception of the last base station observations. After this 60 second period, the position reverts to the best solution available; the degradation in accuracy is reflected in the standard deviation fields. If the system is not operating in an RTK mode, pseudorange differential solutions continue for 300 seconds after loss of the data link, though a different value can be set using the DGPSTIMEOUT command, refer to the *OEMV Family Firmware Reference Manual*.

**Structure:** **Message ID: 423**

**Log Type: Synch**

| Field # | Field type | Data Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | header | Log header | - | H | 0 |
| 2 | Sol Status | Solution status, see *Table 12, Solution Status* on *page 76* | Enum | 4 | H |
| 3 | Pos Type | Position type, see *Table 11, Position or Velocity Type* on *page 74* | Enum | 4 | H+4 |
| 4 | Lat | Latitude | Double | 8 | H+8 |
| 5 | Lon | Longitude | Double | 8 | H+16 |
| 6 | Hgt | Height above mean sea level | Double | 8 | H+24 |
| 7 | Undulation | Undulation | Float | 4 | H+32 |
| 8 | Datum ID | Datum ID (refer to the DATUM command in the *OEMV Family Firmware Reference Manual*) | Enum | 4 | H+36 |
| 9 | Lat σ | Latitude standard deviation | Float | 4 | H+40 |
| 10 | Lon σ | Longitude standard deviation | Float | 4 | H+44 |
| 11 | Hgt σ | Height standard deviation | Float | 4 | H+48 |
| 12 | Stn ID | Base station ID | Char[4] | 4 | H+52 |
| 13 | Diff_age | Differential age | Float | 4 | H+56 |
| 14 | Sol_age | Solution age in seconds | Float | 4 | H+60 |
| 15 | #obs | Number of observations tracked | Uchar | 1 | H+64 |
| 16 | #GPSL1 | Number of GPS L1 ranges used in computation | Uchar | 1 | H+65 |
| 17 | #L1 | Number of GPS L1 ranges above the RTK mask angle | Uchar | 1 | H+66 |
| 18 | #L2 | Number of GPS L2 ranges above the RTK mask angle | Uchar | 1 | H+67 |

*Continued on page 74*

| Field # | Field type | Data Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 19 | Reserved | | Uchar | 1 | H+68 |
| 20 | | | Uchar | 1 | H+69 |
| 21 | | | Uchar | 1 | H+70 |
| 22 | | | Uchar | 1 | H+71 |
| 23 | xxxx | 32-bit CRC (ASCII and Binary only) | Hex | 4 | H+72 |
| 24 | [CR][LF] | Sentence terminator (ASCII only) | - | - | - |

**Recommended Input:**

log bestgpsposa ontime 1

**ASCII Example:**

```
#BESTGPSPOSA,COM1,0,62.5,FINESTEERING,1036,484878.000,00000028,63e2,0;
SOL_COMPUTED,SINGLE,51.11629893124,-114.03820302746,1052.3434,
-16.271287293,61,19.6934,13.1515,23.8561,"",0.0,60.000,10,10,0,0,
0,0,0,0*1051ada9
```

**Table 11: Position or Velocity Type**

| Position Type (binary) | Position Type (ASCII) | Description |
|---|---|---|
| 0 | NONE | No solution |
| 1 | FIXEDPOS | Position has been fixed by the FIX POSITION command or by position averaging |
| 2 | FIXEDHEIGHT | Position has been fixed by the FIX HEIGHT, or FIX AUTO, command or by position averaging |
| 3 | Reserved | |
| 4 | FLOATCONV | Solution from floating point carrier phase ambiguities |
| 5 | WIDELANE | Solution from wide-lane ambiguities |
| 6 | NARROWLANE | Solution from narrow-lane ambiguities |
| 7 | Reserved | |
| 8 | DOPPLER_VELOCITY | Velocity computed using instantaneous Doppler |
| 9-15 | Reserved | |
| 16 | SINGLE | Single point position |
| 17 | PSRDIFF | Pseudorange differential solution |
| 18 | WAAS | Solution calculated using corrections from an SBAS |
| 19 | PROPOGATED | Propagated by a Kalman filter without new observations |

*Continued on the following page.*

| Position Type (binary) | Position Type (ASCII) | Description |
|---|---|---|
| 20 | OMNISTAR | OmniSTAR VBS position (L1 sub-meter) [a] |
| 21-31 | Reserved | |
| 32 | L1_FLOAT | Floating L1 ambiguity solution |
| 33 | IONOFREE_FLOAT | Floating ionospheric-free ambiguity solution |
| 34 | NARROW_FLOAT | Floating narrow-lane ambiguity solution |
| 48 | L1_INT | Integer L1 ambiguity solution |
| 49 | WIDE_INT | Integer wide-lane ambiguity solution |
| 50 | NARROW_INT | Integer narrow-lane ambiguity solution |
| 51 | RTK_DIRECT_INS | RTK status where the RTK filter is directly initialized from the INS filter. [b] |
| 52 | INS | INS calculated position corrected for the antenna [b] |
| 53 | INS_PSRSP | INS pseudorange single point solution - no DGPS corrections [b] |
| 54 | INS_PSRDIFF | INS pseudorange differential solution [b] |
| 55 | INS_RTKFLOAT | INS RTK floating point ambiguities solution [b] |
| 56 | INS_RTKFIXED | INS RTK fixed ambiguities solution [b] |
| 57 | INS_OMNISTAR | INS OmniSTAR VBS position (L1 sub-meter) [ab] |
| 58 | INS_OMNISTAR_HP | INS OmniSTAR high precision solution [ab] |
| 59 | INS_OMNISTAR_XP | INS OmniSTAR extra precision solution [ab] |
| 64 | OMNISTAR_HP | OmniSTAR high precision [a] |
| 65 | OMNISTAR_XP | OmniSTAR extra precision [a] |
| 66 | CDGPS | Position solution using CDGPS corrections [a] |

a. In addition to a NovAtel receiver with L-band capability, a subscription to the OmniSTAR, or use of the free CDGPS, service is required. Contact NovAtel for details.

b. These types appear in position logs such as BESTPOS. Please refer to your *OEMV Family Firmware Reference Manual.*

**Table 12: Solution Status**

| Binary | ASCII | Description |
|---|---|---|
| 0 | SOL_COMPUTED | Solution computed |
| 1 | INSUFFICIENT_OBS | Insufficient observations |
| 2 | NO_CONVERGENCE | No convergence |
| 3 | SINGULARITY | Singularity at parameters matrix |
| 4 | COV_TRACE | Covariance trace exceeds maximum (trace > 1000 m) |
| 5 | TEST_DIST | Test distance exceeded (maximum of 3 rejections if distance > 10 km) |
| 6 | COLD_START | Not yet converged from cold start |
| 7 | V_H_LIMIT | Height or velocity limits exceeded (in accordance with COCOM export licensing restrictions) |
| 8 | VARIANCE | Variance exceeds limits |
| 9 | RESIDUALS | Residuals are too large |
| 10 | DELTA_POS | Delta position is too large |
| 11 | NEGATIVE_VAR | Negative variance |
| 12 | Reserved | |
| 13 | INTEGRITY_WARNING | Large residuals make position unreliable |
| 14 | INS_INACTIVE | INS has not started yet |
| 15 | INS_ALIGNING | INS doing its coarse alignment |
| 16 | INS_BAD | INS position is bad |
| 17 | IMU_UNPLUGGED | No IMU detected |
| 18 | PENDING | When a FIX POSITION command is entered, the receiver computes its own position and determines if the fixed position is valid [a] |
| 19 | INVALID_FIX | The fixed position, entered using the FIX POSITION command, is not valid |

a. PENDING implies there are not enough satellites being tracked to verify if the FIX POSITION entered into the receiver is valid. The receiver needs to be tracking two or more GPS satellites to perform this check. Under normal conditions you should only see PENDING for a few seconds on power up before the GPS receiver has locked onto its first few satellites. If your antenna is obstructed (or not plugged in) and you have entered a FIX POSITION command, then you may see PENDING indefinitely.

## C.2.2 BESTGPSVEL Best Available GPS Velocity Data

This log contains the best available GPS velocity information (without INS) computed by the receiver. In addition, it reports a velocity status indicator, which is useful in indicating whether or not the corresponding data is valid. The velocity measurements sometimes have a latency associated with them. The time of validity is the time tag in the log minus the latency value.

The velocity is typically computed from the average change in pseudorange over the time interval or the RTK Low Latency filter. As such, it is an average velocity based on the time difference between successive position computations and not an instantaneous velocity at the BESTGPSVEL time tag. The velocity latency to be subtracted from the time tag is normally 1/2 the time between filter updates. Under default operation, the positioning filters are updated at a rate of 2 Hz. This translates into a velocity latency of 0.25 second. The latency can be reduced by increasing the update rate of the positioning filter being used by requesting the BESTGPSVEL or BESTGPSPOS messages at a rate higher than 2 Hz. For example, a logging rate of 10 Hz would reduce the velocity latency to 0.005 seconds. For integration purposes, the velocity latency should be applied to the record time tag.

A valid solution with a latency of 0.0 indicates that the instantaneous Doppler measurement was used to calculate velocity.

**Structure:** **Message ID: 506**

**Log Type: Synch**

| Field # | Field type | Data Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | header | Log header | - | H | 0 |
| 2 | Sol Status | Solution status, see *Table 12, Solution Status* on *page 76* | Enum | 4 | H |
| 3 | Vel Type | Velocity type, see *Table 11, Position or Velocity Type* on *page 74* | Enum | 4 | H+4 |
| 4 | Latency | A measure of the latency in the velocity time tag in seconds. It should be subtracted from the time to give improved results. | Float | 4 | H+8 |
| 5 | Age | Differential age | Float | 4 | H+12 |
| 6 | Hor Spd | Horizontal speed over ground, in metres per second | Double | 8 | H+16 |
| 7 | Trk Gnd | Actual direction of motion over ground (track over ground) with respect to True North, in degrees | Double | 8 | H+24 |
| 8 | Vert Spd | Vertical speed, in metres per second, where positive values indicate increasing altitude (up) and negative values indicate decreasing altitude (down) | Double | 8 | H+32 |
| 9 | Reserved | | Float | 4 | H+40 |
| 10 | xxxx | 32-bit CRC (ASCII and Binary only) | Hex | 4 | H+44 |
| 11 | [CR][LF] | Sentence terminator (ASCII only) | - | - | - |

**Recommended Input:**

log bestgpsvela ontime 1

**ASCII Example:**

```
#BESTGPSVELA,COM1,0,62.5,FINESTEERING,1049,247755.000,00000128,f7e3,0;
SOL_COMPUTED,SINGLE,0.250,0.000,0.1744,333.002126,0.3070,6.0082*dfdc635c
```

## C.2.3 BESTLEVERARM IMU to Antenna Lever Arm

This log contains the distance between the SPAN-CPT and the GPS antenna in the IMU enclosure frame and its associated uncertainties. If the you enter the lever arm through the SETIMUTOANTOFFSET command, see *page 60*, these values are reflected in this log.

The default X (pitch), Y (roll) and Z (azimuth) directions of the inertial frame are clearly marked on the IMU, see *Figure 11* on *page 44*.

**Structure:** **Message ID: 674**

**Log Type: Asynch**

| Field | Field Type | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | Log Header | Log Header | - | H | 0 |
| 2 | X Offset | IMU Enclosure Frame (m) | Double | 8 | H |
| 3 | Y Offset | IMU Enclosure Frame (m) | Double | 8 | H+8 |
| 4 | Z Offset | IMU Enclosure Frame (m) | Double | 8 | H+16 |
| 5 | X Uncertainty | IMU Enclosure Frame (m) | Double | 8 | H+24 |
| 6 | Y Uncertainty | IMU Enclosure Frame (m) | Double | 8 | H+32 |
| 7 | Z Uncertainty | IMU Enclosure Frame (m) | Double | 8 | H+40 |
| 8 | iMapping | See *Table 8, Full Mapping Definitions* on *page 59* | Integer | 4 | H+48 |
| 9 | xxxx | 32-bit CRC | Hex | 4 | H+52 |
| 10 | [CR][LF] | Sentence Terminator (ASCII only) | - | - | - |

**Recommended Input:**

log bestleverarma onchanged

**ASCII Example:**

```
#BESTLEVERARMA,COM1,0,83.5,UNKNOWN,0,2.983,00000008,39e4,35484;
0.3934000000000000,-1.2995000000000001,0.0105500000000000,
0.0300000000000000,0.0300000000000000,0.0300000000000000,4*876c47ad
```

## C.2.4 INSATT INS Attitude

This log, and the INSATTS log, contains the most recent attitude measurements corresponding to the SPAN frame axis according to the installation instructions provided in *Section 2.2, Hardware Set-Up* starting on *page 22* and *INS Window in CDU* on *page 32* of this manual. The attitude measurements may not correspond to other definitions of the terms pitch, roll and azimuth. If your SPAN-CPT's z-axis (as marked on the enclosure) is not pointing up, the output attitude will be with respect to the SPAN computational frame, and not the frame marked on the enclosure. See the SETIMUORIENTATION command to determine what the SPAN computation frame will be, given how your IMU is mounted.

**Structure:** **Message ID: 263**

**Log Type: Synch**

| Field # | Field Type | Data Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | Log Header | Log header | - | H | 0 |
| 2 | Week | GPS Week | Ulong | 4 | H |
| 3 | Seconds into Week | Seconds from week start | Double | 8 | H+4 |
| 4 | Roll | Right handed rotation from local level around y-axis in degrees. | Double | 8 | H+12 |
| 5 | Pitch | Right handed rotation from local level around x-axis in degrees. | Double | 8 | H+20 |
| 6 | Azimuth | Left handed rotation around z-axis. Degrees clockwise from North. | Double | 8 | H+28 |
| 7 | Status | INS status, see *Table 1* on *page 34* | Enum | 4 | H+36 |
| 8 | xxxx | 32-bit CRC (ASCII, Binary and Short Binary only) | Hex | 4 | H+40 |
| 9 | [CR][LF] | Sentence terminator (ASCII only) | - | - | - |

**Recommended Input:**

log insatta ontime 1

**ASCII Example:**

```
#INSATTA,COM3,0,0.0,EXACT,1105,425385.000,00040000,0638,0;
1105,425384.996167250,4.822147742,0.035766158,123.262113519,
INSSolutionGood*3563a760
```

✉ The structure of the INSATT log is different in this firmware version (1.3) than in any earlier beta versions. This is because the order of the roll and pitch fields have been reversed.

## C.2.5 INSATTS Short INS Attitude

This is a short header version of the *INSATT* log on *Page 80*.

**Structure:** **Message ID: 319**

**Log Type: Synch**

| Field # | Field Type | Data Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | Log Header | Log header | - | H | 0 |
| 2 | Week | GPS Week | Ulong | 4 | H |
| 3 | Seconds into Week | Seconds from week start | Double | 8 | H+4 |
| 4 | Roll | Right handed rotation from local level around y-axis in degrees. | Double | 8 | H+12 |
| 5 | Pitch | Right handed rotation from local level around x-axis in degrees. | Double | 8 | H+20 |
| 6 | Azimuth | Left handed rotation around z-axis. Degrees clockwise from North. | Double | 8 | H+28 |
| 7 | Status | INS status, see *Table 1* on *page 34*. | Enum | 4 | H+36 |
| 8 | xxxx | 32-bit CRC (ASCII, Binary and Short Binary only) | Hex | 4 | H+40 |
| 9 | [CR][LF] | Sentence terminator (ASCII only) | - | - | - |

**Recommended Input:**

log insattsa ontime 1

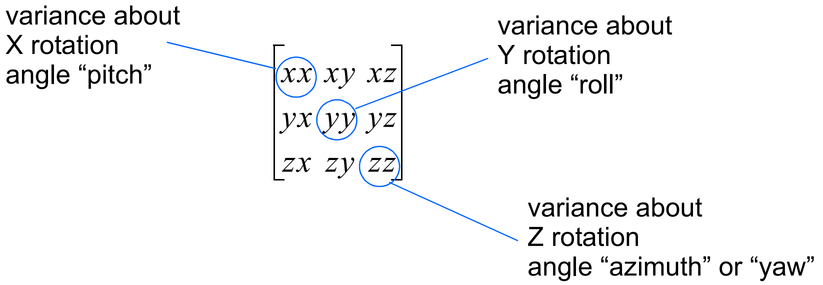
**ASCII Example:**

```
%INSATTSA,1105,425385.000;
1105,425384.996167250,4.822147742,0.035766158,123.262113519,
INSSolutionGood*3563a760
```

⌧ The structure of the INSATTS log is different in this firmware version (1.3) than in any earlier beta versions. This is because the order of the roll and pitch fields have been reversed.

## C.2.6 INSCOV INS Covariance Matrices

The position, attitude, and velocity matrices in this log each contain 9 covariance values, with respect to the SPAN frame axis, as follows:




and are displayed within the log output as:

*…,xx,xy,xz,yx,yy,yz,zx,zy,zz,…*

These values are computed once per second and are only available after alignment. See also *Section 3.3.1, System Start-Up and Alignment Techniques* starting on *page 35*.

**Structure:**

**Message ID: 264**

**Log Type: Asynch**

| Field # | Field Type | Data Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | Log Header | Log header | - | H | 0 |
| 2 | Week | GPS Week | Ulong | 4 | H |
| 3 | Seconds into Week | Seconds from week start | Double | 8 | H+4 |
| 4 | Position Covariance | Position covariance matrix in local level frame (Meters squared) | List of 9 Doubles | 72 | H+12 |
| 5 | Attitude Covariance | Attitude covariance matrix in local level frame. (Degrees squared - rotation around the given axis) | List of 9 Doubles | 72 | H+84 |
| 6 | Velocity Covariance | Velocity covariance matrix in local level frame. (Meters/second squared) | List of 9 Doubles | 72 | H+156 |
| 7 | xxxx | 32-bit CRC (ASCII, Binary and Short Binary only) | Hex | 4 | H+228 |
| 8 | [CR][LF] | Sentence terminator (ASCII only) | - | - | - |

**Recommended Input:**

log inscova onchanged

**ASCII Example:**

```
#INSCOVA,COM3,0,0.0,EXACT,1105,425385.020,00040000,c45c,0;
1105,425385.000000000,
0.0997319969301073,-0.0240959791179416,-0.0133921499963209,
-0.0240959791179416,0.1538605784734939,0.0440068023663888,
-0.0133921499963210,0.0440068023663887,0.4392033415009359,
0.0034190251365443,0.0000759398593357,-0.1362852812808768,
0.0000759398593363,0.0032413999569636,-0.0468473344270137,
-0.1362852812808786,-0.0468473344270131,117.5206493841025100,
0.0004024901765302,-0.0000194916086028,0.0000036582459112,
-0.0000194916086028,0.0004518869575566,0.0000204616202028,
0.0000036582459112,0.0000204616202028,0.0005095575483948*1fc92787
```

## C.2.7 INSCOVS Short INS Covariance Log

This is a short header version of the *INCOV* log on *Page 82*. These values are also computed once per second.

**Structure:** **Message ID: 320**

**Log Type: Asynch**

| Field # | Field Type | Data Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | Log Header | Log header | - | H | 0 |
| 2 | Week | GPS Week | Ulong | 4 | H |
| 3 | Seconds into Week | Seconds from week start | Double | 8 | H+4 |
| 4 | Position Covariance | Position covariance matrix in local level frame. (Meters squared) xx,xy,xz,yx,yy,yz,zx,zy,zz | List of 9 Doubles | 72 | H+12 |
| 5 | Attitude Covariance | Attitude covariance matrix in local level frame. (Degrees squared - rotation around the given axis) xx,xy,xz,yx,yy,yz,zx,zy,zz | List of 9 Doubles | 72 | H+84 |
| 6 | Velocity Covariance | Velocity covariance matrix in local level frame. (Meters/second squared) xx,xy,xz,yx,yy,yz,zx,zy,zz | List of 9 Doubles | 72 | H+156 |
| 7 | xxxx | 32-bit CRC (ASCII, Binary and Short Binary only) | Hex | 4 | H+228 |
| 8 | [CR][LF] | Sentence terminator (ASCII only) | - | - | - |

**Recommended Input:**

log inscovsa onchanged

**ASCII Example:**

```
%INSCOVSA,1105,425385.020;
1105,425385.000000000,
0.0997319969301073,-0.0240959791179416,-0.0133921499963209,
-0.0240959791179416,0.1538605784734939,0.0440068023663888,
-0.0133921499963210,0.0440068023663887,0.4392033415009359,
0.0034190251365443,0.0000759398593357,-0.1362852812808768,
0.0000759398593363,0.0032413999569636,-0.0468473344270137,
-0.1362852812808786,-0.0468473344270131,117.5206493841025100,
0.0004024901765302,-0.0000194916086028,0.0000036582459112,
-0.0000194916086028,0.0004518869575566,0.0000204616202028,
0.0000036582459112,0.0000204616202028,0.0005095575483948*1fc92787
```

## C.2.8 INSPOS INS Position

This log contains the most recent position measurements in WGS84 coordinates and includes an INS status indicator. The log reports the position at the IMU centre, unless you issue the SETINSOFFSET command, see *page 65*.

**Structure:** **Message ID: 265**

**Log Type:Synch**

| Field # | Field Type | Data Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | Log Header | Log header | - | H | 0 |
| 2 | Week | GPS Week | Ulong | 4 | H |
| 3 | Seconds into Week | Seconds from week start | Double | 8 | H+4 |
| 4 | Latitude | Latitude (WGS84) | Double | 8 | H+12 |
| 5 | Longitude | Longitude (WGS84) | Double | 8 | H+20 |
| 6 | Height | Ellipsoidal Height (WGS84) | Double | 8 | H+28 |
| 7 | Status | INS status, see *Table 1* on *page 34* | Enum | 4 | H+36 |
| 8 | xxxx | 32-bit CRC (ASCII, Binary and Short Binary only) | Hex | 4 | H+40 |
| 9 | [CR][LF] | Sentence terminator (ASCII only) | - | - | - |

**Recommended Input:**

log insposa ontime 1

**ASCII Example:**

```
#INSPOSA,COM3,0,0.0,EXACT,1105,425385.000,00040000,323a,0;
1105,425384.996167250,51.058410364,-114.065465722,
1067.791685696,INSSolutionGood*9bfd5a12
```

## C.2.9 INSPOSS Short INS Position

This is a short header version of the *INSPOS* log on *Page 85*.

**Structure:** **Message ID: 321**

**Log Type:Synch**

| Field # | Field Type | Data Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | Log Header | Log header | - | H | 0 |
| 2 | Week | GPS Week | Ulong | 4 | H |
| 3 | Seconds into Week | Seconds from week start | Double | 8 | H+4 |
| 4 | Latitude | Latitude (WGS84) | Double | 8 | H+12 |
| 5 | Longitude | Longitude (WGS84) | Double | 8 | H+20 |
| 6 | Height | Ellipsoidal Height (WGS84) | Double | 8 | H+28 |
| 7 | Status | INS status, see *Table 1* on *page 34* | Enum | 4 | H+36 |
| 8 | xxxx | 32-bit CRC (ASCII, Binary and Short Binary only) | Hex | 4 | H+40 |
| 9 | [CR][LF] | Sentence terminator (ASCII only) | - | - | - |

**Recommended Input:**

log inspossa ontime 1

**ASCII Example:**

```
%INSPOSSA,1105,425385.000;
1105,425384.996167250,51.058410364,-114.065465722,
1067.791685696,INSSolutionGood*9bfd5a12
```

## C.2.10 INSPOSSYNC Time Synchronised INS Position

This log contains the time synchonised INS position. It is synchronised with GPS each second.

**Structure:** **Message ID: 322**

**Log Type: Asynch**

| Field # | Field Type | Data Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | Log Header | Log header | - | H | 0 |
| 2 | Sec | Age of synchronised INS solution (s) | Double | 8 | H |
| 3 | X | ECEF X coordinate | Double | 8 | H+8 |
| 4 | Y | ECEF Y coordinate | Double | 8 | H+16 |
| 5 | Z | ECEF Z coordinate | Double | 8 | H+24 |
| 6 | Cov | ECEF covariance matrix (a 3 x 3 array of length 9).<br>Refer also to the CLOCKMODEL log in the *OEMV Family Firmware Reference Manual*. | Double[9] | 72 | H+32 |
| 7 | xxxx | 32-bit CRC (ASCII, Binary and Short Binary only) | Hex | 4 | H+104 |
| 8 | [CR][LF] | Sentence terminator (ASCII only) | - | - | - |

**Recommended Input:**

log inspossynca onchanged

**ASCII Example:**

```
#INSPOSSYNCA,COM1,0,47.5,FINESTEERING,1332,484154.042,00000000,c98c,34492;
484154.000000000,-1634523.2463,-3664620.7609,4942494.6795,
1.8091616236414247,0.0452272887760925,-0.7438098675219428,
0.0452272887760925,2.9022554471257266,-1.5254793710104819,
-0.7438098675219428,-1.5254793710104819,4.3572293495804546*9fcd6ce1
```

## C.2.11 INSPVA INS Position, Velocity and Attitude

This log allows INS position, velocity and attitude, with respect to the SPAN frame, to be collected in one log, instead of using three separate logs. See the INSATT log, on *page 80*, for an explanation of how the SPAN frame may differ from the IMU enclosure frame.

**Structure:** **Message ID: 507**

**Log Type: Synch**

| Field | Field Type | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | Log Header | Log header | - | H | 0 |
| 2 | Week | GPS Week | Ulong | 4 | H |
| 3 | Seconds | Seconds from week start | Double | 8 | H+4 |
| 4 | Latitude | Latitude (WGS84) | Double | 8 | H+12 |
| 5 | Longitude | Longitude (WGS84) | Double | 8 | H+20 |
| 6 | Height | Ellipsoidal Height (WGS84) | Double | 8 | H+28 |
| 7 | North Velocity | Velocity in a northerly direction (a -ve value implies a southerly direction) | Double | 8 | H+36 |
| 8 | East Velocity | Velocity in an easterly direction (a -ve value implies a westerly direction) | Double | 8 | H+44 |
| 9 | Up Velocity | Velocity in an up direction | Double | 8 | H+52 |
| 10 | Roll | Right handed rotation from local level around y-axis in degrees | Double | 8 | H+60 |
| 11 | Pitch | Right handed rotation from local level around x-axis in degrees | Double | 8 | H+68 |
| 12 | Azimuth | Left handed rotation around z-axis Degrees clockwise from North | Double | 8 | H+76 |
| 13 | Status | INS Status, see *Table 1* on *page 34* | Enum | 4 | H+84 |
| 14 | xxxx | 32-bit CRC | Hex | 4 | H+88 |
| 15 | [CR][LF] | Sentence Terminator (ASCII only) | - | - | - |

**Recommended Input:**

log inspvaa ontime 1

**ASCII Example:**

```
#INSPVAA,COM1,0,31.0,FINESTEERING,1264,144088.000,00040000,5615,1541;
1264,144088.002284950,51.116827527,-114.037738908,401.191547167,
354.846489850,108.429407241,-10.837482850,1.116219952,-3.476059035,
7.372686190,INS_ALIGNMENT_COMPLETE*af719fd9
```

## C.2.12 INSPVAS Short INS Position, Velocity and Attitude

This is a short header version of the INSPVA log on *page 88*.

**Structure:** **Message ID: 508**

**Log Type: Synch**

| Field | Field Type | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | Log Header | Log header | - | H | 0 |
| 2 | Week | GPS Week | Ulong | 4 | H |
| 3 | Seconds | Seconds from week start | Double | 8 | H+4 |
| 4 | Latitude | Latitude (WGS84) | Double | 8 | H+12 |
| 5 | Longitude | Longitude (WGS84) | Double | 8 | H+20 |
| 6 | Height | Ellipsoidal Height (WGS84) | Double | 8 | H+28 |
| 7 | North Velocity | Velocity in a northerly direction (a -ve value implies a southerly direction) | Double | 8 | H+36 |
| 8 | East Velocity | Velocity in an easterly direction (a -ve value implies a westerly direction) | Double | 8 | H+44 |
| 9 | Up Velocity | Velocity in an up direction | Double | 8 | H+52 |
| 10 | Roll | Right handed rotation from local level around y-axis in degrees | Double | 8 | H+60 |
| 11 | Pitch | Right handed rotation from local level around x-axis in degrees | Double | 8 | H+68 |
| 12 | Azimuth | Left handed rotation around z-axis Degrees clockwise from North | Double | 8 | H+76 |
| 13 | Status | INS Status, see *Table 1* on *page 34* | Enum | 4 | H+84 |
| 14 | xxxx | 32-bit CRC | Hex | 4 | H+88 |
| 15 | [CR][LF] | Sentence Terminator (ASCII only) | - | - | - |

**Recommended Input:**

log inspvasa ontime 1

**ASCII Example:**

```
%INSPVASA,1264,144059.000;
1264,144059.002135700,51.116680071,-114.037929194,515.286704183,
277.896368884,84.915188605,-8.488207941,0.759619515,-2.892414901,
6.179554750,INS_ALIGNMENT_COMPLETE*855d6f76
```

## C.2.13 INSSPD INS Speed

This log contains the most recent speed measurements in the horizontal and vertical directions, and includes an INS status indicator.

**Structure:** **Message ID: 266**

**Log Type: Synch**

| Field # | Field Type | Data Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | Log Header | Log header | - | H | 0 |
| 2 | Week | GPS Week | Ulong | 4 | H |
| 3 | Seconds into Week | Seconds from week start | Double | 8 | H+4 |
| 4 | Trk gnd | Actual direction of motion over ground (track over ground) with respect to True North, in degrees | Double | 8 | H+12 |
| 5 | Horizontal Speed | Magnitude of horizontal speed in m/s where a positive value indicates you are moving forward and a negative value indicates you are reversing. | Double | 8 | H+20 |
| 6 | Vertical Speed | Magnitude of vertical speed in m/s where a positive value indicates speed upward and a negative value indicates speed downward. | Double | 8 | H+28 |
| 7 | Status | INS status, see *Table 1* on *page 34* | Enum | 4 | H+36 |
| 8 | xxxx | 32-bit CRC (ASCII, Binary and Short Binary only) | Hex | 4 | H+40 |
| 9 | [CR][LF] | Sentence terminator (ASCII only) | - | - | - |

**Recommended Input:**

log insspda ontime 1

**ASCII Example:**

```
#INSSPDA,COM3,0,0.0,EXACT,1105,425385.000,00040000,efce,0;
1105,425384.996167250,223.766800423,0.019769837,
-0.024795257,INSSolutionGood*15b864f4
```

## C.2.14 INSSPDS Short INS Speed

This is a short header version of the *INSSPD* log on *Page 90*.

**Structure:** **Message ID: 323**

**Log Type: Synch**

| Field # | Field Type | Data Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | Log Header | Log header | - | H | 0 |
| 2 | Week | GPS Week | Ulong | 4 | H |
| 3 | Seconds into Week | Seconds from week start | Double | 8 | H+4 |
| 4 | Trk gnd | Track over ground | Double | 8 | H+12 |
| 5 | Horizontal Speed | Horizontal speed in m/s | Double | 8 | H+20 |
| 6 | Vertical Speed | Vertical speed in m/s | Double | 8 | H+28 |
| 7 | Status | INS status, see *Table 1* on *page 34* | Enum | 4 | H+36 |
| 8 | xxxx | 32-bit CRC (ASCII, Binary and Short Binary only) | Hex | 4 | H+40 |
| 9 | [CR][LF] | Sentence terminator (ASCII only) | - | - | - |

**Recommended Input:**

log insspdsa ontime 1

**ASCII Example:**

```
%INSSPDSA,1105,425385.000;
1105,425384.996167250,223.766800423,0.019769837,
-0.024795257,INSSolutionGood*15b864f4
```

## C.2.15 INSUPDATE INS Update

This log contains the most recent INS update information. It gives you information about what updates were performed in the INS filter at the last update epoch and a wheel sensor status indicator.

**Structure:** **Message ID: 757**

**Log Type: Asynch**

| Field # | Field Type | Data Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | Log Header | Log header | - | H | 0 |
| 2 | Solution Type | Type of GPS solution used for the last update, see *Table 11* on *page 74* | Enum | 4 | H |
| 3 | Reserved | | Integer | 4 | H+4 |
| 4 | #Phase | Number of raw phase observations used in the last INS filter update | Integer | 4 | H+8 |
| 5 | Reserved | | Integer | 4 | H+12 |
| 6 | Zupt Flag | A zero velocity update was performed during the last INS filter update:<br>0 = False<br>1 = True | Boolean | 4 | H+16 |
| 7 | Wheel Status | Wheel status, see *Table 13* below | Ulong | 4 | H+18 |
| 8 | Reserved | | Ulong | 4 | H+22 |
| 9 | xxxx | 32-bit CRC (ASCII, Binary and Short Binary only) | Hex | 4 | H+26 |
| 10 | [CR][LF] | Sentence terminator (ASCII only) | - | - | - |

**Recommended Input:**

log insupdate onchanged

**ASCII Example:**

```
#INSUPDATEA,UNKNOWN,0,32.5,FINESTEERING,1379,339642.042,00040040,3670,2431;
SINGLE,0,6,0,FALSE,WHEEL_SENSOR_UNSYNCED,0*fb5df08b
```

**Table 13: Wheel Status**

| Binary | ASCII |
|---|---|
| 0 | WHEEL_SENSOR_INACTIVE |
| 1 | WHEEL_SENSOR_ACTIVE |
| 2 | WHEEL_SENSOR_USED |
| 3 | WHEEL_SENSOR_UNSYNCED |
| 4 | WHEEL_SENSOR_BAD_MISC |
| 5 | WHEEL_SENSOR_HIGH_ROTATION |

## C.2.16 INSVEL INS Velocity

This log contains the most recent North, East, and Up velocity vector values, with respect to the local level frame, and also includes an INS status indicator.

**Structure:** **Message ID: 267**

**Log Type:Synch**

| Field # | Field Type | Data Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | Log Header | Log header | - | H | 0 |
| 2 | Week | GPS Week | Ulong | 4 | H |
| 3 | Seconds into Week | Seconds from week start | Double | 8 | H+4 |
| 4 | North Velocity | Velocity North in m/s | Double | 8 | H+12 |
| 5 | East Velocity | Velocity East in m/s | Double | 8 | H+20 |
| 6 | Up Velocity | Velocity Up in m/s | Double | 8 | H+28 |
| 7 | Status | INS status, see *Table 1* on *page 34* | Enum | 4 | H+36 |
| 8 | xxxx | 32-bit CRC (ASCII, Binary and Short Binary only) | Hex | 4 | H+40 |
| 9 | [CR][LF] | Sentence terminator (ASCII only) | - | - | - |

**Recommended Input:**

log insvela ontime 1

**ASCII Example:**

```
#INSVELA,COM3,0,0.0,EXACT,1105,425385.000,00040000,7d4a,0;
1105,425384.996167250,-0.014277009,-0.013675287,
-0.024795257,INSSolutionGood*2f3fe011
```

## C.2.17 INSVELS Short INS Velocity

This is a short header version of the *INSVEL* log on *Page 93*.

**Structure:** **Message ID: 324**

**Log Type:Synch**

| Field # | Field Type | Data Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | Log Header | Log header | - | H | 0 |
| 2 | Week | GPS Week | Ulong | 4 | H |
| 3 | Seconds into Week | Seconds from week start | Double | 8 | H+4 |
| 4 | North Velocity | Velocity North m/s | Double | 8 | H+12 |
| 5 | East Velocity | Velocity East m/s | Double | 8 | H+20 |
| 6 | Up Velocity | Velocity Up m/s | Double | 8 | H+28 |
| 7 | Status | INS status, see *Table 1* on *page 34* | Enum | 4 | H+36 |
| 8 | xxxx | 32-bit CRC (ASCII, Binary and Short Binary only) | Hex | 4 | H+40 |
| 9 | [CR][LF] | Sentence terminator (ASCII only) | - | - | - |

**Recommended Input:**

log insvelsa ontime 1

**ASCII Example:**

```
%INSVELSA,1105,425385.000;
1105,425384.996167250,-0.014277009,-0.013675287,
-0.024795257,INSSolutionGood*2f3fe011
```

## C.2.18 MARK1PVA Position, Velocity and Attitude at Mark1

This log outputs position, velocity and attitude information of the system, with respect to the SPAN frame, when an event was received on the Mark 1 input.

**Structure:** **Message ID: 1067**

**Log Type: Synch**

| Field | Field Type | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | Log Header | Log header | - | H | 0 |
| 2 | Week | GPS Week at Mark1 request | Ulong | 4 | H |
| 3 | Seconds | Seconds from week at Mark1 | Double | 8 | H+4 |
| 4 | Latitude | Latitude (WGS84) at Mark1 | Double | 8 | H+12 |
| 5 | Longitude | Longitude (WGS84) at Mark1 | Double | 8 | H+20 |
| 6 | Height | Height (WGS84) at Mark1 | Double | 8 | H+28 |
| 7 | North Velocity | Velocity in a northerly direction (a -ve value implies a southerly direction) at Mark1 | Double | 8 | H+36 |
| 8 | East Velocity | Velocity in an easterly direction (a -ve value implies a westerly direction) at Mark1 | Double | 8 | H+44 |
| 9 | Up Velocity | Velocity in an up direction at Mark1 | Double | 8 | H+52 |
| 10 | Roll | Right handed rotation from local level around y-axis in degrees at Mark1 | Double | 8 | H+60 |
| 11 | Pitch | Right handed rotation from local level around x-axis in degrees at Mark1 | Double | 8 | H+68 |
| 12 | Azimuth | Left handed rotation around z-axis Degrees clockwise from North at Mark1 | Double | 8 | H+76 |
| 13 | Status | INS Status, see *Table 1* on *page 34* at Mark1 | Enum | 4 | H+84 |
| 14 | xxxx | 32-bit CRC | Hex | 4 | H+88 |
| 15 | [CR][LF] | Sentence Terminator (ASCII only) | - | - | - |

**Recommended Input:**

log mark1pva onnew

**Abbreviated ASCII Example:**

```
MARK1PVA USB1 0 51.5 EXACT 1481 251850.001 00040000 46f4 3388

1481 251850.001000000 51.116573435 -114.037237211 1040.805671970 0.000257666
-0.003030102 -0.000089758 3.082229474 -1.019023628 89.253955744
INS_SOLUTION_GOOD
```

## C.2.19 PASHR NMEA, fix and position data

The PASHR log outputs these messages with contents without waiting for a valid almanac. Instead, it uses a UTC time, calculated with default parameters. In this case, the UTC time status is set to WARNING since it may not be 100% accurate. When a valid almanac is available, the receiver uses the real parameters and sets the UTC time to VALID. For more information about NMEA, refer to the *OEMV Firmware Reference Manual* found on our Web site. The PASHR log contains only INS derived attitude information and is only filled when an inertial solution is available.

**Structure:** **Message ID: 1177**

**Log TypeSynch**

| Field | Structure | Feild Description | Symbol | Example |
|---|---|---|---|---|
| 1 | $PASHR | Log Header | --- | $PASHR |
| 2 | Time | UTC Time | hhmmss.ss | 195124.00 |
| 3 | Heading | Heading value in decimal degrees | HHH.HH | 305.30 |
| 4 | True Heading | T displayed if heading is relative to true north. | T | T |
| 5 | Roll | Roll in decimal degrees. The +/- sign will always be displayed. | RRR.RR | +0.05 |
| 6 | Pitch | Pitch in decimal degrees. The +/- sign will always be displayed. | PPP.PP | -0.13 |
| 7 | Reserved | ------ | ---- | ---- |
| 8 | Roll Accuracy | Roll standard deviation in decimal degrees. | rr.rrr | 0.180 |
| 9 | Pitch Accuracy | Pitch standard deviation in decimal degrees. | pp.ppp | 0.185 |
| 10 | Heading Accuracy | Heading standard deviation in decimal degrees. | hh.hhh | 4.986 |
| 11 | GPS Update Quality Flag | 0 = No position<br>1 = All non-RTK fixed integer positions<br>2 = RTK fixed integer position | 1 | 1 |
| 12 | Checksum | Checksum | *XX | *2B |
| 13 | [CR][LF] | Sentence terminator | | [CR][LF] |

**Recommended Input:**

log pashr

**Example:**

```
$PASHR,,,,,,,,,,0*68 (empty)

$PASHR,195124.00,305.30,T,+0.05,-0.13,,0.180,0.185,4.986,1*2B
```

## C.2.20 RAWIMUS Short Raw IMU Data

This log contains an IMU status indicator and the measurements from the accelerometers and gyros with respect to the IMU enclosure frame. This log contains the short header version to reduce the amount of data.

**Structure:** **Message ID: 325**

**Log Type: Asynch**

| Field # | Field Type | Data Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | Log Header | Log header | - | H | 0 |
| 2 | Week | GPS Week | Ulong | 4 | H |
| 3 | Seconds into Week | Seconds from week start | Double | 8 | H+4 |
| 4 | IMU Status | The status of the IMU. This field is given in a fixed length (n) array of bytes in binary but in ASCII or Abbreviated ASCII is converted into 2 character hexadecimal pairs. [a] For more information, *Table 14, SPAN-CPT Status* on *page 98*. | Long | 4 | H+12 |
| 5 | Z Accel Output | Change in velocity count along z axis [a] | Long | 4 | H+16 |
| 6 | - (Y Accel Output) | - (Change in velocity count along y axis) [a, b] | Long | 4 | H+20 |
| 7 | X Accel Output | Change in velocity count along x axis [a] | Long | 4 | H+24 |
| 8 | Z Gyro Output | Change in angle count around z axis [c] Right-handed | Long | 4 | H+28 |
| 9 | - (Y Gyro Output) | - (Change in angle count around y axis) [b, c] Right-handed | Long | 4 | H+32 |
| 10 | X Gyro Output | Change in angle count around x axis [c] Right-handed | Long | 4 | H+36 |
| 11 | xxxx | 32-bit CRC (ASCII, Binary and Short Binary only) | Hex | 4 | H+40 |
| 12 | [CR][LF] | Sentence terminator (ASCII only) | - | - | - |

a. The change in velocity (acceleration) scale factor for each IMU type can be found in *Table 16* on *page 99*. Multiply the scale factor in *Table 16,* by the count in this field, for the velocity increments in m/s.

b. A negative value implies that the output is along the positive Y-axis marked on the IMU. A positive value implies that the change is in the direction opposite to that of the Y-axis marked on the IMU.

c. The change in angle (gyro) scale factor can be found in *Table 16* on *page 99*. Multiply the appropriate scale factor in *Table 16,* by the count in this field, for the angle increments in radians.

**Table 14: SPAN-CPT Status**

| Nibble # | Bit # | Mask | Description | Range Value |
|---|---|---|---|---|
| N0 | 0 | 0x00000001 | Gyro X Status | 1 = Valid, 0 = Invalid |
| | 1 | 0x00000002 | Gyro Y Status | 1 = Valid, 0 = Invalid |
| | 2 | 0x00000004 | Gyro Z Status | 1 = Valid, 0 = Invalid |
| | 3 | 0x00000008 | Unused | Set to 0 |
| N1 | 4 | 0x00000010 | Accelerometer X Status | 1 = Valid, 0 = Invalid |
| | 5 | 0x00000020 | Accelerometer Y Status | 1 = Valid, 0 = Invalid |
| | 6 | 0x00000040 | Accelerometer Z Status | 1 = Valid, 0 = Invalid |
| | 7 | 0x00000080 | Unused | Set to 0 |
| N2 | 8 | 0x00000100 | **Unused** | |
| | 9 | 0x00000200 | | |
| | 10 | 0x00000400 | | |
| | 11 | 0x00000800 | | |
| N3 | 12 | 0x00001000 | | |
| | 13 | 0x00002000 | | |
| | 14 | 0x00004000 | | |
| | 15 | 0x00008000 | | |
| N4 | 16 | 0x00010000 | | |
| | 17 | 0x00020000 | | |
| | 18 | 0x00040000 | | |
| | 19 | 0x00080000 | | |
| N5 | 20 | 0x00100000 | | |
| | 21 | 0x00200000 | | |
| | 22 | 0x00400000 | | |
| | 23 | 0x00800000 | | |
| N6 | 24 | 0x01000000 | | |
| | 25 | 0x02000000 | | |
| | 26 | 0x04000000 | | |
| | 27 | 0x08000000 | | |
| N7 | 28 | 0x10000000 | | |
| | 29 | 0x20000000 | | |
| | 30 | 0x40000000 | | |
| | 31 | 0x80000000 | | |

**Recommended Input:**

log rawimusa onnew

**ASCII Example:**

```
%RAWIMUSA,1105,425384.180;
1105,425384.156166800,00000077,43088060,430312,-3033352,
-132863,186983,823*5aa97065
```

*Table 14* shows how to change the bolded field, IMU Status, in the SPAN-CPT example above into it's binary equivalent, and then how to read *Table 15: SPAN-CPT Status Example*.

**Table 15: SPAN-CPT Status Example**

| Nibble# | N7 | | | | N6 | | | | N5 | | | | N4 | | | | N3 | | | | N2 | | | | N1 | | | | N0 | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0x | 0 | | | | 0 | | | | 0 | | | | 0 | | | | 0 | | | | 0 | | | | 7 | | | | 7 | | | |
| Bit # | 31 | 30 | 29 | 28 | 27 | 26 | 25 | 24 | 23 | 22 | 21 | 20 | 19 | 18 | 17 | 16 | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| Binary | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 |
| Data | Unused | | | | | | | | Unused | | | | | | | | Unused | | | | | | | | | Accelerometer Z Status | Accelerometer Y Status | Accelerometer X Status | | Gyro Z Status | Gyro Y Status | Gyro X Status |
| Value | | | | | | | | | | | | | | | | | | | | | | | | | Unused | Valid | Valid | Valid | Unused | Valid | Valid | Valid |

**Table 16: Raw SPAN-CPT Scale Factors**

| | |
|---|---|
| **Gyroscope Scale Factor** | $\frac{0.1}{(3600.0x256.0)}$ rad/LSB |
| **Acceleration Scale Factor** | $0.05/2^{15}$ m/s/LSB |

## C.2.21 TIMEDWHEELDATA Timed Wheel Data

This log contains wheel sensor data. The time stamp in the header is the time of validity for the wheel data, not the time the TIMEDWHEELDATA log was output.

See also *SPAN-CPT Wheel Sensor* on *page 38*.

**Structure:** **Message ID: 622**

**Log Type: Asynch**

| Field # | Field Type | Data Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | Log Header | Log header | - | H | 0 |
| 2 | Ticks Per Rev | Number of ticks per revolution | Ushort | 2 | H |
| 3 | Wheel Vel | Wheel velocity in counts/s | Ushort | 2 | H+2 |
| 4 | fWheel Vel | Float wheel velocity in counts/s | Float | 4 | H+4 |
| 5 | Reserved | | Ulong | 4 | H+8 |
| 6 | | | Ulong | 4 | H+12 |
| 7 | Ticks Per Second | Cumulative number of ticks per second | long | 4 | H+16 |
| 8 | xxxx | 32-bit CRC (ASCII, Binary and Short Binary only) | Hex | 4 | H+20 |
| 9 | [CR][LF] | Sentance terminator (ASCII only) | - | - | |

**Recommended Input:**

log timedwheeldataa innew

**ASCII Example:**

```
%TIMEDWHEELDATAA, 1393, 411345.001,0,215.814910889,0,0,1942255*3b5fa236
```

## C.2.22 VEHICLEBODYROTATION Vehicle to SPAN frame Rotation

The VEHICLEBODYROTATION log reports the angular offset from the vehicle frame to the SPAN frame. The SPAN frame is defined by the transformed IMU axis with Z pointing up, see the SETIMUORIENTATION on *page 57*.

The VEHICLEBODYROTATION command, see *page 68*, sets the initial estimates for the angular offset. The uncertainty values are optional.

---

🖂 If your SPAN-CPT is mounted with the Z-axis (as marked on the IMU enclosure) pointing up, the IMU enclosure frame is the same as the SPAN frame.

---

**Structure:** **Message ID: 642**

**Log Type: Asynch**

**Recommended Input:**

log vehiclebodyrotationa onchanged

**ASCII Example:**

```
#VEHICLEBODYROTATIONA,COM1,0,36.5,FINESTEERING,1264,144170.094,
00000000,bcf2,1541;1.5869999997474209,2.6639999995760122,77.649999876392343,
2.0000000000000000, 2.0000000000000000, 5.0000000000000000*25f886cc
```

## C.2.23 WHEELSIZE Wheel Size

This log contains wheel sensor information.

The inertial Kalman filter models the size of the wheel to compensate for changes in wheel circumference due to hardware or environmental changes. The default wheel size is 1.96 m. A scale factor to this default size is modeled in the filter and this log contains the current estimate of the wheel size.

**Structure:** **Message ID: 646**

**Log Type: Asynch**

| Field # | Field Type | Data Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | Log Header | Log header | - | H | 0 |
| 2 | Scale | Wheel sensor scale factor | Double | 8 | H |
| 3 | Circum | Wheel circumference (m) | Double | 8 | H+8 |
| 4 | Var | Variance of circumference ($m^2$) | Double | 8 | H+16 |
| 5 | xxxx | 32-bit CRC (ASCII, Binary and Short Binary only) | Hex | 4 | H+24 |
| 6 | [CR][LF] | Sentence terminator (ASCII only) | - | - | - |

**Recommended Input:**

log wheelsizea onnew

**ASCII Example:**

```
#WHEELSIZEA,COM3,0,44.0,EXACT,0,0.000,00000000,85f8,33738;
1.025108123,2.009211922,0.000453791*157fd50b
```

# Appendix D Command Prompt Interface

When the SPAN system turns on, no activity information is transmitted from the serial ports except for the port prompt. A terminal connected to the receiver display a messages on its monitor. For example:

**Com1>** *if connected to COM1 port*

The COM port can be COM1, COM2, COM3, USB1, USB2, USB3, or AUX. Commands are typed at the interfacing terminal's keyboard, and sent after pressing the terminal's <↵> or <Enter> key.

---

☒ Most valid commands do produce a visible response on the screen. The indication that they have been accepted is a return of the port prompt from the receiver.

---

***Example:***

An example of no echo response to an input command is the SETIMUTOANTOFFSET command. It can be entered as follows:

```
COM2>setimutoantoffset 0.1 0.1 0.1[Return]
COM2>
```

The above example illustrates command input to the receiver COM2 serial port, which sets the antenna to IMU offset. However, your only confirmation that the command was actually accepted is the return of the **COM2>** prompt.

If a command is incorrectly entered, the receiver responds with "Invalid Command Name" (or a more detailed error message) followed by the port prompt.

## D.1 DOS

One way to initiate multiple commands and logging from the receiver is to create DOS command files relating to specific functions. This minimizes the time required to set up duplicate test situations. Any convenient text editor can be used to create command text files.

***Example:***

For this example, consider a situation where a laptop computer's appropriately configured COM1 serial port is connected to the receiver's COM1 serial port, and where a rover terminal is connected to the receiver's COM2 serial port. If you wish to monitor the SPAN system activity, the following command file could be used to do this.

1. Open a text editor on the PC and type in the following command sequences:

```
log com2 satvisa ontime 15
log com2 trackstata ontime 15
log com2 rxstatusa ontime 60 5
log com2 bestposa ontime 15
log com2 psrdopa ontime 15
```

2. Save this with a convenient file name (e.g. C:\GPS\BOOT1.TXT) and exit the text editor.
3. Use the DOS *copy* command to direct the contents of the BOOT1.TXT file to the PC's COM1 serial port:

```
C:\GPS>copy boot1.txt com1
1 files(s) copied
C:\GPS>
```

4. The SPAN system is now initialized with the contents of the BOOT1.TXT command file, and logging is directed from the receiver's COM2 serial port to the rover terminal.

## D.2 WINDOWS

As any text editor or communications program can be used for these purposes, the use of Windows 98 is described only as an illustration. The following example shows how Windows 98 accessory programs *Notepad* and *HyperTerminal* can be used to create a hypothetical waypoint navigation file on a laptop computer, and send it to the receiver. It is assumed that the laptop computer's COM1 serial port is connected to the receiver's COM1 serial port, and that a rover terminal is connected to the receiver's COM2 serial port.

***Example:***

1. Open *Notepad* and type in the following command text:

```
setnav 51.111 -114.039 51.555 -114.666 0 start stop
magvar -21
log com1 bestposa ontime 15
log com1 psrvela ontime 15
log com1 navigatea ontime 15
log com2 gprmb ontime 15 5
log com2 gpvtg ontime 15 5
log com2 rxconfiga ontime 60
```

2. Save this with a convenient file name (e.g. C:\GPS\BOOTNAV1.TXT) and exit *Notepad*.
3. Ensure that the *HyperTerminal* settings are correctly set up to agree with the receiver communications protocol; these settings can be saved (e.g. C:\GPS\OEMSETUP.HT) for use in future sessions. You may wish to use XON / XOFF handshaking to prevent loss of data.
4. Select Transfer | Send Text File to locate the file that is to be sent to the receiver. Once you double-click on the file or select Open, *HyperTerminal* sends the file to the receiver.

The above example initializes the SPAN system with origin and destination waypoint coordinates and sets the magnetic variation correction to -21 degrees. The BESTPOSA, PSRVELA, and NAVIGATEA logs have been set to output from the receiver's COM1 serial port at intervals of once every 15 seconds, whereas the GPRMB and GPVTG NMEA logs have been set to be logged out of the receiver's COM2 serial port at intervals of 15 seconds and offset by five seconds. The RXCONFIGA log has been set to output every 60 seconds from its COM2 serial port.

# Appendix E Replacement Parts

The following are a list of the replacement parts available. Should you require assistance, or need to order additional components, please contact your local NovAtel dealer or Customer Service.

## E.1 SPAN-CPT System

| Part Description | NovAtel Part |
|---|---|
| KVH Enclosure | 80023524 |
| KVH Standard Unterminated Cable | 60723107 |
| KVH Development Terminated Cable | 60723108 |
| SPAN-CPT Quickstart Guide | GM-14915081 |
| OEMV, **CDU** and *Convert* disk (refer to *Page 29* of this manual and to the *OEMV Family Installation and Operation User Manual*) | 01017827 |
| SPAN-CPT User Manual | OM-20000122 |
| OEMV Family Installation and Operation User Manual | OM-20000093 |
| OEMV Family Firmware Reference Manual | OM-20000094 |

## E.2 Accessories and Options

| | Part Description | NovAtel Part |
|---|---|---|
| Optional NovAtel GPSAntennas: | Model 532 (for aerodynamic applications) | GPS-532 |
| | Model 702 (for high-accuracy applications) | GPS-702 |
| | Model 702L (for L-band applications) | GPS-702L |
| | Model 533 (for high-performance base station applications) | GPS-533 |
| Optional RF Antenna Cable: | 5 meters | C006 |
| | 15 meters | C016 |

# Appendix F Frequently Asked Questions

## 1 Why don't I hear any sound from my SPAN-CPT?

a. The SPAN-CPT does not make noise. Check that the multi-purpose I/O cable is connected properly.
b. Check the input power supply. A minimum of 12V should be supplied to the system for stable SPAN-CPT performance. The supply should also be able to output at least 12W over the entire operating temperature range.

## 2 Why don't I have any INS logs?

On start-up, the INS logs are not available until the system has solved for time. This requires that an antenna is attached, and satellites are visible, to the system. You can verify that time is solved by checking the time status in the header of any standard header SPAN log such as BESTPOS. When the time status reaches FINESTEERING, the inertial filter starts and INS messages are available.

## 3 How can I access the inertial solution?

The INS/GPS solution is available from a number of specific logs dedicated to the inertial filter. The INSPOS, INSPVA, INSVEL, INSSPD, and INSATT logs are the most commonly used logs for extracting the INS solution. These logs can be logged at any rate up to the rate of the IMU data (100 Hz). Further details on these logs are available in Appendix C, INS Logs starting on Page 65.

## 4 Can I still access the GPS-only solution while running SPAN?

The GPS only solution used when running the OEMV receiver without the IMU is still available when running SPAN. Logs such as PSRPOS, RTKPOS and OMNIPOS are still available. Any non-INS logs should be logged at a maximum rate of 5 Hz when running SPAN. Only INS-specific logs documented in Appendix C, INS Logs starting on Page 65 should be logged at rates higher than 5 Hz when running SPAN.

## 5 What will happen to the INS solution when I lose GPS satellite visibility?

When GPS tracking is interrupted, the INS/GPS solution bridges through the gaps with what is referred to as free-inertial navigation. The IMU measurements are used to propagate the solution. Errors in the IMU measurements accumulate over time to degrade the solution accuracy. For example, after one minute of GPS outage, the horizontal position accuracy is approximately 8.4 m. The SPAN solution continues to be computed for as long as the GPS outage lasts, but the solution uncertainty increases with time. This uncertainty can be monitored using the INSCOV log, see *page 82*.

## 6 All the accels measurements in my RAWIMUS logs are zero and the IMU status shows one or all accels are failing. What is wrong?

Ensure a monotonic power supply on power up of your SPAN-CPT unit. See *Connect Power* on *page 24* for more information.

# Index

Recyclable

Printed in Canada on recycled paper