# OEM6 to OEM7
# Integration Guide

# OEM6 to OEM7 Integration Guide

**Publication Number:** OM-20000164

**Revision Level:** v9D

**Revision Date:** August 2020

**Firmware Version**: 7.07.03 / OM7MR0703RN0000

## Proprietary Notice

# Table of Contents

## Chapter 4 Receiver Technical Specifications

# Figures

# Tables

# Customer Support

## NovAtel Knowledge Base

If you have a technical issue, visit the NovAtel Support page at novatel.com/support. Through the *Support* page, you can contact Customer Support, find papers and tutorials or download current manuals and the latest firmware.

## Before Contacting Customer Support

Before contacting NovAtel Customer Support about a software problem, perform the following steps:

If logging data over an RS-232 serial cable, ensure that the configured baud rate can support the data bandwidth (see **SERIALCONFIG** command). NovAtel recommends a minimum suggested baud rate of 230400 bps.

1. Log the following data to a file on your computer for 15 minutes:

```
LOG RXSTATUSB onchanged
LOG ALMANACB onchanged
LOG RAWEPHEMB onchanged
LOG GLORAWEPHEMB onchanged
LOG TRACKSTATB ontime 1
LOG SATVIS2B ontime 60
LOG BESTPOSB ontime 1
LOG RANGEB ontime 1
LOG RXCONFIGA once
LOG ITDETECTSTATUSB onchanged
LOG VERSIONA once
LOG PORTSTATSB ontime 10
```

   For SPAN systems, add the following logs to the above list in the file created on your computer:

```
LOG RAWIMUSXB onnew
LOG INSUPDATESTATUSB onnew
LOG INSPVAXB ontime 1
LOG INSCONFIGA onchanged
```

   For issues with tracking L-Band or TerraStar (PPP) convergence, add the following logs to the above list in the file created on your computer:

```
LOG IONUTCB onchanged
LOG GLOCLOCKB onchanged
LOG PPPPOSB ontime 1
LOG PPPSATSB ontime 1
LOG LBANDTRACKSTATB ontime 1
LOG TERRASTARINFOA onchanged
LOG TERRASTARSTATUSA onchanged
LOG LBANDBEAMTABLEA onchanged
```

2. Send the data file to NovAtel Customer Support: support.novatel@hexagon.com
3. You can also issue a **FRESET** command to the receiver to clear any unknown settings.

The **FRESET** command will erase all user settings. You should know your configuration (by requesting the RXCONFIGA log) and be able to reconfigure the receiver before you send the **FRESET** command.

If you are having a hardware problem, send a list of the troubleshooting steps taken and the results.

## Contact Information

Log a support request with NovAtel Customer Support using one of the following methods:

**Log a Case and Search Knowledge:**

Website: novatel.com/support

**Log a Case, Search Knowledge and View Your Case History:** (login access required)

Web Portal: https://novatelsupport.force.com/community/login

**E-mail:**

support.novatel@hexagon.com

**Telephone:**

U.S. and Canada: 1-800-NOVATEL (1-800-668-2835)

International: +1-403-295-4900

# Chapter 1 OEM6 to OEM7 Integration

When upgrading from an OEM6 receiver to an OEM7 receiver, there are several hardware and software differences that must be accounted for in the system design.

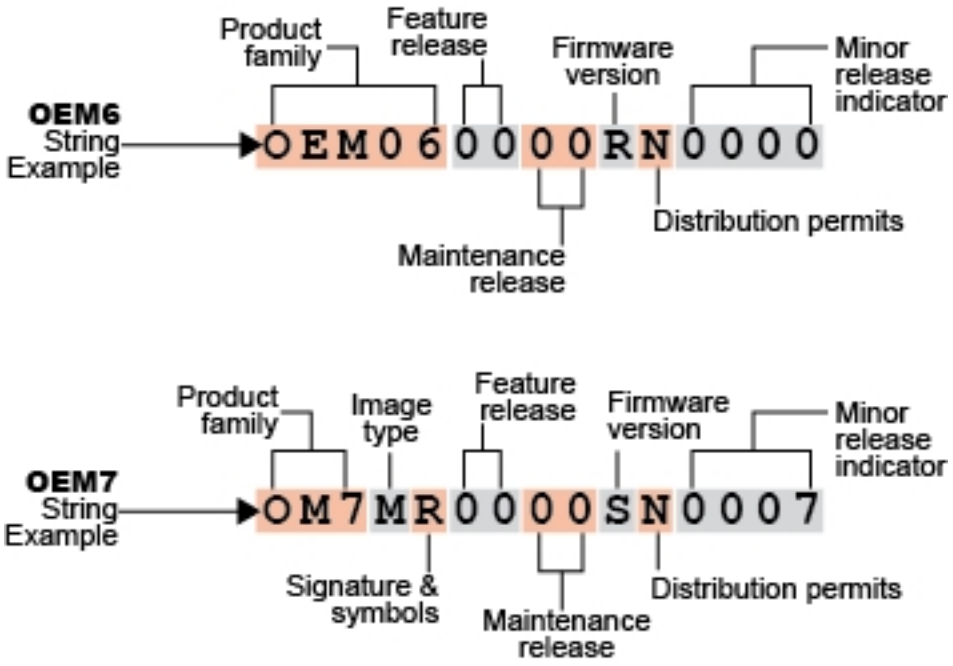
These differences are described in the following chapters:

- *New for OEM7 – Firmware* on page 14
- *OEM6 to OEM7 – Hardware* on page 151
- *Receiver Technical Specifications* on page 174

# Chapter 2 New for OEM7 – Firmware

This chapter describes the new commands, logs, features and functionality available on OEM7 receivers.

## 2.1 Firmware Version Naming




For information about the fields in the firmware version, refer to the **VERSION** log.

## 2.2 OEM6 to OEM7 – Commands

The following sections describe the command changes between OEM6 and OEM7.

### *2.2.1 Deleted Commands*

The following table lists the OEM6 commands that have been deleted or replaced in OEM7.

**Table 1: Deleted and Replaced Commands in OEM7**

| OEM6 | OEM7 |
|---|---|
| APPLYVEHICLEBODYROTATION | Deleted. |
| ASSIGNLBAND | Replaced. Use the existing **ASSIGNLBANDBEAM** command |
| ASSIGNLBAND2 | Replaced. Use the existing **ASSIGNLBANDBEAM** command |
| BASEANTENNAMODEL | Replaced. Use the existing **BASEANTENNAPCO** command and **BASEANTENNAPCV** command |
| COM | Replaced. Use the existing **SERIALCONFIG** command |

| OEM6 | OEM7 |
|---|---|
| COMCONFIG | Replaced. Use the existing **SERIALCONFIG** command, **INTERFACEMODE** command and **ECHO** command |
| DGPSEPHEMDELAY | Deleted. |
| EXTHDGOFFSET | Replaced. Use the new **SETINSROTATION** command (see page 64) |
| GIMBALSPANROTATION | Replaced. Use the new **SETINSROTATION** command (see page 64) |
| INSWHEELUPDATE | Replaced. Use the existing **SETINSUPDATE** command |
| INSZUPTCONTROL | Replaced. Use the existing **SETINSUPDATE** command |
| LEVERARMCALIBRATE | Replaced. Use the new **INSCALIBRATE** command (see page 34) |
| OMNIUSEGLONASS | Deleted. |
| PDPVELOCITYOUT | Replaced. Use the existing **BESTVELTYPE** command |
| RTKCOMMAND | Deleted. |
| RTKELEVMASK | Deleted. |
| RVBCALIBRATE | Replaced. Use the new **INSCALIBRATE** command (see page 34) |
| SETCANNAME | Deleted. |
| SETGIMBALORIENTATION | Replaced. Use the new **SETINSROTATION** command (see page 64) |
| SETIMUORIENTATION | Replaced. Use the new **SETINSROTATION** command (see page 64) |
| SETIMUTOANTOFFSET | Replaced. Use the new **SETINSTRANSLATION** command (see page 66) |
| SETIMUTOANTOFFSET2 | Replaced. Use the new **SETINSTRANSLATION** command (see page 66) |
| SETIMUTOEXTOFFSET | Replaced. Use the new **SETINSTRANSLATION** command (see page 66) |
| SETIMUTOGIMBALOFFSET | Replaced. Use the new **SETINSTRANSLATION** command (see page 66) |
| SETINSOFFSET | Replaced. Use the new **SETINSTRANSLATION** command (see page 66) |
| SETINITATTITUDE | Replaced. Use the existing **SETINITAZIMUTH** command |
| SETINSOFFSETS | Replaced. Use the new **SETINSROTATION** command on page 64 and **SETINSTRANSLATION** command on page 66 |
| SETINSROTATION | Replaced. The existing SETINSROTATION command (Message ID 1796) has removed.<br>A new **SETINSROTATION** command (see page 64) (Message ID 1921) has been added. This new command has enhanced capability and is the recommend command to use. |

| OEM6 | OEM7 |
|---|---|
| SETMARK1OFFSET<br>SETMARK2OFFSET<br>SETMARK3OFFSET<br>SETMARK4OFFSET | Replaced. Use the new **SETINSROTATION** command on page 64 and **SETINSTRANSLATION** command on page 66 |
| SETRTCM16 | Deleted. |
| SETRTCM36 | Deleted. |
| SETRTCMRXVERSION | Deleted. |
| SETRTCMTXVERSION | Deleted. |
| SETWHEELSOURCE | Deleted. |
| VEHICLEBODYROTATION | Replaced. Use the new **SETINSROTATION** command (see page 64) |

### 2.2.2 Revised Commands

Revised commands are listed in the following table.

**Table 2: Revised Commands in OEM7**

| OEM6 | OEM7 |
|---|---|
| ANTENNAPOWER | 3V3 option removed<br><br>If a short circuit or other problem causes an overload of the current supplied to the antenna, the receiver hardware shuts down the power supplied to the antenna. To restore power, power cycle the receiver. The Receiver Status word, available in the **RXSTATUS** log (see page 126), provides more information about the cause of the problem. |
| ASSIGN,<br>ASSIGNALL,<br>LOCKOUT,<br>SBASCONTROL,<br>TRACKSV,<br>UNLOCKOUT | The OEM7 tracks SBAS PRNs 120-158 and 183-187.<br>The OEM6 tracked 120-138 and 183-187 |
| ASSIGNLBANDBEAM | This command now supports multiple L-Band channels.<br>The Frequency field is entered only in Hz. |

<table>
<tr><th>OEM6</th><th>OEM7</th></tr>
<tr><td>AUTH</td><td>New options were added for the State parameter
<table>
<tr><th>ASCII</th><th>Binary</th><th>Description</th></tr>
<tr><td>ERASE_TABLE</td><td>7</td><td>Erase all auth codes from the system. Requires a special auth code to prevent against accidental erasing</td></tr>
<tr><td>CLEAN_TABLE</td><td>8</td><td>Remove all invalid auth codes from the system</td></tr>
</table>
Special auth code for the ERASE_TABLE case:<br>PW5W2B,WW5TM9,WW2PCZ,WW3M4H,WW4HPG,ERASE_AUTH<br>Special auth code for the CLEAN_TABLE case:<br>4DR69H,G369W8,34MNJJ,5NHXCJ,GW7C75,CLEAN_AUTH</td></tr>
<tr><td>COMCONTROL</td><td>The COM ports available have changed to reflect the OEM7 receivers.<br>See the COMCONTROL command on page 24</td></tr>
<tr><td>CONNECTIMU</td><td>Added the EPSON G320N IMU.<br>Removed the Litef LCI-1 IMU.<br>For the IMU Type parameter, the ASCII values for IMU no longer use the "IMU_" prefix. However, the legacy ASCII values that contain the "IMU_" prefix are still supported. The binary IMU type values have not changed.<br>See the CONNECTIMU command on page 27</td></tr>
<tr><td>ETHCONFIG</td><td>Ethernet interface hardware now automatically connects properly to the other physical device</td></tr>
<tr><td>EXTERNALPVAS</td><td>The format of several of the options in this command have been changed from Double to Float.<br>See the EXTERNALPVAS command on page 29</td></tr>
<tr><td>FRESET</td><td>Added new FRESET targets.
<table>
<tr><th>Binary</th><th>ASCII</th><th>Description</th></tr>
<tr><td>10</td><td>USERDATA</td><td>Resets the user data saved using the NVMUSERDATA command (see page 52)</td></tr>
</table></td></tr>
<tr><td>INTERFACEMODE</td><td>Added new Interface Mode options:
<table>
<tr><th>Binary</th><th>ASCII</th><th>Description</th></tr>
<tr><td>49</td><td>NOVATELMINBINARY</td><td>NovAtel binary message with a minimal header. Only available for CCOM ports.</td></tr>
</table></td></tr>
</table>

| OEM6 | OEM7 |
|---|---|
| LOG | The optional parameters Period, Offset and Hold are only used when the Trigger is ONTIME. Previous versions of the firmware accepted (but ignored) non-zero values for Period and Offset even if the Trigger was not ONTIME. The firmware on OEM7 receivers will reject such commands and return an error.<br>It is especially important that these values are zero when entering a LOG command in Binary or ASCII format. |
| POSAVE | “state” field no longer optional; default ON. A value must now be entered |
| SETINITAZIMUTH | The range for the standard deviation parameter has changed to 1 to 25 degrees. |

### *2.2.3 New Commands*

The following table lists the commands added in OEM7.

**Table 3: New Commands in OEM7**

| OEM7 | Description |
|---|---|
| CANCONFIG | Configures the CAN port parameters<br>See the **CANCONFIG** command on page 20 |
| CCOMCONFIG | Binds a CAN port to a J1939 node configures the CAN protocol used by the port.<br>See the **CCOMCONFIG** command on page 22 |
| FORCEGALE6CODE | Forces Galileo E6 channels to track E6B or E6C<br>See the **FORCEGALE6CODE** command on page 33 |
| INSCALIBRATE | Initiates the calibration of INS offsets<br>See the **INSCALIBRATE** command on page 34 |
| INSSEED | Enables or disables saving and restoring the last known SPAN solution<br>See the **INSSEED** command on page 36 |
| ITBANDPASSCONFIG | Configures a bandpass filter on the receiver<br>See the **ITBANDPASSCONFIG** command on page 37 |
| ITDETECTCONFIG | Enables interference detection<br>See the **ITDETECTCONFIG** command on page 38 |
| ITFRONTENDMODE | Configures the front end mode for each RF path<br>See the **ITFRONTENDMODE** command on page 39 |
| ITPROGFILTCONFIG | Configures filtering on the receiver<br>See the **ITPROGFILTCONFIG** command on page 41 |

| OEM7 | Description |
|---|---|
| ITSPECTRALANALYSIS | Enables and configures spectral analysis on the receiver<br>See the **ITSPECTRALANALYSIS** command on page 43 |
| J1939CONFIG | Configures the CAN J1939 network-level parameters<br>See the **J1939CONFIG** command on page 47 |
| NMEAFORMAT | Customizes the NMEA output<br>See the **NMEAFORMAT** command on page 49 |
| NVMUSERDATA | Writes the data provided to NVM<br>See the **NVMUSERDATA** command on page 52 |
| PPPBASICCONVERGEDCRITERIA | Sets the convergence threshold for lower accuracy PPP solutions.<br>See the **PPPBASICCONVERGEDCRITERIA** command on page 53 |
| PPPDYNAMICSEED | Seed the PPP filter in any platform motion state<br>See the **PPPDYNAMICSEED** command on page 54 |
| PPPRESET | Resets the PPP filter.<br>See the **PPPRESET** command on page 56 |
| REFERENCESTATIONTIMEOUT | Sets a timeout for removing previously stored base stations<br>See the **REFERENCESTATIONTIMEOUT** command on page 57 |
| RFINPUTGAIN | Selects the mode of setting the CAG for the purpose of interference detection.<br>See the **RFINPUTGAIN** command on page 58 |
| RTKPORTMODE | Assigns a port on the rover receiver for the RTK and ALIGN information being received<br>See the **RTKPORTMODE** command on page 60 |
| RTKRESET | Resets the RTK filter.<br>See the **RTKRESET** command on page 62 |
| SETINSPROFILE | Set the filter behavior to optimize the output for a specific environment<br>See the **SETINSPROFILE** command on page 63 |
| SETINSROTATION | Specifies the rotational offsets between the IMU frame and other reference frames, such as the vehicle frame or an ALIGN baseline<br>see the **SETINSROTATION** command on page 64 |
| SETINSTRANSLATION | Specifies the translational offsets between the IMU frame and other reference frames, including GNSS antennas or the desired output frame<br>See the **SETINSTRANSLATION** command on page 66 |

## 2.2.4 CANCONFIG

### Configure CAN ports

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7, SPAN CPT7, SMART7, SMART2**

Use the CANCONFIG command to configure the hardware parameters of the CAN ports.

**Message ID: 884**

**Abbreviated ASCII Syntax:**

```
CANCONFIG port switch [speed]
```

**Factory Default:**

```
CANCONFIG CAN1 OFF 250K
CANCONFIG CAN2 OFF 250K
```

**ASCII Example:**

```
CANCONFIG CAN1 OFF 500K
```

| Field | Field Type | ASCII Value | Binary Value | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 1 | CANCONFIG header | - | - | Command header. See Messages for more information. | - | H | 0 |
| 2 | port | CAN1 | 1 | Physical CAN port ID<br>**Note**: The OEM7500 supports the CAN2 port only. | Enum | 4 | H |
| | | CAN2 | 2 | | | | |
| 3 | switch | ON | 1 | Sets the port to be On or Off the CAN bus | Enum | 4 | H+4 |
| | | OFF | 0 | | | | |
| 4 | speed | See *Table 4: CAN Port Speed* below | | Physical CAN port speed (bits per second)<br>(default = 250K | Enum | 4 | H+8 |

The CAN port must be set to OFF (using `CANCONFIG <port> OFF`) before the port speed can be changed.

**Table 4: CAN Port Speed**

| ASCII Value | Binary Value |
|---|---|
| 10K | 0 |
| 20K | 1 |
| 50K | 2 |

| ASCII Value | Binary Value |
|---|---|
| 100K | 3 |
| 125K | 4 |
| 250K | 5 |
| 500K | 6 |
| 1M | 7 |

### *2.2.5 CCOMCONFIG*

#### Configure the CAN COM port

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7, SPAN CPT7, SMART7**

Bind a CAN communication port to a J1939 node (see **J1939CONFIG** command on page 47) and specify the CAN protocol, PGN, priority and address for messages transmitted and received over the CCOM port.

**Message ID: 1902**

**Abbreviated ASCII Syntax:**

```
CCOMCONFIG port node protocol [pgn [priority [address]]]
```

**Factory Default:**

```
CCOMCONFIG ccom1 node1 J1939 61184 7 fe
CCOMCONFIG ccom2 node2 J1939 61184 7 fe
CCOMCONFIG ccom3 node1 J1939 126720 7 fe
CCOMCONFIG ccom4 none none 0 0 0
CCOMCONFIG ccom5 none none 0 0 0
CCOMCONFIG ccom6 none none 0 0 0
```

**ASCII Example :**

```
ccomconfig ccom1 node1 j1939 1792 6 1b
```

| Field | Field Type | ASCII Value | Binary Value | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 1 | CCOMCONFIG Header | - | - | Command header. See Messages for more information. | - | H | 0 |
| 2 | port | CCOM1 | 38 | Name of CCOM port. | Enum | 4 | H |
| | | CCOM2 | 39 | | | | |
| | | CCOM3 | 40 | | | | |
| | | CCOM4 | 41 | | | | |
| | | CCOM5 | 42 | | | | |
| | | CCOM6 | 43 | | | | |
| 3 | node | NODE1 | 1 | The J1939 node to use. This binds a CCOM port to the CAN NAME/address associated with the node. | Enum | 4 | H+4 |
| | | NODE2 | 2 | | | | |
| 4 | protocol | See *Table 5: CAN Protocol* on the next page | | CAN transport protocol to use. | Enum | 4 | H+8 |

| Field | Field Type | ASCII Value | Binary Value | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 5 | pgn | 0 - 131071 | | Any valid PGN as defined by the J1939 protocol.<br>All messages transmitted over this CCOM port will contain this PGN value.<br>Only messages with this PGN will be received on this CCOM port.<br>**Note**: This value is ignored if the protocol is NMEA2000. | Ulong | 4 | H+12 |
| 6 | priority | 0-7 | | Default CAN message priority for transmitted messages. (Priority 0 is the highest priority)<br>**Note**: This value is ignored if the protocol is NMEA2000. | Uchar | 1 | H+16 |
| 7 | address | 00 – FF | | **00 – FD**:<br>Transmit and receive messages to/from this address only.<br>**FE**:<br>Transmit and receive message to/from the address of the first message received.<br>**FF**:<br>Broadcast messages and receive messages from all addresses.<br>**Note**: This value is ignored if the protocol is NMEA2000. | Hex | 1 | H+17 |

**Table 5: CAN Protocol**

| Binary | ASCII | Description |
|---|---|---|
| 2 | J1939 | J1939 single packet |
| 3 | NMEA2000 | NMEA2000 (single packet, multi-packet, fast packet) |
| 5 | ISO11783 | ISO 11783 transport protocol |

## 2.2.6 COMCONTROL

### Controls the serial port hardware control lines

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720**

This command is used to control the hardware control lines of the serial communication (COM) ports. The TOGGLEPPS mode of this command is typically used to supply a timing signal to a host PC computer by using the RTS and DTR lines. The accuracy of controlling the COM control signals is better than 900 µs. The other modes are typically used to control custom peripheral devices.

1. If handshaking is disabled, any of these modes can be used without affecting regular serial communications through the selected COM port. However, if handshaking is enabled, it may conflict with handshaking of the selected COM port, causing unexpected results.
2. The PULSEPPSLOW control type cannot be issued for a TX signal.
3. Only PULSEPPSHIGH, FORCEHIGH and FORCELOW control types can be used for a TX signal.
4. To use the COM2 flow control signals, COM5 must be disabled. See OEM7600, OEM7700 and OEM7720 Multiplexed Port in the OEM7 Installation and Operation User Manual for more information.

**Message ID: 431**

**Abbreviated ASCII Syntax:**

```
COMCONTROL [port] [signal] [control]
```

**Factory Default:**

```
COMCONTROL COM1 RTS DEFAULT
COMCONTROL COM2 RTS DEFAULT
COMCONTROL COM3 RTS DEFAULT
COMCONTROL COM4 RTS DEFAULT
COMCONTROL COM5 RTS DEFAULT
```

**ASCII Example 1:**

```
SERIALCONFIG COM1 9600 N 8 1 N
```
(to disable handshaking)
```
COMCONTROL COM1 RTS FORCELOW
```

**ASCII Example 2:**

```
COMCONTROL COM1 RTS TOGGLEPPS
COMCONTROL COM2 RTS TOGGLEPPS
```

**ASCII Example 3:**

To set a break condition on COM1:

```
COMCONTROL COM1 TX FORCELOW
```

A break condition remains in effect until it is cleared. To clear a break condition on COM1:

```
COMCONTROL COM1 TX DEFAULT
```

or

```
COMCONTROL COM1 TX FORCEHIGH
```

| Field | Field Type | ASCII Value | Binary Value | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 1 | COM CONTROL header | - | - | Command header. See Messages for more information. | - | H | 0 |
| 2 | port | COM1 | 1 | Serial port to control. | Enum | 4 | H |
| | | COM2 | 2 | | | | |
| | | COM3 | 3 | | | | |
| | | COM4 | 19 | | | | |
| | | COM5 | 31 | | | | |
| 3 | signal | RTS | 0 | COM signal to control. The controllable COM signals are RTS, DTR and TX. (Default = RTS) See also *Table 6: Tx, DTR and RTS Availability* on the next page | Enum | 4 | H+4 |
| | | DTR | 1 | | | | |
| | | TX | 2 | | | | |
| 4 | control | DEFAULT | 0 | Disables this command and returns the COM signal to its default state (Default) | Enum | 4 | H+8 |
| | | FORCEHIGH | 1 | Immediately forces the signal high | | | |
| | | FORCELOW | 2 | Immediately forces the signal low | | | |
| | | TOGGLE | 3 | Immediately toggles the current state of the signal | | | |
| | | TOGGLEPPS | 4 | Toggles the state of the selected signal within 900 µs after each 1PPS event. The state change of the signal lags the 1PPS by an average value of 450 µs. The delay of each pulse varies by a uniformly random amount less than 900 µs | | | |
| | | PULSEPPSLOW | 5 | Pulses the line low at a 1PPS event and to high 1 ms after it. Not for TX | | | |
| | | PULSEPPSHIGH | 6 | Pulses the line high for 1 ms at the time of a 1PPS event | | | |

**Table 6: Tx, DTR and RTS Availability**

| | Tx Available On | DTR Available On | RTS Available On |
|---|---|---|---|
| **OEM719** | COM1, COM2, COM3 | N/A | N/A |
| **OEM729** | COM1, COM2, COM3 | N/A | COM1 and COM2 |
| **OEM7600** | COM1, COM2, COM3, COM4, COM5 | N/A | COM1 and COM2 |
| **OEM7700** | COM1, COM2, COM3, COM4, COM5 | N/A | COM1 and COM2 |
| **OEM7720** | COM1, COM2, COM3, COM4, COM5 | N/A | COM1 and COM2 |

## 2.2.7 CONNECTIMU

### Connects an IMU to a Port

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7**

Use this command to specify the type of IMU connected to the receiver and the receiver port used by the IMU.

**Message ID: 1428**

**Abbreviated ASCII Syntax:**

```
CONNECTIMU IMUPort IMUType
```

**Abbreviated ASCII Example:**

```
CONNECTIMU COM2 LN200
```

| Field | Field Type | ASCII Value | Binary Value | Description | Binary Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 1 | CONNECTIMU header | - | - | Command header. See Messages for more information. | - | H | 0 |
| 2 | IMUPort | COM1 | 1 | IMU Port is COM port 1 | Enum | 4 | H |
| | | COM2 | 2 | IMU Port is COM port 2 | | | |
| | | COM3 | 3 | IMU Port is COM port 3 | | | |
| | | SPI | 7 | IMU Port is the SPI port | | | |
| | | COM4 | 19 | IMU Port is COM port 4 | | | |
| | | COM5 | 31 | IMU Port is COM port 5 | | | |
| 3 | IMUType | See *Table 7: IMU Type* on the next page | | IMU Type | Enum | 4 | H+4 |

SPI is available only on the OEM7500, OEM7600, OEM7700, and OEM7720.

COM4 and COM5 are available only on the OEM7600, OEM7700 and OEM7720.

The IMU-ISA-100C, IMU-FSAS, IMU-HG1900, IMU-LN200, IMU-µIMU, IMU-CPT and IMU-KVH1750 use RS-422 protocol and must be connected to a receiver port that is configured to use RS-422. Refer to the OEM7 Installation and Operation User Manual for information about which receiver ports support RS-422 and instructions for enabling RS-422.

**Table 7: IMU Type**

| Binary | ASCII | Description |
|---|---|---|
| 0 | UNKNOWN | Unknown IMU type (default) |
| 1 | HG1700_AG11 | Honeywell HG1700 AG11 |
| 4 | HG1700_AG17 | Honeywell HG1700 AG17 |
| 5 | HG1900_CA29 | Honeywell HG1900 CA29 |
| 8 | LN200 | Northrop Grumman LN200/LN200C |
| 11 | HG1700_AG58 | Honeywell HG1700 AG58 |
| 12 | HG1700_AG62 | Honeywell HG1700 AG62 |
| 13 | IMAR_FSAS | iMAR iIMU-FSAS |
| 16 | KVH_COTS | KVH CPT IMU |
| 20 | HG1930_AA99 | Honeywell HG1930 AA99 |
| 26 | ISA100C | Northrop Grumman Litef ISA-100C |
| 27 | HG1900_CA50 | Honeywell HG1900 CA50 |
| 28 | HG1930_CA50 | Honeywell HG1930 CA50 |
| 31 | ADIS16488 | Analog Devices ADIS16488 |
| 32 | STIM300 | Sensonor STIM300 |
| 33 | KVH_1750 | KVH1750 IMU |
| 41 | EPSON_G320 | Epson G320N |
| 52 | LITEF_MICROIMU | Northrop Grumman Litef µIMU-IC |
| 56 | STIM300D | Sensonor STIM300, Direct Connection |
| 58 | HG4930_AN01 | Honeywell HG4930 AN01 |
| 61 | EPSON_G370 | Epson G370N |
| 62 | EPSON_G320_200HZ | Epson G320N – 200 Hz |

The IMU Type field also supports the legacy ASCII values that contain the "IMU_" prefix. For example, *LN200* or *IMU_LN200*.
IMUs recently added as SPAN supported devices, such as the LITEF_MICROIMU and STIM300D, do not support the "IMU_" prefix.

Values not shown in this table are reserved.

## 2.2.8 EXTERNALPVAS

### Enter PVA Update

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7, SPAN CPT7, SMART7-S, SMART7-SI**

This command should only be used by advanced users of GNSS+INS.

The standard deviations entered using this command must be representative of actual input error.

The **EXTERNALPVAS** command uses a short header if the command is entered in ASCII or Binary.

This command allows the user to provide their own update for INS in full position, velocity, attitude and other updates, and in any combination. The user can also provide height or attitude only updates, along with Zero Velocity Updates (ZUPTs). The position and velocity updates can be entered in local level or ECEF.

The **EXTERNALPVAS** command is designed to provide a method for additional sensor information to be input into the SPAN filter, specifically during GNSS denied environments. This will provide a method to constrain the error growth that is typical in an Inertial Kalman Filter when GNSS observations are unavailable (environments such as: urban canyon, tunnels, jamming etc.). It is important to ensure that the external update and its corresponding standard deviations are accurate and input with minimal latency to provide optimal effectiveness. Entering an external update or its standard deviation inappropriately may have an adverse effect on the SPAN solution.

The default input frame is ECEF. Updates are entered in ECEF unless Local Level is specified using the OptionsMask parameter.

**Message ID: 1463**

**Abbreviated ASCII Syntax:**

```
EXTERNALPVAS Position1 Position2 Position3 Velocity1 Velocity2 Velocity3
Attitude1 Attitude2 Attitude3 PosStdDev1 PosStdDev2 PosStdDev3 VelStdDev1
VelStdDev2 VelStdDev3 AttStdDev1 AttStdDev2 AttStdDev3 UpdateMask
OptionsMask
```

**Abbreviated ASCII Example:** (Local Level Velocity Update)

```
EXTERNALPVAS 0.0 0.0 0.0 0.4502 10.54 -0.09598 0.0 0.0 0.0 0.0 0.0 0.0 0.01
0.01 0.01 0.0 0.0 0.0 4000 10
```

**Full ASCII Example:** (Local Level Velocity Update)

```
%EXTERNALPVASA,2051,493713.000;0.00000000000,0.00000000000,0.00000000000,
0.1333,17.1424,-0.3311,0.000000000,0.000000000,0.000000000,0.0000,0.0000,
0.0000,0.0119,0.0096,0.0132,0.0000,0.0000,0.0000,00004000,00000010*edced535
```

| Field | Field Type | ASCII Value | Binary Value | Description | Binary Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 1 | EXTERNALPVAS header | - | - | Command header. See Messages for more information. | - | H | 0 |
| 2 | Position1 | | | Latitude (degrees), ECEF X-coordinate (m) or X relative coordinate (m) | Double | 8 | H |
| 3 | Position2 | | | Longitude (degrees), ECEF Y-coordinate (m) or Y relative coordinate (m) | Double | 8 | H+8 |
| 4 | Position3 | | | Height, ECEF Z-coordinate or Z relative coordinate (m) | Double | 8 | H+16 |
| 5 | Velocity1 | | | North velocity or velocity along the X-axis (m/s) | Float | 4 | H+24 |
| 6 | Velocity2 | | | East velocity or velocity along the Y-axis (m/s) | Float | 4 | H+28 |
| 7 | Velocity3 | | | Up velocity or velocity along the Z-axis (m/s) | Float | 4 | H+32 |
| 8 | Attitude1 | | | Pitch – Local Level to SPAN User Output Frame or relative delta from SPAN User Output Frame (degrees) | Float | 4 | H+36 |
| 9 | Attitude2 | | | Roll – Local Level to SPAN User Output Frame or relative delta from SPAN User Output Frame (degrees) | Float | 4 | H+40 |
| 10 | Attitude3 | | | Azimuth – Local Level to SPAN User Output Frame or relative delta from SPAN User Output Frame (degrees) | Float | 4 | H+44 |
| 11 | PosStdDev1 | | | Position1 standard deviation (m) | Float | 4 | H+48 |
| 12 | PosStdDev2 | | | Position2 standard deviation (m) | Float | 4 | H+52 |
| 13 | PosStdDev3 | | | Position3 standard deviation (m) | Float | 4 | H+56 |
| 14 | VelStdDev1 | | | Velocity1 standard deviation (m/s) | Float | 4 | H+60 |
| 15 | VelStdDev2 | | | Velocity2 standard deviation (m/s) | Float | 4 | H+64 |
| 16 | VelStdDev3 | | | Velocity3 standard deviation (m/s) | Float | 4 | H+68 |
| 17 | AttStdDev1 | | | Attitude1 standard deviation (degrees) | Float | 4 | H+72 |

| Field | Field Type | ASCII Value | Binary Value | Description | Binary Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 18 | AttStdDev2 | | | Attitude2 standard deviation (degrees) | Float | 4 | H+76 |
| 19 | AttStdDev3 | | | Attitude3 standard deviation (degrees) | Float | 4 | H+80 |
| 20 | UpdateMask | | | This mask selects which updates are applied. Setting a bit applies the update. More than one update can be applied at one time.<br>See *Table 8: EXTERNALPVAS Updates Mask* below for the external update bits that can be used. | HEX Ulong | 4 | H+84 |
| 21 | OptionsMask | | | This mask selects the update options for various updates, such as using an ECEF or Local Level system for the position updates.<br>See *Table 9: EXTERNALPVAS Options Mask* on the next page for details. | HEX Ulong | 4 | H+88 |

**Table 8: EXTERNALPVAS Updates Mask**

| Bit | Mask | Description |
|---|---|---|
| 0 | 0x00001 | Reserved |
| 1 | 0x00002 | Reserved |
| 2 | 0x00004 | ZUPT Update. No fields are required in the **EXTERNALPVAS** command for this update. |
| 3 | 0x00008 | Reserved |
| 4 | 0x00010 | Reserved |
| 5 | 0x00020 | External Position Update.<br>This update is entered using Position1 to Position3 in the **EXTERNALPVAS** command. |
| 6 | 0x00040 | Reserved |
| 7 | 0x00080 | Reserved |
| 8 | 0x00100 | Reserved |
| 9 | 0x00200 | Reserved |
| 10 | 0x00400 | Reserved |
| 11 | 0x00800 | Reserved |
| 12 | 0x01000 | Reserved |

| Bit | Mask | Description |
|---|---|---|
| 13 | 0x02000 | Reserved |
| 14 | 0x04000 | External Velocity Update.<br>This update is entered using Velocity1 to Velocity3 in the **EXTERNALPVAS** command. |
| 15 | 0x08000 | External Attitude Update.<br>This update is entered using Attitude1 to Attitude3 in the **EXTERNALPVAS** command. |
| 16 | 0x10000 | External Heading Update.<br>This update is entered using Attitude3 in the **EXTERNALPVAS** command. |
| 17 | 0x20000 | External Height Update.<br>This update is entered using Position3 in the **EXTERNALPVAS** command. |

If both the External Position Update and External Height Update bits are set, only the External Position Update will be applied.

If both the External Attitude Update and External Heading Update bits are set, only the External Attitude Update will be applied.

**Table 9: EXTERNALPVAS Options Mask**

| Bit | Mask | Description | Range Value |
|---|---|---|---|
| 0 | 0x00000001 | Reserved | |
| 1 | 0x00000002 | Reserved | |
| 2–3 | 0x0000000C | Position Input Frame | 00 – ECEF |
| | | | 01 – LLH |
| 4–5 | 0x00000030 | Velocity Input Frame | 00 – ECEF |
| | | | 01 – LLH |
| 6 | 0x00000040 | Position Update Type | 0 – Absolute |
| | | | 1 – Relative |
| 7 | 0x00000080 | Attitude Update Type | 0 – Absolute |
| | | | 1 – Relative |

## 2.2.9 FORCEGALE6CODE

### Force receiver to track Galileo E6C or E6B signal

**Platform: OEM719, OEM729, OEM7700, PwrPak7**

Use this command to force Galileo E6 channels to track E6B or E6C.

**Message ID: 2222**

**Abbreviated ASCII Syntax:**

```
FORCEGALE6CODE E6codetype
```

**Factory Default:**

```
FORCEGALE6CODE E6C
```

| Field | Field Type | ASCII Value | Binary Value | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 1 | FORCEGALE6CODE | - | - | Command header. See Messages for more information. | - | H | 0 |
| 2 | E6codetype | E6B | 0 | Galileo E6 code type (default = E6C) | Enum | 4 | H |
| | | E6C | 1 | | | | |

## 2.2.10 INSCALIBRATE

### Initiate calibration of the INS offsets

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7, SPAN CPT7, SMART7-S, SMART7-SI**

Use this command to initiate the calibration of INS offsets.

The RBV calibration requires a valid RBV estimate to be entered prior to initializing the calibration. See the **SETINSROTATION** command on page 64 for details on entering a RBV estimate.

For optimal SPAN performance when using Dual Antenna with SPAN, an ALIGN offset calibration is required for each unique installation. This calibration refines the IMU to antenna baseline angular offset from the initial estimate derived from the input lever arms.

**Message ID: 1882**

**Abbreviated ASCII Syntax:**

```
INSCALIBRATE Offset [Trigger] [SDThreshold]
```

**Abbreviated ASCII Example:**

```
INSCALIBRATE RBV NEW 1.0
```

| Field | Field Type | ASCII Value | Binary Value | Description | Binary Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 1 | INSCALIBRATE header | - | - | Command header. See Messages for more information. | - | H | 0 |
| 2 | Offset | ANT1 | 1 | Use this option to set the INS calibration offset from the IMU to the primary GNSS antenna<br>**Note**: The ANT1 option is available only on IMU Grade 2 or higher IMUs. See Models and Features in the OEM7 SPAN Installation and Operation User Manual. | Enum | 4 | H |
| | | ALIGN | 8 | Use this option to set the INS calibration offset from the IMU Body frame to ALIGN frame rotation. | | | |
| | | RBV | 11 | Use this option to set the INS calibration offset from the IMU Body frame to Vehicle frame rotation. | | | |

| Field | Field Type | ASCII Value | Binary Value | Description | Binary Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 3 | Trigger | STOP | 0 | Ends the INS calibration and uses the current estimate for the RBV offsets | Enum | 4 | H+4 |
| | | NEW | 1 | Begins a new single line calibration, overwriting any previous input or cumulative average offset values | | | |
| | | ADD | 2 | Adds a new path.<br>Only valid for multi-path RBV calibrations | | | |
| | | RESET | 3 | Resets the calibration process and restores the RBV offsets to previous user input values | | | |
| 4 | SDThreshold | | | Standard Deviation Threshold<br>(default for lever arm calibration = 0.10 m)<br>(default for RBV calibration = 0.5 degrees) | Float | 4 | H+8 |

## 2.2.11 INSSEED

### INS Seed Configuration

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7, SPAN CPT7, SMART7-S, SMART7-SI**

This command enables or disables the saving and restoration of the last known SPAN solution from NVM.

**Message ID: 1906**

**Abbreviated ASCII Syntax:**

```
INSSEED Command [Validation]
```

**Abbreviated ASCII Example:**

```
INSSEED ENABLE
```

| Field | Field Type | ASCII Value | Binary Value | Description | Binary Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 1 | INSSEED Header | - | - | Command header. See Messages for more information. | - | H | 0 |
| 2 | Command | DISABLE | 0 | Disable the INS seed functionality | Enum | 4 | H |
| | | ENABLE | 1 | Enable the INS seed functionality | | | |
| | | CLEAR | 2 | Clear the currently saved seed value so it will not be used until re-saved | | | |
| 3 | Validation | VALIDATE | 0 | Validate INS Seed data using GNSS solution before injecting (default) | Enum | 4 | H+4 |
| | | INJECT | 1 | Force an NVM seed value (if available) to be used, without any validation.<br>Using this option to force the seed to be used can result in an unstable INS solution if the vehicle has moved. For advanced users only. | | | |
| 4 | Reserved | | | | Ulong | 4 | H+8 |
| 5 | Reserved | | | | Ulong | 4 | H+12 |

## 2.2.12 ITBANDPASSCONFIG

### Enable and configure bandpass filter on receiver

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7, SPAN CPT7, SMART7**

Use this command to apply a bandpass filter at a certain frequency to mitigate interference in the pass band of GNSS signals. The **ITBANDPASSBANK** log (see page 92) provides information on the allowable configuration settings for each frequency band. The bandpass filter is symmetrical in nature, which means that specifying one cutoff frequency will apply a cutoff on both the low side and high side of the signal's center frequency. Only one filter can be applied for each signal.

On OEM7720, PwrPak7D, PwrPak7D-E1, PwrPak7D-E2 and SPAN CPT7 receivers, any filter enabled for GPS L2 or GLONASS L2 on the secondary antenna will be applied to both GPS L2 and GLONASS L2. For this reason, care must be taken to avoid attenuating the signals with a bandpass filter that is too narrow in bandwidth. The recommended maximum lower cutoff frequency is 1221 MHz. The recommended minimum upper cutoff frequency is 1254 MHz.

**Message ID: 1999**

**Abbreviated ASCII Syntax:**

```
ITBANDPASSCONFIG frequency switch [cutofffrequency]
```

**ASCII Example:**

```
ITBANDPASSCONFIG gpsl5 enable 1165.975
```

| Field | Field Type | ASCII Value | Binary Value | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 1 | ITBANDPASSCONFIG header | - | - | Command header. See Messages for more information. | - | H | 0 |
| 2 | frequency | See *Table 16: Frequency Types* on page 44 | | Set the frequency band on which to apply the filter | Enum | 4 | H |
| 3 | switch | DISABLE | 0 | Disable filter | Enum | 4 | H+4 |
| | | ENABLE | 1 | Enable filter | | | |
| 4 | cutofffrequency | | | Cut off frequency for band pass filter (MHz). (default = 0) Refer to **ITBANDPASSBANK** log (see page 92) for the allowable values. | Float | 4 | H+8 |

## 2.2.13 ITDETECTCONFIG

### Enable interference detection on receiver

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7, SPAN CPT7, SMART7**

This command is used to enable or disable interference detection on the receiver. It is applicable to both Spectral Analysis Detection and Statistical Analysis Detection at the same time. Detection can be enabled on all RF paths, only one RF path (L1, L2, or L5), or no RF paths. By default, only the RF paths connecting to the first antenna are enabled.

**Message ID: 2143**

**Abbreviated ASCII Syntax:**

```
ITDETECTCONFIG RFPath [reserved1] [reserved2] [reserved3]
```

**Factory Default:**

```
ITDETECTCONFIG all
```

**ASCII Example:**

```
ITDETECTCONFIG L1
ITDETECTCONFIG none
```

| Field | Field Type | ASCII Value | Binary Value | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 1 | ITDETECTCONFIG header | - | - | Command header. See Messages for more information. | - | H | 0 |
| 2 | RFPath | See *Table 10: RF Path Selection* below | | RF path selected for detection. By default, all paths are turned on. The receiver will cycle through all active paths. | Enum | 4 | H |
| 3 | reserved1 | 0 | | Reserved parameter | Ulong | 4 | H+4 |
| 4 | reserved2 | 0 | | Reserved parameter | Ulong | 4 | H+8 |
| 5 | reserved3 | 0 | | Reserved parameter | Ulong | 4 | H+12 |

**Table 10: RF Path Selection**

| ASCII Value | Binary Value | Description |
|---|---|---|
| NONE | 0 | Turn off detection on all paths |
| ALL | 1 | Turn on detection on all paths (cycle through all active paths) |
| L1 | 2 | Turn on detection only on L1 path |
| L2 | 3 | Turn on detection only on L2 path |
| L5 | 4 | Turn on detection only on L5 path |

## 2.2.14 ITFRONTENDMODE

### Configure the front end mode settings

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7, SPAN CPT7, SMART7**

Use this command to configure the front end mode for the L1, L2 and L5 RF paths to use the default third-order CIC mode or HDR (High Dynamic Range) mode. The HDR mode is used in an interference environment to obtain best interference rejection in general. However, the power consumption will increase in this mode.

**Message ID: 2039**

**Abbreviated ASCII Syntax:**

```
ITFRONTENDMODE frequency mode
```

**Factory Default**

```
ITFRONTENDMODE L1 cic3
ITFRONTENDMODE L2 cic3
ITFRONTENDMODE LBAND cic3
ITFRONTENDMODE L5 cic3
```

**ASCII Example:**

```
ITFRONTENDMODE L1 hdr
```

On the OEM7500, the default mode for all frequency bands is HDR.

| Field | Field Type | ASCII Value | Binary Value | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 1 | ITFRONTENDMODE header | - | - | Command header. See Messages for more information. | - | H | 0 |
| 2 | frequency | See *Table 11: Frequency Bands* below | | Set the frequency band for adjustment | Enum | 4 | H |
| 3 | mode | See *Table 12: Mode* on the next page | | Select the desired mode | Enum | 4 | H+4 |

**Table 11: Frequency Bands**

| Binary Value | ASCII Value | Description |
|---|---|---|
| 2 | L1 | Selects the L1 frequency |
| 3 | L2 | Selects the L2 frequency |
| 4 | LBAND | Selects the L-Band frequency |
| 5 | L5 | Selects the L5 frequency |

**Table 12: Mode**

| Binary Value | ASCII Value | Description |
|---|---|---|
| 0 | CIC3 | 3rd order CIC (CIC3) mode (default) |
| 1 | HDR | High Dynamic Range (HDR) mode |

## 2.2.15 ITPROGFILTCONFIG

### Enable and configure filtering on the receiver

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7, SPAN CPT7, SMART7**

Use this command to set the programmable filter to be either a notch filter or a bandpass filter to mitigate interference in the pass band of GNSS signals. The notch filter is used to attenuate a very narrow band of frequencies (specified by the notch width) around the configured center frequency.

The bandpass filter is symmetrical in nature, which means that specifying one cutoff frequency will apply a cutoff on both the low side and high side of the spectrum center frequency.

The **ITPROGFILTBANK** log (see page 100) provides information on the allowable configuration settings for the programmable filter (i.e. the allowable settings for the notch filter and bandpass filter) for each frequency band. Only one filter can be applied for each frequency.

**Message ID: 2000**

**Abbreviated ASCII Syntax:**

```
ITPROGFILTCONFIG frequency filterid switch [filtermode] [cutofffreq]
[notchwidth]
```

**ASCII Example:**

```
ITPROGFILTCONFIG gpsl1 pf0 enable notchfilter 1580 1
```

| Field | Field Type | ASCII Value | Binary Value | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 1 | ITPROGFILTCONFIG header | - | - | Command header. See Messages for more information. | - | H | 0 |
| 2 | frequency | See *Table 16: Frequency Types* on page 44 | | Set the frequency band on which to apply the filter | Enum | 4 | H |
| 3 | filterid | See *Table 13: Programmable Filter ID* on the next page | | Select the filter ID to use | Enum | 4 | H+4 |
| 4 | switch | DISABLE | 0 | Disable the filter | Enum | 4 | H+8 |
| | | ENABLE | 1 | Enable the filter | | | |
| 5 | filtermode | See *Table 14: Programmable Filter Mode* on the next page | | Configure the type of filter to use (default = NONE) | Enum | 4 | H+12 |

| Field | Field Type | ASCII Value | Binary Value | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 6 | cutofffreq | | | Center frequency for notch filter or cut off frequency for bandpass filter (MHz).<br>Refer to **ITPROGFILTBANK** log (see page 100) for the allowable values.<br>(default = 0) | Float | 4 | H+16 |
| 7 | notchwidth | | | Notch width (MHz).<br>Refer to **ITPROGFILTBANK** log (see page 100) for the allowable values.<br>(default = 0) | Float | 4 | H+20 |

**Table 13: Programmable Filter ID**

| Binary Value | ASCII Value | Description |
|---|---|---|
| 0 | PF0 | Programmable Filter 0 |
| 1 | PF1 | Programmable Filter 1 |

**Table 14: Programmable Filter Mode**

| Binary Value | ASCII Value | Description |
|---|---|---|
| 0 | NOTCHFILTER | Configure the filter as a notch filter |
| 1 | BANDPASSFILTER | Configure the filter as a bandpass filter |
| 2 | NONE | Turn off filter<br>If the switch parameter is set to ENABLED while the filtermode parameter is set to NONE, the system will return a parameter out of range message. |

## 2.2.16 ITSPECTRALANALYSIS

### Enable and configure spectral analysis on receiver

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7, SPAN CPT7, SMART7**

Use this command to view the spectrum in a range of frequencies. The **ITSPECTRALANALYSIS** command enables and configures the spectral analysis. The spectrum is viewed by plotting the PSD samples in the **ITPSDFINAL** log (see page 102).

Decreasing the update period or increasing the FFT size will impact receiver idle time. The idle time should be monitored to prevent adverse effects on receiver performance.

**Message ID: 1967**

**Abbreviated ASCII Syntax:**

```
ITSPECTRALANALYSIS mode [frequency] [updateperiod] [FFTsize] [timeavg]
[integration_window]
```

**Factory Default:**

```
ITSPECTRALANALYSIS off
```

**ASCII Example:**

```
ITSPECTRALANALYSIS predecimation gps11 500 1K 20 1
```

| Field | Field Type | ASCII Value | Binary Value | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 1 | ITSPECTRALANALYSIS header | - | - | Command header. See Messages for more information. | - | H | 0 |
| 2 | mode | See *Table 15: Data Sources for PSD Samples* on the next page | | Set the view mode. | Enum | 4 | H |
| 3 | frequency | See *Table 16: Frequency Types* on the next page | | Set the frequency band to view. | Enum | 4 | H+4 |
| 4 | updateperiod | 50 to 100000 | | The spectrum update rate in milliseconds.<br>The update period is limited by the FFT size chosen. For 32k the minimum update period is 100 ms and for 64k the minimum update period is 200 ms.<br>(default = 1000) | Ulong | 4 | H+8 |

| Field | Field Type | ASCII Value | Binary Value | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 5 | FFTsize | See *Table 17: FFT Sizes* on the next page | | The frequency resolution of the spectrum.<br>(default = 1k) | Enum | 4 | H+12 |
| 6 | timeavg | 0 to 50 | | Time averaging window in seconds. 0 means no time averaging.<br>(default = 10) | Ulong | 4 | H+16 |
| 7 | integration window | 1 to 1024 | | The integration window size of FFT samples. 1 means no integration.<br>(default = 5) | Ulong | 4 | H+20 |

**Table 15: Data Sources for PSD Samples**

| Binary Value | ASCII Value | Description |
|---|---|---|
| 0 | OFF | Disable spectral analysis |
| 1 | PREDECIMATION | Perform spectrum analysis on the pre-decimated spectrum.<br>This can be used to see a wide view of the spectrum for an RF path (L1, L2 or L5). |
| 2 | POSTDECIMATION | Perform spectrum analysis on the post-decimated spectrum.<br>This is narrower than predecimation and is used to see the spectrum for a given signal. |
| 3 | POSTFILTER | Perform spectrum analysis on the post-filtered spectrum.<br>This can be used when either bandpass or notch filters have been enabled to see the spectrum after the filters are applied. |

The post-filter spectrum is not available for the Galileo AltBOC frequency. Only the pre-decimation and post-decimation spectrums are available for Galileo AltBOC.

**Table 16: Frequency Types**

| Binary Value | ASCII Value | Description |
|---|---|---|
| 0 | GPSL1 | GPS L1 frequency |
| 1 | GPSL2 | GPS L2 frequency |
| 2 | GLONASSL1 | GLONASS L1 frequency |
| 3 | GLONASSL2 | GLONASS L2 frequency |

| Binary Value | ASCII Value | Description |
|---|---|---|
| 4 | Reserved | |
| 5 | GPSL5 | GPS L5 frequency |
| 6[1] | LBAND | Inmarsat L-Band frequency |
| 7 | GALILEOE1 | Galileo E1 frequency |
| 8 | GALILEOE5A | Galileo E5A frequency |
| 9 | GALILEOE5B | Galileo E5B frequency |
| 10 | GALILEOALTBOC | Galileo AltBOC frequency |
| 11 | BEIDOUB1 | BeiDou B1 frequency |
| 12 | BEIDOUB2 | BeiDou B2 frequency |
| 13 | QZSSL1 | QZSS L1 frequency |
| 14 | QZSSL2 | QZSS L2 frequency |
| 15 | QZSSL5 | QZSS L5 frequency |
| 16 | QZSSL6 | QZSS L6 frequency |
| 17 | GALILEOE6 | Galileo E6 frequency |
| 18 | BEIDOUB3 | BeiDou B3 frequency |
| 19 | GLONASSL3 | GLONASS L3 frequency |
| 20 | NAVICL5 | NavIC L5 frequency |
| 21 | BEIDOUB1C | BeiDou B1C frequency |
| 22 | BEIDOUB2A | BeiDou B2a frequency |

**Table 17: FFT Sizes**

| Binary Value | ASCII Value | Description |
|---|---|---|
| 0 | 1K | 1K FFT, 1024 samples |
| 1 | 2K | 2K FFT, 2048 samples |
| 2 | 4K | 4K FFT, 4096 samples |
| 3 | 8K | 8K FFT, 8192 samples |
| 4 | 16K | 16K FFT, 16384 samples |
| 5 | 32K | 32K FFT, 32768 samples |
| 6 | 64K | 64K FFT, 65536 samples |

[1]Must first enable L-Band using the ASSIGNLBANDBEAM command.

The 64k FFT is not available in post-decimation or post-filter modes.

## 2.2.17 J1939CONFIG

### Configure CAN network-level parameters

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7, SPAN CPT7, SMART7, SMART2**

Use this command to configure the CAN J1939 network-level parameters (NAME, etc).

Issuing this command may initiate a CAN 'Address Claim' procedure. The status of the node and address claim are reported in the **J1939STATUS** log (see page 105).

Once a "node" is configured using **J1939CONFIG**, and the "port" is configured to ON using CANCONFIG "port" ON, J1939CONFIG "node" cannot be entered again until the "port" is configured to "OFF" using CANCONFIG "port" OFF. (See the **CANCONFIG** command on page 20

**Message ID: 1903**

**Abbreviated ASCII Syntax:**

```
J1939CONFIG node port [pref_addr [alt_addr_range_start] [alt_addr_range_end]
[mfgcode] [industry] [devclass] [devinstance] [func] [funcinstance]
[ECUinstance]]
```

**Factory Default:**

```
J1939CONFIG NODE1 CAN1 1C 0 FD 305 2 0 0 23 0 0
J1939CONFIG NODE2 CAN2 1C 0 FD 305 2 0 0 23 0 0
```

**ASCII Example :**

```
J1939CONFIG NODE1 CAN1 AA 0 FD 305 2 0 0 23 0 0
```

| Field | Field Type | ASCII Value | Binary Value | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 1 | J1939CONFIG header | - | - | Command header. See Messages for more information. | - | H | 0 |
| 2 | node | NODE1 | 1 | Identifies the J1939 Node (i.e., CAN NAME) | Enum | 4 | H |
| | | NODE2 | 2 | | | | |
| 3 | port | CAN1 | 1 | Physical CAN port to use | Enum | 4 | H+4 |
| | | CAN2 | 2 | | | | |
| 4 | pref_addr | 0x0 - 0xFD | | Preferred CAN address. The receiver attempts to claim this address<br>(default = 0x0) | Ulong | 4 | H+8 |
| 5 | alt_addr_ range_start | 0x0 - 0xFD | | When the pref_addr cannot be claimed, the receiver attempts to claim an address from this range.<br>(default: 0x0) | Ulong | 4 | H+12 |

| Field | Field Type | ASCII Value | Binary Value | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 6 | alt_addr_ range_end | 0x0 - 0xFD | | End of alternative address range. (default: 0xFD) | Ulong | 4 | H+16 |
| 7 | mfgcode | 0-2047 | | NAME: Manufacturer Code. Refer to ISO 11783-5. (default: 0) | Ulong | 4 | H+20 |
| 8 | industry | 0 - 7 | | NAME: Industry Group (default: 2) | Ulong | 4 | H+24 |
| 9 | devclass | 0 - 127 | | NAME: Device Class (default: 0) | Ulong | 4 | H+28 |
| 10 | devinstance | 0 - 15 | | NAME: Device Class Instance (default: 0) | Ulong | 4 | H+32 |
| 11 | func | 0 - 255 | | NAME: Function (default: 23) | Ulong | 4 | H+36 |
| 12 | funcinstance | 0 - 31 | | NAME: Function instance (default: 0) | Ulong | 4 | H+40 |
| 13 | ECUinstance | 0 - 7 | | NAME: ECU Instance (default: 0) | Ulong | 4 | H+44 |

Due to current limitations in the CAN stack, NODE1 can only be associated with CAN1 and NODE2 can only be associated with CAN2. A mismatch combination results in an 'invalid parameter' error.

Node statistics are reported in the **J1939STATUS** log (see page 105).

## 2.2.18 NMEAFORMAT

### Customize NMEA output

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7, SPAN CPT7, SMART7, SMART2**

Use the NMEAFORMAT command to customize the NMEA GPGGA and GPGGALONG output.

Modifying the NMEA output will make it not compliant with the NMEA standard.

**Message ID: 1861**

**Abbreviated ASCII Syntax:**

```
NMEAFORMAT field format
```

**Factory Default:**

```
NMEAFORMAT GGA_LATITUDE 9.4
NMEAFORMAT GGA_LONGITUDE 10.4
NMEAFORMAT GGA_ALTITUDE .2
NMEAFORMAT GGALONG_LATITUDE 12.7
NMEAFORMAT GGALONG_LONGITUDE 13.7
NMEAFORMAT GGALONG_ALTITUDE .3
```

**Example:**

The following settings increase the precision of the GPGGA latitude and longitude fields:

```
NMEAFORMAT GGA_LATITUDE 11.6
NMEAFORMAT GGA_LONGITUDE 12.6
```

The following settings decrease the precision of the GPGGALONG latitude and longitude fields:

```
NMEAFORMAT GGALONG_LATITUDE 11.6
NMEAFORMAT GGALONG_LONGITUDE 12.6
```

The following setting stops the undulation fields of the GPGGALONG log being filled, making a log like the GPGGARTK log that was in NovAtel's OEM6 firmware:

```
NMEAFORMAT GGALONG_UNDULATION !0
```

| Field | Field Type | ASCII Value | Binary Value | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 1 | NMEA FORMAT Header | - | - | Command header. See Messages for more information. | - | H | 0 |

| Field | Field Type | ASCII Value | Binary Value | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 2 | Field | GGA_ LATITUDE | 0 | GPGGA latitude field | Enum | 4 | H |
| | | GGA_ LONGITUDE | 1 | GPGGA longitude field | | | |
| | | GGA_ ALTITUDE | 2 | GPGGA altitude (height) field | | | |
| | | GGA_ UNDULATION | 3 | GPGGA undulation field | | | |
| | | GGALONG_ LATITUDE | 10 | GPGGALONG latitude field | | | |
| | | GGALONG_ LONGITUDE | 11 | GPGGALONG longitude field | | | |
| | | GGALONG_ ALTITUDE | 12 | GPGGALONG altitude (height) field | | | |
| | | GGALONG_ UNDULATION | 13 | GPGGALONG undulation field | | | |

| Field | Field Type | ASCII Value | Binary Value | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 3 | Format | Char[8] | | The Format field has a syntax similar to the printf function commonly found in programming languages. The format is:<br>!x.y<br>Where:<br>**y** is the number of digits to display after the decimal point<br>**x** sets the minimum field width including the decimal point. X is optional if ! is not used. If the value requires fewer digits than x, leading zeros are added to the output.<br>**!** forces the field width to x. ! is optional. If a value exceeds the permitted width, the value will be saturated. If ! is used, y must be less than x.<br>Examples (GGA_LATITUDE):<br>.5 = 5106.98120<br>2.3 = 5106.981<br>7.1 = 05107.0<br>!7.2 = 5106.98<br>!7.3 = 999.999 | Char[8] | 8 | H+4 |

## 2.2.19 NVMUSERDATA

### Write User Data to NVM

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7, SPAN CPT7, SMART7, SMART2**

This command writes the data provided in the data array to NVM. This data can be retrieved by issuing the command **LOG NVMUSERDATA**.

The user data is maintained through power cycles and a standard **FRESET** command. To clear the user data, use the **FRESET USERDATA** command.

The user data may be deleted if the **NVMRESTORE** command is sent. NVMRESTORE should be used with caution and is meant for use only in the event of a NVM receiver error.

**Message ID: 1970**

**Abbreviated ASCII Syntax:**

```
NVMUSERDATA #bytes data
```

| Field | Field Type | Binary Value | Description | Binary Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|
| 1 | NVMUSERDATA header | - | Command header. See Messages for more information. | - | H | 0 |
| 2 | #bytes | - | Number of bytes of data to follow | Ulong | 4 | H |
| 3 | data | - | User input data up to a maximum of 2000 bytes.<br>Data is entered in hexadecimal values with no separators between the values. For example, 1a2b3c4e | Uchar | 2000 | H+4 |

## 2.2.20 PPPBASICCONVERGEDCRITERIA

### Configures decision for PPP Basic convergence

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7, SPAN CPT7, SMART7, SMART2**

The **PPPBASICCONVERGEDCRITERIA** command sets the threshold that determines if the solution has converged for lower accuracy PPP solutions. These are the PPP solutions reported with the PPP_BASIC and PPP_BASIC_CONVERGING position types.

The convergence threshold for high-accuracy PPP solutions (reported with PPP and PPP_CONVERGING position types) is set using the **PPPCONVERGEDCRITERIA** command.

Relaxing the convergence threshold shortens the time before a PPP solution is reported as converged. However, it does not alter solution behavior. During the initial PPP solution period, the positions can have decimeter error variation. Only relax the convergence threshold if the application can tolerate higher solution variability.

**Message ID: 1949**

**Abbreviated ASCII Syntax:**

```
PPPBASICCONVERGEDCRITERIA criteria tolerance
```

**Factory Default:**

```
PPPBASICCONVERGEDCRITERIA horizontal_stddev 0.60
```

**ASCII Example:**

```
PPPBASICCONVERGEDCRITERIA total_stddev 0.45
```

| Field | Field Type | ASCII Value | Binary Value | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 1 | PPPBASIC CONVERGED CRITERIA header | - | - | Command header. See Messages for more information. | - | H | 0 |
| 2 | Criteria | TOTAL_ STDDEV | 1 | Use the total, 3D, standard deviation | Enum | 4 | H |
| | | HORIZONTAL_ STDDEV | 2 | Use the horizontal, 2D, standard deviation | | | |
| 3 | Tolerance | | | Tolerance (m) | Float | 4 | H+4 |

### 2.2.21 PPPDYNAMICSEED

**Seed the PPP filter in any platform motion state**

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7, SPAN CPT7**

This command enables seeding of the PPP engine regardless of the receiver motion state. Accurate seeds can be used to improve initial PPP convergence and re-convergence following signal outages.

The seed position given by the **PPPDYNAMICSEED** command must be in a datum consistent with the PPP corrections that are in use. For TerraStar corrections, the datum is ITRF2014. The dynamic seed's time must refer to receiver time and cannot be more than 15 seconds in the past. A valid PPP solution (the **PPPPOS** log solution status is SOL_COMPUTED) must have been computed for the same epoch as the seed in order for the seed to be used.

See the **PPPSEED** command for stationary-only seeding and for other control over seeding.

**Message ID: 2071**

**Abbreviated ASCII Syntax:**

```
PPPDYNAMICSEED week seconds latitude longitude height northing_std_dev
easting_std_dev height_std_dev [northing_easting_covariance] [northing_
height_covariance] [easting_height_covariance]
```

**Example :**

```
PPPDYNAMICSEED 1817 247603 51.2086442297 -113.9810263055 1071.859 0.02 0.02
0.04
```

| Field | Field Type | ASCII Value | Binary Value | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 1 | PPPDYNAMICSEED header | - | - | Command header. See Messages for more information. | | H | 0 |
| 2 | week | 0-9999 | | GPS Week number | Ulong | 4 | H |
| 3 | seconds | 0-604800 | | Number of seconds into GPS week | Ulong | 4 | H+4 |
| 4 | latitude | ±90 | | Latitude (degrees) | Double | 8 | H+8 |
| 5 | longitude | ±180 | | Longitude (degrees) | Double | 8 | H+16 |
| 6 | height | > -2000.0 | | Ellipsoidal height (meters) | Double | 8 | H+24 |
| 7 | northing_std_dev | | | Northing standard deviation (meters) | Float | 4 | H+32 |
| 8 | easting_std_dev | | | Easting standard deviation (meters) | Float | 4 | H+36 |
| 9 | height_std_dev | | | Ellipsoidal height standard deviation (meters) | Float | 4 | H+40 |

| Field | Field Type | ASCII Value | Binary Value | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 10 | northing_easting_ covariance | | | Covariance between northing and easting components (meters) | Float | 4 | H+44 |
| 11 | northing_height_ covariance | | | Covariance between northing and height components (meters) | Float | 4 | H+48 |
| 12 | easting_height_ covariance | | | Covariance between easting and height components (meters) | Float | 4 | H+52 |

## 2.2.22 PPPRESET

### Reset the PPP filter

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7, SPAN CPT7, SMART7, SMART2**

This command resets the PPP filter. After a reset, the PPP filter is restored to its initial state and PPP convergence will start over.

If deletion of the NVM-saved PPP seed information is also required, then a **PPPSEED CLEAR** command must be applied before the PPPRESET command. See the **PPPSEED** command.

**Message ID: 1542**

**Abbreviated ASCII Syntax:**

```
PPPRESET [Option]
```

**ASCII Example :**

```
PPPRESET
```

| Field | Field Type | ASCII Value | Binary Value | Description | Binary Bytes | Binary Format | Binary Offset |
|---|---|---|---|---|---|---|---|
| 1 | PPPRESET header | - | - | Command header. See Messages for more information. | - | H | 0 |
| 2 | Option | FILTER | 1 | Reset the PPP filter.<br>(default = FILTER) | 4 | Enum | H |
| | | ALL | 10 | Reset the PPP filter and Corrections | | | |

## 2.2.23 REFERENCESTATIONTIMEOUT

### Sets timeout for removing previously stored base stations

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7, SPAN CPT7, SMART7**

This command sets how long the receiver will retain RTK base station co-ordinates. Shorter durations might be required if the receiver is operating in a VRS RTK network that recycles base station IDs quickly.

**Message ID: 2033**

**Abbreviated ASCII Syntax:**

```
REFERENCESTATIONTIMEOUT option [timeout]
```

**Factory Default:**

```
REFERENCESTATIONTIMEOUT AUTO
```

**ASCII Example:**

```
REFERENCESTATIONTIMEOUT SET 90
```

| Field | Field Type | ASCII Value | Binary Value | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 1 | REFERENCESTATION TIMEOUT header | - | - | Command header. See Messages for more information. | - | H | 0 |
| 2 | option | AUTO | 1 | Sets the Timeout to 90 seconds[1]<br>The Timeout field is optional for AUTO and has no effect | Enum | 4 | H |
| | | SET | 2 | Must set the timeout value using the Timeout field<br>0 is not accepted when using the SET option | | | |
| 3 | timeout | 1 to 3600 s | | Specify the time | Ulong | 4 | H+4 |

[1]This behavior is subject to change.

## 2.2.24 RFINPUTGAIN

### Configure the Calibrated Antenna Gain (CAG)

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7, SPAN CPT7**

Use this command to select the mode (AUTO or MANUAL) of setting the CAG for the purpose of interference detection.

If auto mode is used, the receiver will automatically compute the CAG at start up. In this case it is assumed that the receiver is powered up with its antenna connected and no interference is present.

If the antenna is changed, either reset the receiver or reissue this command to allow receiver to re-compute the CAG.

If manual mode is used, the CAG input by the user is used by the receiver to detect interference.

The CAG is defined to be the cascaded RF gain before receiver input plus LNA noise figure (NF), counting active antenna LNA gain, in-line amplifier, RF cable or distribution loss prior to receiver input connector.

A typical GNSS active antenna (of reasonable quality) has a noise figure of ~2dB (dominated by the LNA in an active antenna).

RFINPUTGAIN = Cascaded Gain before receiver + LNA NF

**For advanced users**.
If using this command in manual mode, the antenna gain must be accurately measured when the system is not experiencing any interference. If an erroneous CAG is injected, the interference detection performance can be degraded.

**Message ID: 1658**

**Abbreviated ASCII Syntax:**

```
RFINPUTGAIN RFPath [mode] [CAG]
```

**Factory Default:**

```
RFINPUTGAIN L1 AUTO
RFINPUTGAIN L2 AUTO
RFINPUTGAIN L5 AUTO
```

**ASCII Example:**

```
RFINPUTGAIN L1 MANUAL 30
RFINPUTGAIN L2 30
```

| Field | Field Type | ASCII Value | Binary Value | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 1 | RFINPUTGAIN header | - | - | Command header. See Messages for more information. | - | H | 0 |

| Field | Field Type | ASCII Value | Binary Value | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 2 | RFPath | L1 | 2 | RF path selection | Enum | 4 | H |
| | | L2 | 3 | | | | |
| | | L5 | 5 | | | | |
| 3 | mode | AUTO | 0 | Calibrated Antenna Gain (CAG) mode.<br>Default = MANUAL | Enum | 4 | H+4 |
| | | MANUAL | 1 | | | | |
| 4 | CAG | 0.0-100.0 | | Calibrated Antenna Gain value<br>If the mode is MANUAL, a value for CAG must be entered. | Float | 4 | H+8 |

## 2.2.25 RTKPORTMODE

### Assigns the port for RTK and ALIGN messages

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7, SPAN CPT7, SMART7**

This command only applies to receivers with both RTK and ALIGN enabled.

A rover receiver with RTK and ALIGN enabled can receive RTK and ALIGN corrections at the same time. However, the two different sources (RTK and ALIGN) must be sent to different ports.

Use the RTKPORTMODE command to route correction feeds to different ports. RTK and ALIGN can be routed to any user specified ports.

Failing to specify the mode for the incoming source could cause unexpected behavior of RTK or ALIGN.

Ports configured using the RTKPORTMODE command must also be configured using the **INTERFACEMODE** command.

**Message ID: 1936**

**Abbreviated ASCII Syntax:**

```
RTKPORTMODE [port] mode
```

**Factory Default:**

```
RTKPORTMODE COM1 RTK
RTKPORTMODE COM2 RTK
RTKPORTMODE COM3 RTK
RTKPORTMODE COM4 RTK
RTKPORTMODE COM5 RTK
RTKPORTMODE COM6 RTK
RTKPORTMODE ICOM1 RTK
RTKPORTMODE ICOM2 RTK
RTKPORTMODE ICOM3 RTK
RTKPORTMODE ICOM4 RTK
RTKPORTMODE ICOM5 RTK
RTKPORTMODE ICOM6 RTK
RTKPORTMODE ICOM7 RTK
RTKPORTMODE NCOM1 RTK
RTKPORTMODE NCOM2 RTK
RTKPORTMODE NCOM3 RTK
RTKPORTMODE USB1 RTK
RTKPORTMODE USB2 RTK
```

```
RTKPORTMODE USB3 RTK
RTKPORTMODE WCOM1 RTK
RTKPORTMODE BT1 RTK
RTKPORTMODE AUX RTK
RTKPORTMODE CCOM1 RTK
RTKPORTMODE CCOM2 RTK
RTKPORTMODE CCOM3 RTK
RTKPORTMODE CCOM4 RTK
RTKPORTMODE CCOM5 ALIGN
RTKPORTMODE CCOM6 RTK
```

**ASCII Example:**

```
RTKPORTMODE COM2 RTK
RTKPORTMODE COM3 ALIGN
```

| Field | Field Type | ASCII Value | Binary Value | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 1 | RTKPORTMODE header | - | - | Command header. See Messages for more information. | - | H | 0 |
| 2 | Port | See Table: Communications Port Identifiers | | Port identifier (default = THISPORT) | Enum | 4 | H |
| 3 | Mode | RTK | 0 | Mode for this port | Enum | 4 | H+4 |
| | | ALIGN | 1 | | | | |

### 2.2.26 RTKRESET

**Reset the RTK filter**

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7, SPAN CPT7, SMART7**

This command resets the RTK filter and causes the AdVanceRTK filter to undergo a complete reset, forcing the system to restart the ambiguity resolution calculations.

**Message ID: 2082**

**Abbreviated ASCII Syntax:**

```
RTKRESET [Switch]
```

**Example :**

```
RTKRESET
```

| Field | Field Type | ASCII Value | Binary Value | Description | Format | Binary Byte | Binary Offset |
|---|---|---|---|---|---|---|---|
| 1 | RTKRESET header | - | - | Command header. See Messages for more information. | - | H | 0 |
| 2 | Switch | FILTER | 1 | Reset the RTK filter.<br>This is an optional parameter | Enum | 4 | H |

## 2.2.27 SETINSPROFILE

### Sets filter behavior depending on system environment

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7, SPAN CPT7, SMART7-S, SMART7-SI**

This command sets specific filter behavior depending on the environment the system is installed in. The DEFAULT profile is the legacy setting from earlier SPAN products. The other profiles make changes specific to that environment.

See the OEM7 SPAN Installation and Operation User Manual for a detailed description of each profile's effect.

**Message ID: 1944**

**Abbreviated ASCII Syntax:**

```
SETINSPROFILE profile
```

**Abbreviated ASCII Example:**

```
SETINSPROFILE LAND
```

| Field | Field Type | ASCII Value | Binary Value | Description | Binary Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 1 | SETINS PROFILE Header | - | - | Command header. See Messages for more information. | - | H | 0 |
| 2 | Profile | Default | 0 | Default INS profile with standard SPAN behavior. | Enum | 4 | H |
| | | LAND | 1 | INS profile for land vehicles | | | |
| | | MARINE | 2 | INS profile for marine vehicles | | | |
| | | FIXEDWING | 3 | INS profile for fixed wing aircraft | | | |
| | | FOOT | 4 | INS profiles for walking/backpack applications | | | |
| | | VTOL | 5 | INS profile for vertical takeoff and landing vehicles (UAVs, helicopters, etc.) | | | |
| | | RAIL | 6 | INS profile for trains | | | |
| | | AGRICULTURE | 7 | INS profile for agriculture applications | | | |

## 2.2.28 SETINSROTATION

**Specifies rotational offsets between the IMU frame and other reference frames**

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7, SPAN CPT7, SMART7-S, SMART7-SI**

Use the **SETINSROTATION** command to specify rotational offsets between the IMU frame and other reference frames, such as the vehicle frame or an ALIGN baseline. Offsets must be entered as the rotation from the IMU body frame, to the frame of interest. The order of rotations is Z, X, Y. All rotations are right handed.

It is very important to follow the order of rotations (Z, X, Y) when determining the rotations from IMU body frame to frame of interest.

To specify translational offsets between frames, see the **SETINSTRANSLATION** command on page 66.

**Message ID: 1921**

**Abbreviated ASCII Syntax:**

```
SETINSROTATION INSRotation XRotation YRotation ZRotation [XRotationSD]
[YRotationSD] [ZRotationSD]
```

**Abbreviated ASCII Example:**

```
SETINSROTATION RBV 0 0 90 3.0 3.0 3.0
```

| Field | Field Type | ASCII Value | Binary Value | Description | Binary Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 1 | SETINSROTATION Header | - | - | Command header. See Messages for more information. | - | H | 0 |
| 2 | INS Rotation | *Table 18: Rotational Offset Types* on the next page | | Rotational offset to be set. | Enum | 4 | H |
| 3 | XRotation | ±180 | | X rotation offset from IMU origin (degrees) | Float | 4 | H+4 |
| 4 | YRotation | ±180 | | Y rotation offset from IMU origin (degrees) | Float | 4 | H+8 |
| 5 | ZRotation | ±180 | | Z rotation offset from IMU origin (degrees) | Float | 4 | H+12 |
| 6 | XRotationSD | 0 to 45 | | Optional X rotation offset standard deviation (degrees) Default: 0.0 | Float | 4 | H+16 |

| Field | Field Type | ASCII Value | Binary Value | Description | Binary Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 7 | YRotationSD | 0 to 45 | | Optional Y translation offset standard deviation (degrees) Default: 0.0 | Float | 4 | H+20 |
| 8 | ZRotationSD | 0 to 45 | | Optional Z translation offset standard deviation (degrees) Default: 0.0 | Float | 4 | H+24 |
| 9 | Reserved | | | | Long | 4 | H+28 |

**Table 18: Rotational Offset Types**

| ASCII Value | Binary Value | Description |
|---|---|---|
| USER | 4 | Rotation from the IMU body frame to the user output frame.<br>This offset shifts the attitude information in the INSPVA, INSPOS, INSVEL, INSATT, and INSSPD logs, along with their short header and extended versions. |
| MARK1 | 5 | Rotation from the IMU body frame to the desired output for MARK1.<br>This offset rotates the attitude information in the MARK1PVA log. |
| MARK2 | 6 | Rotation from the IMU body frame to the desired output for MARK2.<br>This offset rotates the attitude information in the MARK2PVA log. |
| ALIGN | 8 | Rotation from the IMU body frame to an ALIGN dual antenna solution.<br>When using a dual antenna ALIGN solution with SPAN, this offset will be calculated automatically if translational offsets to both the primary and secondary GNSS antennas are provided using the **SETINSTRANSLATION** command on the next page. |
| MARK3 | 9 | Rotation from the IMU body frame to the desired output for MARK3.<br>This offset rotates the attitude information in the MARK3PVA log. |
| MARK4 | 10 | Rotation from the IMU body frame to the desired output for MARK4.<br>This offset rotates the attitude information in the MARK4PVA log. |
| RBV | 11 | Rotation from the IMU body frame to the vehicle frame. |
| RBM | 12 | Rotation from the IMU body frame to the gimbal mount body frame. |

## 2.2.29 SETINSTRANSLATION

**Specifies translational offsets between the IMU frame and other reference frames**

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7, SPAN CPT7, SMART7-S, SMART7-SI**

Use the **SETINSTRANSLATION** command to specify translational offsets between the IMU frame and other reference frames, including GNSS antennas or the desired output frame. Offsets must be entered as the vector from the IMU, to the frame or position of interest. Offsets can be entered either in the IMU body frame, or the vehicle frame; offsets in the vehicle frame will be automatically rotated into the IMU body frame using the best available IMU Body to Vehicle Rotation (RBV).

For details on entering the RBV rotation or other angular offsets, see the **SETINSROTATION** command on page 64.

**Message ID: 1920**

**Abbreviated ASCII Syntax:**

```
SETINSTRANSLATION INSTranslation XTranslation YTranslation ZTranslation
[XTranslationSD] [YTranslationSD] [ZTranslationSD] [InputFrame]
```

**Abbreviated ASCII Example:**

```
SETINSTRANSLATION USER 1.0 2.0 3.0 0.05 0.05 0.05 VEHICLE
```

| Field | Field Type | ASCII Value | Binary Value | Description | Binary Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 1 | SETINS TRANSLATION Header | - | - | Command header. See Messages for more information. | - | H | 0 |
| 2 | InsTranslation | See *Table 19: Translation Offset Types* on the next page | | Translation offset to be set | Enum | 4 | H |
| 3 | XTranslation | ±100 | | X translation offset from IMU origin (m) | Float | 4 | H+4 |
| 4 | YTranslation | ±100 | | Y translation offset from IMU origin (m) | Float | 4 | H+8 |
| 5 | ZTranslation | ±100 | | Z translation offset from IMU origin (m) | Float | 4 | H+12 |
| 6 | XTranslationSD | 0 to 10 | | Optional X translation offset standard deviation (m) | Float | 4 | H+16 |
| 7 | YTranslationSD | 0 to 10 | | Optional Y translation offset standard deviation (m) | Float | 4 | H+20 |

| Field | Field Type | ASCII Value | Binary Value | Description | Binary Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|---|---|
| 8 | ZTranslationSD | 0 to 10 | | Optional Z translation offset standard deviation (m) | Float | 4 | H+24 |
| 9 | InputFrame | *Table 20: Translation Input Frame* on the next page | | Optional input frame for translation offset values | Enum | 4 | H+48 |

For the ANT1, ANT2, EXTERNAL and GIMBAL translations, the standard deviation defaults are set to 10% of the translation value (up to a max of 10 meters).

Large standard deviations can lead to an inaccurate position solution. Therefore, it is highly recommended to measure translation offsets as accurately as possible and to manually enter translation offset standard deviations that reflect that accuracy.

**Table 19: Translation Offset Types**

| ASCII Value | Binary Value | Description |
|---|---|---|
| ANT1 | 1 | Offset from the IMU center of navigation to the phase center of the primary GNSS antenna. |
| ANT2 | 2 | Offset from the IMU center of navigation to the phase center of the secondary GNSS antenna. |
| EXTERNAL | 3 | Offset from the IMU center of navigation to the external position source location.<br>This offset type is for use with the **EXTERNALPVAS** command (see page 29). |
| USER | 4 | Translation from the IMU center of navigation to the user output location.<br>This offset shifts the position and velocity information in the INSPVA, INSPOS, INSVEL, INSATT, and INSSPD logs, along with their short header and extended versions. |
| MARK1 | 5 | Translation from the IMU center of navigation to the MARK1 output location.<br>This offset shifts the position and velocity information in the MARK1PVA log. |
| MARK2 | 6 | Translation from the IMU center of navigation to the MARK2 output location.<br>This offset shifts the position and velocity information in the MARK2PVA log. |
| GIMBAL | 7 | Translation from the IMU center of navigation to the gimbal mount center of rotation. |
| MARK3 | 9 | Translation from the IMU center of navigation to the MARK3 output location.<br>This offset shifts the position and velocity information in the MARK3PVA log. |
| MARK4 | 10 | Translation from the IMU center of navigation to the MARK4 output location.<br>This offset shifts the position and velocity information in the MARK4PVA log. |

**Table 20: Translation Input Frame**

| ASCII Value | Binary Value | Description |
|---|---|---|
| IMUBODY | 0 | Offset is provided in the IMU enclosure frame.<br>Default: IMUBODY |
| VEHICLE | 1 | Offset is provided in the vehicle frame.<br>Offsets entered in the vehicle frame will be automatically rotated into the IMU frame using the best available RBV (rotation from IMU Body to Vehicle) information when required.<br>Vehicle frame offsets should only be used if the RBV is known accurately, either though user measurement or calibration.<br>The order of entry for vehicle frame offsets and the RBV rotation does not matter. |

## 2.3 OEM6 to OEM7 – Logs

The following sections describe the log changes between OEM6 and OEM7.

### *2.3.1 Deleted Logs*

The following table lists the OEM6 logs that have been deleted or replaced in OEM7.

**Table 21: Deleted and Replaced Logs in OEM7**

| OEM6 | OEM7 |
|---|---|
| BESTLEVERARM | Replaced. Use the new **INSCONFIG** log (see page 81) |
| BESTLEVERARM2 | Replaced. Use the new **INSCONFIG** log (see page 81) |
| CMR Data logs | Deleted the logs *CMRDESC*, *CMRGLOOBS*, *CMROBS*, *CMRPLUS*, *CMRREF* |
| GALEPHEMERIS | Replaced with the **GALFNAVEPHEMERIS** log for Galileo FNAV ephemeris information and **GALINAVEPHEMERIS** log for Galileo INAV ephemeris information |
| GPGGARTK | Deleted<br>Refer to the **NMEAFORMAT** command on page 49 for information about accessing the information that was provided by this log. |
| HEADING | Replaced. Use the existing **HEADING2** log |
| IMUTOANTOFFSETS | Replaced. Use the new **INSCONFIG** log (see page 81) |
| INSCOV | Replaced. Use the new **INSSTDEV** log (see page 84) |
| INSCOVS | Replaced. Use the new **INSSTDEVS** log (see page 86) |
| INSUPDATE | Replaced. Use the new **INSUPDATESTATUS** log (see page 88) |
| LBANDINFO | Replaced.<br>Use the existing **TERRASTARINFO** log or **VERIPOSINFO** log to log subscription information<br>Use the existing **TERRASTARSTATUS** log or **VERIPOSSTATUS** log to log decoder and subscription status information |
| LBANDSTAT | |
| OMNIHPPOS | |
| OMNIHPSATS | |
| OMNIVIS | |
| PSRTIME | Deleted |
| RAWLBANDFRAME | Deleted |
| RAWLBANDPACKET | Deleted |
| SATVIS | Replaced. Use the existing **SATVIS2** log |
| RTCA Data Logs | Deleted the logs: *RTCA1*, *RTCAEPHEM*, *RTCAOBS*, *RTCAOBS2*, *RTCAOBS3*, *RTCAREF*, *RTCAREFEXT* |

| OEM6 | OEM7 |
|---|---|
| RTCM Data Logs | Deleted the logs: *RTCM1, RTCM3, RTCM9, RTCM15, RTCM16, RTCM16T, RTCM1819, RTCM2021, RTCM22, RTCM23, RTCM24, RTCM31, RTCM32, RTCM36, RTCM36T, RTCM59, RTCM59GLO, RTCMOMNI1* |
| WAAS0, WAAS1, WAAS2, etc. | Replaced. Use the existing **SBAS0** log, **SBAS1** log, **SBAS2** log, etc. |

### *2.3.2 Revised Logs*

Revised logs are listed in the table below.

**Table 22: Revised Logs in OEM7**

<table>
<tr><th>OEM6</th><th>OEM7</th></tr>
<tr><td>AUTHCODES</td><td>Added auth code types for Reserved and High Speed Signatures:<br>4 = RESERVED<br>5 = HIGH_SPEED</td></tr>
<tr><td>CHANCONFIGLIST</td><td>The following signal types have been added to Table: CHANCONFIGLIST Signal Type
<table>
<tr><th>Value</th><th>Name</th><th>Description</th></tr>
<tr><td>32</td><td>BEIDOUB1B3</td><td>BeiDou B1, B3</td></tr>
<tr><td>33</td><td>BEIDOUB3</td><td>BeiDou B3</td></tr>
<tr><td>34</td><td>BEIDOUB1B2B3</td><td>BeiDou B1, B2 and B3</td></tr>
<tr><td>35</td><td>GALE1E5AE5BALTBOCE6</td><td>Galileo E1, E5A, E5B, AltBOC, E6</td></tr>
<tr><td>36</td><td>GPSL1L2PL2CL5L1C</td><td>GPS L1CA, L2P, L2C, L5, L1C</td></tr>
<tr><td>37</td><td>QZSSL1CAL2CL5L1C</td><td>QZSS L1CA, L2C, L5, L1C</td></tr>
<tr><td>38</td><td>QZSSL1CAL2CL5L1CL6</td><td>QZSS L1CA, L2C, L5, L1C, L6</td></tr>
<tr><td>39</td><td>GLOL1L3</td><td>GLONASS L1CA, L3</td></tr>
<tr><td>40</td><td>GLOL3</td><td>GLONASS L3</td></tr>
<tr><td>41</td><td>GLOL1L2PL2CL3</td><td>GLONASS L1CA, L2P, L2CA, L3</td></tr>
<tr><td>42</td><td>GPSL1L2PL2CL1C</td><td>GPS L1CA, L2P, L2C, L1C</td></tr>
<tr><td>43</td><td>QZSSL1CAL2CL1C</td><td>QZSS L1CA, L2C, L1C</td></tr>
</table></td></tr>
<tr><td>GPHDT</td><td>The GPHDT log can only be logged using the ONCHANGED trigger. Other triggers, such as ONTIME are not accepted.</td></tr>
<tr><td>HWMONITOR</td><td>A new status, Peripheral Core Voltage, was added for OEM7 receivers. Also, the status options supported by OEM7 receivers are indicated.</td></tr>
</table>

<table>
<tr><th>OEM6</th><th>OEM7</th></tr>
<tr><td>INSATTX,<br>INSPOSX,<br>INSPVAX,<br>INSUPDATESTATUS,<br>INSVELX,<br>RELINSPVA,<br>SYNCRELINSPVA</td><td>Additional status information has been added to the Extended Solution Status word. The Extended Solution Status word is available in these logs.<br>See the INSATTX log on page 75 for a description of the new status values available.</td></tr>
<tr><td>GIMBALLEDPVA,<br>IMURATEPVA,<br>IMURATEPVAS,<br>INSATT, INSATTS,<br>INSATTX, INSPOS,<br>INSPOSS, INSPOSX,<br>INSPVA, INSPVAS,<br>INSPVAX, INSSPD,<br>INSSPDS, INSVEL,<br>INSVELS, INSVELX,<br>MARKxPVA,<br>RELINSPVA,<br>SYNCRELINSPVA</td><td>The following statuses have been added to the Inertial Solution Status table.
<table>
<tr><th>Binary</th><th>ASCII</th><th>Description</th></tr>
<tr><td>10</td><td>WAITING_<br>AZIMUTH</td><td>The INS filer has orientation, initial biases, initial position and valid roll/pitch estimated. Will not proceed until initial azimuth is entered.</td></tr>
<tr><td>11</td><td>INITIALIZING_<br>BIASES</td><td>The INS filter is estimating initial biases during the first 10 seconds of stationary data.</td></tr>
<tr><td>12</td><td>MOTION_<br>DETECT</td><td>The INS filter has not completely aligned, but has detected motion.</td></tr>
</table></td></tr>
<tr><td>LBANDTRACKSTAT</td><td>This log now reports the L-Band tracking and Viterbi status for more than one beam.</td></tr>
<tr><td>RANGECMP</td><td>GLONASS frequency number added to the Range Record Format table. The changes are shown below:
<table>
<tr><th>Data</th><th>Bits first to last</th><th>Length (bits)</th><th>Scale Factor</th><th>Units</th></tr>
<tr><td>C/No</td><td>165-169</td><td>5</td><td>(20+n)</td><td>dB-Hz</td></tr>
<tr><td>GLONASS Frequency Number</td><td>170-175</td><td>n+7</td><td>1</td><td></td></tr>
<tr><td>Reserved</td><td>176-191</td><td>16</td><td></td><td></td></tr>
</table></td></tr>
<tr><td>RANGE, RANGECMP,<br>RANGECMP2</td><td>OEM7 tracks additional GNSS signals:<br>GPS L1C, GLONASS L3, QZSS L1C and QZSS L6</td></tr>
<tr><td>RANGE, RANGECMP,<br>RANGECMP2,<br>RANGECMP4,<br>RANGEGPSL1, SATVIS2,<br>TRACKSTAT</td><td>The OEM7 tracks SBAS PRNs 120-158 and 183-192.<br>The OEM7 tracks QZSS PRNs 193-202.<br>The OEM6 tracks SBAS PRNs 120-138 and 183-187.<br>The OEM6 tracks QZSS PRNs 193-197.</td></tr>
</table>

| OEM6 | OEM7 |
|---|---|
| RXSTATUS | New OEM7 error and status states have been added to the Receiver Error, Receiver Status, Auxiliary 1 Status and Auxiliary 2 Status words.<br>See the **RXSTATUS** log on page 126<br>The receiver family from which the RXSTATUS log was generated must be known to properly interpret the information provided by the Error and Status words. To determine the receiver family, use the Version Bits (25 and 26) of the Receiver Status word.<br>If bit 25 is 1 and bit 26 is 0, the RXSTATUS log is from an OEM7 receiver.<br>If bit 25 is 0 and bit 26 is 0, the RXSTATUS log is from an OEM6 or earlier receiver. |
| TRACKSTAT | The OEM7 has channel state 23 for BOC sidepeak check. |
| VARIABLELEVERARM | The computed lever arm offset is now reported in the IMU Body frame, rather than the SPAN computation frame. |

### *2.3.3 New Logs*

The table below lists the logs added in OEM7.

**Table 23: New Logs in OEM7**

| OEM7 | Description |
|---|---|
| GALCNAVRAWPAGE | Provides Galileo raw C-NAV page data from Galileo E6 signals.<br>See the **GALCNAVRAWPAGE** log on page 74 |
| INSCALSTATUS | Provides the status and estimated values for the currently running offset calibration.<br>See the **INSCALSTATUS** log on page 79 |
| INSCONFIG | Provides the current configuration of the SPAN system.<br>See the **INSCONFIG** log on page 81 |
| INSSTDEV | Provides the INS PVA standard deviations.<br>See the **INSSTDEV** log on page 84 |
| INSSTDEVS | Provides the INS PVA standard deviations and has a short header.<br>See the **INSSTDEVS** log on page 86 |
| INSUPDATESTATUS | Provides the most recent INS update information<br>See the **INSUPDATESTATUS** log on page 88 |
| ITBANDPASSBANK | Provides the allowable bandpass filter configurations<br>See the **ITBANDPASSBANK** log on page 92 |

| OEM7 | Description |
|---|---|
| ITDETECTSTATUS | Provides a list of detected interference<br>See the **ITDETECTSTATUS** log on page 94 |
| ITFILTTABLE | Provides the filter configuration for each frequency<br>See the **ITFILTTABLE** log on page 96 |
| ITPROGFILTBANK | Provides the allowable filter configurations<br>See the **ITPROGFILTBANK** log on page 100 |
| ITPSDFINAL | Provides the processed power spectral density information<br>See the **ITPSDFINAL** log on page 102 |
| J1939STATUS | Provides the status of the J1939 node<br>See the **J1939STATUS** log on page 105 |
| MODELFEATURES | Clearly states the features available for the current loaded model<br>See the **MODELFEATURES** log on page 107 |
| RANGECMP4 | Highly compressed version of the RANGE log<br>See the **RANGECMP4** log on page 110 |
| RAWSBASFRAME2 | Provides the raw SBAS frame data and the transmitted frequency.<br>See the **RAWSBASFRAME2** log on page 122 |
| RTKASSISTSTATUS | Provides information on the state of RTK ASSIST<br>See the **RTKASSISTSTATUS** log on page 124 |
| SAFEMODESTATUS | Provides the Safe Mode status<br>See the **SAFEMODESTATUS** log on page 139 |

### 2.3.4 GALCNAVRAWPAGE

#### Galileo raw CNAV page

**Platform: OEM719, OEM729, OEM7500, OEM7700, OEM7720, PwrPak7, SPAN CPT7, SMART7, SMART7-I, SMART7-S, SMART7-SI, SMART7-W**

This log provides Galileo raw C/NAV page data from Galileo E6 signals.

The **GALCNAVRAWPAGE** log is not output by default. To receive this log, data decoding for E6B/E6C must be enabled using the **DATADECODESIGNAL** command the specific signal.

**Message ID: 2239**

**Log Type: Asynch**

**Recommended Input:**

```
log galcnavrawpage onnew
```

**Abbreviated ASCII Example:**

```
<GALCNAVRAWPAGE USB1 0 49.5 SATTIME 1997 145162.000 02040020 ab53 32768
<     319 30 2761 2b26bcef0c04f6711bf86137086a12c14f87c07b4c6aa4de04bceb8612c34
c691bfabceceb86bced4f851bfb0c074c68613604bff48448d33487
```

| Field | Field Type | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | GALCNAVRAWPAGE header | Log header. See Messages for more information. | - | H | 0 |
| 2 | signal channel | Signal channel providing the bits | Ulong | 4 | H |
| 3 | PRN | Satellite PRN number | Ulong | 4 | H+4 |
| 4 | Page ID | The page ID | Ulong | 4 | H+8 |
| 5 | data | Raw CNAV page data | HEX [58] | 58 | H+12 |
| 6 | xxxx | 32-bit CRC (ASCII and Binary only) | Hex | 4 | h+70 |
| 7 | [CR][LF] | Sentence terminator (ASCII only) | - | - | - |

### 2.3.5 INSATTX

#### Inertial Attitude – Extended

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7, SPAN CPT7, SMART7-S, SMART7-SI**

This log includes the information from the **INSATT** log, as well as information about the attitude standard deviation. The position type and solution status fields indicate whether or not the corresponding data is valid.

The INSATTX log is a large log and is not recommend for high rate logging.

If you want to use high rate logging, log the **INSATTS** log at a high rate and the **INSSTDEVS** log ontime 1.

**Message ID: 1457**

**Log Type: Synch**

**Recommended Input:**

```
log insattxa ontime 1
```

**ASCII Example:**

```
#INSATTXA,COM4,0,48.5,FINESTEERING,2088,169973.000,02004020,5d25,15823;
INS_SOLUTION_GOOD,INS_PSRSP,-0.064981993,0.506340505,90.280586875,0.5448,
0.5359,0.0834,13000044,0*15aeac66
```

| Field | Field Type | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | INSATTX Header | Log header. See Messages for more information. | | H | 0 |
| 2 | INS Status | Solution status<br>See Table: Inertial Solution Status | Enum | 4 | H |
| 3 | Pos Type | Position type<br>See Table: Position or Velocity Type | Enum | 4 | H+4 |
| 4 | Roll | Roll in Local Level (degrees) | Double | 8 | H+8 |
| 5 | Pitch | Pitch in Local Level (degrees) | Double | 8 | H+16 |
| 6 | Azimuth | Azimuth in Local Level (degrees)<br>This is the inertial azimuth calculated from the IMU gyros and the SPAN filters. | Double | 8 | H+24 |
| 7 | Roll σ | Roll standard deviation (degrees) | Float | 4 | H+32 |
| 8 | Pitch σ | Pitch standard deviation (degrees) | Float | 4 | H+36 |
| 9 | Azimuth σ | Azimuth standard deviation (degrees) | Float | 4 | H+40 |

| Field | Field Type | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 10 | Ext sol stat | Extended solution status<br>See *Table 24: Extended Solution Status* below | Hex | 4 | H+44 |
| 11 | Time Since Update | Elapsed time since the last ZUPT or position update (seconds) | Ushort | 2 | H+48 |
| 12 | xxxx | 32-bit CRC (ASCII and Binary only) | Hex | 4 | H+50 |
| 13 | [CR][LF] | Sentence terminator (ASCII only) | - | - | - |

**Table 24: Extended Solution Status**

| Nibble | Bit | Mask | Description | Range Value |
|---|---|---|---|---|
| N0 | 0 | 0x00000001 | Position update | 0 = Unused<br>1 = Used |
| | 1 | 0x00000002 | Phase update | 0 = Unused<br>1 = Used |
| | 2 | 0x00000004 | Zero velocity update | 0 = Unused<br>1 = Used |
| | 3 | 0x00000008 | Wheel sensor update | 0 = Unused<br>1 = Used |
| N1 | 4 | 0x00000010 | ALIGN (heading) update | 0 = Unused<br>1 = Used |
| | 5 | 0x00000020 | External position update | 0 = Unused<br>1 = Used |
| | 6 | 0x00000040 | INS solution convergence flag | 0 = Not converged<br>1 = Converged |
| | 7 | 0x00000080 | Doppler update | 0 = Unused<br>1 = Used |
| N2 | 8 | 0x00000100 | Pseudorange update | 0 = Unused<br>1 = Used |
| | 9 | 0x00000200 | Velocity update | 0 = Unused<br>1 = Used |
| | 10 | 0x00000400 | Reserved | |
| | 11 | 0x00000800 | Dead reckoning update | 0 = Unused<br>1 = Used |

| Nibble | Bit | Mask | Description | Range Value |
|---|---|---|---|---|
| N3 | 12 | 0x00001000 | Phase wind up update | 0 = Unused<br>1 = Used |
| | 13 | 0x00002000 | Course over ground update | 0 = Unused<br>1 = Used |
| | 14 | 0x00004000 | External velocity update | 0 = Unused<br>1 = Used |
| | 15 | 0x00008000 | External attitude update | 0 = Unused<br>1 = Used |
| N4 | 16 | 0x00010000 | External heading update | 0 = Unused<br>1 = Used |
| | 17 | 0x00020000 | External height update | 0 = Unused<br>1 = Used |
| | 18 | 0x00040000 | Reserved | |
| | 19 | 0x00080000 | Reserved | |
| N5 | 20 | 0x00100000 | Rover position update | 0 = Unused<br>1 = Used |
| | 21 | 0x00200000 | Rover position update type | 0 = Non-RTK update<br>1 = RTK integer update |
| | 22 | 0x00400000 | Reserved | |
| | 23 | 0x00800000 | Reserved | |
| N6 | 24 | 0x01000000 | Turn on biases estimated | 0 = Static turn-on biases not estimated (starting from zero)<br>1 = Static turn-on biases estimated |
| | 25 | 0x02000000 | Alignment direction verified | 0 = Not verified<br>1 = Verified |
| | 26 | 0x04000000 | Alignment Indication 1 | 0 = Not set, 1 = Set<br>Refer to *Table 25: Alignment Indication* on the next page |
| | 27 | 0x08000000 | Alignment Indication 2 | 0 = Not set, 1 = Set<br>Refer to *Table 25: Alignment Indication* on the next page |

| Nibble | Bit | Mask | Description | Range Value |
|---|---|---|---|---|
| N7 | 28 | 0x10000000 | Alignment Indication 3 | 0 = Not set, 1 = Set<br>Refer to *Table 25: Alignment Indication* below |
| | 29 | 0x20000000 | NVM Seed Indication 1 | 0 = Not set, 1 = Set<br>Refer to *Table 26: NVM Seed Indication* below |
| | 30 | 0x40000000 | NVM Seed Indication 2 | 0 = Not set, 1 = Set<br>Refer to *Table 26: NVM Seed Indication* below |
| | 31 | 0x80000000 | NVM Seed Indication 3 | 0 = Not set, 1 = Set<br>Refer to *Table 26: NVM Seed Indication* below |

**Table 25: Alignment Indication**

| Bits 28-26 Values[1] | Hex Value | Completed Alignment Type |
|---|---|---|
| 000 | 0x00 | Incomplete Alignment |
| 001 | 0x01 | Static |
| 010 | 0x02 | Kinematic |
| 011 | 0x03 | Dual Antenna |
| 100 | 0x04 | User Command |
| 101 | 0x05 | NVM Seed |

**Table 26: NVM Seed Indication**

| Bit 31-29 Values[2] | Hex Value | NVM Seed Type |
|---|---|---|
| 000 | 0x00 | NVM Seed Inactive |
| 001 | 0x01 | Seed stored in NVM is invalid |
| 010 | 0x02 | NVM Seed failed validation check |
| 011 | 0x03 | NVM Seed is pending validation (awaiting GNSS) |
| 100 | 0x04 | NVM Seed Injected (includes error model data) |
| 101 | 0x05 | NVM Seed data ignored due to a user-commanded filter reset or configuration change |
| 110 | 0x06 | NVM Seed error model data injected |

1 The values from left to right are Bit 28, Bit 27 and Bit 26.

2 The values from left to right are Bit 31, Bit 30 and Bit 29.

### 2.3.6 INSCALSTATUS

#### Offset calibration status

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7, SPAN CPT7, SMART7-S, SMART7-SI**

This log reports the status and estimated values of the currently running offset calibration.

**Message ID: 1961**

**Log Type: Asynch**

**Abbreviated ASCII Syntax:**

```
log inscalstatus onchanged
```

**ASCII Example:**

```
#INSCALSTATUSA,COM1,0,80.0,FINESTEERING,1880,317815.012,02000000,a4f2,32768;RBV
,0.0000,-180.0000,-90.0000,45.0000,45.0000,45.0000,INS_CONVERGING,1*e0b3152d
```

| Field | Field Type | Description | Binary Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | INSCALSTATUS header | Log header. See Messages for more information. | - | H | 0 |
| 2 | Offset Type | Type of offset (see *Table 27: Offset Type* on the next page). | Enum | 4 | H |
| 3 | X axis offset | IMU body frame X-axis offset (m/degrees). | Float | 4 | H+4 |
| 4 | Y axis offset | IMU body frame Y-axis offset (m/degrees). | Float | 4 | H+8 |
| 5 | Z axis offset | IMU body frame Z-axis offset (m/degrees). | Float | 4 | H+12 |
| 6 | X uncertainty | IMU body frame X-axis offset uncertainty (m/degrees). | Float | 4 | H+16 |
| 7 | Y uncertainty | IMU body frame Y-axis offset uncertainty (m/degrees). | Float | 4 | H+20 |
| 8 | Z uncertainty | IMU body frame Z-axis offset uncertainty (m/degrees). | Float | 4 | H+24 |
| 9 | Source Status | Source from which offset values originate (see *Table 28: Source Status* on the next page). | Enum | 4 | H+28 |
| 10 | Multi-line Calibration Count | Counter for number of completed calibrations cumulatively averaged. | Ulong | 4 | H+32 |
| 11 | xxxx | 32-bit CRC (ASCII and Binary only). | Hex | 4 | H+36 |
| 12 | [CR][LF] | Sentence terminator (ASCII only). | - | - | - |

Units for the axis offset and uncertainty values (fields 3-8) are in meters for translational offset components and degrees for rotational offset components.

**Table 27: Offset Type**

| Binary | ASCII | Description |
|---|---|---|
| 1 | ANT1 | Primary IMU to antenna lever arm |
| 8 | ALIGN | Align offset |
| 11 | RBV | IMU body to vehicle offset |

**Table 28: Source Status**

| Binary | ASCII | Description |
|---|---|---|
| 1 | FROM_NVM | Offset values originate from saved parameters in NVM |
| 2 | CALIBRATING | Offset values originate from a currently running calibration process |
| 3 | CALIBRATED | Offset values originate from a completed calibration process |
| 4 | FROM_ COMMAND | Offset values originate from a user command |
| 5 | RESET | Offset values originate from a system reset |
| 6 | FROM_DUAL_ ANT | Offset values originate from a dual antenna Align solution |
| 7 | INS_ CONVERGING | Offset values originate from initial input values. Calibration process on hold until INS solution is converged. |
| 8 | INSUFFICIENT_ SPEED | Offset values originate from a currently running calibration process. Further estimation on hold due to insufficient speed. |
| 9 | HIGH_ ROTATION | Offset values originate from a currently running calibration process. Further estimation on hold due to high vehicle rotations. |

### 2.3.7 INSCONFIG

**Determine required settings for post-processing or system analysis**

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7, SPAN CPT7, SMART7-S, SMART7-SI**

This log is the single message required to determine all required settings for post-processing or system analysis. This log is asynchronous and published for any change to the included fields. It is intended to be recorded occasionally though it could be updated frequently at system startup. The **INSCONFIG** log may also be logged synchronously with the ONTIME trigger, but this log should **NOT** be requested at a high rate (limited to 1 Hz).

**Message ID: 1945**

**Log Type: Asynch**

**Recommended Input:**

```
log insconfig onchanged
```

**ASCII Example:**

```
#INSCONFIGA,COM1,0,71.0,COARSESTEERING,1931,517331.006,02400000,6d7a,32768;
EPSON_G320,6,50,20,DEFAULT,00ffd1bf,AUTOMATIC,ROVER,FALSE,00000000,0,0,0,0,0,0,
0,0,0,1,ANT1,IMUBODY,0.0540,0.0699,-0.0346,0.0200,0.0200,0.0200,FROM_NVM,1,RBV,
IMUBODY,180.0000,0.0000,90.0000,5.0000,5.0000,5.0000,FROM_COMMAND*b1233ac4
```

| Field | Field Type | Description | Binary Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | INSCONFIG Header | Command header. See Messages for more information. | - | H | 0 |
| 2 | IMU Type | IMU type. See *Table 7: IMU Type* on page 28. | Enum | 4 | H |
| 3 | Mapping | Mapping / Orientation | Uchar | 1 | H+4 |
| 4 | Initial Alignment Velocity | Minimum Alignment Velocity entered by the user (see the **SETALIGNMENTVEL** command).<br>**Note**: Velocity (m/s) is scaled by 10 for 10cm/s precision | Uchar | 1 | H+5 |
| 5 | Heave Window | Length of the heave window in seconds (if set using the **SETHEAVEWINDOW** command) | Ushort | 2 | H+6 |
| 6 | Profile | Profile setting (see the **SETINSPROFILE** command on page 63) | Enum | 4 | H+8 |
| 7 | INS Enabled Updates | Enabled INS updates (see *Table 31: INS Update Values* on page 90) | Hex | 4 | H+12 |
| 8 | Alignment Mode | Alignment mode configured on the system (see the **ALIGNMENTMODE** command) | Enum | 4 | H+16 |

| Field | Field Type | Description | Binary Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 9 | Relative INS Output Frame | The user specified output frame of the Relative INS Vector (see **SETRELINSOUTPUTFRAME** command)<br>If not specified, the default value appears. | Enum | 4 | H+20 |
| 10 | Relative INS Output Direction | The User specified Output direction of the Relative INS Vector (From or To Master-Rover) (see the **SETRELINSOUTPUTFRAME** command).<br>If not specified, the default value appears. TRUE if From Master, FALSE (Default) if From Rover | Bool | 4 | H+24 |
| 11 | INS Receiver Status | Lower byte- INS Reset. Corresponds numerically to the INS Reset as described by the INSResetEnum<br>Second byte-<br>= 0x01 if an IMU Communication Error (Receiver status bit 17).<br>= 0x00 otherwise.<br>Other values are reserved for future use.<br>Upper 2 bytes - reserved. | Hex | 4 | H+28 |
| 12 | INS Seed Enabled | INS Seed Enable setting<br>(see the **INSSEED** command on page 36)<br>Enabled = 1, Disabled = 0 | Uchar | 1 | H+32 |
| 13 | INS Seed Validation | INS Seed Validation setting<br>(see the **INSSEED** command on page 36) | Uchar | 1 | H+33 |
| 14 | Reserved 1 | | N/A | 2 | H+34 |
| 15 | Reserved 2 | | N/A | 4 | H+36 |
| 16 | Reserved 3 | | N/A | 4 | H+40 |
| 17 | Reserved 4 | | N/A | 4 | H+44 |
| 18 | Reserved 5 | | N/A | 4 | H+48 |
| 19 | Reserved 6 | | N/A | 4 | H+52 |
| 20 | Reserved 7 | | N/A | 4 | H+56 |
| 21 | Number of Translations | Number of translation entries to follow | Ulong | 4 | H+60 |
| 22 | Translation | Translation to follow (see *Table 19: Translation Offset Types* on page 67) | Enum | 4 | variable |
| 23 | Frame | Frame of translation (IMUBODY or VEHICLE) | Enum | 4 | variable |
| 24 | X Offset | X Offset | Float | 4 | variable |

| Field | Field Type | Description | Binary Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 25 | Y Offset | Y Offset | Float | 4 | variable |
| 26 | Z Offset | Z Offset | Float | 4 | variable |
| 27 | X Uncertainty | X Uncertainty | Float | 4 | variable |
| 28 | Y Uncertainty | Y Uncertainty | Float | 4 | variable |
| 29 | Z Uncertainty | Z Uncertainty | Float | 4 | variable |
| 30 | Translation Source | Source of translation (see *Table 28: Source Status* on page 80) | Enum | 4 | variable |
| | Next Translation | | | | |
| variable | Number of Rotations | Number of rotation entries to follow | Ulong | 4 | variable |
| variable | Rotation | Rotation to follow (see *Table 18: Rotational Offset Types* on page 65) | Enum | 4 | variable |
| variable | Frame | Frame of rotation (IMUBODY or VEHICLE) | Enum | 4 | variable |
| variable | X Rotation | X Rotation | Float | 4 | variable |
| variable | Y Rotation | Y Rotation | Float | 4 | variable |
| variable | Z Rotation | Z Rotation | Float | 4 | variable |
| variable | X Rotation Std Dev | X Rotation offset standard deviation (degrees) | Float | 4 | variable |
| variable | Y Rotation STD Dev | Y Rotation offset standard deviation (degrees) | Float | 4 | variable |
| variable | Z Rotation STD Dev | Z Rotation offset standard deviation (degrees) | Float | 4 | variable |
| variable | Rotation Source | Source of rotation (see *Table 28: Source Status* on page 80) | Enum | 4 | variable |
| | Next Rotation | | | | |
| variable | xxxx | 32-bit CRC (ASCII and Binary only) | Hex | 4 | variable |
| | [CR][LF] | Sentence terminator (ASCII only) | - | - | - |

### 2.3.8 INSSTDEV

#### INS PVA standard deviations

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7, SPAN CPT7, SMART7-S, SMART7-SI**

This log displays the INS PVA standard deviations.

**Message ID: 2051**

**Log Type: Synch**

**Abbreviated ASCII Syntax:**

```
log insstdev ontime 1
```

**ASCII Example:**

```
#INSSTDEVA,COM1,0,78.0,FINESTEERING,1907,233990.000,02000020,3e6d,32768;0.4372,
0.3139,0.7547,0.0015,0.0015,0.0014,3.7503,3.7534,5.1857,26000005,0,0,01ffd1bf,0
*3deca7d2
```

| Field | Field Type | Description | Binary Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | INSSTDEV Header | Log header. See Messages for more information. | - | H | 0 |
| 2 | Latitude σ | Latitude standard deviation (m) | Float | 4 | H |
| 3 | Longitude σ | Longitude standard deviation (m) | Float | 4 | H+4 |
| 4 | Height σ | Height standard deviation (m) | Float | 4 | H+8 |
| 5 | North Velocity σ | North velocity standard deviation (m/s) | Float | 4 | H+12 |
| 6 | East Velocity σ | East velocity standard deviation (m/s) | Float | 4 | H+16 |
| 7 | Up Velocity σ | Up velocity standard deviation (m/s) | Float | 4 | H+20 |
| 8 | Roll σ | Roll standard deviation (degrees) | Float | 4 | H+24 |
| 9 | Pitch σ | Pitch standard deviation (degrees) | Float | 4 | H+28 |
| 10 | Azimuth σ | Azimuth standard deviation (degrees) | Float | 4 | H+32 |
| 11 | Ext sol stat | Extended solution status<br>See *Table 24: Extended Solution Status* on page 76 | Ulong | 4 | H+36 |
| 12 | Time Since Update | Elapsed time since the last ZUPT or position update (seconds) | Ushort | 2 | H+40 |
| 13 | Reserved | | Ushort | 2 | H+42 |

| Field | Field Type | Description | Binary Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 14 | Reserved | | Ulong | 4 | H+44 |
| 15 | Reserved | | Ulong | 4 | H+48 |
| 16 | xxxx | 32-bit CRC (ASCII and Binary only). | Hex | 4 | H+52 |
| 17 | [CR][LF] | Sentence terminator (ASCII only). | - | - | - |

### 2.3.9 INSSTDEVS

#### Short INS PVA standard deviations

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7, SPAN CPT7, SMART7-S, SMART7-SI**

This log is the short header version of the **INSSTDEV** log (see page 84).

**Message ID: 2052**

**Log Type: Synch**

**Abbreviated ASCII Syntax:**

```
log insstdevs ontime 1
```

**ASCII Example:**

```
%INSSTDEVSA,1907,233990.000;0.4372,0.3139,0.7547,0.0015,0.0015,0.0014,3.7503,3.
7534,5.1857,26000005,0,0,01ffd1bf,0*2c967ced
```

| Field | Field Type | Description | Binary Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | INSSTDEV Header | Log header. See Messages for more information. | - | H | 0 |
| 2 | Latitude σ | Latitude standard deviation (m) | Float | 4 | H |
| 3 | Longitude σ | Longitude standard deviation (m) | Float | 4 | H+4 |
| 4 | Height σ | Height standard deviation (m) | Float | 4 | H+8 |
| 5 | North Velocity σ | North velocity standard deviation (m/s) | Float | 4 | H+12 |
| 6 | East Velocity σ | East velocity standard deviation (m/s) | Float | 4 | H+16 |
| 7 | Up Velocity σ | Up velocity standard deviation (m/s) | Float | 4 | H+20 |
| 8 | Roll σ | Roll standard deviation (degrees) | Float | 4 | H+24 |
| 9 | Pitch σ | Pitch standard deviation (degrees) | Float | 4 | H+28 |
| 10 | Azimuth σ | Azimuth standard deviation (degrees) | Float | 4 | H+32 |
| 11 | Ext sol stat | Extended solution status<br>See *Table 24: Extended Solution Status* on page 76 | Ulong | 4 | H+36 |
| 12 | Time Since Update | Elapsed time since the last ZUPT or position update (seconds) | Ushort | 2 | H+40 |
| 13 | Reserved | | Ushort | 2 | H+42 |
| 14 | Reserved | | Ulong | 4 | H+44 |

| Field | Field Type | Description | Binary Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 15 | Reserved | | Ulong | 4 | H+48 |
| 16 | xxxx | 32-bit CRC (ASCII and Binary only). | Hex | 4 | H+52 |
| 17 | [CR][LF] | Sentence terminator (ASCII only). | - | - | - |

## 2.3.10 INSUPDATESTATUS

### INS Update Status

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7, SPAN CPT7, SMART7-S, SMART7-SI**

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

**Message ID: 1825**

**Log Type: Asynch**

**Recommended Input:**

```
log insupdatestatus onchanged
```

**ASCII Example:**

```
#INSUPDATESTATUSA,COM2,0,76.0,FINESTEERING,1934,149288.000,02000000,78f1,32768;
SINGLE,0,0,0,INACTIVE,INACTIVE,00000005,00ffd1bf,0,0*d6b7ee02
```

| Field | Field Type | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | INSUPDATE STATUS Header | Log header. See Messages for more information. | - | H | 0 |
| 2 | PosType | Type of GNSS solution used for the last INS filter update.<br>See Table: Position or Velocity Type | Enum | 4 | H |
| 3 | NumPSR | Number of raw pseudorange observations used in the last INS filter update. | Integer | 4 | H+4 |
| 4 | NumADR | Number of raw phase observations used in the last INS filter update. | Integer | 4 | H+8 |
| 5 | NumDOP | Number of raw doppler observations used in the last INS filter update. | Integer | 4 | H+12 |
| 6 | DMI Update Status | Distance measurement instrument (wheel sensor) status.<br>See *Table 29: DMI Update Status* on the next page. | Enum | 4 | H+16 |
| 7 | Heading Update Status | Status of the heading update during the last INS filter update.<br>See *Table 30: Heading Update Values* on the next page | Enum | 4 | H+20 |
| 8 | Ext sol stat | Extended solution status<br>See *Table 24: Extended Solution Status* on page 76 | Hex | 4 | H+24 |

| Field | Field Type | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 9 | INS Enabled Updates | Enabled INS updates.<br>See *Table 31: INS Update Values* on the next page | Hex | 4 | H+28 |
| 10 | Reserved | | Ulong | 4 | H+32 |
| 11 | Reserved | | Ulong | 4 | H+36 |
| 12 | xxxx | 32-bit CRC (ASCII, Binary and Short Binary only) | Hex | 4 | H+40 |
| 13 | [CR][LF] | Sentence terminator (ASCII only) | - | - | - |

**Table 29: DMI Update Status**

| Binary | ASCII | Description |
|---|---|---|
| 0 | INACTIVE | The DMI sensor is not connected. |
| 1 | ACTIVE | The DMI sensor is connected but not used. |
| 2 | USED | The DMI sensor is connected and used in the INS solution. |
| 3 | RESERVED | Reserved |
| 4 | BAD_MISC | The measurement of the DMI sensor resulted in bad misclosure and was rejected. |
| 5 | HIGH_ ROTATION | The measurement of the DMI sensor was not used because the system is undergoing high rotation. |
| 6 | DISABLED | DMI updates are disabled. |
| 7 | ZUPT | The DMI update reported zero velocity. |

**Table 30: Heading Update Values**

| Binary | ASCII | Description |
|---|---|---|
| 0 | INACTIVE | A heading update was not available. |
| 1 | ACTIVE | Heading updates are running, but the epoch is not used as an update. When all other rejection criteria pass, a heading update will still only be applied once every 5 seconds (20 seconds when stationary). |
| 2 | USED | The update for that epoch was taken. |
| 5 | HEADING_ UPDATE_ BAD_MISC | Heading updates are running, but was not performed this epoch due to a large dis-agreement with filter estimates. |

**Table 31: INS Update Values**

| Nibble | Bit | Mask | Description | Range Value |
|---|---|---|---|---|
| N0 | 0 | 0x00000001 | Position update | 0 = Disabled<br>1 = Enabled |
| | 1 | 0x00000002 | Phase update | 0 = Disabled<br>1 = Enabled |
| | 2 | 0x00000004 | Zero velocity update | 0 = Disabled<br>1 = Enabled |
| | 3 | 0x00000008 | Wheel sensor update | 0 = Disabled<br>1 = Enabled |
| N1 | 4 | 0x00000010 | ALIGN (heading) update | 0 = Disabled<br>1 = Enabled |
| | 5 | 0x00000020 | External position update | 0 = Disabled<br>1 = Enabled |
| | 6 | 0x00000040 | Reserved | |
| | 7 | 0x00000080 | Doppler update | 0 = Disabled<br>1 = Enabled |
| N2 | 8 | 0x00000100 | Pseudorange update | 0 = Disabled<br>1 = Enabled |
| | 9 | 0x00000200 | Velocity update | 0 = Disabled<br>1 = Enabled |
| | 10 | 0x00000400 | Reserved | |
| | 11 | 0x00000800 | Dead reckoning update | 0 = Disabled<br>1 = Enabled |
| N3 | 12 | 0x00001000 | Phase wind up update | 0 = Disabled<br>1 = Enabled |
| | 13 | 0x00002000 | Course over ground update | 0 = Disabled<br>1 = Enabled |
| | 14 | 0x00004000 | External velocity update | 0 = Disabled<br>1 = Enabled |
| | 15 | 0x00008000 | External attitude update | 0 = Disabled<br>1 = Enabled |

| Nibble | Bit | Mask | Description | Range Value |
|---|---|---|---|---|
| N4 | 16 | 0x00010000 | External heading update | 0 = Disabled<br>1 = Enabled |
| | 17 | 0x00020000 | External height update | 0 = Disabled<br>1 = Enabled |
| | 18 | 0x00040000 | Reserved | |
| | 19 | 0x00080000 | Reserved | |

### 2.3.11 ITBANDPASSBANK

#### Allowable band pass filter configurations

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7, SPAN CPT7, SMART7, SMART2**

The ITBANDPASSBANK log provides information on the allowable configurations for each frequency when applying a bandpass filter. The current filters in use can be seen with the **ITFILTTABLE** log on page 96.

**Message ID: 2022**

**Log Type: Asynch**

**Recommended Input:**

```
log itbandpassbanka once
```

**Abbreviated ASCII Example:**

```
<ITBANDPASSBANK USB1 0 87.5 FINESTEERING 1933 346809.694 12000020 fb2e 14137
5
GPSL5 1164.3750 1173.1250 1178.1250 1186.8750 0.05
GALILEOE5B 1195.6250 1204.3750 1209.3750 1218.1250 0.05
BEIDOUB1 1551.2500 1560.0000 1565.0000 1573.7500 0.05
BEIDOUB2 1195.6250 1204.3750 1209.3750 1218.1250 0.05
QZSSL5 1164.3750 1173.1250 1178.1250 1186.8750 0.05
```

| Field | Field Type | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | ITBANDPASSBANK header | Log header. See Messages for more information. | - | H | 0 |
| 2 | # of entries | Number of entries to follow | Ulong | 4 | H |
| 3 | frequency | See *Table 16: Frequency Types* on page 44 | Enum | 4 | H+4 |
| 4 | min lower frequency cutoff | The minimum frequency cutoff at the lower end (MHz) | Float | 4 | H+8 |
| 5 | max lower frequency cutoff | The maximum frequency cutoff at the lower end (MHz) | Float | 4 | H+12 |
| 6 | min upper frequency cutoff | The minimum frequency cutoff at the upper end (MHz) | Float | 4 | H+16 |
| 7 | max upper frequency cutoff | The maximum frequency cutoff at the upper end (MHz) | Float | 4 | H+20 |
| 8 | frequency step | The minimum cut off frequency resolution (MHz) | Float | 4 | H+24 |
| 9 | Next entry offset = H + 4 + (#entries * 24) | | | | |

| Field | Field Type | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 10 | xxxx | 32-bit CRC (ASCII and Binary only) | Ulong | 4 | H+4+ (#entries * 24) |
| 11 | [CR][LF] | Sentence terminator (ASCII only) | - | - | - |

## 2.3.12 ITDETECTSTATUS

### Interference detection status

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7, SPAN CPT7, SMART7, SMART2**

This log lists all of the detected interferences from all active paths where interference detection is enabled.

This log should be used with the **onchanged** trigger only.

**Message ID: 2065**

**Log Type: Asynch**

**Recommended Input:**

```
log itdetectstatusa onchanged
```

**ASCII Example**

```
#ITDETECTSTATUSA,USB2,0,74.0,FINESTEERING,1982,430605.267,0200c000,7fdb,32768;
3,
L1,STATISTICANALYSIS,-0.718,29.167,0.126,12.797,00000000,00000000,00000000,
L2,SPECTRUMANALYSIS,1249.961,71.191,-56.769,-132.907,00000000,00000000,
00000000,
L2,SPECTRUMANALYSIS,1289.512,1.978,-75.967,-138.493,00000000,00000000,
00000000*5e83b175
```

| Field | Field Type | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | ITDETECTSTATUS header | Log header. See Messages for more information. | - | H | 0 |
| 2 | # of entries | Number of interferences to follow | Ulong | 4 | H |
| 3 | RF Path | RF path for this entry.<br>2 = L1<br>3 = L2<br>5 = L5 | Enum | 4 | H+4 |
| 4 | Interference detection type | Interference detection type for this entry.<br>0 = SPECTRALANALYSIS<br>1 = STATISTICALANALYSIS | Enum | 4 | H+8 |

| Field | Field Type | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 5 | Parameter 1 | The first parameter of the interference.<br>For SPECTRALANALYSIS type, this is the center frequency in MHz.<br>For STATISTICALANALYSIS type, this is reserved. | Float | 4 | H+12 |
| 6 | Parameter 2 | The second parameter of the interference.<br>For SPECTRALANALYSIS type, this is the bandwidth in MHz.<br>For STATISTICALANALYSIS type, this is reserved. | Float | 4 | H+16 |
| 7 | Parameter 3 | The third parameter of the interference.<br>For SPECTRALANALYSIS type, this is the estimated power in dBm of the interference.<br>For STATISTICALANALYSIS type, this is reserved. | Float | 4 | H+20 |
| 8 | Parameter 4 | The fourth parameter of the interference.<br>For SPECTRALANALYSIS type, this is the highest estimated power spectrum density in dBmHz of the interference.<br>For STATISTICALANALYSIS type this is reserved. | Float | 4 | H+24 |
| 9 | Reserved 1 | Reserved | Ulong | 4 | H+28 |
| 10 | Reserved 2 | Reserved | Ulong | 4 | H+32 |
| 11 | Reserved 3 | Reserved | Ulong | 4 | H+36 |
| 12 | Next interference signal offset = H + 4 + (#entries * 36) | | | | |
| 13 | xxxx | 32-bit CRC (ASCII and Binary only) | Ulong | 4 | H=4+ (#entries * 36) |
| 14 | [CR][LF] | Sentence terminator (ASCII only) | - | - | - |

## 2.3.13 ITFILTTABLE

### Filter configuration for each frequency

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7, SPAN CPT7, SMART7, SMART2**

The ITFILTTABLE log contains the filter configuration summary for each frequency. It lists which bandpass or notch filters are enabled and how each is configured.

**Message ID: 1991**

**Log Type: Asynch**

**Recommended Input:**

```
log itfilttablea once
```

**ASCII Example:**

```
#ITFILTTABLEA,USB2,0,80.5,FINESTEERING,1923,232588.825,12000000,35d0,32768;
13,
GPSL1,8,CIC3,00000001,DISABLE,0.0000,0.0000,1,
ENABLE,PF0,NOTCHFILTER,1572.2500,1577.7500,1.000,
GPSL2,4,CIC3,00000000,DISABLE,0.0000,0.0000,0,
GLONASSL1,9,CIC3,00000000,DISABLE,0.0000,0.0000,0,
GLONASSL2,5,CIC3,00000000,DISABLE,0.0000,0.0000,0,
GPSL5,0,CIC3,00000000,DISABLE,0.0000,0.0000,0,
…
QZSSL1,8,CIC3,00000001,DISABLE,0.0000,0.0000,1,
ENABLE,PF0,NOTCHFILTER,1572.2500,1577.7500,1.000,
QZSSL2,4,CIC3,00000000,DISABLE,0.0000,0.0000,0,
QZSSL5,0,CIC3,00000000,DISABLE,0.0000,0.0000,0*3ca84167
```

| Field | Field Type | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | ITFILTTABLE header | Log header. See Messages for more information. | - | H | 0 |
| 2 | # entries | Number of records with information to follow | Ulong | 4 | H |
| 3 | frequency | The frequency at which the filter is applied. See *Table 16: Frequency Types* on page 44 | Enum | 4 | H+4 |
| 4 | Encoder ID | ID of the digital path used by this frequency | Ulong | 4 | H+8 |
| 5 | DDC filter type | The DDC filter type (see *Table 32: DDC Filter Type* on the next page) | Enum | 4 | H+12 |
| 6 | status word | Filter warning limit status. Raise a warning flag if the filter is placed too close to the center frequency of the GNSS signal (see *Table 33: ITFILTTable Status Word* on page 98) | Ulong | 4 | H+16 |

| Field | Field Type | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 7 | switch | Filter is enabled or disabled (see *Table 34: Filter Switches* on page 99) | Enum | 4 | H+20 |
| 8 | lower cut off frequency | Cut off frequency at the lower end (MHz) | Float | 4 | H+24 |
| 9 | upper cut off frequency | Cut off frequency at the upper end (MHz) | Float | 4 | H+28 |
| 10 | # prog filters | Number of programmable filters applied | Ulong | 4 | H+32 |
| 11 | switch | Filter is enabled or disabled (see *Table 34: Filter Switches* on page 99) | Enum | 4 | H+36 |
| 12 | prog filter ID | The programmable filter ID (see *Table 13: Programmable Filter ID* on page 42) | Enum | 4 | H+40 |
| 13 | mode | Programmable filter mode (notch filter or bandpass) (see *Table 14: Programmable Filter Mode* on page 42) | Enum | 4 | H+44 |
| 14 | lower cut off frequency | Cut off frequency at the lower end (MHz) | Float | 4 | H+48 |
| 15 | upper cut off frequency | Cut off frequency at the upper end (MHz) | Float | 4 | H+52 |
| 16 | notch width | Width of notch filter (MHz) | Float | 4 | H+56 |
| 17 | Next programmable filter – variable binary offset | | | | |
| 18 | Next frequency – variable binary offset | | | | |
| 19 | xxxx | 32-bit CRC (ASCII and Binary only) | Ulong | 4 | variable |
| 20 | [CR][LF] | Sentence terminator (ASCII only) | - | - | - |

**Table 32: DDC Filter Type**

| Binary | ASCII |
|---|---|
| 0 | PASSTHROUGH |
| 1 | CIC1 |
| 2 | CIC2 |
| 3 | CIC3 |
| 4 | HALFBAND |

**Table 33: ITFILTTable Status Word**

| Nibble | Bit | Mask | Description | Range Value |
|---|---|---|---|---|
| N0 | 0 | 0x00000001 | First enabled filter | 0 = Within acceptable limit<br>1 = Warning |
| | 1 | 0x00000002 | | |
| | 2 | 0x00000004 | | |
| | 3 | 0x00000008 | | |
| N1 | 4 | 0x00000010 | | |
| | 5 | 0x00000020 | | |
| | 6 | 0x00000040 | | |
| | 7 | 0x00000080 | | |
| N2 | 8 | 0x00000100 | Second enabled filter | 0 = Within acceptable limit<br>1 = Warning |
| | 9 | 0x00000200 | | |
| | 10 | 0x00000400 | | |
| | 11 | 0x00000800 | | |
| N3 | 12 | 0x00001000 | | |
| | 13 | 0x00002000 | | |
| | 14 | 0x00004000 | | |
| | 15 | 0x00008000 | | |
| N4 | 16 | 0x00010000 | Third enabled filter | 0 = Within acceptable limit<br>1 = Warning |
| | 17 | 0x00020000 | | |
| | 18 | 0x00040000 | | |
| | 19 | 0x00080000 | | |
| N5 | 20 | 0x00100000 | | |
| | 21 | 0x00200000 | | |
| | 22 | 0x00400000 | | |
| | 23 | 0x00800000 | | |

| Nibble | Bit | Mask | Description | Range Value |
|---|---|---|---|---|
| N6 | 24 | 0x01000000 | Fourth enabled filter | 0 = Within acceptable limit<br>1 = Warning |
| | 25 | 0x02000000 | | |
| | 26 | 0x04000000 | | |
| | 27 | 0x08000000 | | |
| N7 | 28 | 0x10000000 | | |
| | 29 | 0x20000000 | | |
| | 30 | 0x40000000 | | |
| | 31 | 0x80000000 | | |

**Table 34: Filter Switches**

| Binary Value | ASCII Value | Description |
|---|---|---|
| 0 | DISABLE | Filter disabled |
| 1 | ENABLE | Filter enabled |

## 2.3.14 ITPROGFILTBANK

### Allowable filter configurations

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7, SPAN CPT7, SMART7, SMART2**

The ITPROGFILTBANK log provides information on the allowable configurations for the programmable filter for each frequency when applying either a notch filter or bandpass filter. The current filters in use can be seen with the **ITFILTTABLE** log on page 96.

**Message ID: 2023**

**Log Type: Asynch**

**Recommended Input:**

```
log itprogfiltbanka once
```

**Abbreviated ASCII Example:**

```
<ITPROGFILTBANK USB1 0 88.0 FINESTEERING 1933 346362.985 12000020 3696 14137
12
GPSL1 5
NOTCHFILTER 1563.0000 1574.0000 1576.0000 1587.0000 0.05 0.15
NOTCHFILTER 1563.7500 1573.6000 1576.4000 1586.2500 0.05 0.50
NOTCHFILTER 1564.0500 1573.3000 1576.7000 1585.9500 0.05 1.00
NOTCHFILTER 1565.7500 1571.7000 1578.3000 1584.2500 0.05 2.50
BANDPASSFILTER 1563.7500 1572.5000 1577.5000 1586.2500 0.05 0.00
GPSL2 5
NOTCHFILTER 1215.5000 1226.5000 1228.5000 1239.5000 0.05 0.15
NOTCHFILTER 1216.2500 1226.1000 1228.9000 1238.7500 0.05 0.50
NOTCHFILTER 1216.5500 1225.8000 1229.2000 1238.4500 0.05 1.00
NOTCHFILTER 1218.2500 1224.2000 1230.8000 1236.7500 0.05 2.50
BANDPASSFILTER 1216.2500 1225.0000 1230.0000 1238.7500 0.05 0.00
GLONASSL1 5
NOTCHFILTER 1589.5625 1600.5625 1602.5625 1613.5625 0.05 0.15
NOTCHFILTER 1590.3125 1600.1625 1602.9625 1612.8125 0.05 0.50
NOTCHFILTER 1590.6125 1599.8625 1603.2625 1612.5125 0.05 1.00
NOTCHFILTER 1592.3125 1598.2625 1604.8625 1610.8125 0.05 2.50
BANDPASSFILTER 1590.3125 1599.0625 1604.0625 1612.8125 0.05 0.00
...
```

| Field | Field Type | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | ITPROGFILTBANK header | Log header. See Messages for more information. | - | H | 0 |
| 2 | # entries | Number of entries to follow | Ulong | 4 | H |
| 3 | frequency | See *Table 16: Frequency Types* on page 44 | Enum | 4 | H+4 |

| Field | Field Type | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 4 | # prog filters | Number of programmable filters applied with information to follow | Ulong | 4 | H+8 |
| 5 | mode | Programmable filter mode (notch filter or bandpass)<br>(see *Table 14: Programmable Filter Mode* on page 42) | Enum | 4 | H+12 |
| 6 | min lower frequency cutoff | The minimum frequency cutoff at the lower end (MHz) | Float | 4 | H+16 |
| 7 | max lower frequency cutoff | The maximum frequency cutoff at the lower end (MHz) | Float | 4 | H+20 |
| 8 | min upper frequency cutoff | The minimum frequency cutoff at the upper end (MHz) | Float | 4 | H+24 |
| 9 | max upper frequency cutoff | The maximum frequency cutoff at the upper end (MHz) | Float | 4 | H+28 |
| 10 | frequency step | The minimum cut off frequency resolution (MHz) | Float | 4 | H+32 |
| 11 | notch width | Width of notch filter (MHz) | Float | 4 | H+36 |
| 12 | Next programmable filter – variable binary offset | | | | |
| 13 | Next frequency – variable binary offset | | | | |
| 14 | xxxx | 32-bit CRC (ASCII and Binary only) | Ulong | 4 | variable |
| 15 | [CR][LF] | Sentence terminator (ASCII only) | | | |

### 2.3.15 ITPSDFINAL

**Processed power spectral density**

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7, SPAN CPT7, SMART7, SMART2**

The ITPSDFINAL log contains the samples for the spectral analysis. The rate and size is set by the update period and the FFT size respectively when issuing the **ITSPECTRALANALYSIS** command (see page 43).

PSD samples are compressed into 2 byte samples to reduce log sizes. The range of values that can be displayed is -200 dBm to +56 dBm with a 1/256 resolution. The following steps should be performed on the PSD samples in this log to convert them back into dBm units for display purposes:

1. Divide the sample by 256.0
2. Subtract 200

The number of samples are calculated according to the following table. The maximum number of samples in one ITPSDFINAL log is 1024. That means if the number of samples is less than 1024, one log is enough to output them. However, if the number of samples is larger than 1024, more than one ITPSDFINAL log is needed. For example, in postdecimation mode with the FFT size of 8K and subcarrier integration of 5, there is one log with 1024 samples and another log with 614 samples. The output logs can be grouped together through the sequence number of the log header.

| Data Source | Number of Samples |
|---|---|
| PREDECIMATION | FFTsize/(2*subcarrier_integration ) |
| POSTDECIMATION | FFTsize/subcarrier_integration |
| POSTFILTER | FFTsize/subcarrier_integration |

As the data rate for the ITPSDFINAL log is dictated by the updateperiod parameter in the **ITSPECTRALANALYSIS** command (see page 43), do not use ONTIME to log this message. Instead use ONNEW to log ITPSDFINAL.

The pre-decimation spectrum shows the absolute power in dBm which is proportional to the resolution bandwidth (RBW). The post-decimation and post-filter spectrum shows the signal magnitude in relative power (dB).

**Message ID: 1968**

**Log Type: Asynch**

**Recommended Input:**

```
log itpsdfinala onnew
```

**ASCII Example**

```
#ITPSDFINALA,UNKNOWN,0,66.0,FINESTEERING,1891,166978.221,02040000,b79a,32768;
1310752,1531.250,195312.500,512,28033,30370,30225,29190,27254,29521,32694,
33025,28553,28902,29060,26663,30267,30054,
…
```

```
34027,38038,31082,29418,28805,27373,27869,28847,28331,31901,30251,33625,33625
*000b928d
```

| Field | Field Type | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | ITPSDFINAL header | Log header. See Messages for more information. | - | H | 0 |
| 2 | status word | Status word containing information about the configuration of the spectral analysis (see *Table 35: Spectral Analysis Status Word* below) | Ulong | 4 | H |
| 3 | frequency start | Frequency represented by first data sample (MHz) | Float | 4 | H+4 |
| 4 | resolution bandwidth | The resolution bandwidth (Hz) | Float | 4 | H+8 |
| 5 | # samples | Number of spectral density samples | Ulong | 4 | H+12 |
| 6 | sample | Power spectral density sample | Ushort | 2 | H+16 |
| 7 | Next sample = H+16+(2*#samples) | | | | |
| 8 | xxxx | 32-bit CRC (ASCII and Binary only) | Ulong | 4 | H+16+ (2*# samples) |
| 9 | [CR][LF] | Sentence terminator (ASCII only) | - | - | - |

**Table 35: Spectral Analysis Status Word**

| Nibble | Bit | Mask | Description | Range Value |
|---|---|---|---|---|
| N0 | 0 | 0x00000001 | Frequency | 0 – 20<br>See *Table 16: Frequency Types* on page 44 |
| | 1 | 0x00000002 | | |
| | 2 | 0x00000004 | | |
| | 3 | 0x00000008 | | |
| N1 | 4 | 0x00000010 | | |
| | 5 | 0x00000020 | Data Source | 0 – 3<br>See *Table 15: Data Sources for PSD Samples* on page 44 |
| | 6 | 0x00000040 | | |
| | 7 | 0x00000080 | | |

| Nibble | Bit | Mask | Description | Range Value |
|---|---|---|---|---|
| N2 | 8 | 0x00000100 | FFT Size | 0 – 6<br>See *Table 17: FFT Sizes* on page 45 |
| | 9 | 0x00000200 | | |
| | 10 | 0x00000400 | | |
| | 11 | 0x00000800 | Integration Window | 1 – 1024 samples |
| N3 | 12 | 0x00001000 | | |
| | 13 | 0x00002000 | | |
| | 14 | 0x00004000 | | |
| | 15 | 0x00008000 | | |
| N4 | 16 | 0x00010000 | | |
| | 17 | 0x00020000 | Time Average Window | 0 – 100 seconds |
| | 18 | 0x00040000 | | |
| | 19 | 0x00080000 | | |
| N5 | 20 | 0x00100000 | | |
| | 21 | 0x00200000 | | |
| | 22 | 0x00400000 | | |
| | 23 | 0x00800000 | | |
| N6 | 24 | 0x01000000 | | |
| | 25 | 0x02000000 | Reserved | |
| | 26 | 0x04000000 | | |
| | 27 | 0x08000000 | | |
| N7 | 28 | 0x10000000 | Reserved | |
| | 29 | 0x20000000 | | |
| | 30 | 0x40000000 | | |
| | 31 | 0x80000000 | | |

### 2.3.16 J1939STATUS

**Status of CAN J1939 Node**

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7, SPAN CPT7, SMART7, SMART2**

This logs reports the status of J1939 node, specifically J1939 Address Claim function (initiated using the **J1939CONFIG** command (see page 47)).

This log displays the status only for nodes that have been set.

**Message ID: 1907**

**Log Type: Asynch**

**Recommended Input:**

```
LOG J1939STATUSA ONCHANGED
```

**ASCII Examples:**

```
#J1939STATUSA,COM1,1,81.0,UNKNOWN,0,0.000,02004020,e9ce,32768;NODE1,DISABLED,0,
FE*637c7f
```

```
#J1939STATUSA,COM1,0,81.0,UNKNOWN,0,0.000,02004020,e9ce,32768;NODE2,DISABLED,0,
FE*c41af5ee
```

| Field | Field Type | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | J1939STATUS header | Log header. See Messages for more information. | - | H | 0 |
| 2 | node | J1939 Node. The node can be either NODE1 or NODE2. | Enum | 4 | H |
| 3 | status | Node status. See *Table 36: Node Status* below | Enum | 4 | H+4 |
| 4 | count | Number of attempts that were made to claim address.<br>This will be 1 when the preferred address is used and may be more if the alternate range is used. | Ulong | 4 | H+8 |
| 5 | address | Claimed CAN Address.<br>0xFE (NULL address) if the address could not be negotiated. | Uchar | 1 | H+12 |
| 6 | xxxx | 32-bit CRC (ASCII and Binary only) | Hex | 4 | H+13 |
| 7 | [CR][LF] | Sentence Terminator (ASCII only) | - | - | - |

**Table 36: Node Status**

| Value | ASCII | Description |
|---|---|---|
| 1 | DISABLED | Address claim activity is not taking place. The node does not have J1939 enabled. |

| Value | ASCII | Description |
|---|---|---|
| 2 | CLAIMING | Address claim procedure is in progress. |
| 3 | CLAIMED | Address claimed successfully. Ready for data transfer. |
| 4 | FAILED | Address claim was not successful. No further activity is taking place. |

## 2.3.17 MODELFEATURES

### States features available for current loaded model

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7, SPAN CPT7, SMART7, SMART2**

The MODELFEATURES log states the features available for the current loaded model.

To see which satellite systems are available for the current model, use the **CHANCONFIGLIST** log.

Most features have a boolean state: authorized or unauthorized. However, some have more complex licensed states with varying degrees of capability.

This log is best viewed in Abbreviated ASCII.

**Message ID: 1329**

**Log Type: Polled**

**Recommended Input:**

```
log modelfeatures once
```

**Abbreviated ASCII Example:**

```
<MODELFEATURES COM1 0 92.5 COARSESTEERING 2007 237316.648 02400000 141a 14898
<     20
<          1HZ MAX_MSR_RATE
<          0HZ MAX_POS_RATE
<          SINGLE ANTENNA
<          AUTHORIZED NTRIP
<          UNAUTHORIZED IMU
<          UNAUTHORIZED INS
<          UNAUTHORIZED MEAS_OUTPUT
<          UNAUTHORIZED DGPS_TX
<          UNAUTHORIZED RTK_TX
<          UNAUTHORIZED RTK_FLOAT
<          UNAUTHORIZED RTK_FIXED
<          UNAUTHORIZED PPP
<          UNAUTHORIZED LOW_END_POSITIONING
<          UNAUTHORIZED RAIM
<          UNAUTHORIZED ALIGN_HEADING
<          UNAUTHORIZED ALIGN_RELATIVE_POS
<          UNAUTHORIZED API
<          UNAUTHORIZED INTERFERENCE_MITIGATION
<          UNAUTHORIZED RTKASSIST
<          UNAUTHORIZED SCINTILLATION
```

| Field | Field type | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | MODELFEATURES header | Log header. See Messages for more information. | - | H | 0 |

| Field | Field type | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 2 | # Feature | Number of features in list | Ulong | 4 | H |
| 3 | Feature Status | Licensing status of feature<br>See *Table 37: Feature Status* below | Enum | 4 | H+4 |
| 4 | Feature Type | Type of feature<br>See *Table 38: Feature Type* on the next page | Enum | 4 | H+8 |
| 5... | Next feature = H+4+(# Feature x 8) | | | | |
| 6 | xxxx | 32-bit CRC (ASCII and Binary only) | Ulong | 4 | H+4+(# Feature x 8) |
| 7 | [CR][LF] | Sentence terminator (ASCII only) | - | - | - |

**Table 37: Feature Status**

| Value | Name | Description |
|---|---|---|
| 0 | AUTHORIZED | The feature is authorized |
| 1 | UNAUTHORIZED | The feature is unauthorized |
| 2 | 0Hz | Disables output of POS logs |
| 6 | 20Hz | Maximum logging rate for POS or MSR logs is 20 Hz |
| 8 | 100Hz | Maximum logging rate for POS or MSR logs is 100 Hz |
| 9 | RATE_INVALID | Option bits don't correspond to a valid rate |
| 15 | STANDARD | SPAN Standard Model |
| 20 | COMMERCIAL_MEMS | IMU Grade-Commercial MEMS |
| 21 | TACTICAL | IMU Grade-Tactical |
| 22 | HIGH_GRADE_TACTICAL | IMU Grade-High Grade Tactical |
| 23 | NAVIGATION | IMU Grade-Navigation |
| 25 | SINGLE | Single antenna |
| 26 | DUAL | Dual antenna |
| 30 | LITE | SPAN Lite Model |
| 33 | CONSUMER_MEMS | IMU Grade-Consumer MEMS |

**Table 38: Feature Type**

| Value | Name | Description |
|---|---|---|
| 0 | MAX_MSR_RATE | Maximum measurement logging rate |
| 1 | MAX_POS_RATE | Maximum position logging rate |
| 3 | MEAS_OUTPUT | Output of raw measurements (phase and pseudorange) |
| 4 | DGPS_TX | Transmission of DGPS (non RTK) corrections |
| 5 | RTK_TX | Transmission of RTK corrections |
| 6 | RTK_FLOAT | RTK float positioning |
| 7 | RTK_FIXED | RTK fixed positioning |
| 8 | RAIM | Extended RAIM |
| 9 | LOW_END_POSITIONING | GLIDE and TerraStar-L positioning |
| 10 | ALIGN_HEADING | Heading |
| 11 | ALIGN_RELATIVE_POS | Heading and Relative Positioning |
| 12 | API | Lua Scripted User Interface (formerly User Application API) |
| 15 | NTRIP | NTRIP Server/Client |
| 19 | PPP | TerraStar-C, TerraStar-C PRO, or TerraStar-X positioning |
| 20 | SCINTILLATION | Scintillation |
| 22 | INS | Inertial (SPAN) |
| 23 | IMU | IMU Grade |
| 26 | FEATURE_ INTERFERENCE_ MITIGATION | Interference Mitigation |
| 28 | ANTENNA | Number of antenna enabled on the receiver |
| 29 | GENERIC_IMU | SPAN Generic IMU Interface |
| 30 | INS_PLUS_PROFILES | SPAN Plus Profiles |
| 31 | HEAVE | SPAN Heave Option |
| 32 | RELATIVE_INS | SPAN Relative INS |
| 999 | MODEL_INVALID | If a bad model is loaded, MODELFEATURES will contain one entry: MODEL_INVALID STATUS_INVALID |

### 2.3.18 RANGECMP4

**Highly compressed version of the RANGE log**

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7, SPAN CPT7, SMART7, SMART2**

This log contains the RANGE data in a more heavily compressed format compared to the RANGECMP2 log.

For dual antenna receivers, a RANGECMP4_1 log can be requested to get RANGECMP4 data from the second antenna. As described in Table: Binary Message Header Structure, the message type indicates the log is from the second antenna. To request an ASCII log enter RANGECMP4A_1, and for a binary log enter RANGECMP4B_1.

1. Channels that have been manually assigned to a PRN with **ASSIGN** or **ASSIGNALL** are not reported in the **RANGECMP4** log.
2. L-Band channels are not reported in the **RANGECMP4** log.
3. **RANGECMP4** is a complex log. For more information about decoding the **RANGECMP4** log, refer to Example of Bit Parsing a RANGECMP4 Log.

**Message ID: 2050**

**Log Type: Synch**

**Recommended Input:**

```
log rangecmp4a ontime 10
```

**Example:**

```
#RANGECMP4A,COM1,0,81.5,FINESTEERING,1921,228459.000,00000020,fb0e,32768;627,
6300320908510000000009200dbbf7d8306f822d0a3b2bc897f0010d350428cf31228ea9f7300040
050ff5e641cb7c7463d2a00b6a4644f6e5ee2a0fe530a00fe1f829dcfe4cf30d52abaf37f94e016
21cd8d8c04a0bafcaf00e43b0761690064e7bfe90f11ce8710a4eb2b573202607403fc28e647c6f
e9f550118007a9d839c2680ebfedff6876be81150411adbc972feef4686c483f30a09f01773ff0b
0050d8b8a843f41576b94100440e1e4f59ace54fffca2700fc1f62e14720f4facba64affbf9c52f
f39ce4b3eef9f14fd0f00244387d00d80fefabfeb0fb3cf456ae97542d410fc9ffab7f601e73580
e5efdaff0f00a0b33991fc072ccbaa99ff134efa9fd0dc684bfc61f0fffeff60b02000000000800
4c0ff3fa0b2f724f7e1eee889e9fb9f3977c0437391ab135877fe0b00301edf93f4bd63c62850fd
bf8527e6e5cd438e3a208400e0ff43bb6f5fc2101c75b058daff375c5ea4378f51940022eefff0
fe1c97dcda81887c83a63007c9d5a7ed65ce6f901427bffff3f9c04f735db1d55294a3bfc5f35cc
c66df318c412181400140060eedbd7285feaf6a653f9bf9fc7fe27cd653633c0b5fcffff03197b4
f8228d4e59d0cfbffa731b2f73b07e9b68078f47f0000a9be7dcdcc51898da269fe839b6191ab9c
c67701f21000fc3f0001a1000000008002c03fb4362793b9bfeb657dfcffe6badabb9a4375b77f5
bff1fed87bce64454a98ae16c14ff4fec6f7a48f3206b03e8040138fbd0023d225492cd7679a4ff
a5623b08810e42bf05fce17fa41f9a9ccfc8e2626231edf2ff208a1225ce6150204067febfef030
1000000000000028000ca9cc8728bb3306e68af97f921cfce3e632f0d1cf8300c8f701*6de99eb7
```

| Field | Field Type | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | RANGECMP4 header | Log header. See Messages for more information. | - | H | 0 |
| 2 | # bytes | Number of bytes in the compressed binary Range Data. | Uchar | 4 | H |
| 3 | Range Data | The compressed binary range data is organized into satellite system blocks which break down into measurement blocks for each active signal within each system. Refer to the following tables for more details about this format:<br>*Table 39: Header* on the next page (sent once)<br>*Table 40: Satellite and Signal Block* on the next page (sent once per satellite system bit set to 1 in the GNSS Field found in *Table 39: Header* on the next page)<br>*Table 41: Measurement Block Header* on page 113 (sent once for each bit set to 1 in the Satellites Field found in *Table 40: Satellite and Signal Block* on the next page)<br>*Table 42: Primary Reference Signal Measurement Block* on page 114 and *Table 43: Secondary Reference Signals Measurement Block* on page 115, or *Table 44: Primary Differential Signal Measurement Block* on page 116 and *Table 45: Secondary Differential Signals Measurement Block* on page 117, Measurement Block (sent for each bit set to 1 in the Included Signals Field for a given satellite found in *Table 40: Satellite and Signal Block* on the next page)<br>The byte data is received MSB first so each group of bytes (as defined by the number of needed bits) must be swapped prior to processing. | Uchar | # bytes | H+4 |
| 4 | xxxx | 32-bit CRC (ASCII only) | Hex | 4 | H+4+ (# bytes) |
| 5 | [CR][LF] | Sentence terminator (ASCII only) | - | - | - |

**Table 39: Header**

| Data Name | Description | Bits | Scale Factor |
|---|---|---|---|
| GNSS | Indicates which satellite system data is encoded and in what order. When the bit is set the satellite data is included. Data for each system is encoded sequentially:<br>Bit 0 = GPS<br>Bit 1 = GLONASS<br>Bit 2 = SBAS<br>Bit 5 = Galileo<br>Bit 6 = BeiDou<br>Bit 7 = QZSS<br>Bit 9 = NavIC | 16 | 1 |
| | Bit Sum: | 16 | |

This block is sent once per message

**Table 40: Satellite and Signal Block**

| Data Name | Range | Description | Bits | Scale Factor |
|---|---|---|---|---|
| Satellites | 0… 1.84467E+19 | Indicates which satellites are present for this system and their order in the message. Each PRN is represented by a bit. (Bit 0 = PRN 1, Bit 1 = PRN 2, …)<br>**Notes**:<br>• GLONASS Satellite: This value represents the Slot ID of the satellite (range of 1 to 24 where Bit 0 = Slot ID 1). In the event the Slot ID is between 43 and 63, the actual GLONASS Slot ID has not yet been determined and has been replaced with a temporary Slot ID calculated using the GLONASS Frequency Number. See the GLONASS Frequency Number field in *Table 41: Measurement Block Header* on the next page for more details.<br>• SBAS Satellite PRNs 120 to 158 are offset by 120. (Bit 0 = PRN 120, Bit 1 = 121, …)<br>• SBAS Satellite PRNs 183 to 191 are offset by 130<br>• QZSS Satellite PRNs are offset by 193 | 64 | 1 |
| Signals | 0… 65535 | Indicates which signals are present for this system and their order in the message. Each signal is represented by a bit as defined in *Table 46: Signal Bit Mask* on page 119. | 16 | 1 |

| Data Name | Range | Description | Bits | Scale Factor |
|---|---|---|---|---|
| Included Signals | 0… mxn | A two dimensional field to tell the decoder which signals are present for each of the satellites.<br>**m** = The number of rows equals the number of bits set to 1 found in the Satellites field. (Maximum number of PRNs in the satellite system)<br>**n** = The number of columns equals the number of bits set to 1 found in the Signals field. (Maximum number of Signals in the satellite system) | mxn | |
| | | Bit Sum: | 80 + mxn | |

This block is sent once for each bit set to 1 in the GNSS field found in *Table 39: Header* on the previous page.

**Table 41: Measurement Block Header**

| Data Name | Range | Description | Bits | Scale Factor |
|---|---|---|---|---|
| Data Format Flag | 0… 1 | Identifies what type of Measurement Block will be used:<br>0 = Reference<br>(*Table 42: Primary Reference Signal Measurement Block* on the next page and *Table 43: Secondary Reference Signals Measurement Block* on page 115)<br>1 = Differential<br>(*Table 44: Primary Differential Signal Measurement Block* on page 116 and *Table 45: Secondary Differential Signals Measurement Block* on page 117) | 1 | 1 |
| Ref Data Block ID | 0… 7 | This ID identifies to which reference data the Differential Data is linked. This value is incremented by 1 each time a new Reference Measurement Block is used. | 3 | 1 |

| Data Name | Range | Description | Bits | Scale Factor |
|---|---|---|---|---|
| GLONASS Frequency Number | 0… 20 (-7 to +13) | These bits are only present for GLONASS satellites in the Reference Data. This represents the GLONASS Frequency Number which identifies the frequency offset of the carrier frequency. The value will appear as a number between 0 and 20 which directly translates into a frequency offset number between -7 to +13.<br>If the GLONASS Slot ID is unknown, a temporary Slot ID for this satellite will be set between 43 and 63 based on the GLONASS Frequency Number:<br>PRN = 63 – GLONASS Frequency Number<br>The GLONASS Frequency Number used in this calculation is the 0 to 20 value, not the adjusted -7 to +13 value. | 5 | 1 |
| | | Bit Sum: | 4 (Non-GLONASS)<br>9 (GLONASS) | |
| This block is sent once for each bit set to 1 in the Satellites field found in *Table 40: Satellite and Signal Block* on page 112. | | | | |

**Table 42: Primary Reference Signal Measurement Block**

| Data Name | Range | Description | Bits | Scale Factor |
|---|---|---|---|---|
| Parity Flag | 0… 1 | 0 = Parity Unknown<br>1 = Parity Known | 1 | 1 |
| ½ Cycle Flag | 0… 1 | 0 = Half Cycle Not Added<br>1 = Half Cycle Added | 1 | 1 |
| C/No | 0… 63.95 | C/No | 11 | 0.05 dBHz |
| Lock Time | 0… 15 | The Lock Time – See *Table 47: Lock Time* on page 119 | 4 | 1 |
| Pseudorange Std Dev | 0… 15 | The Pseudorange Standard Deviation (m) – See *Table 49: Pseudorange Std Dev* on page 121 | 4 | |
| ADR Std Dev | 0… 15 | The ADR Standard Deviation (cycles) – See *Table 48: ADR Std Dev* on page 120 | 4 | |

| Data Name | Range | Description | Bits | Scale Factor |
|---|---|---|---|---|
| Primary Pseudorange | 0… 68719476.74 | The Pseudo Range of the 1st signal (Signals field in *Table 40: Satellite and Signal Block* on page 112).<br>If this value equals ($2^{37}$-1) = 137438953471, it represents a signal that is not locked. | 37 | 0.0005 m |
| PhaseRange – Primary Pseudorange | ±419.4303 | (2's Complement)<br>If this value equals –($2^{23}$-1) = -4194304, it represents the signal is not locked. | 23 | 0.0001 m |
| Primary Doppler | ±3355.4431 | (2's Complement)<br>If this value equals –($2^{26}$-1) = -33554432, it represents an invalid Doppler. | 26 | 0.0001 m/s |
| | | Bit Sum: | 111 | |

This block is sent once for the first bit set to 1 in the Included Signals field found in *Table 40: Satellite and Signal Block* on page 112.
For any bits set to 1 after the first bit set to 1, refer to *Table 43: Secondary Reference Signals Measurement Block* below.

This table is for Reference blocks only, as indicated by the Data Format Flag (see *Table 41: Measurement Block Header* on page 113).

**Table 43: Secondary Reference Signals Measurement Block**

| Data Name | Range | Description | Bits | Scale Factor |
|---|---|---|---|---|
| Parity Flag | 0… 1 | 0 = Parity Unknown<br>1 = Parity Known | 1 | 1 |
| ½ Cycle Flag | 0… 1 | 0 = Half Cycle Not Added<br>1 = Half Cycle Added | 1 | 1 |
| C/No Indicator | 0… 63.95 | C/No | 11 | 0.05 dBHz |
| Lock Time | 0… 15 | The Lock Time – See *Table 47: Lock Time* on page 119 | 4 | 1 |
| Pseudorange Std Dev | 0… 15 | The Pseudorange Standard Deviation (m) – See *Table 49: Pseudorange Std Dev* on page 121 | 4 | |

| Data Name | Range | Description | Bits | Scale Factor |
|---|---|---|---|---|
| ADR Std Dev | 0… 15 | The ADR Standard Deviation (cycles) – See *Table 48: ADR Std Dev* on page 120 | 4 | |
| Pseudorange – Primary Signal Pseudorange | ±262.1435 | (2's Complement)<br>If this value equals $-(2^{20}-1) = -524288$, it indicates the signal is not locked. | 20 | 0.0005 m |
| Phaserange – Pseudorange | ±419.4303 | (2's Complement)<br>If this value equals $-(2^{23}-1) = -4194304$, it indicates the signal is not locked. | 23 | 0.0001 m |
| Doppler – Primary Doppler | ±0.8191 | (2's Complement)<br>If this value equals $-(2^{14}-1) = -8192$, it indicates an invalid Doppler. | 14 | 0.0001 m/s |
| | | Bit Sum: | 82 | |

This block is sent once for each bit set to 1 after the first bit set to 1 in the Included Signals field found in *Table 40: Satellite and Signal Block* on page 112.

This table is for Reference blocks only, as indicated by the Data Format Flag (see *Table 41: Measurement Block Header* on page 113).

**Table 44: Primary Differential Signal Measurement Block**

| Data Name | Range | Description | Bits | Scale Factor |
|---|---|---|---|---|
| Parity Flag | 0… 1 | 0 = Parity Unknown<br>1 = Parity Known | 1 | 1 |
| ½ Cycle Flag | 0… 1 | 0 = Half Cycle Not Added<br>1 = Half Cycle Added | 1 | 1 |
| C/No | 0… 63.95 | C/No | 11 | 0.05 dBHz |
| Lock Time | 0… 15 | The Lock Time – See *Table 47: Lock Time* on page 119 | 4 | 1 |
| Pseudorange Std Dev | 0… 15 | The Pseudorange Standard Deviation (m) – See *Table 49: Pseudorange Std Dev* on page 121 | 4 | |
| ADR Std Dev | 0… 15 | The ADR Standard Deviation (cycles) – See *Table 48: ADR Std Dev* on page 120 | 4 | |

| Data Name | Range | Description | Bits | Scale Factor |
|---|---|---|---|---|
| Pseudorange – Predicted Pseudorange | ±131.0715 | (2's Complement)<br>If this value equals $-(2^{19}-1) = -262144$, it indicates a signal that is not locked.<br>The Predicted Pseudorange = reference pseudorange plus (the reference doppler x time difference between the reference log and the differential log). The Reference log and Differential logs used must contain matching Ref Data Block ID references (*Table 41: Measurement Block Header* on page 113). | 19 | 0.0005 m |
| Phaserange – Predicted Phaserange | ±3.2767 | (2's Complement)<br>If this value equals $-(2^{16}-1) = -32768$, it indicates the signal is not locked.<br>The Predicted Phaserange = reference phaserange plus (the reference doppler x time difference between the reference log and the differential log). The Reference log and Differential logs used must contain matching Ref Data Block ID references (*Table 41: Measurement Block Header* on page 113). | 16 | 0.0001 m |
| Doppler – Reference Doppler | ±13.1071 | (2's Complement)<br>If this value equals $-(2^{18}-1) = -131072$, it indicates an invalid Doppler.<br>The Reference Doppler is the Doppler for that PRN and for that signal from the Reference log. The Reference log and Differential logs used must contain matching Ref Data Block ID references (*Table 41: Measurement Block Header* on page 113). | 18 | 0.0001 m/s |
| | | Bit Sum: | 78 | |

This block is sent once for each bit set to 1 after the first bit set to 1 in the Included Signals field found in *Table 40: Satellite and Signal Block* on page 112.
For any bits set to 1 after the first bit set to 1, refer to *Table 45: Secondary Differential Signals Measurement Block* below.

This table is for Differential blocks only, as indicated by the Data Format Flag (see *Table 41: Measurement Block Header* on page 113).

**Table 45: Secondary Differential Signals Measurement Block**

| Data Name | Range | Description | Bits | Scale Factor |
|---|---|---|---|---|
| Parity Flag | 0… 1 | 0 = Parity Unknown<br>1 = Parity Known | 1 | 1 |

| Data Name | Range | Description | Bits | Scale Factor |
|---|---|---|---|---|
| ½ Cycle Flag | 0… 1 | 0 = Half Cycle Not Added<br>1 = Half Cycle Added | 1 | 1 |
| C/No | 0… 63.95 | C/No | 11 | 0.05 dBHz |
| Lock Time | 0… 15 | The Lock Time – See *Table 47: Lock Time* on the next page | 4 | 1 |
| Pseudorange Std Dev | 0… 15 | The Pseudorange Standard Deviation (m) – See *Table 49: Pseudorange Std Dev* on page 121 | 4 | 1 |
| ADR Std Dev | 0… 15 | The ADR Std Dev (cycles)– See *Table 48: ADR Std Dev* on page 120 | 4 | 1 |
| Pseudorange – Predicted Pseudorange | ±131.0715 | (2's Complement)<br>If this value equals $-(2^{19}-1) = -262144$, it indicates the signal is not locked.<br>The Predicted Pseudorange = reference pseudorange plus (the reference doppler x time difference between the reference log and the differential log). The Reference log and Differential logs used must contain matching Ref Data Block ID references (*Table 41: Measurement Block Header* on page 113). | 19 | 0.0005 m |
| Phaserange – Predicted Phaserange | ±3.2767 | (2's Complement)<br>If this value equals $-(2^{16}-1) = -32768$, it indicates the signal is not locked.<br>The Predicted Phaserange = reference phaserange plus (the reference doppler x time difference between the reference log and the differential log). The Reference log and Differential logs used must contain matching Ref Data Block ID references (*Table 41: Measurement Block Header* on page 113). | 16 | 0.0001 m |
| Doppler – Reference Doppler | ±13.1071 | (2's Complement)<br>If this value equals $-(2^{14}-1) = -8192$, it indicates an invalid Doppler.<br>The Reference Doppler is the Doppler for that PRN and for that signal from the Reference log. The Reference log and Differential logs used must contain matching Ref Data Block ID references (*Table 41: Measurement Block Header* on page 113). | 14 | 0.0001 m/s |
| | | Bit Sum: | 74 | |

This block is sent once for each bit set to 1 after the first bit set to 1 in the Included Signals field found in *Table 40: Satellite and Signal Block* on page 112.

This table is for Differential blocks only, as indicated by the Data Format Flag (see *Table 41: Measurement Block Header* on page 113).

**Table 46: Signal Bit Mask**

| | GPS | GLONASS | SBAS | Galileo | BeiDou | QZSS | NavIC |
|---|---|---|---|---|---|---|---|
| **Bit 1** | L1CA | L1CA | L1CA | E1 | B1I | L1CA | L5SPS |
| **Bit 2** | | | L5I | E5A | B1GEO | | |
| **Bit 3** | | L2CA | | E5B | B2I | L2C | |
| **Bit 4** | L2Y | L2P | | ALTBOC | B2GEO | L5Q | |
| **Bit 5** | L2C | | | E6C | B3I | | |
| **Bit 6** | L2P | L3 | | | B3GEO | | |
| **Bit 7** | L5Q | | | | B1CP | | |
| **Bit 8** | | | | | | L1C | |
| **Bit 9** | | | | | B2AP | | |
| **Bit 10** | | | | | | | |
| **Bit 11** | | | | | | L6P | |
| **Bit 12** | | | | E6B | | | |
| **Bit 13** | | | | | | | |
| **Bit 14** | | | | | | | |
| **Bit 15** | L1C | | | | | | |

**Table 47: Lock Time**

| Indicator (i) | Minimum Lock Time (ms) | Range of Indicated Lock Times (t represents the Lock Time) (ms) |
|---|---|---|
| 0 | 0 | $0 \le t < 16$ |
| 1 | 16 | $16 \le t < 32$ |
| 2 | 32 | $32 \le t < 64$ |
| 3 | 64 | $64 \le t < 128$ |
| 4 | 128 | $128 \le t < 256$ |
| 5 | 256 | $256 \le t < 512$ |
| 6 | 512 | $512 \le t < 1024$ |
| 7 | 1024 | $1024 \le t < 2048$ |
| 8 | 2048 | $2048 \le t < 4096$ |

| Indicator (i) | Minimum Lock Time (ms) | Range of Indicated Lock Times (t represents the Lock Time) (ms) |
|---|---|---|
| 9 | 4096 | 4096 ≤ t < 8192 |
| 10 | 8192 | 8192 ≤ t < 16384 |
| 11 | 16384 | 16384 ≤ t < 32768 |
| 12 | 32768 | 32768 ≤ t < 65536 |
| 13 | 65536 | 65536 ≤ t < 131072 |
| 14 | 131072 | 131072 ≤ t < 262144 |
| 15 | 262144 | 262144 ≤ t |

**Table 48: ADR Std Dev**

| ADR Std Dev (cycles) | |
|---|---|
| 0 | ≤ 0.0039 |
| 1 | ≤ 0.0052 |
| 2 | ≤ 0.0070 |
| 3 | ≤ 0.0093 |
| 4 | ≤ 0.0124 |
| 5 | ≤ 0.0165 |
| 6 | ≤ 0.0221 |
| 7 | ≤ 0.0295 |
| 8 | ≤ 0.0393 |
| 9 | ≤ 0.0525 |
| 10 | ≤ 0.0701 |
| 11 | ≤ 0.0935 |
| 12 | ≤ 0.1248 |
| 13 | ≤ 0.1666 |
| 14 | ≤ 0.2223 |
| 15 | > 0.2223 |

**Table 49: Pseudorange Std Dev**

| Pseudorange Std Dev (m) | |
|---|---|
| 0 | ≤ 0.020 |
| 1 | ≤ 0.030 |
| 2 | ≤ 0.045 |
| 3 | ≤ 0.066 |
| 4 | ≤ 0.099 |
| 5 | ≤ 0.148 |
| 6 | ≤ 0.220 |
| 7 | ≤ 0.329 |
| 8 | ≤ 0.491 |
| 9 | ≤ 0.732 |
| 10 | ≤ 1.092 |
| 11 | ≤ 1.629 |
| 12 | ≤ 2.430 |
| 13 | ≤ 3.625 |
| 14 | ≤ 5.409 |
| 15 | > 5.409 |

## 2.3.19 RAWSBASFRAME2

### Raw SBAS frame data 2

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7, SPAN CPT7, SMART7, SMART2**

This log contains the raw SBAS frame data of 226 bits (8-bit preamble, 6-bit message type and 212 bits of data but without a 24-bit CRC). It also contains the transmitted frequency. Only frame data with a valid preamble and CRC are reported.

**Message ID: 2185**

**Log Type: Asynch**

**Recommended Input:**

```
log rawsbasframe2a onnew
```

**ASCII Example:**

```
#RAWSBASFRAME2A,COM1,0,77.5,SATTIME,1977,514394.000,02000020,b39f,32768;135,209
,2,1,0,3,c60d4009ffc018001ffc005ffdfffffbff9ffc00bfed79db9bb95b9540*9a75ce69

#RAWSBASFRAME2A,COM1,0,77.5,SATTIME,1977,514394.000,02000020,b39f,32768;138,207
,2,1,0,4,c6125ffdffc005fffffffbfe3fb9ffdffdffdffdfffba3956abffffc0*9324a574

#RAWSBASFRAME2A,COM1,0,77.5,SATTIME,1977,514395.000,02000020,b39f,32768;135,208
,1,0,0,4,53125ffdffc011ffc000007fe3fb5ffdffdffdffdfffba3956abffffc0*69490ac5

#RAWSBASFRAME2A,COM1,0,78.5,SATTIME,1977,514395.000,02000020,b39f,32768;138,206
,1,0,0,3,530c7ff9ffc017ff9fffff9ffdfffffbfedffc003fe579db9bb95b9540*c7ca1531
```

The **RAWSBASFRAME2** log output contains all the raw data required for an application to compute its own SBAS correction parameters.

| Field | Field Type | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | RAWSBASFRAME2 header | Log header. See Messages for more information. | | H | 0 |
| 2 | PRN | SBAS satellite PRN number | Ulong | 4 | H |
| 3 | signal channel | Signal channel number that the frame was decoded on | Ulong | 4 | H+4 |
| 4 | SBAS Signal Source | Identifies the source of the SBAS signal:<br>1 – SBASL1CA<br>2 – SBASL5I | Uchar | 1 | H+8 |

| Field | Field Type | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 5 | SBAS Preamble Type | Identifies what preamble was used when tracking the SBAS signal:<br>0 – SBASL1CA 8-bit Preamble<br>1 – SBASL5I 8-bit Preamble | Uchar | 1 | H+9 |
| 6 | Reserved | | Ushort | 2 | H+10 |
| 7 | SBAS frame ID | SBAS frame ID | Ulong | 4 | H+12 |
| 8 | data | Raw SBAS frame data. There are 226 bits of data and 6 bits of padding. | Hex [29] | 32[1] | H+16 |
| 9 | xxxx | 32-bit CRC (ASCII and Binary only) | Ulong | 4 | H+48 |
| 10 | [CR][LF] | Sentence terminator (ASCII only) | - | - | - |

[1] In the binary log case, an additional 3 bytes of padding are added to maintain 4-byte alignment.

## 2.3.20 RTKASSISTSTATUS

### RTK ASSIST status

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7, SPAN CPT7, SMART7**

This log provides information on the state of RTK ASSIST.

RTK ASSIST operates in two modes: coast and full assist. The RTKASSISTSTATUS log reports which mode is currently available. Coast mode is available as soon as the RTK ASSIST corrections are received from the L-Band satellite, while full assist mode requires a convergence period. In coast mode, position error growth during RTK correction outages is slightly worse than in full assist mode and RTK will not resume following a full signal outage until after RTK corrections are restored. Full assist gives the lowest position error growth during RTK correction outages, and makes it possible for RTK to resume even if there are complete GNSS signal outages during the RTK ASSIST period.

The RTK ASSIST ACTIVE state reported in the RTKASSISTSTATUS log is also reported in the RTKPOS and BESTPOS extended solution status field. See Table: Extended Solution Status.

The RTKASSISTSTATUS log reports the time remaining in the RTK ASSIST ACTIVE state. Once RTK ASSIST becomes active, the remaining time will count down from the time out set by the **RTKASSISTTIMEOUT** command .

The corrections age reported in the RTKASSISTSTATUS log should typically be below 30 seconds. If the age exceeds this value, then L-Band tracking is likely degraded. The most likely cause of degraded L-Band tracking are obstructions between the antenna and the L-Band satellite.

**Message ID: 2048**

**Log Type: Synch**

**Recommended Input:**

```
log rtkassiststatusa ontime 5
```

**ASCII Example:**

```
#RTKASSISTSTATUSA,COM1,0,80.0,FINESTEERING,1932,491359.000,02000020,80fe,46672;
ACTIVE,ASSIST,969.0,14.0*26e32616
```

| Field | Field type | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | RTKASSISTSTATUS header | Log header. See Messages for more information. | | H | 0 |
| 2 | State | State:<br>INACTIVE (0)<br>ACTIVE (1) | Enum | 4 | H |

| Field | Field type | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 3 | Mode | Mode:<br>UNAVAILABLE (0)<br>COAST (1)<br>ASSIST (2) | Enum | 4 | H+4 |
| 4 | Remaining time | Time remaining in seconds | Float | 4 | H+8 |
| 5 | Corrections age | Age of the RTK ASSIST corrections in seconds. Maximum value of 120 seconds. | Float | 4 | H+12 |
| 6 | xxxx | 32-bit CRC (ASCII and Binary only) | Hex | 4 | H+16 |
| 7 | [CR][LF] | Sentence terminator (ASCII only) | - | - | - |

## 2.3.21 RXSTATUS

### Receiver status

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7, SPAN CPT7, SMART7, SMART2**

This log conveys various status parameters of the GNSS receiver system. These include the Receiver Status and Error words which contain several flags specifying status and error conditions. If an error occurs (shown in the Receiver Error word), the receiver idles all channels, turns off the antenna and disables the RF hardware as these conditions are considered to be fatal errors. The log contains a variable number of status words to allow for maximum flexibility and future expansion. For information about resolving an error, refer to Examining the RXSTATUS Log in the OEM7 Installation and Operation User Manual, PwrPak7 Installation and Operation User Manual or SPAN CPT7 Installation and Operation User Manual.

The receiver gives the user the ability to determine the importance of the status bits. In the case of the Receiver Status, setting a bit in the priority mask causes the condition to trigger an error. This causes the receiver to idle all channels, turn off the antenna and disable the RF hardware, the same as if a bit in the Receiver Error word is set. Setting a bit in an Auxiliary Status priority mask causes that condition to set the bit in the Receiver Status word corresponding to that Auxiliary Status. See also the **STATUSCONFIG** command.

Field #4, the receiver status word as represented in *Table 51: Receiver Status* on page 130, is also in Field #8 of the header. See the ASCII Example below and *Table 51: Receiver Status* on page 130 for clarification.

Refer also to Built-In Status Tests in the OEM7 Installation and Operation User Manual, PwrPak7 Installation and Operation User Manual or SPAN CPT7 Installation and Operation User Manual.

**Message ID: 93**

**Log Type: Asynch**

**Recommended Input:**

```
log rxstatusa onchanged
```

**Abbreviated ASCII Example:**

```
#RXSTATUS USB1 0 67.0 FINESTEERING 2070 159546.023 02000000 2ae1 15824
00000000 5    (Receiver Error)
     02000000 00000000 00000000 00000000    (Receiver Status)
     00040000 00001008 00000000 00000000    (Aux1 Status)
     00000000 00000000 80000000 00000000    (Aux2 Status)
     80000000 00000000 00000000 00000000    (Aux3 Status)
     0030c000 00000000 ffffffff 00000000    (Aux4 Status)
```

Receiver errors automatically generate event messages. These event messages are output in RXSTATUSEVENT logs. It is also possible to have status conditions trigger event messages to be generated by the receiver. This is done by setting/clearing the appropriate bits in the event set/clear masks. The set mask tells the receiver to generate an event message when the bit becomes set. Likewise, the clear mask causes messages to be generated when a bit is cleared. See the **STATUSCONFIG** command for details.

If you wish to disable all these messages without changing the bits, simply UNLOG the **RXSTATUSEVENT** logs on the appropriate ports. See the **UNLOG** command.

| Field | Field type | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | RXSTATUS header | Log header. See Messages for more information. | | H | 0 |
| 2 | error | Receiver error (see *Table 50: Receiver Error* on the next page). A value of zero indicates no errors | Ulong | 4 | H |
| 3 | # stats | Number of status codes (including Receiver Status). Each status code consists of 4 fields, the status, priority mask, event set mask and event clear mask. Each set is repeated for each status type.<br>Note that for clarity, the Receiver Status, Auxiliary 1 Status, Auxiliary 2 Status, Auxiliary 3 Status and Auxiliary 4 status are listed separately in this message | Ulong | 4 | H+4 |
| 4 | rxstat | Receiver status word (see *Table 51: Receiver Status* on page 130) | Ulong | 4 | H+8 |
| 5 | rxstat pri | Receiver status priority mask, which can be set using the **STATUSCONFIG** command | Ulong | 4 | H+12 |
| 6 | rxstat set | Receiver status event set mask, which can be set using the **STATUSCONFIG** command | Ulong | 4 | H+16 |
| 7 | rxstat clear | Receiver status event clear mask, which can be set using the **STATUSCONFIG** command | Ulong | 4 | H+20 |
| 8 | aux1stat | Auxiliary 1 status word (see *Table 53: Auxiliary 1 Status* on page 132) | Ulong | 4 | H+24 |
| 9 | aux1stat pri | Auxiliary 1 status priority mask, which can be set using the **STATUSCONFIG** command | Ulong | 4 | H+28 |
| 10 | aux1stat set | Auxiliary 1 status event set mask, which can be set using the **STATUSCONFIG** command | Ulong | 4 | H+32 |
| 11 | aux1stat clear | Auxiliary 1 status event clear mask, which can be set using the **STATUSCONFIG** command | Ulong | 4 | H+36 |
| 12 | aux2stat | Auxiliary 2 status word (see *Table 54: Auxiliary 2 Status* on page 133) | Ulong | 4 | H+40 |
| 13 | aux2stat pri | Auxiliary 2 status priority mask, which can be set using the **STATUSCONFIG** command | Ulong | 4 | H+44 |
| 14 | aux2stat set | Auxiliary 2 status event set mask, which can be set using the **STATUSCONFIG** command | Ulong | 4 | H+48 |
| 15 | aux2stat clear | Auxiliary 2 status event clear mask, which can be set using the **STATUSCONFIG** command | Ulong | 4 | H+52 |

| Field | Field type | Description | Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 16 | aux3stat | Auxiliary 3 status word (see *Table 55: Auxiliary 3 Status* on page 135) | Ulong | 4 | H+56 |
| 17 | aux3stat pri | Auxiliary 3 status priority mask, which can be set using the **STATUSCONFIG** command | Ulong | 4 | H+60 |
| 18 | aux3stat set | Auxiliary 3 status event set mask, which can be set using the **STATUSCONFIG** command | Ulong | 4 | H+64 |
| 19 | aux3stat clear | Auxiliary 3 status event clear mask, which can be set using the **STATUSCONFIG** command | Ulong | 4 | H+68 |
| 20 | aux4stat | Auxiliary 4 status word (see *Table 57: Auxiliary 4 Status* on page 137) | Ulong | 4 | H+72 |
| 21 | aux4stat pri | Auxiliary 4 status priority mask, which can be set using the **STATUSCONFIG** command | Ulong | 4 | H+76 |
| 22 | aux4stat set | Auxiliary 4 status event set mask, which can be set using the **STATUSCONFIG** command | Ulong | 4 | H+80 |
| 23 | aux4stat clear | Auxiliary 4 status event clear mask, which can be set using the **STATUSCONFIG** command | Ulong | 4 | H+84 |
| 24 | xxxx | 32-bit CRC (ASCII and Binary only) | Ulong | 4 | H+88 |
| 25 | [CR][LF] | Sentence terminator (ASCII only) | - | - | - |

**Table 50: Receiver Error**

| Nibble | Bit | Mask | Description | Bit = 0 | Bit = 1 |
|---|---|---|---|---|---|
| N0 | 0 | 0x00000001 | Dynamic Random Access Memory (DRAM) status<br>RAM failure on an OEM7 card may also be indicated by a flashing red LED. | OK | Error |
| | 1 | 0x00000002 | Invalid firmware | OK | Error |
| | 2 | 0x00000004 | ROM status | OK | Error |
| | 3 | 0x00000008 | Reserved | | |
| N1 | 4 | 0x00000010 | Electronic Serial Number (ESN) access status | OK | Error |
| | 5 | 0x00000020 | Authorization code status | OK | Error |
| | 6 | 0x00000040 | Reserved | | |
| | 7 | 0x00000080 | Supply voltage status | OK | Error |

| Nibble | Bit | Mask | Description | Bit = 0 | Bit = 1 |
|---|---|---|---|---|---|
| N2 | 8 | 0x00000100 | Reserved | | |
| | 9 | 0x00000200 | Temperature status (as compared against acceptable limits) | OK | Error |
| | 10 | 0x00000400 | MINOS status | OK | Error |
| | 11 | 0x00000800 | PLL RF status. Error with an RF PLL. See AUX2 status bits (*Table 54: Auxiliary 2 Status* on page 133) for individual PLL status | OK | Error |
| N3 | 12 | 0x00001000 | Reserved | | |
| | 13 | 0x00002000 | | | |
| | 14 | 0x00004000 | | | |
| | 15 | 0x00008000 | NVM status | OK | Error |
| N4 | 16 | 0x00010000 | Software resource limit exceeded | OK | Error |
| | 17 | 0x00020000 | Model invalid for this receiver | OK | Error |
| | 18 | 0x00040000 | Reserved | | |
| | 19 | 0x00080000 | | | |
| N5 | 20 | 0x00100000 | Remote loading has begun | No | Yes |
| | 21 | 0x00200000 | Export restriction | OK | Error |
| | 22 | 0x00400000 | Safe Mode | OK | Error |
| | 23 | 0x00800000 | Reserved | | |
| N6 | 24 | 0x01000000 | | | |
| | 25 | 0x02000000 | | | |
| | 26 | 0x04000000 | | | |
| | 27 | 0x08000000 | | | |
| N7 | 28 | 0x10000000 | | | |
| | 29 | 0x20000000 | | | |
| | 30 | 0x40000000 | | | |
| | 31 | 0x80000000 | Component hardware failure | OK | Error |

**Table 51: Receiver Status**

| Nibble | Bit | Mask | Description | Bit = 0 | Bit = 1 |
|---|---|---|---|---|---|
| N0 | 0 | 0x00000001 | Error flag, see *Table 50: Receiver Error* on page 128 | No error | Error |
| | 1 | 0x00000002 | Temperature status | Within specifications | Warning |
| | 2 | 0x00000004 | Voltage supply status | OK | Warning |
| | 3 | 0x00000008 | Primary antenna power status<br>See the **ANTENNAPOWER** command | Powered | Not powered |
| N1 | 4 | 0x00000010 | LNA Failure | OK | Failure |
| | 5 | 0x00000020 | Primary antenna open circuit flag<br>This flag is only available on certain products. | OK | Open, antenna disconnected |
| | 6 | 0x00000040 | Primary antenna short circuit flag<br>This flag is only available on certain products. | OK | Short circuit detected |
| | 7 | 0x00000080 | CPU overload flag<br>This flag is only available on certain products. | No overload | Overload |
| N2 | 8 | 0x00000100 | COM port transmit buffer overrun.<br>See AUX2 status bits (*Table 54: Auxiliary 2 Status* on page 133) for individual COM port status | OK | COM buffer overrun |
| | 9 | 0x00000200 | Reserved | | |
| | 10 | 0x00000400 | | | |
| | 11 | 0x00000800 | Link overrun flag<br>This flag indicates if any of the USB, ICOM, CCOM, NCOM or File ports are overrun. See AUX1, AUX2 and AUX3 status bits (*Table 53: Auxiliary 1 Status* on page 132, *Table 54: Auxiliary 2 Status* on page 133 or *Table 55: Auxiliary 3 Status* on page 135) for the specific port for which the buffer is overrun. | No overrun | Overrun |

| Nibble | Bit | Mask | Description | Bit = 0 | Bit = 1 |
|---|---|---|---|---|---|
| N3 | 12 | 0x00001000 | Input overrun flag<br>This flag is set if any of the receiver ports (COM, USB, ICOM or NCOM) experience an input overrun. | No overrun | Overrun |
| | 13 | 0x00002000 | Aux transmit overrun flag | No overrun | Overrun |
| | 14 | 0x00004000 | Antenna gain state<br>See the AUX3 status bits (*Table 55: Auxiliary 3 Status* on page 135) for the antenna gain status. | OK | Out of range |
| | 15 | 0x00008000 | Jammer Detected<br>See the AUX1 status bits (*Table 53: Auxiliary 1 Status* on the next page) for individual RF status | OK | Jammer Detected |
| N4 | 16 | 0x00010000 | INS reset flag | No INS reset | INS reset |
| | 17 | 0x00020000 | IMU communication failure | No error | No IMU communication |
| | 18 | 0x00040000 | GPS almanac flag/UTC known | Valid | Invalid |
| | 19 | 0x00080000 | Position solution flag | Valid | Invalid |
| N5 | 20 | 0x00100000 | Position fixed flag, see the **FIX** command | Not fixed | Fixed |
| | 21 | 0x00200000 | Clock steering status | Enabled | Disabled |
| | 22 | 0x00400000 | Clock model flag | Valid | Invalid |
| | 23 | 0x00800000 | External oscillator locked flag | Unlocked | Locked |
| N6 | 24 | 0x01000000 | Software resource | OK | Warning |
| | 25 | 0x06000000 | Version bit 0 | See *Table 52: Version Bits* on the next page | |
| | 26 | | Version bit 1 | See *Table 52: Version Bits* on the next page | |
| | 27 | 0x08000000 | Tracking mode | Normal tracking | HDR tracking |
| N7 | 28 | 0x10000000 | Digital Filtering Enabled | Disabled | Enabled |
| | 29 | 0x20000000 | Auxiliary 3 status event flag | No event | Event |
| | 30 | 0x40000000 | Auxiliary 2 status event flag | No event | Event |
| | 31 | 0x80000000 | Auxiliary 1 status event flag | No event | Event |

**Table 52: Version Bits**

| Bit 25 | Bit 26 | Description |
|---|---|---|
| 0 | 0 | Interpret Status/Error Bits as OEM6 or earlier format |
| 1 | 0 | Interpret Status/Error Bits as OEM7 format |
| 0 | 1 | Reserved for a future version |
| 1 | 1 | Reserved for a future version |

**Table 53: Auxiliary 1 Status**

| Nibble | Bit | Mask | Description | Bit = 0 | Bit = 1 |
|---|---|---|---|---|---|
| N0 | 0 | 0x00000001 | Jammer detected on RF1 | OK | Jammer detected |
| | 1 | 0x00000002 | Jammer detected on RF2 | OK | Jammer detected |
| | 2 | 0x00000004 | Jammer detected on RF3 | OK | Jammer detected |
| | 3 | 0x00000008 | Position averaging | Off | On |
| N1 | 4 | 0x00000010 | Jammer detected on RF4 | OK | Jammer detected |
| | 5 | 0x00000020 | Jammer detected on RF5 | OK | Jammer detected |
| | 6 | 0x00000040 | Jammer detected on RF6 | OK | Jammer detected |
| | 7 | 0x00000080 | USB connection status | Connected | Not connected |
| N2 | 8 | 0x00000100 | USB1 buffer overrun flag | No overrun | Overrun |
| | 9 | 0x00000200 | USB2 buffer overrun flag | No overrun | Overrun |
| | 10 | 0x00000400 | USB3 buffer overrun flag | No overrun | Overrun |
| | 11 | 0x00000800 | Reserved | | |
| N3 | 12 | 0x00001000 | Profile Activation Bit | OK | Error |
| | 13 | 0x00002000 | Throttled Ethernet Reception | OK | Throttled |
| | 14 | 0x00004000 | Reserved | | |
| | 15 | 0x00008000 | Reserved | | |

| Nibble | Bit | Mask | Description | Bit = 0 | Bit = 1 |
|---|---|---|---|---|---|
| N4 | 16 | 0x00010000 | Reserved | | |
| | 17 | 0x00020000 | Reserved | | |
| | 18 | 0x00040000 | Ethernet not connected | Connected | Not connected |
| | 19 | 0x00080000 | ICOM1 buffer overrun flag | No overrun | Overrun |
| N5 | 20 | 0x00100000 | ICOM2 buffer overrun flag | No overrun | Overrun |
| | 21 | 0x00200000 | ICOM3 buffer overrun flag | No overrun | Overrun |
| | 22 | 0x00400000 | NCOM1 buffer overrun flag | No overrun | Overrun |
| | 23 | 0x00800000 | NCOM2 buffer overrun flag | No overrun | Overrun |
| N6 | 24 | 0x01000000 | NCOM3 buffer overrun flag | No overrun | Overrun |
| | 25 | 0x02000000 | Reserved | | |
| | 26 | 0x04000000 | Reserved | | |
| | 27 | 0x08000000 | Reserved | | |
| N7 | 28 | 0x10000000 | Reserved | | |
| | 29 | 0x20000000 | Reserved | | |
| | 30 | 0x40000000 | Status error reported by the IMU. May be treated as a notice unless the issue persists. | OK | Status Error Reported |
| | 31 | 0x80000000 | IMU measurement outlier detected. Indicates when the SPAN system has detected an outlier in the IMU performance. May be treated as a notice unless the issue persists. | OK | Outlier detected |

**Table 54: Auxiliary 2 Status**

| Nibble | Bit | Mask | Description | Bit = 0 | Bit = 1 |
|---|---|---|---|---|---|
| N0 | 0 | 0x00000001 | SPI Communication Failure | OK | Error |
| | 1 | 0x00000002 | $I^2C$ Communication Failure | OK | Error |
| | 2 | 0x00000004 | COM4 buffer overrun flag | No overrun | Buffer Overrun |
| | 3 | 0x00000008 | COM5 buffer overrun flag | No overrun | Buffer Overrun |

| Nibble | Bit | Mask | Description | Bit = 0 | Bit = 1 |
|---|---|---|---|---|---|
| N1 | 4 | 0x00000010 | Reserved | | |
| | 5 | 0x00000020 | Reserved | | |
| | 6 | 0x00000040 | Reserved | | |
| | 7 | 0x00000080 | Reserved | | |
| N2 | 8 | 0x00000100 | Reserved | | |
| | 9 | 0x00000200 | COM1 buffer overrun flag | OK | Buffer Overrun |
| | 10 | 0x00000400 | COM2 buffer overrun flag | OK | Buffer Overrun |
| | 11 | 0x00000800 | COM3 buffer overrun flag | OK | Buffer Overrun |
| N3 | 12 | 0x00001000 | PLL RF1 unlock flag | OK | PLL Unlock |
| | 13 | 0x00002000 | PLL RF2 unlock flag | OK | PLL Unlock |
| | 14 | 0x00004000 | PLL RF3 unlock flag | OK | PLL Unlock |
| | 15 | 0x00008000 | PLL RF4 unlock flag | OK | PLL Unlock |
| N4 | 16 | 0x00010000 | PLL RF5 unlock flag | OK | PLL Unlock |
| | 17 | 0x00020000 | PLL RF6 unlock flag | OK | PLL Unlock |
| | 18 | 0x00040000 | CCOM1 buffer overrun | OK | Buffer Overrun |
| | 19 | 0x00080000 | CCOM2 buffer overrun | OK | Buffer Overrun |
| N5 | 20 | 0x00100000 | CCOM3 buffer overrun | OK | Buffer Overrun |
| | 21 | 0x00200000 | CCOM4 buffer overrun | OK | Buffer Overrun |
| | 22 | 0x00400000 | CCOM5 buffer overrun | OK | Buffer Overrun |
| | 23 | 0x00800000 | CCOM6 buffer overrun | OK | Buffer Overrun |
| N6 | 24 | 0x01000000 | ICOM4 buffer overrun | OK | Buffer Overrun |
| | 25 | 0x02000000 | ICOM5 buffer overrun | OK | Buffer Overrun |
| | 26 | 0x04000000 | ICOM6 buffer overrun | OK | Buffer Overrun |
| | 27 | 0x08000000 | ICOM7 buffer overrun | OK | Buffer Overrun |

| Nibble | Bit | Mask | Description | Bit = 0 | Bit = 1 |
|---|---|---|---|---|---|
| N7 | 28 | 0x10000000 | Secondary antenna power status<br>See the **ANTENNAPOWER** command | Powered | Not Powered |
| | 29 | 0x20000000 | Secondary antenna open circuit<br>This flag is only available on certain products | OK | Open, antenna disconnected |
| | 30 | 0x40000000 | Secondary antenna short circuit<br>This flag is only available on certain products | OK | Short circuit detected |
| | 31 | 0x80000000 | Reset loop detected | OK | Reset Detected |

**Table 55: Auxiliary 3 Status**

| Nibble | Bit | Mask | Description | Bit = 0 | Bit = 1 |
|---|---|---|---|---|---|
| N0 | 0 | 0x00000001 | SCOM buffer overrun flag.<br>This flag is set if any of the SCOM ports (SCOM1 – SCOM4) experience overrun. | No overrun | Overrun |
| | 1 | 0x00000002 | WCOM1 buffer overrun flag | No overrun | Overrun |
| | 2 | 0x00000004 | FILE buffer overrun flag | No overrun | Overrun |
| | 3 | 0x00000008 | Reserved | | |
| N1 | 4 | 0x00000010 | Antenna 1 Gain State | *Table 56: Antenna Gain State* on the next page | |
| | 5 | 0x00000020 | | | |
| | 6 | 0x00000040 | Antenna 2 Gain State | *Table 56: Antenna Gain State* on the next page | |
| | 7 | 0x00000080 | | | |
| N2 | 8 | 0x00000100 | Reserved | | |
| | 9 | 0x00000200 | | | |
| | 10 | 0x00000400 | | | |
| | 11 | 0x00000800 | | | |
| N3 | 12 | 0x00001000 | Reserved | | |
| | 13 | 0x00002000 | | | |
| | 14 | 0x00004000 | | | |
| | 15 | 0x00008000 | | | |

| Nibble | Bit | Mask | Description | Bit = 0 | Bit = 1 |
|---|---|---|---|---|---|
| N4 | 16 | 0x00010000 | Reserved | | |
| | 17 | 0x00020000 | | | |
| | 18 | 0x00040000 | | | |
| | 19 | 0x00080000 | | | |
| N5 | 20 | 0x00100000 | Reserved | | |
| | 21 | 0x00200000 | | | |
| | 22 | 0x00400000 | | | |
| | 23 | 0x00800000 | | | |
| N6 | 24 | 0x01000000 | Reserved | | |
| | 25 | 0x02000000 | | | |
| | 26 | 0x04000000 | | | |
| | 27 | 0x08000000 | | | |
| N7 | 28 | 0x10000000 | Reserved | | |
| | 29 | 0x20000000 | Web content is corrupt or does not exist | Content is OK | Error with content |
| | 30 | 0x40000000 | RF Calibration Data is present and in error | Data is OK | Data has an error |
| | 31 | 0x80000000 | RF Calibration Data is present | No data found | Data exists and has no errors |

**Table 56: Antenna Gain State**

| Bits 4-5 or Bits 6-7 | Description |
|---|---|
| 00 | Antenna Gain in range |
| 01 | Antenna Gain Low<br>This state indicates that the input signal is very weak (under -160 dBm/Hz). It can indicate the antenna is not operating correctly, the antenna is not suitable for NovAtel receivers, or there is no antenna connected. |
| 10 | Antenna Gain High.<br>This state indicates that the input signal is very strong (above -120 dBm/Hz). This can be caused by a strong in-band interference or by too much signal gain or too many LNAs cascaded in the path. |
| 11 | Antenna Gain Anomaly.<br>This state indicates that an anomaly has been detected for the input signal. It can be caused by strong in-band or out-of-band interference, or by the antenna being disconnected/changed during operation. |

**Table 57: Auxiliary 4 Status**

| Nibble | Bit | Mask | Description | Bit = 1 |
|---|---|---|---|---|
| N0 | 0 | 0x00000001 | GNSS Tracked Status | <60% of available satellites are tracked well |
| | 1 | 0x00000002 | | <15% of available satellites are tracked well |
| | 2 | 0x00000004 | Reserved | |
| | 3 | 0x00000008 | | |
| N1 | 4 | 0x00000010 | Reserved | |
| | 5 | 0x00000020 | | |
| | 6 | 0x00000040 | | |
| | 7 | 0x00000080 | | |
| N2 | 8 | 0x00000100 | Reserved | |
| | 9 | 0x00000200 | | |
| | 10 | 0x00000400 | | |
| | 11 | 0x00000800 | | |
| N3 | 12 | 0x00001000 | Clock freewheeling due to bad position integrity | Clock freewheeling |
| | 13 | 0x00002000 | Reserved | |
| | 14 | 0x00004000 | Usable RTK Corrections<br>Percentage of expected measurements which have timely RTK corrections (latency <20 seconds) | <60% of expected corrections available |
| | 15 | 0x00008000 | | <15% of expected corrections available |
| N4 | 16 | 0x00010000 | Bad RTK Geometry | PDOP >5.0 |
| | 17 | 0x00020000 | Reserved | |
| | 18 | 0x00040000 | | |
| | 19 | 0x00080000 | Long RTK Baseline | Baseline >50 km |

| Nibble | Bit | Mask | Description | Bit = 1 |
|---|---|---|---|---|
| N5 | 20 | 0x00100000 | Poor RTK COM Link (poor correction quality) | Corrections quality ≤60% |
| | 21 | 0x00200000 | Poor ALIGN COM Link (poor correction quality) | Corrections quality ≤60% |
| | 22 | 0x00400000 | GLIDE Not Active | GLIDE not active |
| | 23 | 0x00800000 | Bad PDP Geometry | PDOP >5.0 |
| N6 | 24 | 0x01000000 | No TerraStar Subscription | No subscription |
| | 25 | 0x02000000 | Reserved | |
| | 26 | 0x04000000 | | |
| | 27 | 0x08000000 | | |
| N7 | 28 | 0x10000000 | Bad PPP Geometry | PDOP >5.0 |
| | 29 | 0x20000000 | Reserved | |
| | 30 | 0x40000000 | No INS Alignment | No alignment |
| | 31 | 0x80000000 | INS not converged | Not converged |

Only GPS and GLONASS are considered in the Auxiliary 4 status word states.

For bits relating to RTK, ALIGN or INS, the bits will only be set if the receiver has that type of positioning is enabled via Auth Code.

## 2.3.22 SAFEMODESTATUS

### Safe Mode Status

**Platform: OEM719, OEM729, OEM7500, OEM7600, OEM7700, OEM7720, PwrPak7, SPAN CPT7, SMART7, SMART2**

This log provides additional information about the state of the receiver in the event that the *Safe Mode* error bit and/or *Reset Loop Detected* status bit are set in the **RXSTATUS** log (see page 126).

The data within this log is set at receiver start up and will not change over time.

For information about *Safe Mode*, see Safe Mode in the OEM7 Installation and Operation User Manual, PwrPak7 Installation and Operation User Manual or SPAN CPT7 Installation and Operation User Manual.

**Message ID: 2060**

**Log Type: Asynch**

**Recommended Input:**

```
log SAFEMODESTATUSA once
```

**Abbreviated ASCII Example:**

```
#SAFEMODESTATUSA,COM1,0,89.0,UNKNOWN,0,0.000,024c0020,8e55,32768;SAFE_MODE_
OK,0,"Normal Operation."*29c7d28a
```

| Field | Field Type | Description | Binary Format | Binary Bytes | Binary Offset |
|---|---|---|---|---|---|
| 1 | SAFEMODESTATUS header | Log header. See Messages for more information. | - | H | 0 |
| 2 | Status | Safe Mode State. See *Table 58: Safe Mode States* on the next page | Enum | 4 | H |
| 3 | Reset Count | Number of resets since power up or a successful boot | Ulong | 4 | H+4 |
| 4 | Description | String for additional information about the Safe Mode State | String | 80 | H+8 |
| 5 | xxxx | 32-bit CRC (ASCII and Binary only) | Ulong | 4 | H+88 |
| 6 | [CR][LF] | Sentence terminator (ASCII only) | - | - | - |

**Table 58: Safe Mode States**

| Value | State | Safe Mode Error Bit | Reset Loop Detected Bit | Notes | Recovery Steps |
|---|---|---|---|---|---|
| 0 | SAFE_MODE_OK | 0 | 0 | Normal Operation. No reset loop detected. | No action required |
| 1 | SAFE_MODE_ WARNING | 0 | 1 | An unexpected reset was detected. The receiver will operate as normal | No action required |
| 2 | SAFE_MODE_ DISABLE_ SATELLITE_DATA | 0 | 1 | Satellite Navigation Data previously saved to NVM is ignored in this state. As the receiver continues to track GNSS satellites, new data will be downloaded. There may be some delay in initial satellite acquisition as this will effectively be a Cold Start, but the receiver will otherwise operate as normal. | No action required |
| 3 | SAFE_MODE_ DISABLE_ NON_ COMMUNICATION_ NVM | 1 | 1 | All data previously saved to NVM that is not related to communication is ignored in this state.<br>Communication ports (COM, USB, ICOM, etc.) will remain in the configuration previously saved by SAVECONFIG allowing the user to take corrective action. | Depending on what NVM data is causing the problem, a **FRESET** may resolve the issue.<br>If a standard **FRESET** does not resolve the issue, see the **FRESET** command for other NVM targets that may be causing the issue and could be removed. |
| 4 | SAFE_MODE_ DISABLE_ ALL_NVM | 1 | 1 | All data previously saved to NVM is ignored in this state. | See recovery steps for SAFE_MODE_ DISABLE_ NON_ COMMUNICATION_ NVM. |

| Value | State | Safe Mode Error Bit | Reset Loop Detected Bit | Notes | Recovery Steps |
|---|---|---|---|---|---|
| 5 | SAFE_MODE_ DISABLE_ AUTH | 1 | 1 | All data previously saved to NVM and all Auth Codes are ignored in this state. | Use the **AUTH REMOVE** command to remove the offending Auth Code. The **AUTHCODES** log can be used to determine what Auth Codes are currently loaded. |
| 6 | SAFE_MODE_ FAILED | 1 | 1 | All data previously saved to NVM and all Auth Codes are ignored in this state. | This state is unexpected. The recovery steps for other states may apply. |
| 7 | SAFE_MODE_ UNEXPECTED_ MAIN_FIRMWARE | 1 | 0 or 1 | An error related to main firmware loading occurred. | Reload the main firmware. |

## 2.4 OEM6 to OEM7 – SPAN

The commands used to enter translational offsets and rotational offsets have changed in OEM7. For information about translations and rotations, see SPAN Translations and Rotations.

### *2.4.1 Translational Offsets*

Determining the translational offsets in OEM7 is similar to the process in OEM6. Three dimensional distances from IMU to the other SPAN system device (e.g. GNSS antenna) are entered into the SPAN receiver. The differences are the commands used to enter the offsets have changed and the distances can be entered in the IMU frame or the vehicle frame.

See *Table 59: Commands for Entering SPAN Offsets and Rotations* below for the changes in the commands used for entering translational offsets. See the **SETINSTRANSLATION** command (see page 66) for more information.

The OEM6 offset commands required that all translational offsets be entered in the IMU Body Frame, so to translate to the new commands, the same values can be entered within the new structure.

### *2.4.2 Rotational Offsets*

Determining rotational offsets in OEM7 has changed significantly. With the OEM6 commands, rotations usually referenced the SPAN Computation Frame. Now, all rotations are given as the rotation from the IMU Body Frame, to the frame of interest. For IMUs installed with their Z axis pointed upwards, rotation angles will be similar as those used with the legacy commands. For IMUs installed in other orientations, angles will be very different.

See *Table 59: Commands for Entering SPAN Offsets and Rotations* below for the changes in the commands used for entering rotational offsets. See the **SETINSROTATION** command (see page 64) for more information.

**Table 59: Commands for Entering SPAN Offsets and Rotations**

| OEM6 Command | OEM7 Command | Notes |
|---|---|---|
| SETIMUTOANTOFFSET | **SETINSTRANSLATION** ANT1 | |
| SETIMUTOANTOFFSET2 | **SETINSTRANSLATION** ANT2 | |
| SETIMUTOEXTOFFSET | **SETINSTRANSLATION** EXTERNAL | |
| SETIMUTOGIMBALOFFSET | **SETINSTRANSLATION** GIMBAL | |
| SETINSOFFSET | **SETINSTRANSLATION** USER | |
| SETMARKxOFFSET | **SETINSTRANSLATION** MARKx<br>**SETINSROTATION** MARKx | Translational and rotational offsets are now entered separately.<br>Markx can be MARK1, MARK2, MARK3 or MARK4. |

| OEM6 Command | OEM7 Command | Notes |
|---|---|---|
| VEHICLEBODYROTATION<br>SETIMUORIENTATION | **SETINSROTATION**<br>RBV | The information provided by the OEM6 commands is now combined in the RBV rotation. |
| GIMBALSPANROTATION<br>SETGIMBALORIENTATION | **SETINSROTATION**<br>RBM | The information provided by the OEM6 commands is now combined in the RBM rotation.<br>The RBV rotation must also be specified for correct operation. |
| EXTHDGOFFSET | **SETINSROTATION**<br>ALIGN | This offset continues to be automatically calculated if the translational offsets to the primary and secondary GNSS antennas are provided. |
| APPLYVEHICLEBODYROTATION | **SETINSROTATION**<br>RBV | Default attitude output is now in the Vehicle Frame, as long as an RBV rotation is provided. This is overridden if a USER rotational offset is provided. |

### 2.4.3 INS Profiles

INS Profiles are a method to improve the performance of the system in different conditions. When a receiver is configured with an INS Profile, the default receiver settings are modified to optimize performance in the selected condition. See *INS Profiles* on page 145 for more information.

### 2.4.4 INS Seed

The INS Seed functionality is an alignment method whereby INS alignment information from a previous powerup can be injected into the system at startup to achieve an INS alignment very quickly. This is especially useful for systems that previously required a kinematic alignment. See *INS Seed / Fast INS Initialization* on page 148 for more information.

### 2.4.5 ALIGN Calibration

For optimal SPAN performance when using Dual Antenna with SPAN, an ALIGN offset calibration is required for each unique installation. This calibration refines the IMU to antenna baseline angular offset from the initial estimate derived from the input lever arms. Refer to **INSCALIBRATE** command for details.

### 2.4.6 Multi-Line Body to Vehicle Calibration

The Body to Vehicle frame offset calibration feature assumes flat ground when estimating the roll offsets between the IMU Body and Vehicle frames. For applications where vehicle roll must be well known, or pass-to-pass accuracy of a highly offset position is especially important, the Multi-Line IMU Body to Vehicle frame offset calibration routine offers higher calibrated accuracy in roll. This is done by running independent calibrations over the same ground path in opposite directions; averaging the results cancels out the effect of any terrain induced roll. See *Multi-Line Body to Vehicle Frame Rotation Calibration Routine* on page 146 for more information.

### 2.4.7 KVH1750 Baud Rate Conversion

If purchased directly from KVH, the KVH1750 must be configured before it can be used with an OEM7 receiver.

The KVH1750 natively communicates at 921,600 bps, however the maximum baud rate of the COM ports on an OEM7 receiver is 460,800 bps. Thus, KVH1750 IMUs purchased from KVH must be modified to use 460,800 bps. Also, the MSYNC setting on the KVH1750 must be set to EXT.

Though slower, 460,800 bps is more than adequate to transmit the data at the required 200 Hz without any impact to performance.

There are two methods to change the KVH1750 IMU baud rate. If an OEM6 receiver is available, a command is available that will allow the receiver to change the IMU baud rate. If not, a direct RS-422 connection at 921,600 baud is required to change the IMU baud rate.

#### Connect Directly to the IMU

This method requires a direct connection to the IMU at 921,600 baud over RS-422. Make sure the RS-422 link used is capable of 921,600 baud as not all serial ports are capable of that rate.

1. Open a command terminal at 921,600 baud via an RS-422 serial link.

   By default, the IMU will output 0.5 Hz binary messages which will help confirm you are connected properly.
2. Issue the following two commands in sequence to enter configuration mode and change the baud rate.

   =CONFIG,1

   =MSYNC,EXT

   =BAUD,460800
3. Upon completion of these commands, power cycle the IMU to boot up in the new baud rate.
4. Confirm the change by connecting to the IMU at 460,800 baud and verifying the incoming messages.

The new baud rate configuration is saved to the IMU NVM automatically so the process is complete and the IMU is ready for use with an OEM7 receiver.

#### Use an OEM6 Receiver Command

OEM6 firmware as of version OEM060631RN0000 (Dec 2016) includes the **IMUCONFIGURATION** command that internally configures the KVH1750 IMU for SPAN communication. Three sets of configuration options are available, two of which allow changing the baud rate of the IMU to either 921,600 or 460,800. This provides customers upgrading to OEM7 an easy method to modify the baud rate. To change the IMU to 460800 baud, use the following procedure:

1. Connect the KVH-1750 IMU to an RS-422 capable OEM6 receiver communication port.
2. Issue the following command.

   ```
   CONNECTIMU COM# IMU_KVH_1750
   ```
3. Issue the following command.

   ```
   IMUCONFIGURATION IMU_KVH_1750 2
   ```

The **IMUCONFIGURATION** command can be used to configure a KVH17xx IMU depending on the value of the Option field.

```
IMUCONFIGURATION IMUType [Option]
```

| IMU Type | Option | Configuration Details |
|---|---|---|
| IMU_KVH_1750 | 0 | Configure KVH options for NovAtel communication. |
| | 1 | Change KVH baud rate to 921,600 |
| | 2 | Change KVH baud rate to 460,800 |

### *2.4.8 INS Profiles*

INS Profiles provide two major functions; simplified configuration and enhanced performance. Each INS Profile sets specific filter behavior to enhance system performance in particular environments.

Basic enhancements provided by the INS Profiles are available to all SPAN enabled models and allow for simple configuration rather than sending many discreet commands.

The enhanced profile settings are enabled by the receiver model. The enhanced profile adds enhanced profile behavior such as Dead Reckoning for land and Heave for marine. A receiver model with an INS Mode of **P** or **R** is required to enable Enhanced INS Profiles.

**Table 60: OEM7 INS Profiles**

| Profile | Description | Basic Profile Settings | Enhanced Profile Settings |
|---|---|---|---|
| DEFAULT | The default SPAN profile.<br>This profile has legacy SPAN default settings. | None | N/A |
| LAND | INS Profile for fixed axle land vehicles. Enables robust kinematic alignment routines and configures the update profile. | Enables direction detection on kinematic alignment routine<br>Enables course over ground attitude updates | Enables intelligent vehicle dynamics modeling for land vehicles (Dead Reckoning) |
| MARINE | INS profile for marine vessels. Enacts changes to disable static behavior and make the marine solution more robust. | Disable Zero Velocity Updates<br>Disable turn-on bias estimation | Enables Heave algorithms |
| FIXEDWING | INS profile for fixed wing aircraft | None | None |
| FOOT | INS profile for walking or backpack applications | None | None |
| VTOL | INS profile for Vertical Take Off and Landing aircraft (helicopter, quadcopter, etc.) | None | None |

| Profile | Description | Basic Profile Settings | Enhanced Profile Settings |
|---|---|---|---|
| RAIL | INS profile for railway applications | Enable direction detection on kinematic alignment routine<br>Enable course over ground attitude updates | Enables intelligent vehicle dynamics modeling for rail vehicles (Dead Reckoning) |
| AGRICULTURE | INS profile for agriculture applications | Enables direction detection on kinematic alignment routine<br>Enables course over ground attitude updates | Enables intelligent vehicle dynamics modeling for agricultural vehicles (Dead Reckoning) |

#### Use

To use an INS Profile, send the **SETINSPROFILE** command at startup (or save to NVM) to activate the mode. For example:

```
SETINSPROFILE LAND
```

The type of profile activated, basic or enhanced, is determined by the SPAN model.

The currently selected profile option is available in the **INSCONFIG** log.

#### RBV Calibration

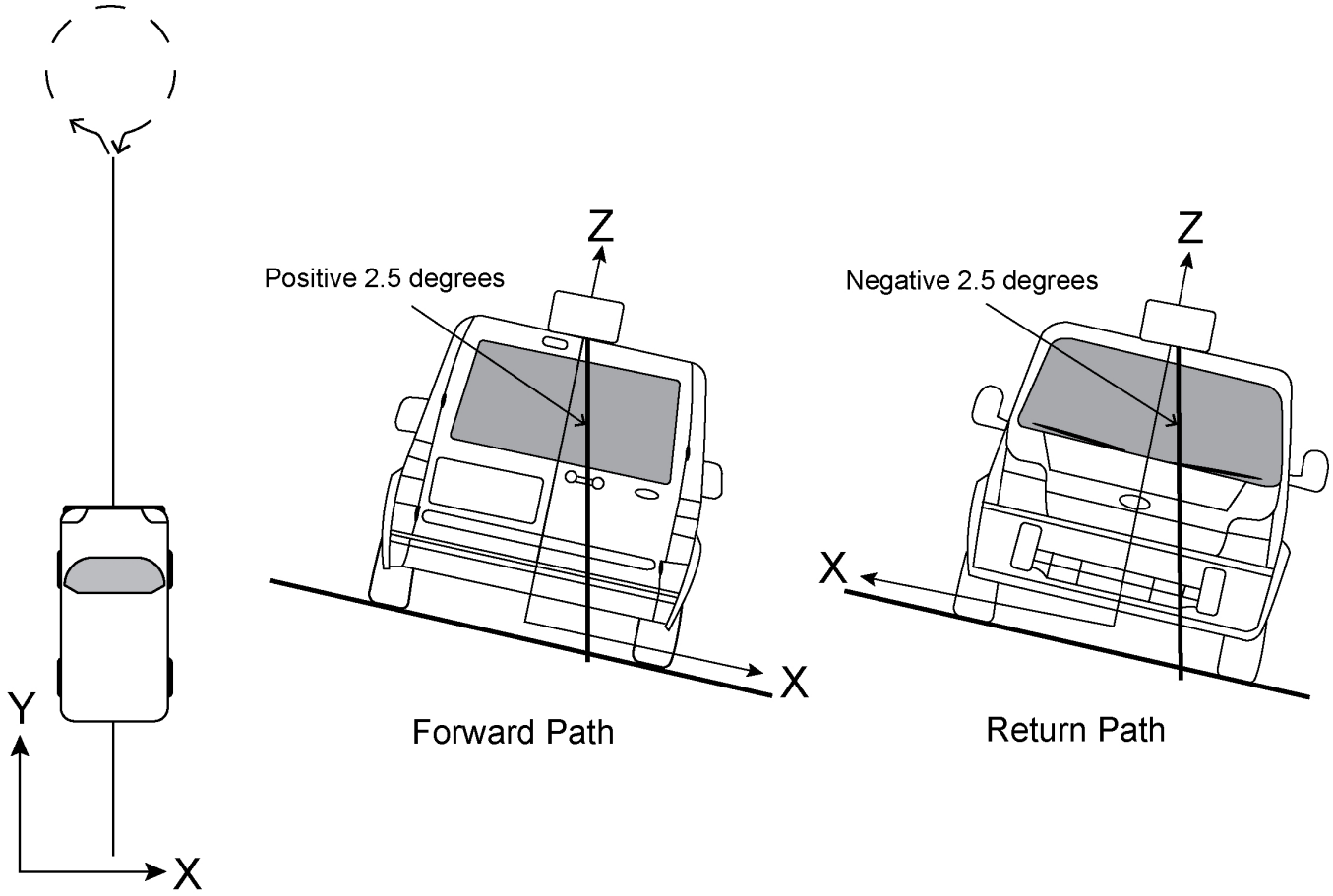
For the profiles to provide the greatest benefit, the rotation offset of the IMU body to the vehicle frame (RBV) should be known as accurately as possible. Since constraints happen at the vehicle level, any error in the offset will translate into worse performance. These parameters are estimated in the background when the system is fully converged, however it is always better to have this measured as precisely as possible before mission critical data is collected.

The INSCALIBRATE RBV functionality can be used to estimate and save these offsets to NVM in a controlled manner.

### *2.4.9 Multi-Line Body to Vehicle Frame Rotation Calibration Routine*

The single pass Body to Vehicle frame offset calibration feature assumes flat ground when estimating the roll offsets between the IMU Body and Vehicle frames, as this component of the rotational offset is difficult to observe through typical ground vehicle motion. In practice this can result in an outstanding roll offset error as high as several degrees. For applications where vehicle roll must be well known, or pass-to-pass accuracy of a highly offset position is especially important, the Multi-Line IMU Body to Vehicle frame offset calibration routine offers higher calibrated accuracy in roll. This is done by running independent calibrations over the same ground path in opposite directions; averaging the results cancels out the effect of any terrain induced roll. A minimum of two iterations are required. Additional iteration pairs can be executed for increased reliability.

**Figure 1: Multi-Line IMU Body to Vehicle Calibration**




The steps for the calibration routine are:

1. Apply power to the .
2. Configure the SPAN system, see .
3. Perform an initial system alignment using one of the methods described in System Start-Up and Alignment Techniques. Ensure the system reaches Solution Good.
4. Enable a new calibration using the **INSCALIBRATE** command, with the optional standard deviation target omitted:

   ```
   INSCALIBRATE RBV NEW
   ```

5. Start to move the vehicle. Drive a straight course, at a vehicle speed greater than 5 m/s (18 km/hr). A distance of approximately 250 meters is recommended.

   To monitor the calibration, log **INSCALSTATUS** using the ONCHANGED trigger.

6. When the end of the available distance is reached, stop the vehicle. Halt the ongoing calibration by sending the **INSCALIBRATE** command again:

   ```
   INSCALIBRATE RBV STOP
   ```

7. At this point **INSCALSTATUS** will report a source status of CALIBRATED. Turn the vehicle around and come to a stop on the finishing point of the previous line.
8. Initiate the second pass of the calibration by issuing the **INSCALIBRATE** command with the **ADD**

parameter:

```
INSCALIBRATE RBV ADD
```

9. Drive the vehicle back along the original course, until you reach the starting point for the first pass. Stop the vehicle, and re-issue the **INSCALIBRATE** command, with the **STOP** parameter. **INSCALSTATUS** will again indicate a source status of CALIBRATED.

   At this point the multi-line calibration can be considered complete, but more iterations can be repeated until the desired accuracy level is reached.

10. When the calibration is complete, the calibrated offset value can be viewed in either the **INSCALSTATUS** log or the **INSCONFIG** log.

To save a calibrated rotation for subsequent start ups, issue the **SAVECONFIG** command after calibration is complete. Each time the IMU is re-mounted this calibration should be performed again.

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

For information about the logs and commands used in this procedure, refer to the OEM7 Commands and Logs Reference Manual.

### 2.4.10 INS Seed / Fast INS Initialization

The INS Seed functionality is an alignment method whereby INS alignment information from a previous powerup can be injected into the system at startup to achieve an INS alignment very quickly. This is especially useful for systems that previously required a kinematic alignment.

When INS Seed is enabled, alignment data is automatically saved when appropriate, with no specific action required from the user. During subsequent power ups, the saved solution is compared to the current position of the system, and if valid, the saved solution is used to immediately initialize the inertial filter, making it ready for use.

Error model information is also saved with the seed data. Even if full alignment cannot be achieved, this information will still be used to aid filter convergence, although another method of alignment will be required.

This feature is enabled via the **INSSEED** command. This command setting must be saved to NVM via the **SAVECONFIG** command to be used upon next system boot-up.

#### Criteria

For the INS seed functionality to be enabled, the **INSSEED ENABLE** command must be sent.

#### Saving

The following criteria must be met for valid alignment data to be saved into receiver NVM.

- The INS solution status must be converged, as indicated by the INS solution convergence flag in the inertial Extended Solution Status.
- The system velocity is less than 0.2 m/s.

When these conditions are met, the required information is automatically saved to NVM. If these conditions cease to be met, the saved alignment data is automatically discarded. If the alignment data is discarded, error model information is automatically retained as appropriate.

### Use at Boot-up

Upon boot, several checks are done to verify that any seed data existing in NVM is appropriate for use. These checks require a GNSS position to be computed, so an NVM seed will never be injected prior to achieving a GNSS position. The following criteria must be met in order to use a seed:

- Seed data from NVM must be valid
- The system must be stationary
- The system must not have moved significantly from the seed position: <10 meters position and 10 degrees in roll and pitch

If a valid error model is present in the seed data from NVM, this will always be used, even if the system is not stationary or exceeded the movement thresholds.

**INJECT Option (Advanced Users Only)**

There is an advanced option available to skip the second and third validation steps described in the boot-up section above. This can be used if GNSS is not available on power-up or if speed to achieve an alignment is imperative.

This option is for advanced users only. Forcing an injection of an NVM seed without the validation checks can cause an unstable INS solution if the vehicle has moved.

Apart from injecting the seed at startup without validation, this will also remove some requirements typically asserted to allow the filter to converge. This will allow the system to run without GNSS adequately at startup.

## Example Usage

Enabling functionality after FRESET:

1. Send the following command.

```
INSSEED ENABLE VALIDATE
```

2. Send the following command.

```
SAVECONFIG
```

Saving valid INS information:

1. Operate the system as normal, until the filter completes convergence.
2. Bring the system to a complete stop before powering off or resetting.

Using valid INS information on start-up:

1. Remain static while initializing.
2. Wait for the system to receive a valid position from GNSS.
3. If valid, alignment data will be injected.

## System Indicators

The **INSSEEDSTATUS** log reports the seed injection status for the current power up and information on current seed validity. Refer the OEM7 Commands and Logs Reference Manual for more information about the **INSSEEDSTATUS** log.

The Extended Solution Status field has bits that indicate the Alignment type that occurred and NVM seed injection status.

**Alignment Type: Bits 26-28**

These bits indicate how the INS system reached INS_ALIGNMENT_COMPLETE. When the INS system aligns using saved INS solution data from NVM, bits 26-28 will be set to 101

**NVM Seed Status: Bits 29-31**

These bits indicate the current status of the seeding process at start-up, as described in the table below:

**Table 61: NVM Seed Indication**

| Bit 31-29 Values[1] | Hex Value | NVM Seed Type |
|---|---|---|
| 000 | 0x00 | NVM Seed Inactive |
| 001 | 0x01 | Seed stored in NVM is invalid |
| 010 | 0x02 | NVM Seed failed validation check |
| 011 | 0x03 | NVM Seed is pending validation (awaiting GNSS) |
| 100 | 0x04 | NVM Seed Injected (includes error model data) |
| 101 | 0x05 | NVM Seed data ignored due to a user-commanded filter reset or configuration change |
| 110 | 0x06 | NVM Seed error model data injected |

1. The values from left to right are Bit 31, Bit 30 and Bit 29.

# Chapter 3 OEM6 to OEM7 – Hardware

The following OEM7 GNSS receivers are discussed in the following sections.

- OEM719
- OEM729

## 3.1 Physical changes

Two of the OEM7 receivers are form factor replacements for OEM6 receivers. The OEM719 replaces the OEM615 and the OEM729 replaces the OEM628. Form factor replacement means that the OEM7 receivers:

- have the same physical dimensions as the OEM6 receivers
- have the same keep-out zones as the OEM6 receivers
- use the same locations for the mounting holes as the OEM6 receivers
- use the same connectors as the OEM6 receivers
- use the same locations for the connectors as the OEM6 receivers

The physical differences between the OEM7 and OEM6 receivers are described in the following sections.

### *3.1.1 Shielding*

The OEM719 and OEM729 have an additional shield compared to the OEM615 and OEM628. However, all of the shields on the OEM719 are within the keep-out zone defined for the OEM615 and all of the shields on the OEM729 are within the keep-out zone defined for the OEM628. Designs that respected the keep-out zones for the OEM6 receivers will not be affected by the shields on the OEM719 and OEM729.

For information about the OEM615 and OEM628 keep-out zones, see the OEM6 Family Installation and Operation User Manual (OM-20000128). For information about the OEM719 and OEM729, see *Figure 10: OEM719 Keep-outs* on page 179 and *Figure 17: OEM729 Keep-outs* on page 198.

### *3.1.2 Mounting*

The OEM719 and OEM729 can be mounted using standoffs or bosses that have the same spacing required for OEM6 receivers. See *Figure 9: OEM719 Dimensions* on page 178 and *Figure 16: OEM729 Dimensions* on page 197 for the exact spacing.

OEM7 receivers also have an area on the edge of the cards to allow the use of mounting rails. In *Figure 15: OEM719 Mounting Surface* on page 184 and *Figure 18: OEM729 Mounting Surfaces* on page 199, this is the area outside of the keep-out zone. On the OEM719, this area is 1.75 mm wide. On the OEM729, this area is 2.5 mm wide.

Using rails to mount the OEM7 receiver improves the thermal and vibration performance of the receiver. Securing the OEM7 receiver to mounting rails using clamping bars provides the most secure configuration for aggressive thermal and vibration use cases.

For more details about mounting OEM7 receiver cards, refer to the *OEM7 Receiver Card Mechanical Integration Application Note* (D19021).

### *3.1.3 Weight*

The OEM7 receivers are slightly heavier than the OEM6 receivers they replace.

| Receiver | Weight | | Receiver | Weight |
|---|---|---|---|---|
| OEM615 | 24 g | | OEM628 | 37 g |
| OEM719 | 31 g | | OEM729 | 48 g |

## 3.2 Electrical changes

### *3.2.1 COM Port Data Rate*

The maximum data rate for COM ports has changed. On OEM6 receivers, the maximum data rate was 921,600 bps. On OEM7 receivers, the maximum data rate is 460,800 bps.

### *3.2.2 Power Supply*

The power consumption of OEM7 receivers is dependent on receiver configuration and feature use. Enabling more features (up to 555 channels, interference resistance, etc) will increase the power required.

- The OEM719 power consumption is approximately 10% higher than a similarly configured OEM615.
- The OEM729 power consumption is approximately 10% lower than a similarly configured OEM628.

A monotonic rise of the supply voltage input is required to guarantee a proper power-on reset sequence. The maximum rise time is 100 ms.

Bulk supply bypassing (approx 10 µF) and high-frequency bypassing (0.1 µF and 220 pF) near the supply pin is recommended for optimal performance.

On the OEM7 receiver cards, there is an approximate effective input bulk capacitance of 100 µF at room temperature. This is made up of ceramic capacitors with a voltage rating of 6.3 VDC and tolerance of 20%.

OEM7 receivers monitor the supply voltage. If the supply voltage falls outside of the normal operating range, a warning or error is provided in the RXSTATUS log. See the **RXSTATUS** log on page 126 for more information.

#### Inrush Current

The inrush current behavior of the OEM7 receivers is different than the OEM6 receivers.

| OEM6 Receiver | Inrush Current (Typical) | | OEM7 Receiver | Inrush Current (Typical) |
|---|---|---|---|---|
| OEM615 | 6.0 A for less than 60 µs | | OEM719 | 1.71 A for less than 1.5 ms |
| OEM617 | 6.0 A for less than 60 µs | | OEM729 | 2.0 A for less than 1.8 ms |
| OEM617D | 6.0 A for less than 60 µs | | OEM7700 | 2.0 A for less than 1.8 ms |
| OEM628 | 6.6 A for less than 60 µs | | OEM7720 | 2.0 A for less than 1.8 ms |
| OEM638 | 12.0 A for less than 60 µs | | OEM7600 | 2.0 A for less than 1.8 ms |

### *3.2.3 USB Interfaces*

The layout guidelines are identical to those provided for OEM6, but the recommended common-mode choke value has changed. See *USB Interface* on page 161.

The OEM719 and OEM729 USB interfaces operate at 12 Mb/s ("Full-Speed") operation for backwards-compatibility reasons.

### 3.2.4 Input and Output lines

The voltage limits and drive currents on the input and output lines of the OEM7 receivers are different than the levels on the OEM6 receivers. See *Table 100: OEM719 Strobe Electrical Specification* on page 190 and *Table 109: OEM729 Strobe Electrical Specifications* on page 206 for details.

### 3.2.5 LNA Power

If a short circuit or other problem causes an overload of the current supplied to the antenna, the receiver hardware shuts down the power supplied to the antenna. To restore power, power cycle the receiver. The Receiver Status word, available in the **RXSTATUS** log (see page 126), provides more information about the cause of the problem.

**OEM719**

On OEM615 receivers, LNA power is provided by an external power supply connected to pin 1 of the main connector.

On OEM719 receivers, LNA power is generated on the receiver and pin 1 of the main connector has no electrical connection on the board. The OEM719 provides up to 200 mA of current at +5 VDC. The LNA is generated from the 3.3 V supply; therefore the OEM719 will draw more power on the 3.3 V rail than a similarly configured OEM615.

The typical current draw for a NovAtel active antenna is <40 mA. For a typical antenna, this may increase the power drawn from the 3.3 V supply by as much as 70 mA (as the LNA supply is approximately 85-90% efficient).

**OEM729**

On OEM628 receivers, the voltage provided to the LNA could be changed from the default of +5 VDC to +3.3 VDC using the ANTENNAPOWER command.

The voltage for the LNA power is not selectable on OEM729 receivers, but the OEM729 can provide more current to the LNA. The OEM729 provides up to 200 mA of current at +5 VDC.

## 3.3 Environmental changes

### 3.3.1 Thermal dissipation

To ensure functionality and reliability, the OEM7 receiver cards must operate within the specified ambient air temperature limits (-40°C to +85°C).

The OEM7 receivers have been designed to efficiently transfer heat from the receiver components into the printed circuit board. Mounting the OEM7 receiver on rails, or attaching heat sinks to the mounting areas, will transfer the heat from the receiver card to adjacent circuit boards, the enclosure or the air.

The mounting area is the area on the sides of the receiver that are outside of the keep-out zone. See *Figure 15: OEM719 Mounting Surface* on page 184 and *Figure 18: OEM729 Mounting Surfaces* on page 199.

The best thermal mitigation comes from sinking heat through the mounting rails. This provides more effective thermal relief than attaching a heat sink to the MINOS.

For more information about thermal dissipation, refer to the *OEM7 Receiver Card Mechanical Integration Application Note* (D19021).

### 3.3.2 Vibration

On OEM6 receivers, the random vibration specification is dependent on the receiver variant. Standard OEM6 receivers were rated to 7.7g RMS (MIL-STD_810G Method 514.6, Category 24). High vibration OEM6 receivers were rated to 20g RMS (MIL-STD_810G Method 514.6, Category 24).

All OEM7 receivers are rated to 20g RMS (MIL-STD_810G Method 514.6E-1, Category 24). However, for high vibration installations, special considerations are required.

For more information about vibration considerations, refer to the *OEM7 Receiver Card Mechanical Integration Application Note* (D19021).

## 3.4 Interference

Typical GNSS signal levels are in the order of -130 dBm. Spurious emissions from digital logic (especially a microprocessor) or a nearby radio transmitter can easily meet the appropriate EU or FCC emission limits and still be several orders of magnitude above the level of the GNSS signals.

These emissions may desensitize the GNSS receiver and lead to performance degradation (low carrier to noise ratios, position accuracy, or loss of lock in extreme cases).

There are two bands of interest:

- Low Band (1164 MHz to 1300 MHz): contains GPS L2, GPS L5, GLONASS L2, BeiDou B2, BeiDou B3, Galileo E5a, Galileo E5b, Galileo E5 and Galileo E6.
- High Band (1525 MHz to 1610 MHz): contains GPS L1, GLONASS L1, BeiDou B1, Galileo E1, and L-Band.

OEM7 receivers are designed to mitigate RF interference on the GNSS card itself. However, if an emission (fundamental or harmonic) from equipment co-located with the GNSS receiver or antenna falls into one of these bands, there are a few things that can be done to ensure good system performance:

- Place radio transmitters and their associated antennas as far away from the GNSS antenna and OEM7 card as possible (for example, a cellular radio antenna).
- Provision for RF shielding over all high-speed digital logic circuitry (even if the design meets the required emissions regulations). A CPU operating at 400MHz will produce an emission at 1200MHz (3rd harmonic) and another at 1600MHz (4th harmonic) – the emissions may be strong enough to affect the GNSS system performance. Similar provisions should be made for any RF circuitry near the GNSS receiver or antenna.
- Run the OEM7 antenna coaxial cable away from high-speed digital logic, other radio sections and high-current nets (as in switching power supplies).

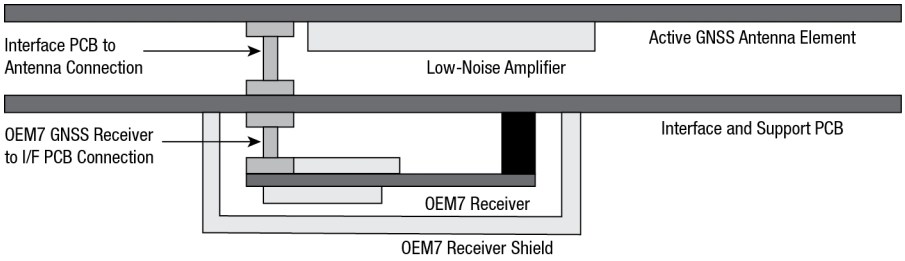
### 3.4.1 Interference from Co-Located LNA

Often a GNSS receiver is placed in the same enclosure as a GNSS antenna and LNA to create an all in one GNSS receiver product (also referred to as a *smart antenna*).

The close proximity of a high-gain amplifier to the GNSS receiver may cause some low-level near-field energy from the receiver to be amplified and fed back into the receiver, causing C/No degradation to one or more constellations.

It is strongly advised in a smart antenna design that the receiver be enclosed in a shield to prevent this near-field energy from coupling into the antenna. This includes connecting the GNSS receiver input ground to the shield surrounding the receiver, similar to the following image. It is also strongly advised to use mounting rail for receiver cards used in this configuration.

**Figure 2: SMART Antenna Interference Example**




The previous figure shows one method of shielding the receiver.

The rail mounting provision on the edges of the receiver card can also used to attach entire board shielding.

## 3.1 Receiver Card Interface Examples

The OEM7 receiver cards provide a number of 3.3V CMOS-level I/O pins for status indication and timing:

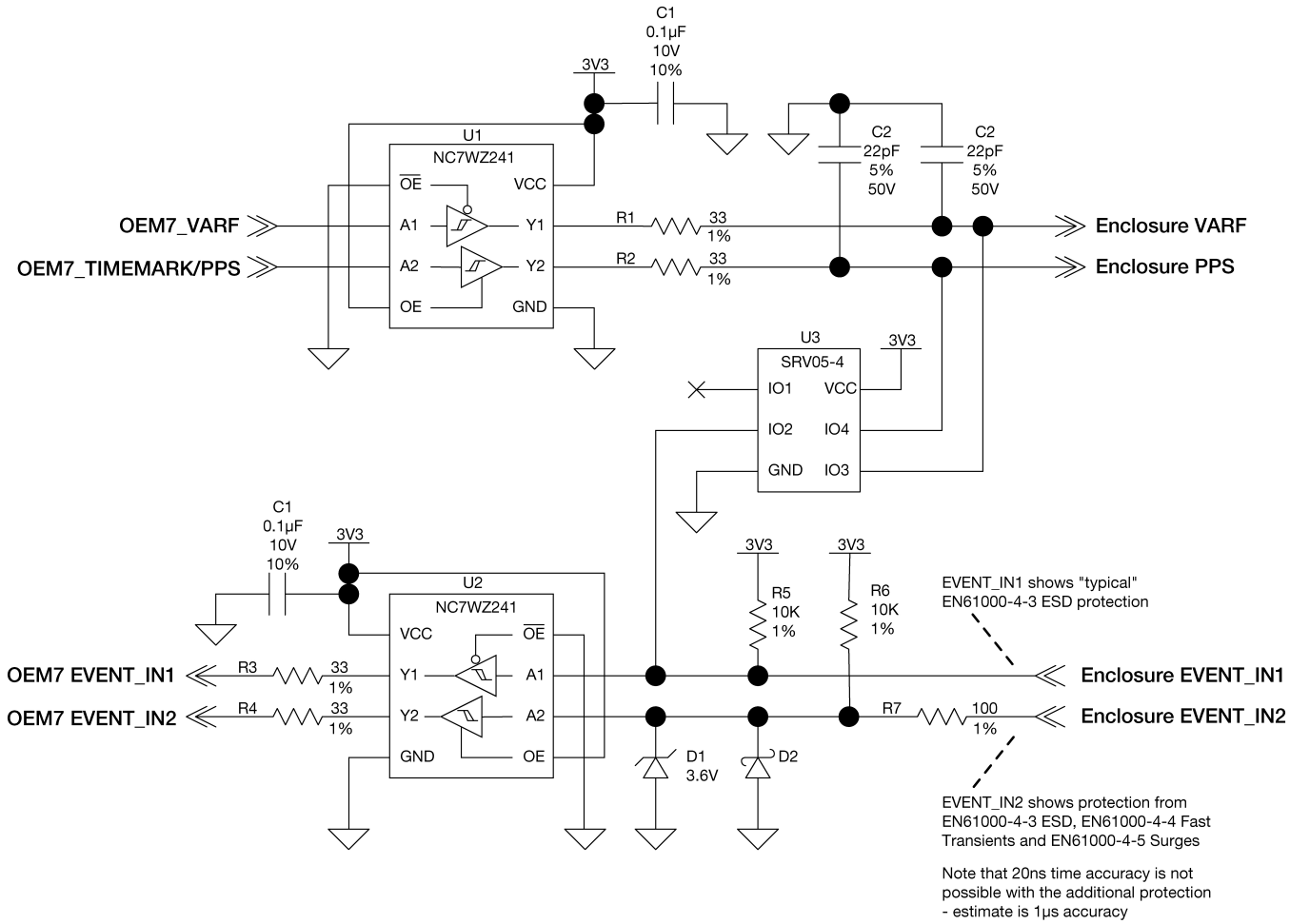
- **COMx**: CMOS-level UART ports
- **Ethernet**: 10/100 Ethernet port
- **CAN1** and **CAN2**: CMOS-level CAN ports (external CAN transceivers required)
- **PPS**: Output pulse providing time reference signal (software configurable output rate, defaults to 1 Pulse Per Second)
- **EVENTx_OUT**: Variable Frequency output (a software-configurable clock output similar to PPS, off by default). (The EVENT1_OUT signal may also be referred to as an VARF.)
- **PV**: Position Valid (This signal drives high when the receiver has calculated a valid position)
- **EVENTx_IN**: Event inputs (with configurable polarity)
- **USERIO**: A user defined general purpose input or output (OEM729 only)

These I/O require additional ESD protection if they are to be routed to enclosure connectors. A protection circuit similar to the following examples must be used on any OEM7 CMOS-level signals that will be routed to an enclosure connector. The ferrite bead and small-value capacitor provide some immunity to electrostatic discharge events, but also serve to reduce radiated and conducted emissions from the enclosure.

### *3.4.2 EVENT_IN, EVENT_OUT and PPS Signal Protection*

The following schematic shows appropriate signal conditioning for EVENT_IN (MKI), EVENT_OUT (VARF) and PPS (TIMEMARK) signals.

**Figure 3: Protection and Buffering for EVENT_IN, PPS and EVENT_OUT signals**




The buffers (U1, U2) are chosen to provide additional drive strength for long cables. The series terminations (R1, R2, R3, R4) are present to allow for signal integrity adjustments (to reduce overshoot/undershoot, etc.). The passive component values may require tuning, depending on the application.

The TVS (U3) provides protection from ESD/EFT events and some induced power line surges. There are two implementations shown for the EVENT_IN signals. EVENT_IN1 is shown with basic ESD/EFT protection, while EVENT_IN2 is shown with additional surge protection for harsh environments (see D1, R7, D2, D3 and R9). If the additional surge protection is to be used on a design, the buffer must be included as well. This design has been used to protect EVENT_IN signals from EN61000-4-5 induced surges up to 2kV on several NovAtel enclosure products. If nanosecond-level timing is critical to the application, this protection circuit may be unsuitable, as it causes a timing shift of a few microseconds to the EVENT signal (heavily dependent on temperature).

The 10 kΩ pullups (R5, R6) are required if the buffers are used on the EVENT_IN lines. If the buffers are not used, the OEM7 cards have built-in 10 kΩ pullup resistors – external pull resistors are not required.

**Table 62: EVENT_IN, EVENT_OUT and PPS Pin Designation**

| | OEM719 | OEM729 | OEM7600 | OEM7700 | OEM7720 |
|---|---|---|---|---|---|
| **Connector** | P1701 | P1802 | P1701 | P2001 | P1901 |

| | OEM719 | OEM729 | OEM7600 | OEM7700 | OEM7720 |
|---|---|---|---|---|---|
| **EVENT_IN1** | 9[1] | 8 | 30 | 30 | 30 |
| **EVENT_IN2** | 7[2] | 7[3] | 29 | 29 | 29 |
| **EVENT_IN3** | – | – | 32 | 32 | 32 |
| **EVENT_IN4** | – | – | 31 | 31 | 31 |
| **PPS** | 19 | 4 | 24 | 24 | 24 |
| **EVENT_ OUT1/VARF** | 6[4] | 3 | 17 | 17 | 17 |
| **EVENT_OUT2** | – | – | 26 | 26 | 26 |
| **EVENT_OUT3** | – | – | 23 | 23 | 23 |
| **EVENT_OUT4** | – | – | 25 | 25 | 25 |

**Table 63: Bill of Materials (critical components)**

| Designator | Manufacturer | Manufacturer Part Number |
|---|---|---|
| D1 | On Semiconductor | MMSZ5227BT1G |
| D2 | Comchip Technology | CDBU00340 |
| U1, U2 | Fairchild Semiconductor | NC7WZ241L8X |
| U3 | Semtech | SRV05-4A.TCT |

### *3.4.3 Position Valid (PV) LED Driver*

The PV signal may be used to indicate that the receiver card has computed a valid position. Many NovAtel enclosure products use it to drive a green LED on the enclosure. It may also be used to monitor the status of the receiver with an external microcontroller.

If the LED requires less than 6 mA to drive, and can be operated from a 3.3 V supply, the PV pin may be used to drive the LED directly (with an appropriate current-limiting resistor). If the LED must be driven from a different supply voltage or requires more than 6 mA, a buffer must be used.

A simple buffer – suitable for driving a high-brightness LED from a 5 V supply – is shown below. The value for R2 was selected to limit the LED current to 10 mA (assuming $V_{CE(Q1,sat)}$=0.25 V, $V_{FWD(D1)}$=3.4 V). This circuit will not compensate for the forward voltage change over temperature (and the resulting brightness changes).

[1]This pin is multiplexed with the COM3 transmit signal and is not available if COM3 is enabled.

[2]This pin is multiplexed with the CAN1 transmit signal and is not available if CAN1 is enabled.

[3]This pin is multiplexed with the COM3 receive signal and is not available when COM3 is enabled. See *P1802 Main Connector 24-Pin Header* on page 207 for more information.

[4]This pin is multiplexed with the CAN1 receive signal and is not available if CAN1 is enabled.

**Figure 4: OEM7 Buffer for Driving High-Brightness LEDs from PV**

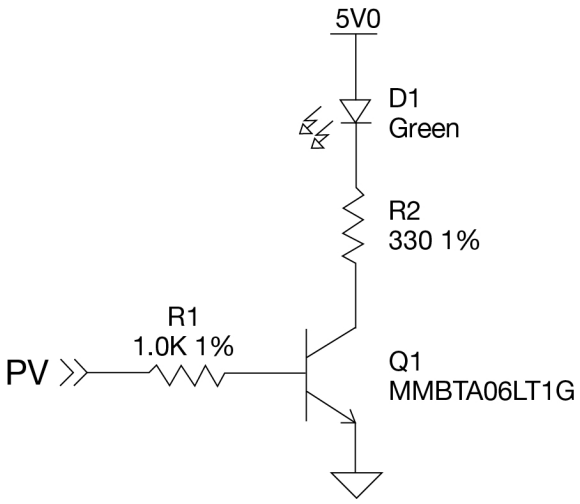



**Table 64: PV Pin Designation**

| | OEM719 | OEM729 | OEM7600 | OEM7700 | OEM7720 |
|---|---|---|---|---|---|
| **Connector** | P1701 | P1802 | P1701 | P2001 | P1901 |
| **PV** | 17 | 10 | 22 | 22 | 22 |

**Table 65: PV LED Driver Bill of Materials (critical components)**

| Designator | Manufacturer | Manufacturer Part Number |
|---|---|---|
| Q1 | ON Semiconductor | MMBTA06LT3G |
| | Fairchild Semiconductor | MMBTA06LT1G |
| | Diodes Inc | MMBTA06-7-F |
| D1 | OSRAM | LT L29S-N1R2-25-Z |

### *3.4.4 Communication Ports*

The COM ports on OEM7 receivers (except COM1 on the OEM729) are CMOS-level I/O pins only. These ports require the addition of an RS-232 or RS-422 transceiver to provide appropriate signal levels. Most RS-232 or RS-422 transceivers will provide some ESD protection (for harsh environments, additional protection may be required).

COM1 on the OEM729 uses RS-232 protocol by default. This port can be changed to RS-422 protocol using the **SERIALPROTOCOL** command.

A combination of a series ferrite bead and small-value shunt capacitor is recommended on any RS-232/R-S422 lines that leave the enclosure (similar to the arrangement used on the I/O ports and CAN interfaces). Many of the available transceivers provide protection for ESD events to ±15kV – if additional protection is required, an external TVS with a working voltage of ±15V to ±25V will be required.

### 3.4.5 CAN Controller Ports

OEM7 receivers provide 3.3 V CMOS-level CAN controller ports. An external CAN transceiver is required. The following figure shows a typical CAN transceiver implementation.

The combination of ferrite beads and small-value capacitors are not necessarily required but may provide improved EMI performance. A low-capacitance TVS device is shown on the schematic to provide ESD protection.

**Figure 5: OEM7 CAN Transceiver Example**

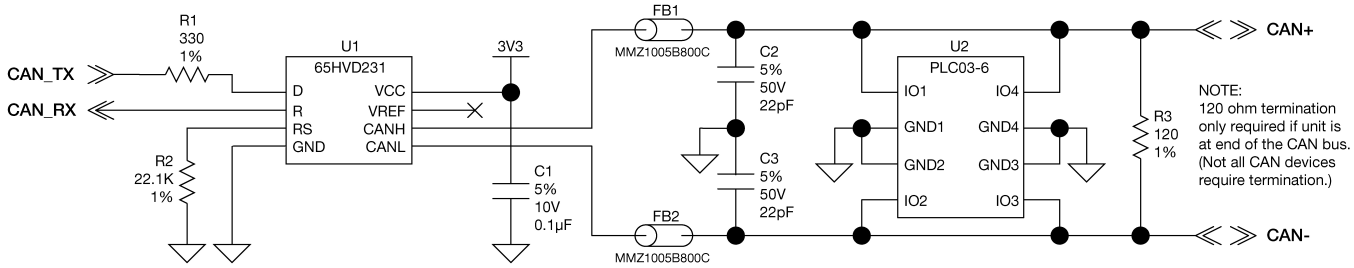



**Table 66: CAN Transceiver Pin Designation**

| | OEM719 | OEM729 | OEM7600 | OEM7700 | OEM7720 |
|---|---|---|---|---|---|
| **Connector** | P1701[1] | P1803 | P1701 | P2001 | P1901 |
| **CAN1TX** | 7 | 10 | 36 | 36 | 36 |
| **CAN1RX** | 6 | 11 | 38 | 38 | 38 |
| **CAN2TX** | 20 | 12 | 37 | 37 | 37 |
| **CAN2RX** | 8 | 13 | 35 | 35 | 35 |

The 120Ω termination resistor should only be used when the CAN device is used at one end of the CAN bus. Multiple terminations along the length of the CAN bus will degrade performance for all CAN devices on that bus.

The slew rate adjustment resistor (R2) value shown sets the slew rate for applications for SAE J1939 agricultural applications. Other applications may require a different slew rate. Refer to the transceiver data sheet for more information.

**Table 67: CAN Transceiver Example Bill of Materials (critical components)**

| Designator | Manufacturer | Manufacturer Part Number |
|---|---|---|
| FB1, FB2 | TDK | MMZ1005B800C |
| U1 | Texas Instruments | SN65HVD231QD |
| U2 | Bourns | CDNBS08-PLC03-6 |

[1]Pins 6 and 7 are multiplexed with other signals and may not be available when certain features are enabled. See *OEM719 Interface Connector* on page 191 for more information.

### 3.4.6 USB Interface

This section describes the interface requirements for an OEM7 receiver card USB port. For general information about the OEM7 receiver card USB ports, see USB Ports.

OEM7 receivers include an USB 2.0 device interface set to either Full-Speed (12 Mb/s) or High-Speed (480 Mb/s) transfer rate. On the OEM719 and OEM729 receivers, the transfer rate for the USB Device port is Full-Speed (12 Mb/s). The USB device interface on the OEM7600, OEM7700 and OEM7720 receivers use the High-Speed (480 Mb/s) transfer rate. The device interface will not auto-negotiate the speed with the Host computer.

It is imperative for signal integrity and EMI reasons that the differential data traces be routed as a 90Ω differential pair. Use of a small-value common-mode choke (as shown in *Figure 6: OEM7 USB Device Interface Example* below) may improve the radiated emissions performance (but should not be necessary).

Any stubs on the traces must be kept as short as possible and it is strongly recommended not to change reference planes. Match the USB differential pair (D+ and D- trace lengths) to within 3 mm or less.

The common-mode choke and ESD protection should be placed as close as possible to the USB connector (J1).

**Figure 6: OEM7 USB Device Interface Example**

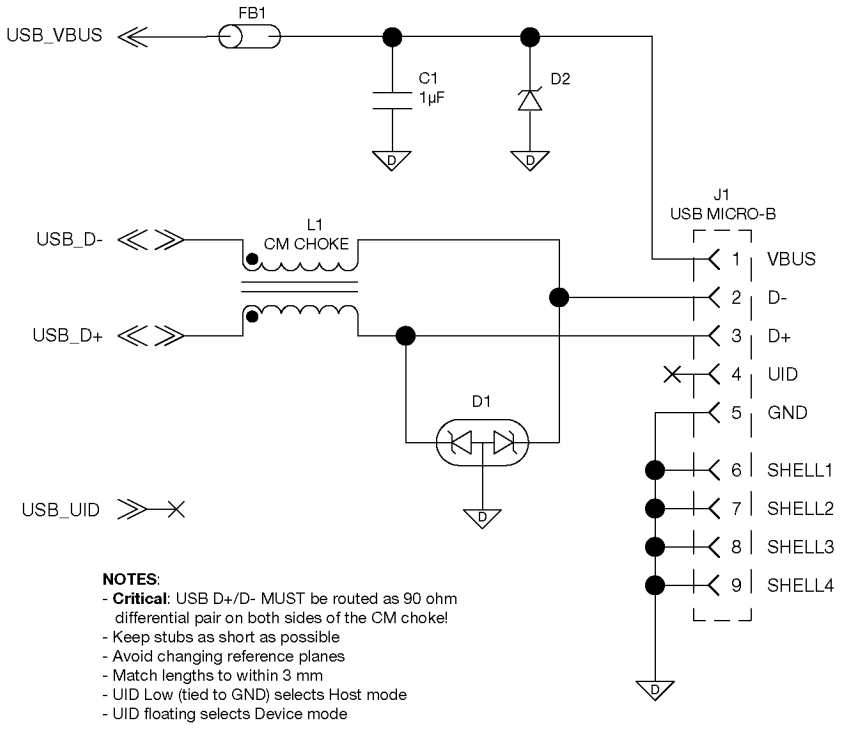



**Table 68: USB Device Interface Pin Designation**

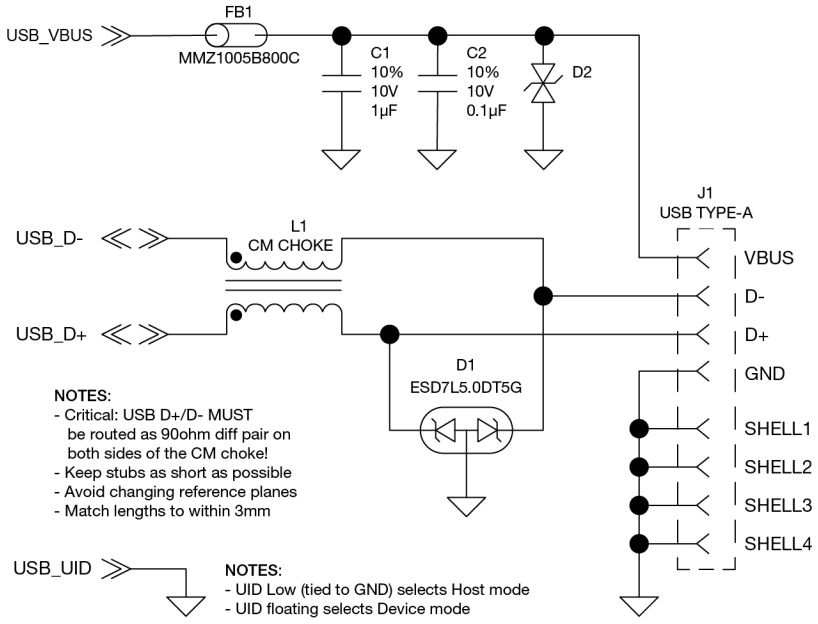
| | OEM719 | OEM729 | | OEM7600 | OEM7700 | OEM7720 |
|---|---|---|---|---|---|---|
| **Connector** | P1701 | P1802 | P1803 | P1701 | P2001 | P1901 |
| **USB0_D-** | 3 | 21 | - | 50 | 50 | 50 |

| | OEM719 | OEM729 | | OEM7600 | OEM7700 | OEM7720 |
|---|---|---|---|---|---|---|
| **USB0_D+** | 4[1] | 22 | - | 48 | 48 | 48 |
| **USB1_D-** | - | - | - | 47 | 47 | 47 |
| **USB1_D+** | - | - | - | 49 | 49 | 49 |
| **USB_ VBUS** | - | - | 15 | 52 | 52 | 52 |
| **UID** | - | - | 14 | 51 | 51 | 51 |

**Table 69: USB Device Interface Example Bill of Materials**

| Designator | Manufacturer | Manufacturer Part Number |
|---|---|---|
| D1 | On Semiconductor | ESD7L5.0DT5G |
| D2 | On Semiconductor | ESD5Z6.0T1G |
| FB1 | TDK | MMZ1005B800C |
| L1 | Wurth Electronics | 744230900 |
| J1 | FCI | 10104110-0001LF |

**Figure 7: OEM7 USB Host Device Interface Example**




[1]This pin is multiplexed with the COM3 receive signal and is not available when COM3 is enabled.

**Table 70: USB Host Interface Pin Designation**

| | OEM719 | OEM729 | | OEM7600 | OEM7700 | OEM7720 |
|---|---|---|---|---|---|---|
| **Connector** | P1701 | P1802 | P1803 | P1701 | P2001 | P1901 |
| **USB0_D-** | 3 | 21 | - | 50 | 50 | 50 |
| **USB0_D+** | 4[1] | 22 | - | 48 | 48 | 48 |
| **USB1_D-** | - | - | - | 47 | 47 | 47 |
| **USB1_D+** | - | - | - | 49 | 49 | 49 |
| **USB_ VBUS** | - | - | 15 | 52 | 52 | 52 |
| **UID** | - | - | 14 | 51 | 51 | 51 |

**Table 71: USB Host Interface Example Bill of Materials**

| Designator | Manufacturer | Manufacturer Part Number |
|---|---|---|
| D1 | On Semiconductor | ESD7L5.0DT5G |
| D2 | Littelfuse | V5.5MLA0402NR |
| FB1 | TDK | MMZ1005B800C |
| J1 | Molex | 0482040001 |
| L1 | Wurth Electronics | 744230450 |

### *3.4.7 Ethernet Port*

OEM7 receivers, except the OEM719, provide a 10/100 Ethernet port with auto-negotiation. The Ethernet interface is disabled by default and must be configured. See the Ethernet Configuration for instructions on Ethernet device configuration. The PHY layer is based on the Texas Instruments TKL106 Ethernet PHY.

PHY terminations are provided on the OEM7 receiver card and a 3.3 V output is presented to bias the Ethernet magnetics. The 3.3 V power supplied by the OEM7 receiver card is not to be used for any purposes other than biasing the Ethernet magnetics.

A reference schematic is shown in *Figure 8: Ethernet Reference Schematic* on the next page. The ferrite bead is included as an EMI de-risk contingency and may not be necessary.

[1]This pin is multiplexed with the COM3 receive signal and is not available when COM3 is enabled.

**Figure 8: Ethernet Reference Schematic**

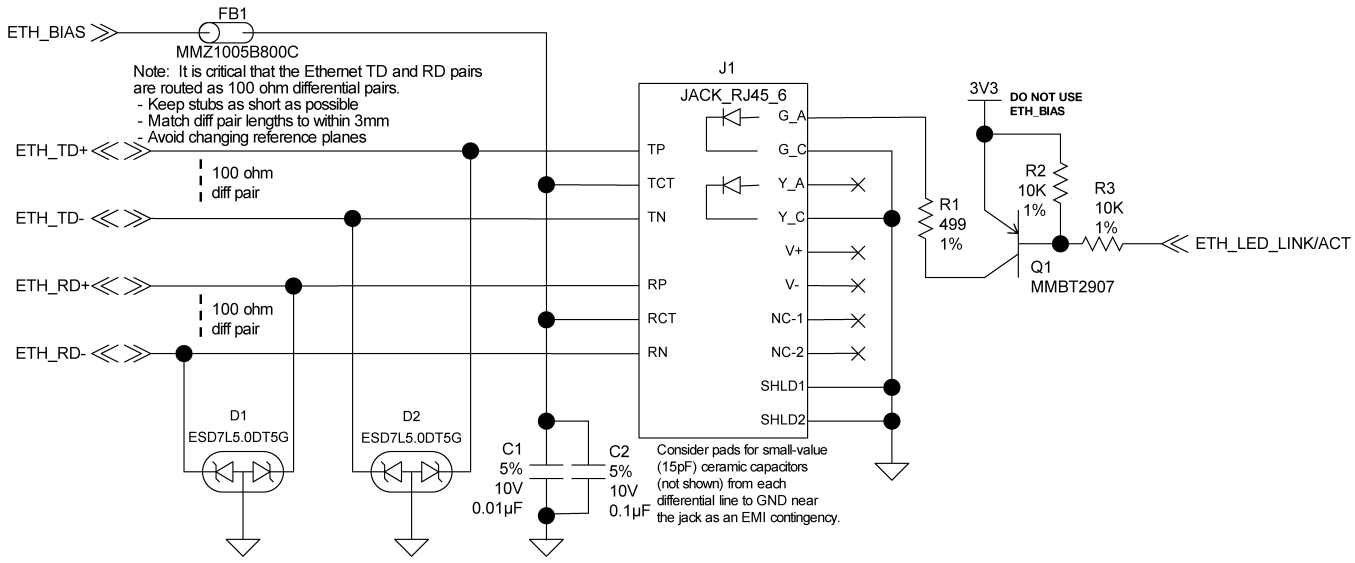



**Table 72: Ethernet Pin Designation**

| | OEM729 | OEM7600 | OEM7700 | OEM7720 |
|---|---|---|---|---|
| **Connector** | P1803 | P1701 | P2001 | P1901 |
| **ETH_ BIAS** | Receive: 3<br>Transmit: 6 | 56 | 56 | 56 |
| **ETH_TD+** | 4 | 58 | 58 | 58 |
| **ETH_TD-** | 5 | 60 | 60 | 60 |
| **ETH_RD+** | 2 | 57 | 57 | 57 |
| **ETH_RD-** | 1 | 59 | 59 | 59 |

Run 100 Ω differential pairs over unbroken reference planes directly to the jack. Ensure the integrated magnetics in the jack meet the specifications in *Table 73: Ethernet Transformer Characteristics* below. Ensure that the jack is no more than 15 cm (6 inches) from the OEM7 connector. Shorter runs are better.

**Table 73: Ethernet Transformer Characteristics**

| Parameter | Value | Test Condition |
|---|---|---|
| Turns ratio | 1 CT : 1 CT | |
| Open-CCT inductance (minimum) | 350 µH | 100 mV, 100 kHz, 8 mA |
| Leakage inductance (maximum) | 0.4 µH | 1 MHz (minimum) |
| Inter-winding capacitance (minimum) | 12 pF | |
| DC resistance (maximum) | 0.9 Ω | |

| Parameter | Value | Test Condition |
|---|---|---|
| Insertion loss (maximum) | 1.0 dB | 0 MHz - 65 MHz |
| HIPOT (minimum) | 1500 Vrms | |

**Table 74: Bill of Materials (critical components)**

| Designator | Manufacturer | Manufacturer Part Number |
|---|---|---|
| FB1 | TDK | MMZ1005B800C |
| D1, D2 | On Semiconductor | ESD7L5.0DT5G |
| J1 | Wurth Electronics | 7499211121A |
| Q1 | On Semiconductor<br>Fairchild Semiconductor | MMBT2907ALT1G<br>MMBT2907A |

## 3.5 Differences Between PwrPak7 and OEM6 Enclosures

The following tables compare the specifications and features of the PwrPak7 with the specifications and features of OEM6 enclosures.

**Table 75: Physical Specifications**

| | PwrPak7-E1<br>PwrPak7D-E1 | PwrPak7<br>PwrPak7D | FlexPak6 | FlexPak6D | ProPak6 |
|---|---|---|---|---|---|
| Dimensions | 147 x 125 x 55 mm | 147 x 125 x 55 mm | 147 x 113 x 45 mm | 147 x 113 x 45 mm | 190 x 185 x 75 mm |
| Weight | 510 g | 500 g | 337 g | 315 g | 1.79 kg |

**Table 76: Power Requirements**

| | PwrPak7<br>(all models) | FlexPak6 | FlexPak6D | ProPak6 |
|---|---|---|---|---|
| Input Voltage | 9 to 36 VDC | 6 to 36 VDC | 6 to 36 VDC | 9 to 36 VDC |
| Power Consumption | 1.8 W | 1.8 W | 1.9 W | 3.5 W |

**Table 77: Signals Tracked**

| | PwrPak7<br>PwrPak7-E1 | PwrPak7D<br>PwrPak7D-E1 | FlexPak6 | FlexPak6D | ProPak6 |
|---|---|---|---|---|---|
| Channels | 555 | 555 | 120 | 120 | 240 |

| | | PwrPak7<br>PwrPak7-E1 | PwrPak7D<br>PwrPak7D-E1 | FlexPak6 | FlexPak6D | ProPak6 |
|---|---|---|---|---|---|---|
| GPS | L1 C/A | Y | Y | Y | Y | Y |
| | L1C | Y | Y | – | – | – |
| | L2C | Y | Y | Y | Y | Y |
| | L2P | Y | Y | Y | Y | Y |
| | L5 | Y | Y | Y | – | Y |
| GLONASS | L1 C/A | Y | Y | Y | Y | Y |
| | L2 C/A | Y | Y | Y | Y | Y |
| | L2P | Y | Y | Y | – | Y |
| | L3 | Y | Y | – | – | – |
| | L5 | Y | Y | – | – | – |
| Galileo | E1 | Y | Y | Y | Y | Y |
| | E5a | Y | Y | Y | – | Y |
| | E5b | Y | Y | Y | Y | Y |
| | E5 AltBOC | Y | Y | Y | | Y |
| | E6 | Y | – | – | – | – |
| BeiDou | B1I | Y | Y | Y | Y | Y |
| | B1C | Y | Y | – | – | – |
| | B2I | Y | Y | Y | Y | Y |
| | B2a | Y | Y | – | – | – |
| | B3I | Y | – | – | – | – |
| QZSS | L1 C/A | Y | Y | Y | Y | Y |
| | L1C | Y | Y | – | – | – |
| | L2C | Y | Y | Y | Y | Y |
| | L5 | Y | Y | Y | – | Y |
| | L6 | Y | – | – | – | – |
| NavIC | L5 | Y | Y | – | – | – |
| SBAS | L1 | Y | Y | Y | Y | Y |
| | L5 | Y | Y | – | – | – |
| L-Band | | 5 channels | 5 channels | 1 channel | – | 1 channel |

**Table 78: Horizontal Position Accuracy (RMS)**

| | PwrPak7 (all models) | FlexPak6 | FlexPak6D | ProPak6 |
|---|---|---|---|---|
| Single Point L1 | 1.5 m | 1.5 m | 1.5 m | 1.5 m |
| Single Point L1/L2 | 1.2 m | 1.2 m | 1.2 m | 1.2 m |
| SBAS | 60 cm | 60 cm | 60 cm | 60 cm |
| DGPS | 40 cm | 40 cm | 40 cm | 40 cm |
| TerraStar-L | 40 cm | 40 cm | – | – |
| TerraStar-C | 4 cm | 4 cm | – | 4 cm |
| TerraStar-C PRO | 2.5 cm | – | – | – |
| RTK | 1 cm + 1ppm | 1 cm + 1ppm | 1 cm + 1ppm | 1 cm + 1ppm |
| RTK ASSIST | Y | Y | – | – |
| RTK ASSIST PRO | Y | – | – | – |

**Table 79: Communications Ports**

| | | PwrPak7 (all models) | FlexPak6 | FlexPak6D | ProPak6 |
|---|---|---|---|---|---|
| Serial Ports | COM1 | RS-232/RS-422 | RS-232/RS-422 | RS-232/RS-422 | RS-232/RS-422 |
| | COM2 | RS-232/RS-422 | RS-232/RS-422 | RS-232/RS-422 | RS-232/RS-422 |
| | COM3 | RS-232 | – | – | RS-232/RS-422 |
| | COM7[1] | – | – | – | RS-232* |
| | COM8[1] | – | – | – | RS-232* |
| | COM9[1] | – | – | – | RS-232* |
| | COM10[1] | – | – | – | RS-232* |
| USB Ports | Device | 1 | 1 | 1 | 1 |
| | Host | 1 | – | – | 1 |
| CAN Bus Ports | | 1 | 1 | – | 2 |
| Ethernet Port (10BASE-T/100BASE-TX) | | 1 | 1 | – | 1 |
| Wi-Fi | Access Point | 1 | – | – | 1 |
| | Client | – | – | – | 1 |

| | PwrPak7 (all models) | FlexPak6 | FlexPak6D | ProPak6 |
|---|---|---|---|---|
| Cellular (CDMA or GSM/GPRS/HSDPA) | – | – | – | 1 |
| Bluetooth | – | – | – | 1 |

1. Only available when using the ProPak6 Expansion cable. The ProPak6 Expansion cable multiplexes four serial COM ports onto a single USB port.

**Table 80: RF Input**

| | | PwrPak7 PwrPak7-E1 | PwrPak7D PwrPak7D-E1 | FlexPak6 | FlexPak6D | ProPak6 Dual Antenna | ProPak6 |
|---|---|---|---|---|---|---|---|
| GNSS Antenna Connector | Primary | TNC | SMA | TNC | SMA | TNC | TNC |
| | Secondary | – | SMA | – | SMA | TNC | – |
| LNA | Primary | +5 VDC ±5%; 200 mA | +5 VDC ±5%; 200 mA | +5 VDC ±5%; 100 mA | +5 VDC ±5%; 100 mA | +5 VDC ±5%; 200 mA | +5 VDC ±5%; 200 mA |
| | Secondary | – | +5 VDC ±5%; 200 mA | – | +5 VDC ±5%; 100 mA | +5 VDC ±5%; 100 mA | – |
| External Oscillator Connector | | – | – | – | – | – | BNC |
| Cellular Antenna Connector | | – | – | – | – | TNC | TNC |

**Table 81: I/O Strobes**

| | PwrPak7 (all models) | FlexPak6 | FlexPak6D | ProPak6 |
|---|---|---|---|---|
| Event In | 3 | 2 | 2 | 4 |
| Event Out | 3 | 1 | 1 | 3 |
| PPS | 1 | 1 | 1 | 1 |
| ERROR | – | 1 | – | * |
| Position Valid (PV) | – | 1 | 1 | * |
| Wheel Sensor | 1 | – | – | – |

* The ProPak6 I/O strobes can be configured to change an Event line to this signal.

**Table 82: Features**

| | | PwrPak7-E1 PwrPak7D-E1 | PwrPak7 PwrPak7D | FlexPak6 | FlexPak6D | ProPak6 |
|---|---|---|---|---|---|---|
| Data Rates | GNSS Measurements | 20 Hz | 100 Hz | 100 Hz | 20 Hz | 100 Hz |
| | GNSS Position | 20 Hz | 100 Hz | 100 Hz | 20 Hz | 100 Hz |
| | INS Position/ Attitude | 200 Hz | * | * | – | * |
| | INS Raw Data | 125 Hz | * | * | – | * |
| SPAN Compatible | | Y | Y | Y | – | Y |
| Integrated IMU | | Y | – | – | – | – |
| Compatible UI | NovAtel Connect | Y | Y | Y | Y | Y |
| | Web UI | Y | Y | – | – | – |
| Onboard Memory | | 16 GB | 16 GB | – | – | 4 GB |
| * INS data rates vary depending on the external IMU connected to the receiver. | | | | | | |

## 3.6 Differences Between SMART7 and SMART6

The following tables compare the specifications and features of the SMART7 with the specifications and features of the SMART6 and SMART6-L.

**Table 83: Physical Specifications**

| | SMART7 (all models) | SMART6 | SMART6-L |
|---|---|---|---|
| Dimensions | 220 x 192 x 66 mm | 155 (diameter) x 81 mm | 155 (diameter) x 81 mm |
| Weight | <1.1 kg | <520 g | <570 g |

**Table 84: Power Requirements**

| | SMART7 (all models) | SMART6 | SMART6-L |
|---|---|---|---|
| Input Voltage | +7 to +30 VDC | +8 to +36 VDC | +8 to +36 VDC |
| Power Consumption (typical) | 4 W | 3.5 W | 2.9 W |

**Table 85: Signals Tracked**

| | SMART7 (all models) | SMART6 | SMART6-L |
|---|---|---|---|
| Channels | 555 | 120 | 120 |

| | | SMART7 (all models) | SMART6 | SMART6-L |
|---|---|---|---|---|
| GPS | L1 C/A | Y | Y | Y |
| | L1C | Y | – | – |
| | L2C | Y | Y | Y |
| | L2P | Y | Y | Y |
| | L5 | Y | – | – |
| GLONASS | L1 C/A | Y | Y | Y |
| | L2 C/A | Y | Y | Y |
| | L2P | Y | – | – |
| | L3 | Y | – | – |
| Galileo | E1 | Y | Y | Y |
| | E5a | Y | – | – |
| | E5b | Y | – | – |
| | E5 AltBOC | Y | – | – |
| BeiDou | B1I | Y | Y | Y |
| | B1C | Y | – | – |
| | B2I | Y | – | – |
| | B2a | Y | – | – |
| QZSS | L1 C/A | Y | – | – |
| | L1C | Y | – | – |
| | L2C | Y | – | – |
| | L5 | Y | – | – |
| SBAS | L1 | Y | Y | Y |
| | L5 | Y | – | – |
| L-Band | | 5 channels | – | Y |

**Table 86: Horizontal Position Accuracy**

| | SMART7 (all models) | SMART6 | SMART6-L |
|---|---|---|---|
| Single Point L1 | 1.5 m | 1.5 m | 1.5 m |

| | SMART7 (all models) | SMART6 | SMART6-L |
|---|---|---|---|
| Single Point L1/L2 | 1.2 m | 1.2 m | 1.2 m |
| SBAS | 60 cm | 60 cm | 60 cm |
| DGPS | 40 cm | 40 cm | 40 cm |
| TerraStar-L | 40 cm | – | 40 cm |
| TerraStar-C | – | – | 4 cm |
| TerraStar-C PRO | 2.5 cm | – | – |
| RTK | 1 cm + 1ppm | 1 cm + 1ppm | 1 cm + 1ppm |
| RTK ASSIST | – | – | Y |
| RTK ASSIST PRO | Y | – | – |

**Table 87: Communications Ports**

| | | SMART7 SMART7-S | SMART7-I | SMART7-W | SMART6 | SMART6-L |
|---|---|---|---|---|---|---|
| Serial Ports | COM1 | RS-232 | RS-232 | RS-232 | RS-232 | RS-232 |
| | COM2 | RS-232 | RS-232 | RS-232 | RS-232 | RS-232 |
| | COM3 | RS-232 | RS-232 | RS-232 | RS-232 | RS-232 |
| CAN Bus Ports | | 1 | 1 | 1 | 1 | 1 |
| Ethernet Port 10BASE-T/100BASE-TX | | – | 1 | – | – | – |
| Wi-Fi | Access Point | – | Y | Y | – | – |
| | Client | – | Y | Y | – | – |
| Bluetooth | | – | – | – | Y | – |

**Table 88: I/O Strobes**

| | SMART7 (all models) | SMART6 | SMART6-L |
|---|---|---|---|
| Event In | – | 1 | 1 |
| PPS | 1 | 1 | 1 |
| Emulated Radar | 1 | 1 | 1 |

**Table 89: Features**

| | | SMART7 | SMART7-I SMART7-W | SMART7-S | SMART6 | SMART6-L |
|---|---|---|---|---|---|---|
| Data Rates | GNSS Measurements | 20 Hz | 20 Hz | 20 Hz | 20 Hz | 50 Hz |
| | GNSS Position | 20 Hz | 20 Hz | 20 Hz | 20 Hz | 50 Hz |
| | INS Position/ Attitude | * | * | 200 Hz | – | * |
| | INS Raw Data | * | * | 125 Hz | – | * |
| SPAN Compatible | | Y | Y | Y | – | Y |
| Integrated IMU | | – | – | Y | – | – |
| Compatible UI | NovAtel Connect | Y | Y | Y | Y | Y |
| | Web UI | – | Y | – | – | – |
| * INS data rates vary depending on the external IMU connected to the receiver. | | | | | | |

## 3.7 Differences between RELAY7 and RELAY

The following tables compare the specifications and features of the RELAY7 with the specifications and features of the RELAY (for the SMART6-L).

**Table 90: Physical Specifications**

| | RELAY7 | RELAY |
|---|---|---|
| Dimensions | 186.5 x 199.5 x 76.4 mm | 290 x 275 x 80 mm |
| Weight | <850 g | 2.0 kg |

**Table 91: Power Requirements**

| | RELAY7 | RELAY |
|---|---|---|
| Input Voltage | +7 to +30 VDC | +9 to +36 VDC |
| Power Consumption (typical) | 2 W | 10.5 W – UHF 400 MHz radio<br>8.5 W – UHF 900 MHz radio<br>5.5 W – HSPA radio |

**Table 92: Supported Radios**

| | RELAY7 | RELAY |
|---|---|---|
| 400 MHz | Y | Y |
| 900 MHz | Y | Y |

| | RELAY7 | RELAY |
|---|---|---|
| HSPA | – | Y |
| Wi-Fi | – | Y |
| The RELAY7 will have either a 400 MHz or 900 MHz radio.<br>The RELAY will have Wi-Fi and either a 400 MHz, 900 MHz or HSPA radio. | | |

# Chapter 4 Receiver Technical Specifications

The following sections detail the technical specifications of the OEM7 family receivers.

## 4.1 OEM719 Technical Specifications

**Table 93: OEM719 Physical Description**

| | |
|---|---|
| Size | 46 mm x 71 mm x 11 mm |
| Weight | 31 grams |
| NovAtel Part Number | Generic assembly OEM719 01019520 |

See the following sections for more information about the OEM719:

### 4.1.1 OEM719 Performance Specifications

All specifications subject to GNSS system characteristics.

**Table 94: OEM719 Receiver Performance**

| | | |
|---|---|---|
| Position Accuracy[1] | Single point L1 | 1.5 m RMS |
| | Single point L1/L2 | 1.2 m RMS |
| | SBAS[2] | 60 cm RMS |
| | DGPS | 40 cm RMS |
| | TerraStar-L[3, 4] | 40 cm RMS |
| | TerraStar-C PRO[3, 4] | 2.5 cm RMS |
| | TerraStar-X[3, 4] | 2.0 cm RMS |
| | RTK | 1 cm + 1 ppm RMS |
| Signals Tracked | GPS | L1 C/A, L1C, L2C, L2P, L5 |
| | GLONASS[5] | L1 C/A, L2 C/A, L2P, L3, L5 |
| | BeiDou | B1I, B1C, B2I, B2a, B3I |
| | Galileo | E1, E5 AltBOC, E5a, E5b, E6 |
| | NavIC (IRNSS) | L5 |
| | QZSS | L1 C/A, L1C, L2C, L5, L6 |
| | SBAS | L1, L5 |
| | L-Band[6] | Up to 5 channels |
| Time to First Fix | Hot: <20 s (Almanac and recent ephemeris saved and approximate position and time entered)<br>Cold: <39 s (No almanac or ephemeris and no approximate position or time) | |

[1]Typical values. All position and velocity RMS values are based on Horizontal position accuracy. Performance specifications are subject to GNSS system characteristics, Signal-in-Space (SIS) operational degradation, ionospheric and tropospheric conditions, satellite geometry, baseline length, multipath effects and the presence of intentional or unintentional interference sources.

[2]GPS-only.

[3]Requires a TerraStar subscription which is available direct from NovAtel novatel.com/products/correction-services/terrastar-correction-services.

[4]Performance dependent on local observing conditions.

[5]Although hardware capable, GLONASS L5 is currently not available.

[6]Currently the receiver can track up to 3 L-Band channels.

| | | | | |
|---|---|---|---|---|
| Signal Reacquisition | <0.5 s L1 (typical)<br><1.0 s L2 and L5 (typical) | | | |
| Data Rates | Measurements | up to 100 Hz | | |
| | Position | up to 100 Hz | | |
| Time Accuracy[1] | 20 ns RMS | | | |
| Velocity Accuracy | <0.03 m/s RMS | | | |
| Measurement Precision | | | Code | Carrier |
| | GPS | L1 C/A | 4 cm | 0.5 mm |
| | | L2 P(Y) | 8 cm | 1.0 mm |
| | | L2C | 8 cm | 0.5 mm |
| | | L5 | 3 cm | 0.5 mm |
| | GLONASS | L1 C/A | 8 cm | 1.0 mm |
| | | L2 P | 8 cm | 1.0 mm |
| | | L2 C/A | 8 cm | 1.0 mm |
| | Galileo | E1 | 3 cm | 0.5 mm |
| | | E5a | 3 cm | 0.75 mm |
| | | E5b | 3 cm | 0.75 mm |
| | | E5 AltBOC | 3 cm | 0.75 mm |
| | | E6 | 3 cm | 0.75 mm |
| | BeiDou | B1I | 4 cm | 0.5 mm |
| | | B1C | 3 cm | 0.5 mm |
| | | B2I | 4 cm | 0.5 mm |
| | | B2a | 3 cm | 0.5 mm |
| | | B3I | 4 cm | 0.5 mm |
| Velocity Limit[2] | 515 m/s | | | |

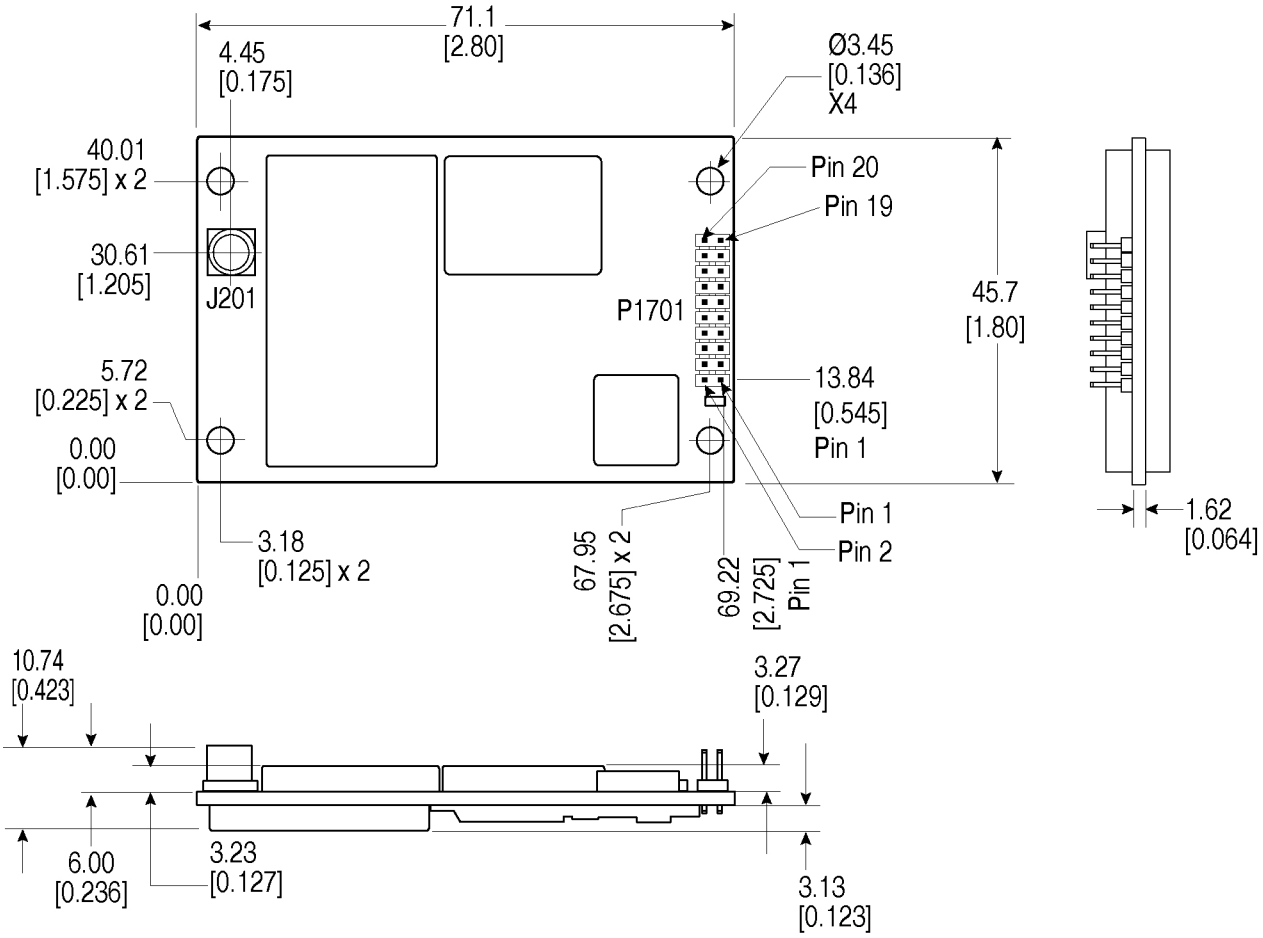
[1]Time accuracy does not include biases due to RF or antenna delay.

[2]Export licensing restricts operation to a maximum of 515 meters per second, message output impacted above 500 m/s.

### 4.1.2 OEM719 Mechanical Specifications

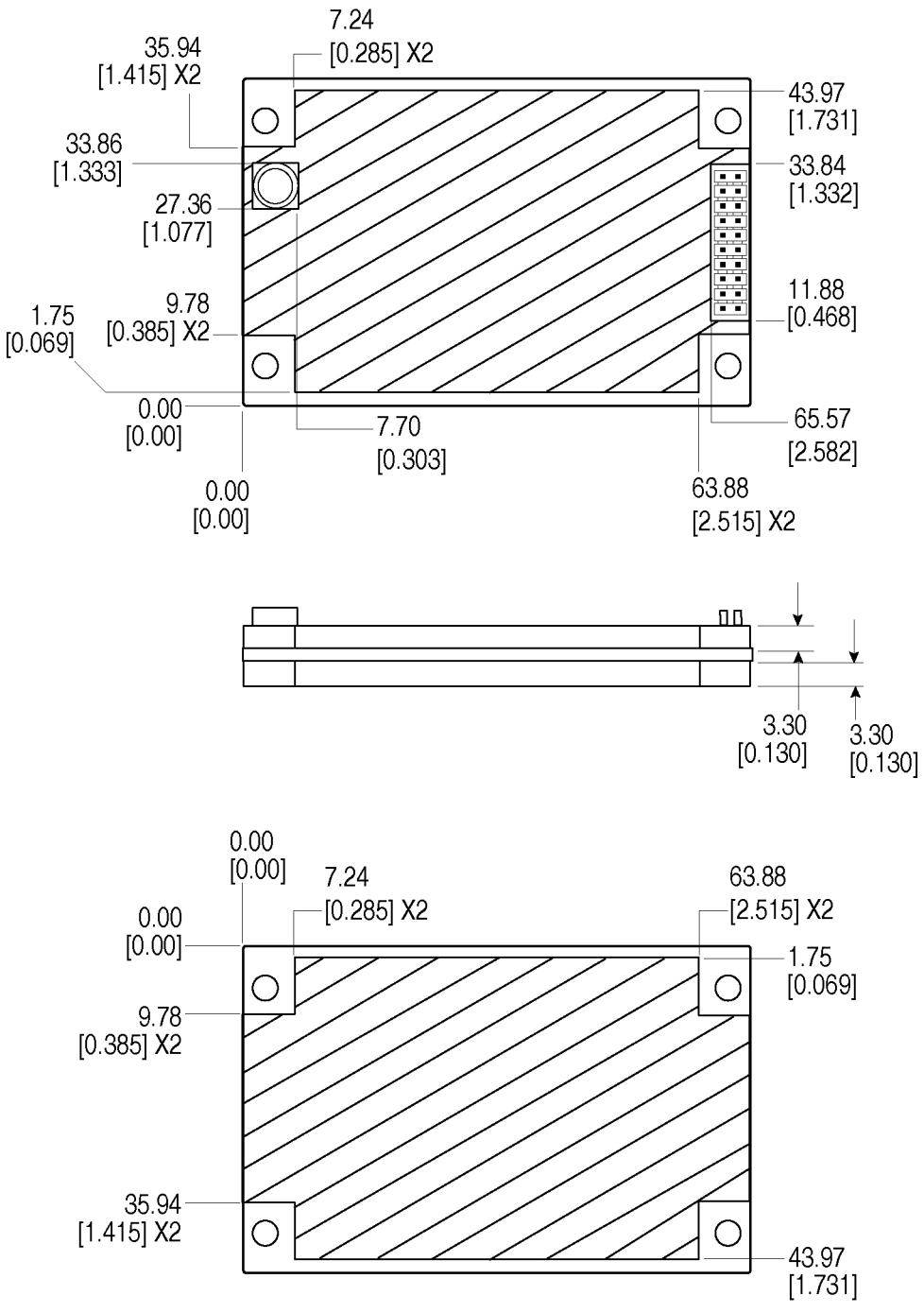
- *Figure 9: OEM719 Dimensions* below
- *Figure 10: OEM719 Keep-outs* on the next page
- *Figure 11: OEM719A Dimensions* on page 180
- *Figure 12: OEM719A Keep-outs* on page 181
- *Figure 13: OEM719B Dimensions* on page 182
- *Figure 14: OEM719B Keep-outs* on page 183
- *Figure 15: OEM719 Mounting Surface* on page 184

**Figure 9: OEM719 Dimensions**



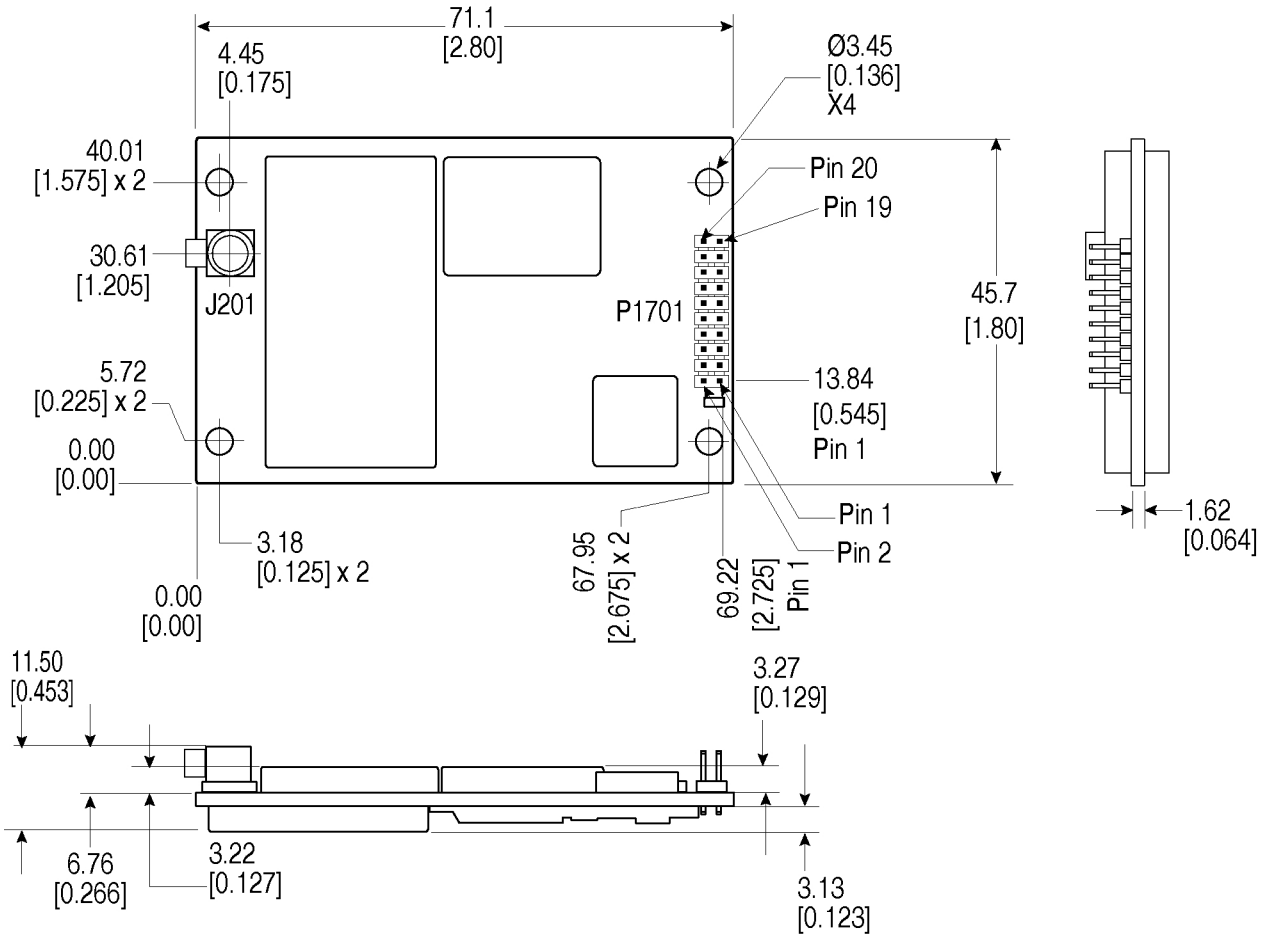

Notes: 719
Dimensions are in millimetres [inches].
Connectors:
(a) J201 MCX jack receptacle, straight (Johnson P/N 133-3711-202 or Samtec P/N MCX-J-P-HST-SMI or equivalent)
(b) P1701 2x10 header, 2 mm pitch (Samtec P/N TMM-110-03-G-D)

**Figure 10: OEM719 Keep-outs**



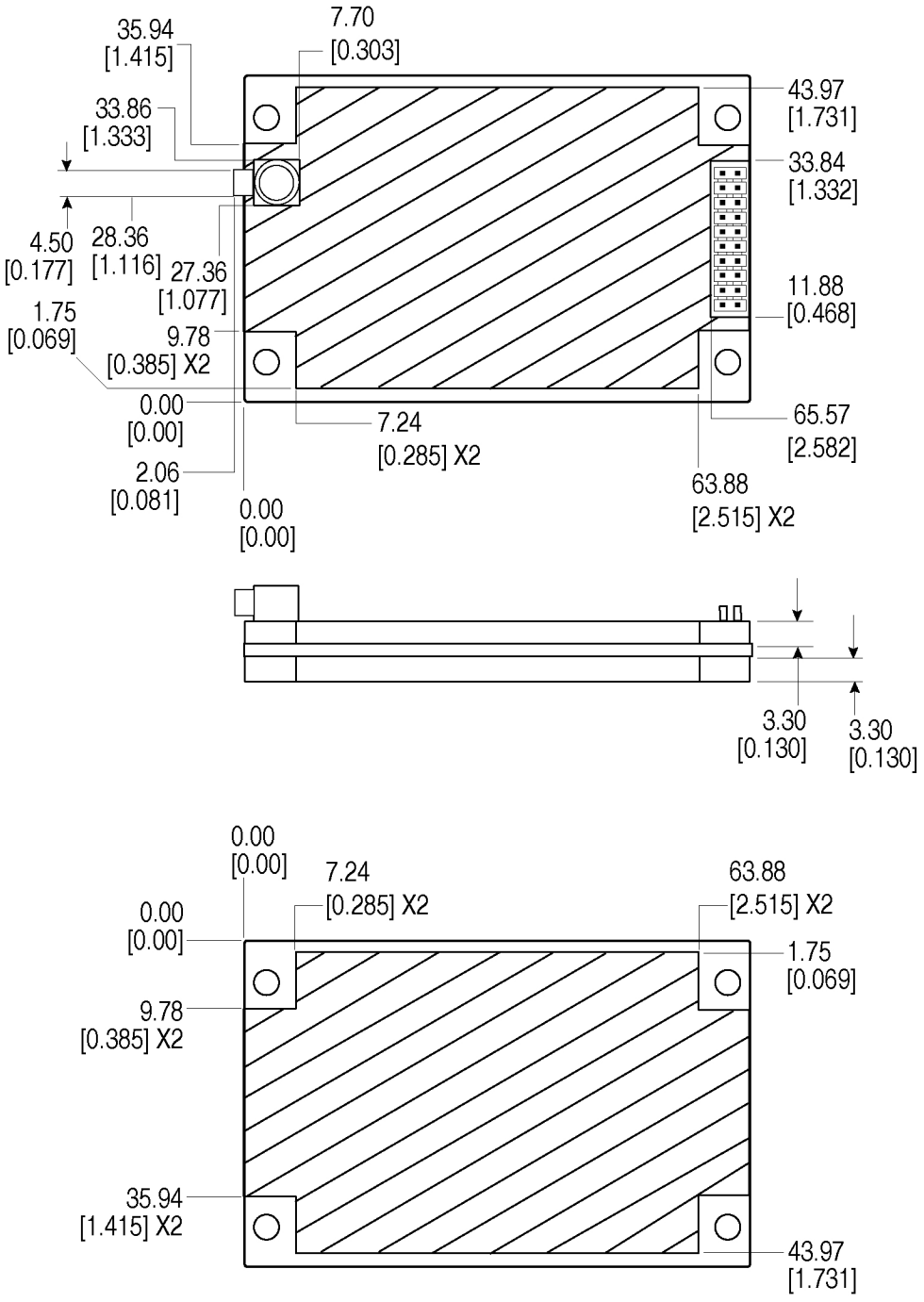

Dimensions are in millimetres [inches].

**Figure 11: OEM719A Dimensions**



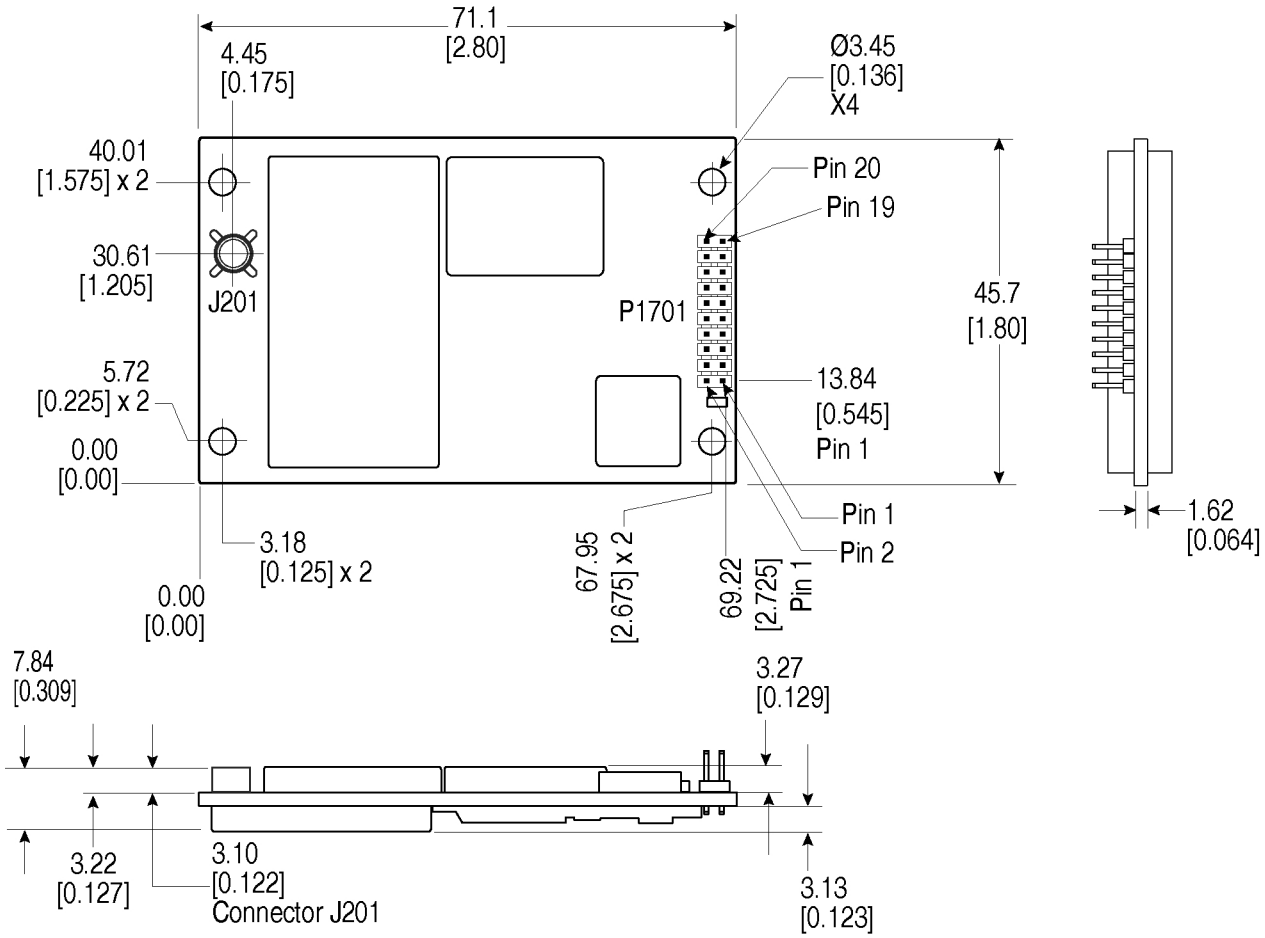

Notes: 719A
Dimensions are in millimetres [inches].
Connectors:
(a) J201 MCX jack receptacle, right angle (Samtec P/N MCX-J-P-H-RA-SMI or equivalent)
(b) P1701 2x10 header, 2 mm pitch (Samtec P/N TMM-110-03-G-D)

**Figure 12: OEM719A Keep-outs**



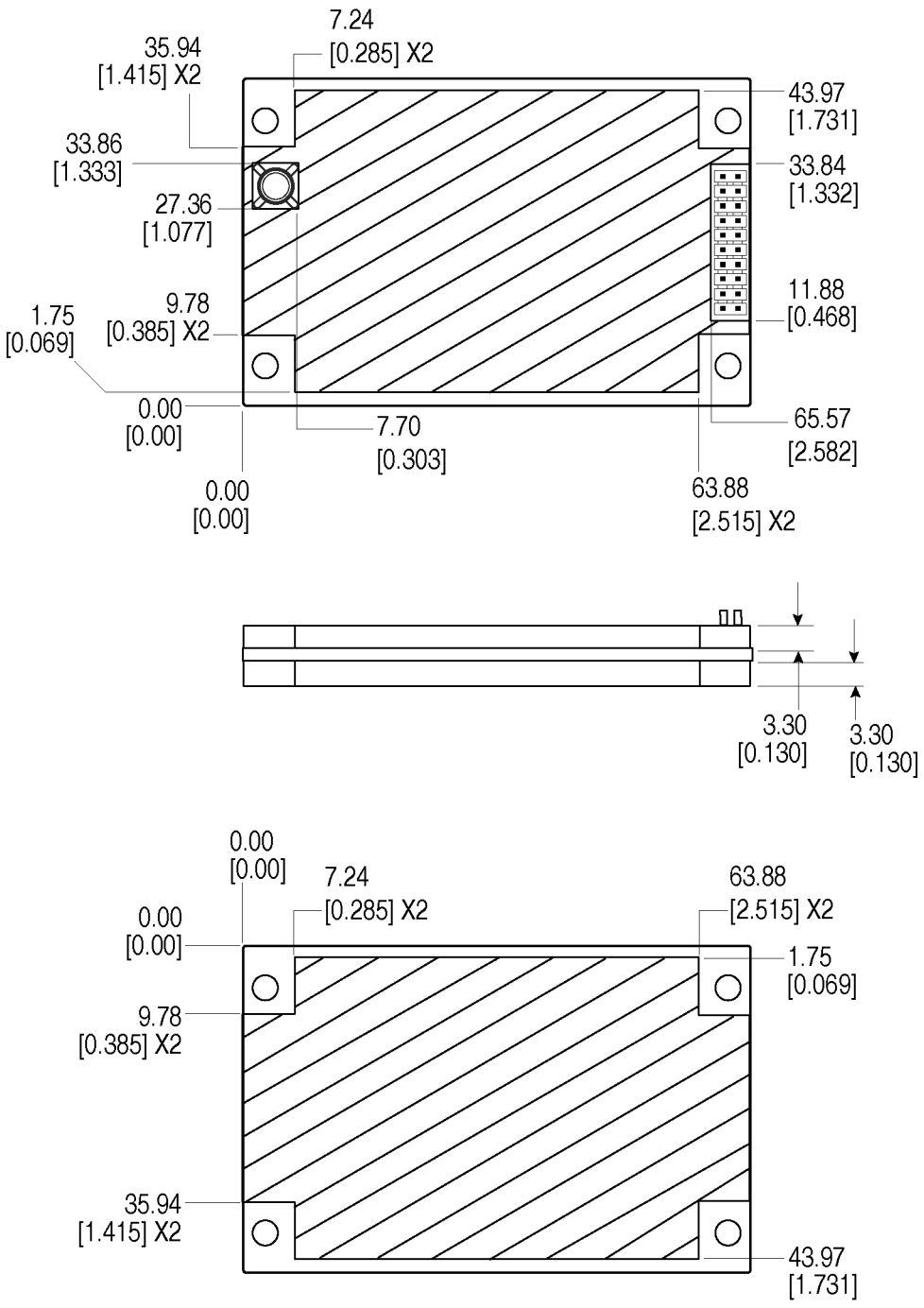

Dimensions are in millimetres [inches].

**Figure 13: OEM719B Dimensions**



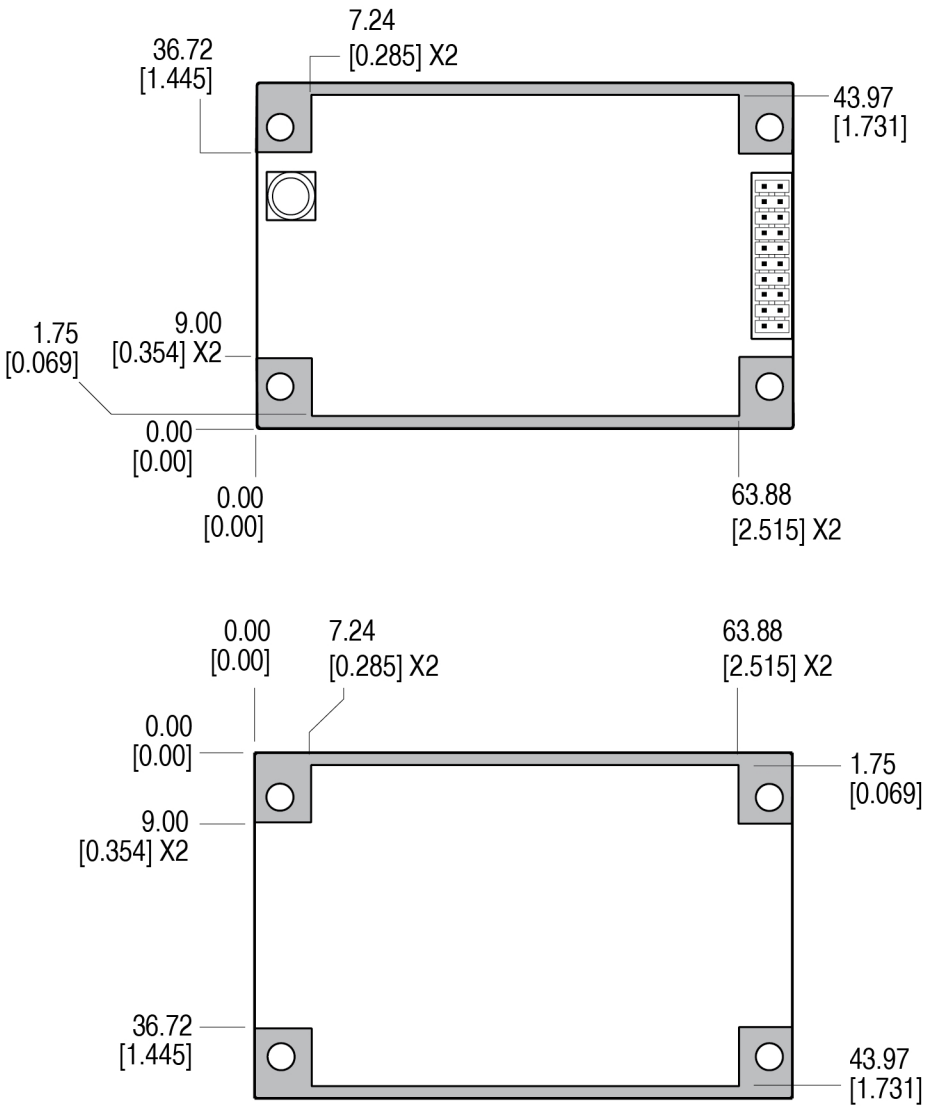

Notes: 719B
Dimensions are in millimetres [inches].
Connectors:
(a) J201: MMBX jack receptible, straight (Huber+Shuhner 82-MMBX-S50-0-1/111_NM)
(b) P1701: 2x10 header, 2 mm pitch (Samtec P/N TMM-110-03-G-D)

**Figure 14: OEM719B Keep-outs**




Dimensions are in millimetres [inches].

**Figure 15: OEM719 Mounting Surface**




Dimensions are in millimetres [inches].

The mounting surfaces are shown in gray. These mounting surfaces apply to all variants of the OEM719.

These mounting surfaces are designed to work with mounting rails. For information about mounting the OEM719, refer to Mounting the Printed Circuit Board.

Assembly tolerances must be considered when using mounting rail features.

### 4.1.3 OEM719 Electrical and Environmental Specifications

**Table 95: OEM719 Environmental Specifications**

| | |
|---|---|
| Operating Temperature | -40°C to +85°C |
| Storage Temperature | -55°C to +95°C |
| Humidity | 95% non-condensing |
| Random Vibration | MIL-STD-810G, Method 514.6, Category 24 (20 g RMS)[1] |
| Sinusoidal Vibration | IEC 60068-2-6 |
| Bump | ISO 9022-31-06 (25 g) |
| Shock<br>Operating<br>Non-operating | <br>MIL-STD-810G, Method 516.6 (40 g)<br>MIL-STD-810G, Method 516.6 (75 g) |
| Acceleration, Operating | MIL-STD-810G (CH1), Method 513.7 (16 g) |

**Table 96: OEM719 Power Requirements**

| | |
|---|---|
| Voltage | +3.3 VDC ±5% |
| Allowable Input Voltage Ripple | 100 mV p-p maximum |
| Power Consumption | 0.9 W typical, GPS L1 only<br>1.3 W typical, GPS L1/L2, GLONASS L1/L2<br>1.8 W typical, all constellations/all frequencies, plus L-Band<br>These are typical values using serial ports without interference mitigation. Values can change with the number of satellites in view, firmware version, data logging rates and features in use. Use them as a guide for what you might expect but not as absolute values. |
| Inrush Current | 1.71 A for less than 1.5 ms (typical) |

**Table 97: OEM719 RF Input/LNA Power Output**

| | | |
|---|---|---|
| Antenna Connector | OEM719 | MCX female, 50 Ω nominal impedance |
| | OEM719A | Right angle, MCX female, 50 Ω nominal impedance |
| | OEM719B | MMBX female, 50 Ω nominal impedance |

[1]Requires mechanical mounting rails to meet 20 g; meets 7.7 g without rails.

| | | | |
|---|---|---|---|
| Cascaded antenna LNA gain (before receiver) | Firmware 7.04 and later | HDR disabled | +15 dB to +55 dB, 26 dB typical |
| | | HDR enabled | +20 dB to +55 dB, 30 dB typical |
| | Firmware before 7.04 | HDR disabled | +15 dB to +40 dB, 26 dB typical |
| | | HDR enabled | +20 dB to +40 dB, 30 dB typical |
| RF Input Frequencies | GPS L1:<br>GPS L2:<br>GPS L5: | 1575.42 MHz<br>1227.60 MHz<br>1176.45 MHz | |
| | GLONASS L1:<br>GLONASS L2:<br>GLONASS L3: | 1593-1610 MHz<br>1237-1254 MHz<br>1202.025 MHz | |
| | Galileo E1:<br>Galileo E5a:<br>Galileo E5b:<br>Galileo E5:<br>Galileo E6: | 1575.42 MHz<br>1176.45 MHz<br>1207.14 MHz<br>1191.795 MHz<br>1278.75 MHz | |
| | BeiDou B1I:<br>BeiDou B1C:<br>BeiDou B2I:<br>BeiDou B2a:<br>BeiDou B3I: | 1561.098 MHz<br>1575.42 MHz<br>1207.14 MHz<br>1176.45 MHz<br>1268.52 MHz | |
| | L-Band: | 1525 to 1560 MHz | |
| LNA Power | +5.0 VDC ±5%, 0 mA to 200 mA (supplied by card through center conductor of RF connector).<br>LNA Power is generated from the 3.3 V supply input for the OEM719.<br>**Note**: Pin 1 of P1701 is not electrically connected on the OEM719. This was the external LNA power input (LNA-PWR) on the OEM615. | | |

### 4.1.4 OEM719 Data Communication Specifications

**Table 98: OEM719 Data Communication Interfaces**

| | |
|---|---|
| **COM1** | |
| Electrical format | LVCMOS |
| Data rates[1] | 2400, 4800, 9600 (default), 19200, 38400, 57600, 115200, 230400 or 460800 bit/s. |
| Signals supported | COM1_Tx, COM1_Rx |
| Electrostatic discharge protection | No |
| **COM2** | |
| Electrical format | LVCMOS |
| Data rates[1] | 2400, 4800, 9600 (default), 19200, 38400, 57600, 115200, 230400 or 460800 bit/s. |
| Signals supported | COM2_Tx, COM2_Rx |
| Electrostatic discharge protection | No |
| **COM3** | |
| Electrical format | LVCMOS [2] |
| Data rates[1] | 2400, 4800, 9600 (default), 19200, 38400, 57600, 115200, 230400 or 460800 bit/s. |
| Signals supported | COM3_Tx, COM3_Rx |
| Electrostatic discharge protection | No |
| **CAN Bus** | |
| Electrical format | LVCMOS (requires external CAN transceiver)[3] |

[1]Data rates higher than 115200 bit/s are not supported by standard PC hardware. Special computer hardware may be required for higher rates, including 230400 bit/s and 460800 bit/s.

[2]COM3 is disabled by default.

[3]CAN1 Rx and Tx signals are internally multiplexed with VARF and Event2, respectively. Both VARF and Event2 are enabled by default. To enable CAN functionality, the following commands must be issued before configuring the CAN1 port:

**EVENTOUTCONTROL MARK1 DISABLE**
**EVENTINCONTROL MARK2 DISABLE**

These commands, together with the commands to configure the CAN1 port, can be saved using SAVECONFIG.

| | |
|---|---|
| Data rates | 250, 500 or 1000 kb/s<br>CAN Bus throughput is determined by slowest device on the bus |
| Signals supported | CAN1Tx, CAN1Rx, CAN2Tx, CAN2Rx |
| **USB** | |
| Electrical format | Conforms to USB 2.0 |
| Data rates | Full-speed (12 Mb/s) |
| Signals supported | USB D (+), USB D (-) |

### *4.1.5 OEM719 Strobe Specifications*

**Table 99: OEM719 Strobe Description**

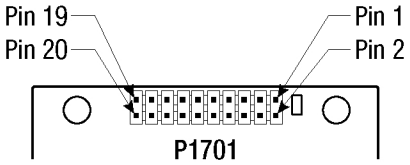
| Strobes | Input/Output | Factory Default | Comment |
|---|---|---|---|
| Event1 (Mark 1) | Input<br>Leading edge triggered | Active low | An input mark for which a pulse greater than 150 ns triggers certain logs to be generated. (Refer to the MARKPOS and MARK1TIME logs and ONMARK trigger.) Polarity is configurable using the **EVENTINCONTROL** command. The mark inputs have 10K pull-up resistors to 3.3 V<br>**Note**: Event1 is the default behavior for pin 9 of connector P1701. The secondary behavior for pin 9 is COM3_Tx. If COM3 is enabled, the Event1 input is disabled. |
| Event2 (Mark 2) | Input<br>Leading edge triggered | Active low | An input mark for which a pulse greater than 150 ns triggers certain logs to be generated (see the MARK2POS and MARK2TIME logs). Polarity is configurable using the **EVENTINCONTROL** command. The mark inputs have 10K pull-up resistors to 3.3 V<br>**Note**: Event2 is the default behavior for pin 7 of connector P1701. The secondary behavior for pin 7 is CAN1Tx. If CAN1 is enabled, the Event2 input is disabled. |
| PPS | Output | Active low | A time synchronization output. This is a pulse where the leading edge is synchronized to receiver-calculated GNSS Time. The polarity, period and pulse width can be configured using the **PPSCONTROL** command |
| PV (Position Valid) | Output | Active high | Indicates a valid GNSS position solution is available. |
| nRESET_IN | Input | Active low | Reset signal input from external system; active low.<br>This pin must be held low for >20 µs while stable power is already applied, to reset the OEM719 card.<br>It is recommended to hold the nRESET_IN pin low for a >150 ms when initially applying power to the card, giving the power supply time to stabilize before the card starts to boot. |
| VARF (Variable Frequency) | Output | Active low | A programmable variable frequency output ranging from 0 - 50 MHz (refer to the **EVENTOUTCONTROL** command)<br>**Note**: VARF is the default behavior for pin 6 of connector P1701. The secondary behavior for pin 6 is CAN1Rx. If CAN1 is enabled, the VARF output is disabled. |

**Table 100: OEM719 Strobe Electrical Specification**

| Strobe | Sym | Min (V) | Typ (V) | Max (V) | Current (mA) |
|---|---|---|---|---|---|
| Event1 (Mark1)<br>Event2 (Mark2) | $V_{IL}$ | | | 0.7 | - |
| | $V_{IH}$ | 2.1 | | | |
| PPS | $V_{OL}$ | | | 0.4 | 16 |
| | $V_{OH}$ | 2.4 | | | |
| PV | $V_{OL}$ | | | 0.4 | 4 |
| | $V_{OH}$ | 2.9 | | | |
| nRESET_IN | $V_{IL}$ | | | 0.8 | - |
| | $V_{IH}$ | 2.3 | | | |
| VARF | $V_{OL}$ | | | 0.4 | 4 |
| | $V_{OH}$ | 2.9 | | | |

### *4.1.6 OEM719 Interface Connector*

#### P1701 Main Connector 20-Pin Header




| Pin | Signal Name | Signal Type | Signal Direction | $V_{IL}$ Max (V) | $V_{IH}$ Min (V) | $V_{OL}$ Max (V) | $V_{OH}$ Min (V) | Drive (mA) | Description |
|---|---|---|---|---|---|---|---|---|---|
| 1 | NC | - | - | - | - | - | - | - | No internal connection on OEM719 receiver.<br>(This pin was the supply feed for the external GNSS antenna on the OEM6 family. The OEM719 generates the antenna power from the 3.3V supply input.) |
| 2 | 3V3 | Power | - | - | - | - | - | - | 3.3 V ±5% supply voltage.<br>This is now monitored by the receiver itself. (Out-of-tolerance supply inputs may generate warning or error messages.) |
| 3 | USB_D- | Analog | Input/ Output | - | - | - | - | - | This is one half of a USB differential pair (pins 3 and 4), match lengths and route as 90 Ω differential pair if USB is used. |
| 4 | USB_D+/ RXD3 | Analog /3.3V CMOS | USB_D+: Input/ Output | - | - | - | - | - | This pin is internally multiplexed. USB_D+ is the default.<br>USB_D+: This is one half of a USB differential pair (pins 3 and 4), match lengths and route as 90 Ω differential pair if USB is used.<br>RXD3: COM3 Receive Data (UART). Internal weak (40 kΩ to 100 kΩ) pullup. |
| | | | RXD3 Input | 0.7 | 2.1 | - | - | - | |

| Pin | Signal Name | Signal Type | Signal Direction | $V_{IL}$ Max (V) | $V_{IH}$ Min (V) | $V_{OL}$ Max (V) | $V_{OH}$ Min (V) | Drive (mA) | Description |
|---|---|---|---|---|---|---|---|---|---|
| 5 | nRESET_IN | 3.3V CMOS | Input | 0.8 | 2.3 | - | - | - | Active Low.<br>Resets the OEM719 receiver card. This pin must be held low for >20 µs while stable power is already applied, to reset the OEM719 card.<br>It is recommended to hold the nRESET_IN pin low for a >150 ms when initially applying power to the card, giving the power supply time to stabilize before the card starts to boot.<br>Internal 10 kΩ pullup. |
| 6 | VARF/ CAN1RX | 3.3V CMOS | VARF: Output | - | - | 0.4 | 2.9 | 4 | This pin is internally multiplexed. VARF is the default.<br>VARF: Variable Frequency Output. Rising or falling edge active.<br>CAN1RX: a CMOS-level signal, requiring an external CAN transceiver.<br>Internal 10 kΩ pullup. |
| | | | CAN1RX: Input | 0.7 | 2.1 | - | - | - | |
| 7 | EVENT2/ CAN1TX | 3.3V CMOS | EVENT2: Input | 0.7 | 2.1 | - | - | - | This pin is internally multiplexed. EVENT2 is the default.<br>EVENT2: Rising edge triggered.<br>CAN1TX is a CMOS-level signal, requiring an external CAN transceiver.<br>Internal 10 kΩ pullup. |
| | | | CAN1TX: Output | - | - | 0.4 | 2.9 | 4 | |
| 8 | CAN2RX | 3.3V CMOS | Input | 0.7 | 2.1 | - | - | - | CAN2RX is a CMOS-level signal, requiring an external CAN transceiver. |
| 9 | EVENT1/ TXD3 | 3.3V CMOS | EVENT1: Input | 0.7 | 2.1 | - | - | - | This pin is internally multiplexed. EVENT1 is the default.<br>EVENT1: Rising edge triggered.<br>TXD3: COM3 Transmit Data (UART)<br>Internal 10 kΩ pullup. |
| | | | TXD3: Output | - | - | 0.4 | 2.9 | 4 | |
| 10 | GND | Power | - | - | - | - | - | - | Supply Return (Ground) |

| Pin | Signal Name | Signal Type | Signal Direction | $V_{IL}$ Max (V) | $V_{IH}$ Min (V) | $V_{OL}$ Max (V) | $V_{OH}$ Min (V) | Drive (mA) | Description |
|---|---|---|---|---|---|---|---|---|---|
| 11 | TXD1 | 3.3V CMOS | Output | - | - | 0.8 | 2.0 | 16 | COM1 Transmit Data (UART)<br>For SPAN applications, this pin can be configured to output a timing signal periodically (generally 1PPS). |
| 12 | RXD1 | 3.3V CMOS | Input | 0.7 | 2.1 | - | - | - | COM1 Receive Data (UART)<br>Internal weak (40 kΩ to 100 kΩ) pullup. |
| 13 | GND | Power | - | - | - | - | - | - | Supply Return (Ground) |
| 14 | TXD2 | 3.3V CMOS | Output | - | - | 0.4 | 2.9 | 4 | COM2 Transmit Data (UART)<br>For SPAN applications, this pin can be configured to output a timing signal periodically (generally 1PPS). |
| 15 | RXD2 | 3.3V CMOS | Input | 0.7 | 2.1 | - | - | - | COM2 Receive Data (UART)<br>Internal weak (40 kΩ to 100 kΩ) pullup. |
| 16 | GND | Power | - | - | - | - | - | - | Supply Return (Ground) |
| 17 | PV | 3.3V CMOS | Output | - | - | 0.4 | 2.9 | 4 | Active High.<br>Position Valid Indicator. Indicates that the receiver has computed a position. Active high output. |
| 18 | GND | Power | - | - | - | - | - | - | Supply Return (Ground) |
| 19 | PPS | 3.3V CMOS | Output | - | - | 0.4 | 2.4 | 16 | Rising or Falling Edge active. (Software-configurable active edge.)<br>This pin can be configured to provide a GNSS-synchronized time output (commonly Pulse Per Second but can operate at other rates as well). |
| 20 | CAN2TX | 3.3V CMOS | Output | - | - | 0.4 | 2.9 | 4 | CAN2TX is a CMOS-level signal, requiring an external CAN transceiver. |

## 4.2 OEM729 Technical Specifications

**Table 101: OEM729 Physical Description**

| | |
|---|---|
| Size | 60 mm x 100 mm x 9 mm |
| Weight | 48 grams |
| NovAtel Part Number | Generic assembly OEM729 01019523 |

See the following sections for more information about the OEM729:

### 4.2.1 OEM729 Performance Specifications

All specifications subject to GNSS system characteristics.

**Table 102: OEM729 Receiver Performance**

| | | |
|---|---|---|
| Position Accuracy[1] | Single point L1 | 1.5 m RMS |
| | Single point L1/L2 | 1.2 m RMS |
| | SBAS[2] | 60 cm RMS |
| | DGPS | 40 cm RMS |
| | TerraStar-L[3, 4] | 40 cm RMS |
| | TerraStar-C PRO[3, 4] | 2.5 cm RMS |
| | TerraStar-X[3, 4] | 2.0 cm RMS |
| | RTK | 1 cm + 1 ppm RMS |
| Signals Tracked | GPS | L1 C/A, L1C, L2C, L2P, L5 |
| | GLONASS[5] | L1 C/A, L2 C/A, L2P, L3, L5 |
| | BeiDou | B1I, B1C, B2I, B2a, B3I |
| | Galileo | E1, E5 AltBOC, E5a, E5b, E6 |
| | NavIC (IRNSS) | L5 |
| | QZSS | L1 C/A, L1C, L2C, L5, L6 |
| | SBAS | L1, L5 |
| | L-Band[6] | Up to 5 channels |
| Time to First Fix | Hot: <20 s (Almanac and recent ephemeris saved and approximate position and time entered)<br>Cold: <39 s (No almanac or ephemeris and no approximate position or time) | |

[1]Typical values. All position and velocity RMS values are based on Horizontal position accuracy. Performance specifications are subject to GNSS system characteristics, Signal-in-Space (SIS) operational degradation, ionospheric and tropospheric conditions, satellite geometry, baseline length, multipath effects and the presence of intentional or unintentional interference sources.

[2]GPS-only.

[3]Requires a TerraStar subscription which is available direct from NovAtel novatel.com/products/correction-services/terrastar-correction-services.

[4]Performance dependent on local observing conditions.

[5]Although hardware capable, GLONASS L5 is currently not available.

[6]Currently the receiver can track up to 3 L-Band channels.

| | | | | |
|---|---|---|---|---|
| Signal Reacquisition | <0.5 s L1 (typical)<br><1.0 s L2 and L5 (typical) | | | |
| Data Rates | Measurements | up to 100 Hz | | |
| | Position | up to 100 Hz | | |
| Time Accuracy[1] | 20 ns RMS | | | |
| Velocity Accuracy | <0.03 m/s RMS | | | |
| Measurement Precision | | | Code | Carrier |
| | GPS | L1 C/A | 4 cm | 0.5 mm |
| | | L2 P(Y) | 8 cm | 1.0 mm |
| | | L2C | 8 cm | 0.5 mm |
| | | L5 | 3 cm | 0.5 mm |
| | GLONASS | L1 C/A | 8 cm | 1.0 mm |
| | | L2 P | 8 cm | 1.0 mm |
| | | L2 C/A | 8 cm | 1.0 mm |
| | Galileo | E1 | 3 cm | 0.5 mm |
| | | E5a | 3 cm | 0.75 mm |
| | | E5b | 3 cm | 0.75 mm |
| | | E5 AltBOC | 3 cm | 0.75 mm |
| | | E6 | 3 cm | 0.75 mm |
| | BeiDou | B1I | 4 cm | 0.5 mm |
| | | B1C | 3 cm | 0.5 mm |
| | | B2I | 4 cm | 0.5 mm |
| | | B2a | 3 cm | 0.5 mm |
| | | B3I | 4 cm | 0.5 mm |
| Velocity Limit[2] | 515 m/s | | | |

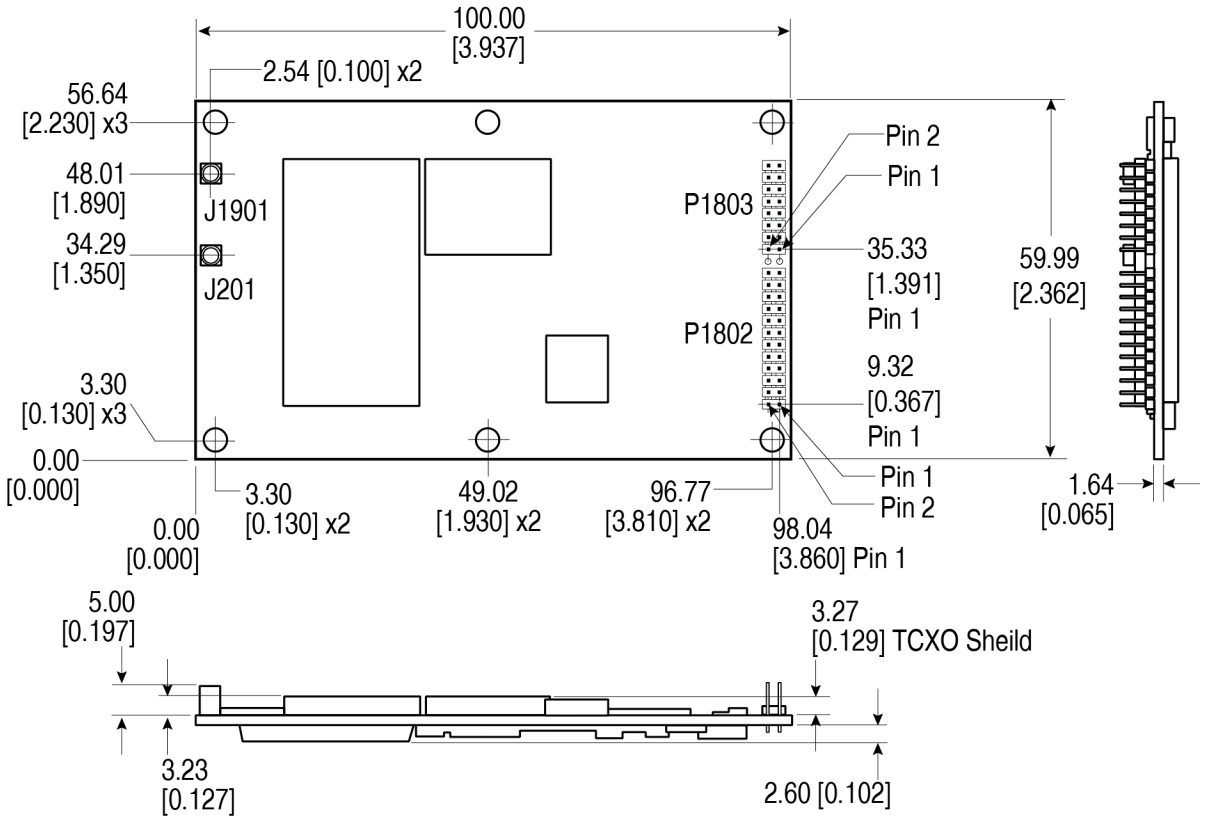
[1]Time accuracy does not include biases due to RF or antenna delay.

[2]Export licensing restricts operation to a maximum of 515 meters per second, message output impacted above 500 m/s.

### 4.2.2 OEM729 Mechanical Specifications

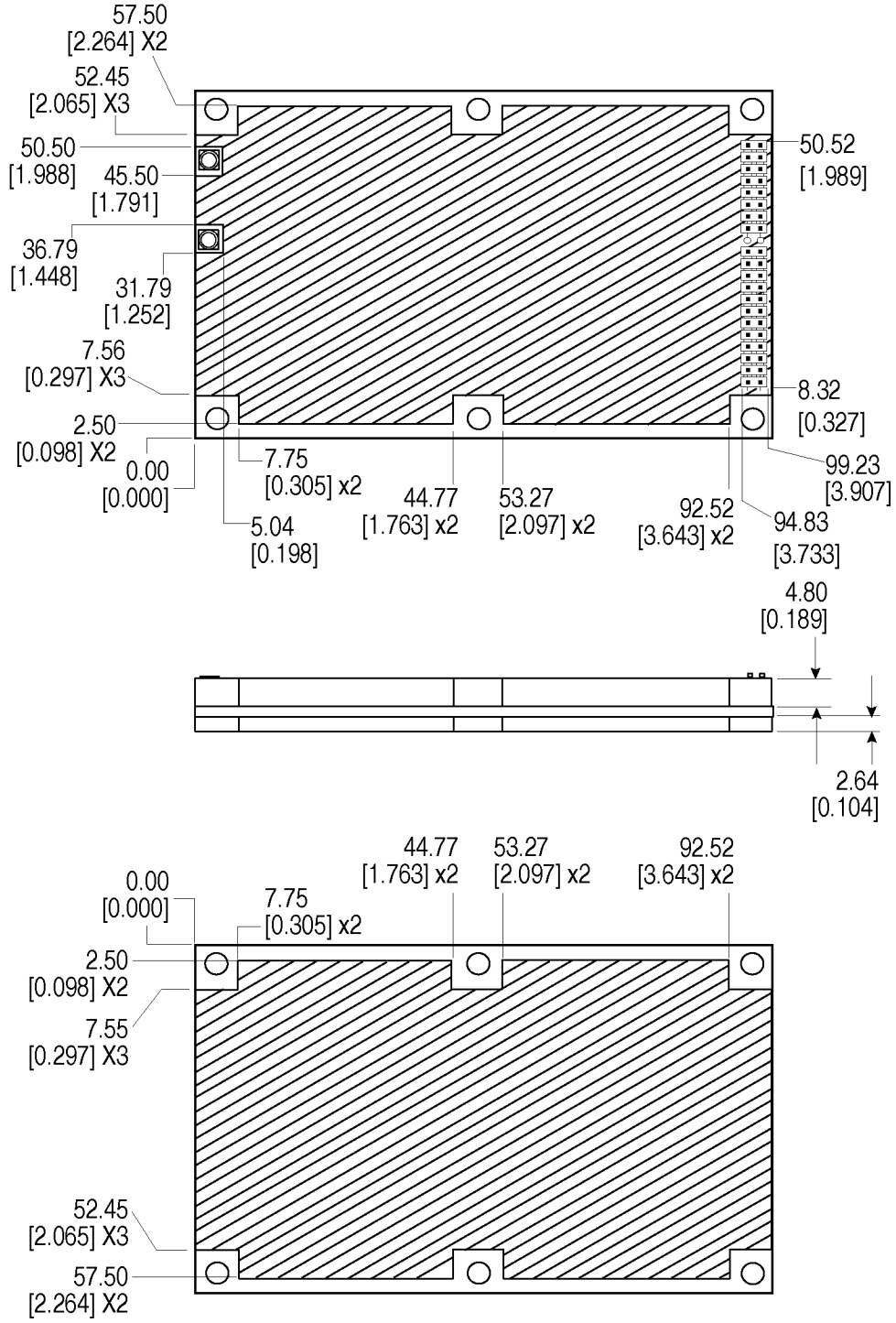
- *Figure 16: OEM729 Dimensions* below
- *Figure 17: OEM729 Keep-outs* on the next page
- *Figure 18: OEM729 Mounting Surfaces* on page 199

**Figure 16: OEM729 Dimensions**



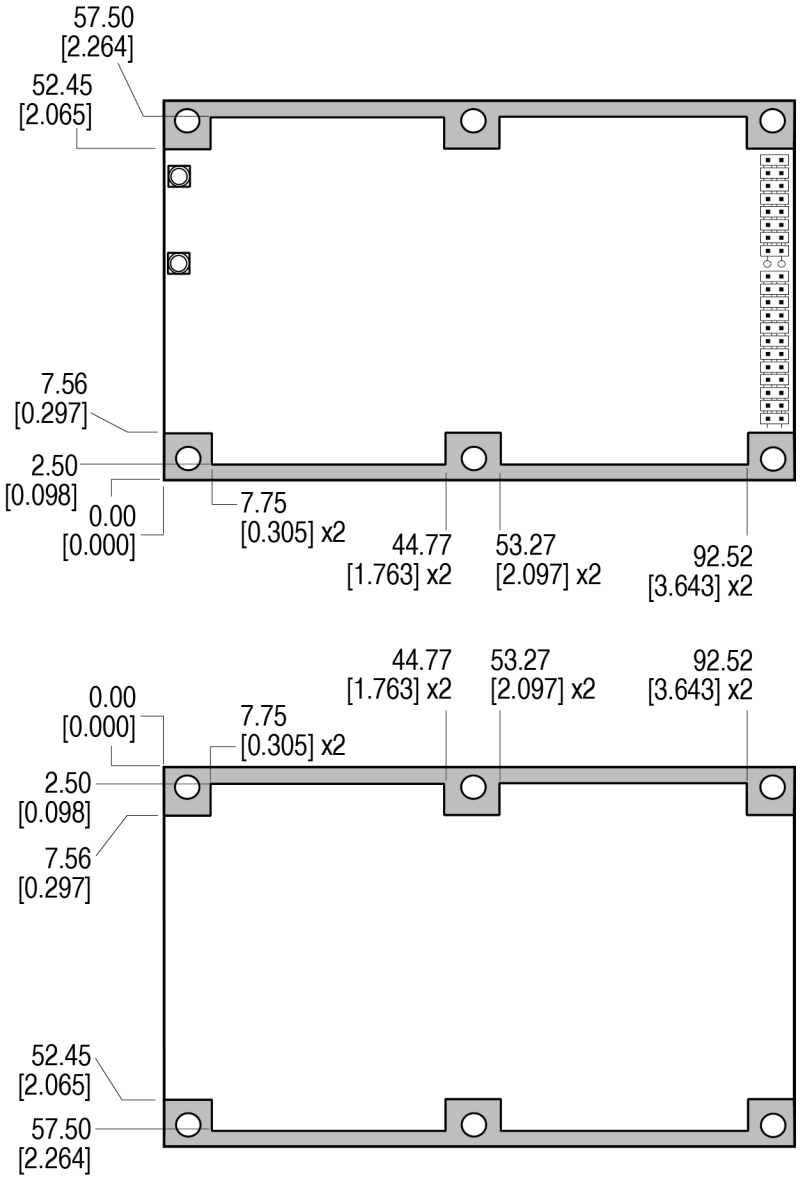

Notes:
Dimensions are in millimetres [inches].
Connectors:
(a) J201 and 1901 MMCX jack receptacle (Johnson P/N 135-3701-201 or Samtec P/N RSP-149374-01 or equivalent)
(b) P1802 2x12 header, 2 mm pitch (Samtec P/N TMM-112-03-G-D)
P1803 2x8 header, 2 mm pitch (Samtec P/N TMM-108-03-G-D)

**Figure 17: OEM729 Keep-outs**




Dimensions are in millimetres [inches].

**Figure 18: OEM729 Mounting Surfaces**




Dimensions are in millimetres [inches].

The mounting surfaces are shown in gray. For information about mounting the OEM729, refer to Mounting the Printed Circuit Board.

Assembly tolerances must be considered when using mounting rail features.

### 4.2.3 OEM729 Electrical and Environmental Specifications

**Table 103: OEM729 Environmental Specifications**

| | |
|---|---|
| Operating Temperature | -40°C to +85°C |
| Storage Temperature | -55°C to +95°C |
| Humidity | 95% non-condensing |
| Random Vibration | MIL-STD-810G, Method 514.6, Category 24 (20 g RMS) |
| Sinusoidal Vibration | IEC 60068-2-6 |
| Bump | ISO 9022-31-06 (25 g) |
| Shock<br>Operating<br>Non-operating | <br>MIL-STD-810G, Method 516.6 (40 g)<br>MIL-STD-810G, Method 516.6 (75 g) |
| Acceleration | MIL-STD-810G (CH1), Method 513.7, (16 g) |

**Table 104: OEM729 Power Requirements**

| | |
|---|---|
| Voltage | +3.3 VDC ±5% |
| Allowable Input Voltage Ripple | 100 mV p-p maximum |
| Power Consumption | 0.9 W typical, GPS L1 only<br>1.3 W typical, GPS L1/L2, GLONASS L1/L2<br>1.8 W typical, all constellations, all frequencies, plus L-Band<br>These are typical values using serial ports without interference mitigation. These values can change with the number of satellites in view, firmware version, data logging rates and features in use. Use them as a guide for what you might expect but not as absolute values |
| Inrush Current | 2.0 A for less than 1.8 ms (typical) |

**Table 105: OEM729 RF Input/LNA Power Output**

| | |
|---|---|
| Antenna Connector | MMCX female, 50 Ω nominal impedance |

| | | | |
|---|---|---|---|
| Cascaded antenna LNA gain (before receiver) | Firmware 7.04 and later | HDR disabled | +15 dB to +55 dB, 26 dB typical |
| | | HDR enabled | +20 dB to +55 dB, 30 dB typical |
| | Firmware before 7.04 | HDR disabled | +15 dB to +40 dB, 26 dB typical |
| | | HDR enabled | +20 dB to +40 dB, 30 dB typical |
| RF Input Frequencies | GPS L1:<br>GPS L2:<br>GPS L5: | 1575.42 MHz<br>1227.60 MHz<br>1176.45 MHz | |
| | GLONASS L1:<br>GLONASS L2:<br>GLONASS L3: | 1593-1610 MHz<br>1237-1254 MHz<br>1202.025 MHz | |
| | Galileo E1:<br>Galileo E5a:<br>Galileo E5b:<br>Galileo E5:<br>Galileo E6: | 1575.42 MHz<br>1176.45 MHz<br>1207.14 MHz<br>1191.795 MHz<br>1278.75 MHz | |
| | BeiDou B1I:<br>BeiDou B1C:<br>BeiDou B2I:<br>BeiDou B2a:<br>BeiDou B3I: | 1561.098 MHz<br>1575.42 MHz<br>1207.14 MHz<br>1176.45 MHz<br>1268.52 MHz | |
| | L-Band: | 1525 to 1560 MHz | |
| LNA Power | +5.0 VDC ±5%, 0 mA to 200 mA (supplied by card through center conductor of RF connector).<br>LNA Power is generated from the 3.3 V supply input for the OEM729. | | |

**Table 106: OEM729 External Oscillator Input**

| | |
|---|---|
| External Oscillator Connector | MMCX female, 50 Ω nominal impedance |
| External Clock input | Refer to the **EXTERNALCLOCK** command |
| Frequency | 5 MHz or 10 MHz |
| Input Impedance | 50 Ohm nominal |

| | |
|---|---|
| Input VSWR | <2:1 |
| Signal Level | 0 dBm minimum to +13.0 dBm maximum |
| Frequency Stability | ±0.5 ppm maximum |
| Wave Shape | Sinusoidal |

### 4.2.4 OEM729 Data Communication Specifications

**Table 107: Data Communications Interface**

| | |
|---|---|
| **COM1** | |
| Electrical format | RS-232/RS-422 |
| Data rates[1] | 2400, 4800, 9600 (default), 19200, 38400, 57600, 115200, 230400 or 460800 bit/s. |
| Signals supported | COM1_Tx, COM1_Rx, COM1_RTS, COM1_CTS |
| Electrostatic discharge protection | Yes |
| **COM2** | |
| Electrical format | LVCMOS |
| Data rates[1] | 2400, 4800, 9600 (default), 19200, 38400, 57600, 115200, 230400 or 460800 bit/s. |
| Signals supported | COM2_Tx, COM2_Rx, COM2_RTS, COM2_CTS |
| Electrostatic discharge protection | No |
| **COM3** | |
| Electrical format | LVCMOS [2, 3] |
| Data rates[1] | 2400, 4800, 9600 (default), 19200, 38400, 57600, 115200, 230400 or 460800 bit/s. |
| Signals supported | COM3_Tx, COM3_Rx |
| Electrostatic discharge protection | No |
| **CAN Bus** | |
| Electrical Format | LVCMOS |
| Data rates | 1 Mbps maximum.<br>CAN Bus throughput is determined by slowest device on the bus |

[1]Data rates higher than 115200 bit/s are not supported by standard PC hardware. Special PC hardware may be required for higher rates, including 230400 bit/s and 460800 bit/s.

[2]Upon power-up, COM3 is enabled by default. COM3 is multiplexed with Event 2

[3]To enable EVENT2, issue the following commands:

**INTERFACEMODE COM3 NONE NONE**
**EVENTINCONTROL MARK2 ENABLE**

| | |
|---|---|
| Signals supported | CAN1 and CAN2 |
| **USB** | |
| Electrical format | Conforms to USB 2.0 |
| Data rates | Full-speed (12 Mb/s) |
| Signals supported | USB D (+), USB D (-) |
| **ETHERNET** | |
| Physical layer | 10BASE-T/100BASE-TX |

### *4.2.5 OEM729 Strobe Specifications*

**Table 108: OEM729 Strobes Description**

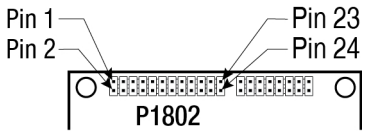
| Strobes | Input/Output | Factory Default | Comment |
|---|---|---|---|
| ERROR | Output | Active high | Normally low.<br>A high output on this pin indicates that the receiver is in an error state. For information about the cause of the error, log the **RXSTATUS** log. |
| Event1 | Input<br>Leading edge triggered | Active low | An input mark for which a pulse greater than 150 ns triggers certain logs to be generated. (Refer to the MARKPOS and MARK1TIME logs and ONMARK trigger.) Polarity is configurable using the **EVENTINCONTROL** command. |
| Event2 (Mark 2) | Input<br>Leading edge triggered | Active low | An input mark for which a pulse greater than 150 ns triggers certain logs to be generated. (Refer to the MARK2POS and MARK2TIME logs.) Polarity is configurable using the **EVENTINCONTROL** command.<br>**Note**: Event2 is the default behavior for pin 7 of connector P1804. The secondary behavior for pin 7 is COM3_Rx. If COM3 is enabled, the Event2 input is disabled. |
| PPS | Output | Active low | A time synchronization output. This is a pulse where the leading edge is synchronized to receiver calculated GNSS Time. The polarity, period and pulse width can be configured using the **PPSCONTROL** command |
| PV (Position Valid) | Output | Active high | Indicates a valid GNSS position solution is available. |
| nRESET_IN | Input | Active low | Reset signal input from external system; active low.<br>This pin must be held low for >20 µs while stable power is already applied, to reset the OEM729 card.<br>It is recommended to hold the nRESET_IN pin low for a >150 ms when initially applying power to the card, giving the power supply time to stabilize before the card starts to boot. |
| VARF (Variable Frequency) | Output | Active low | A programmable variable frequency output ranging from 0 Hz to 50 MHz (refer to the **EVENTOUTCONTROL** command) |

**Table 109: OEM729 Strobe Electrical Specifications**

| Strobe | Sym | Min (V) | Max (V) | Current (mA) |
|---|---|---|---|---|
| ERROR | $V_{OL}$ | | 0.4 | 4 |
| | $V_{OH}$ | 2.9 | | |
| Event1 (Mark ) Event2 (Mark2) | $V_{IL}$ | | 0.7 | - |
| | $V_{IH}$ | 2.1 | | |
| PPS | $V_{OL}$ | | 0.4 | 16 |
| | $V_{OH}$ | 2.4 | | |
| PV | $V_{OL}$ | | 0.4 | 4 |
| | $V_{OH}$ | 2.9 | | |
| nRESET_IN | $V_{IL}$ | | 0.8 | - |
| | $V_{IH}$ | 2.3 | | |
| VARF | $V_{OL}$ | | 0.4 | 4 |
| | $V_{OH}$ | 2.9 | | |

## *4.2.6 OEM729 Interface Connectors*

### P1802 Main Connector 24-Pin Header




| Pin | Signal Name | Signal Type | Signal Direction | $V_{IL}$ Max (V) | $V_{IH}$ Min (V) | $V_{OL}$ Max (V) | $V_{OH}$ Min (V) | Drive (mA) | Description |
|---|---|---|---|---|---|---|---|---|---|
| 1 | GND | PWR | - | - | - | - | - | - | Ground reference |
| 2 | USER1[1] | 3.3V CMOS | Input/ Output | 0.7 | 2.1 | 0.4 | 2.9 | 4 | User GPIO.<br>Internal 10 kΩ pulldown. |
| 3 | VARF | 3.3V CMOS | Output | - | - | 0.4 | 2.9 | 4 | Variable Frequency output<br>Edges can be synchronized to the GNSS time reference.<br>Internal 10 kΩ pullup |
| 4 | PPS | 3.3V CMOS | Output | - | - | 0.4 | 2.4 | 16 | Pulse Per Second output<br>This signal defaults to one pulse per second but may be altered across a wide range of frequencies using software commands. Edges can be synchronized to GNSS time reference. |
| 5 | 3V3 | PWR | - | - | - | - | - | - | 3.3 V ±5% supply input |
| 6 | 3V3 | PWR | - | - | - | - | - | - | 3.3 V ±5% supply input |
| 7 | COM3_ RX/ EVENT2[2] | 3.3V CMOS | Input | 0.7 | 2.1 | - | - | - | This pin is internally multiplexed.<br>COM3_RX: COM3 receive data input.<br>EVENT2 input. Rising or falling edge triggered. This is used to provide position or time data on an external trigger.<br>Internal 10 kΩ pullup. |

[1]On power up, if pin 2 is set LOW or not connected, COM1 will be configured as RS-232. If pin 2 is set high then COM1 will be configured as RS-422

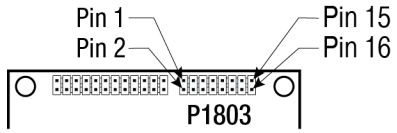
[2]Through firmware, the COM3 UART may be configured on pins 7 and 19.

| Pin | Signal Name | Signal Type | Signal Direction | $V_{IL}$ Max (V) | $V_{IH}$ Min (V) | $V_{OL}$ Max (V) | $V_{OH}$ Min (V) | Drive (mA) | Description |
|---|---|---|---|---|---|---|---|---|---|
| 8 | EVENT1 | 3.3V CMOS | Input | 0.7 | 2.1 | - | - | - | EVENT1 input<br>Rising or falling edge triggered. This input is used to provide a position or time data log based on an external trigger.<br>Internal 10 kΩ pullup. |
| 9 | ERROR | 3.3V CMOS | Output | - | - | 0.4 | 2.9 | 4 | ERROR output<br>Normally low. A high output on this pin indicates that the receiver is in an error state.<br>Internal 10 kΩ pulldown. |
| 10 | PV | 3.3V CMOS | Output | - | - | 0.4 | 2.9 | 4 | Position Valid output<br>A high output on this pin indicates that the receiver has computed a valid GNSS position.<br>Internal 10 kΩ pulldown. |
| 11 | COM2_ CTS | 3.3V CMOS | Input | 0.7 | 2.1 | - | - | - | COM2 Clear To Send input<br>This is an optional flow control signal for the COM2 UART.<br>Internal weak (40 kΩ to 100 kΩ) pullup. |
| 12 | nRESET_ IN | Other | Input | 0.8 | 2.3 | - | - | - | Active Low.<br>Resets the OEM729 receiver card. This pin must be held low for >20 µs while stable power is already applied, to reset the OEM729 card.<br>It is recommended to hold the nRESET_IN pin low for a >150 ms when initially applying power to the card, giving the power supply time to stabilize before the card starts to boot.<br>Internal 10 kΩ pullup. |

| Pin | Signal Name | Signal Type | Signal Direction | $V_{IL}$ Max (V) | $V_{IH}$ Min (V) | $V_{OL}$ Max (V) | $V_{OH}$ Min (V) | Drive (mA) | Description |
|---|---|---|---|---|---|---|---|---|---|
| 13 | COM2_ RTS | 3.3V CMOS | Output | - | - | 0.4 | 2.9 | 4 | COM2 Request To Send output<br>This is an optional flow control signal for the COM2 UART.<br>For SPAN applications, this pin can be configured to output a timing signal periodically (generally 1PPS). |
| 14 | COM2_RX | 3.3V CMOS | Input | 0.7 | 2.1 | - | - | - | COM2 Receive Data input<br>Internal weak (40 kΩ to 100 kΩ) pullup. |
| 15 | COM1_ CTS/ COM1_ RXD- | COM1_ CTS: RS-232 | Input | - | - | - | - | - | This pin is internally multiplexed. COM1_CTS is the default.<br>COM1_CTS: COM1 Clear To Send input. This is an optional flow control signal for the COM1 UART (±25V tolerant).<br>COM1_RXD-: This is one half of the COM1 RS-422 receive differential pair (2V differential typical) |
| | | COM1_ RXD-: RS-422 | Input | - | - | - | - | - | |
| 16 | COM2_TX | 3.3V CMOS | Output | - | - | 0.4 | 2.9 | 4 | COM2 Transmit Data output<br>For SPAN applications, this pin can be configured to output a timing signal periodically (generally 1PPS). |
| 17 | COM1_ RTS/ COM1_ TXD- | COM1_ RTS: RS-232 | Output | - | - | - | - | - | This pin is internally multiplexed. COM1_RTS is the default.<br>COM1_RTS: COM1 Request To Send output. This is an optional flow control signal for the COM1 UART (±25V tolerant).<br>COM1_TXD-: This is one half of the COM1 RS-422 transmit differential pair. (2V differential typical) |
| | | COM1_ TXD-: RS-422 | Output | - | - | - | - | - | |

| Pin | Signal Name | Signal Type | Signal Direction | $V_{IL}$ Max (V) | $V_{IH}$ Min (V) | $V_{OL}$ Max (V) | $V_{OH}$ Min (V) | Drive (mA) | Description |
|---|---|---|---|---|---|---|---|---|---|
| 18 | COM1_ RX/ COM1_ RXD+ | COM1_ RX: RS-232 | Input | 0.7 | 2.1 | - | - | - | This pin is internally multiplexed. COM1_RX is the default.<br>COM1_RX: COM1 Receive Data input (±25V tolerant).<br>COM1_RXD+: This is one half of the COM1 RS-422 receive differential pair (2V differential typical). |
| | | COM1_ RXD+: RS-422 | Input | - | - | - | - | - | |
| 19 | COM3_ TX/USER0 | 3.3V CMOS | COM3_ TX: Output | - | - | 0.4 | 2.9 | 4 | This pin is internally multiplexed. COM3_TX is the default.<br>COM3_TX: COM3 Transmit Data output.<br>USER0: User GPIO. Internal 10 kΩ pulldown. |
| | | | USER0: Input | 0.7 | 2.1 | - | - | - | |
| 20 | COM1_TX/ COM1_ TXD+ | COM1_ TX: RS-232 | Output | - | - | - | - | - | This pin is internally multiplexed. COM1_TX is the default.<br>COM1_TX: COM1 Transmit Data output. (±25V tolerant)<br>For SPAN applications, this pin can be configured to output a timing signal periodically (generally 1PPS).<br>COM1_TXD+: This is one half of the COM1 RS-422 transmit differential pair (2V differential typical) |
| | | COM1_ TXD+: RS-422 | Output | - | - | - | - | - | |
| 21 | USB_D- | Analog | Input/ Output | - | - | - | - | - | USB device signal.<br>This is one half of the USB differential pair. USB_D+ and USB_D- must be length-matched and routed as a 90 Ω differential pair. |
| 22 | USB_D+ | Analog | Input/ Output | - | - | - | - | - | USB device signal.<br>This is one half of the USB differential pair. USB_D+ and USB_D- must be length-matched and routed as a 90 Ω differential pair. |
| 23 | GND | PWR | - | - | - | - | - | - | Ground reference |
| 24 | GND | PWR | - | - | - | - | - | - | Ground reference |

### P1803 Expansion Connector 16-Pin Header




| Pin | Signal Name | Signal Type | Signal Direction | $V_{IL}$ Max (V) | $V_{IH}$ Min (V) | $V_{OL}$ Max (V) | $V_{OH}$ Min (V) | Drive (mA) | Description |
|---|---|---|---|---|---|---|---|---|---|
| 1 | ETH_ RD- | Analog | Input | - | - | - | - | - | This is one half of the Ethernet receive differential pair (100 Ω pair). |
| 2 | ETH_ RD+ | Analog | Input | - | - | - | - | - | This is one half of the Ethernet receive differential pair (100 Ω pair). |
| 3 | ETH_ BIAS | PWR | - | - | - | - | - | - | Center tap power for Ethernet magnetics. |
| 4 | ETH_ TD+ | Analog | Output | - | - | - | - | - | This is one half of the Ethernet transmit differential pair (100 Ω pair). |
| 5 | ETH_ TD- | Analog | Output | - | - | - | - | - | This is one half of the Ethernet transmit differential pair (100 Ω pair). |
| 6 | ETH_ BIAS | PWR | - | - | - | - | - | - | Center tap power for Ethernet magnetics. |
| 7 | LED_A | 3.3V CMOS | Output | - | - | 0.4 | 2.9 | 8 | Activity/Link indicator output.<br>Polarity of the indicator signal is low. When there is an active link, the pin is low. When there is activity on the link, the pin outputs a blink signal. |
| 8 | LED_B | 3.3V CMOS | Output | - | - | 0.4 | 2.9 | 8 | Speed indicator<br>Low = 100 Mbps<br>High = 10 Mbps |
| 9 | GND | PWR | - | - | - | - | - | - | Ground reference |
| 10 | CAN1TX | 3.3V CMOS | Output | - | - | - | - | - | CAN1 Transmit data |
| 11 | CAN1RX | 3.3V CMOS | Input | - | - | - | - | - | CAN1 Receive data |
| 12 | CAN2TX | 3.3V CMOS | Output | - | - | - | - | - | CAN2 Transmit data |
| 13 | CAN2RX | 3.3V CMOS | Input | - | - | - | - | - | CAN2 Receive data |

| Pin | Signal Name | Signal Type | Signal Direction | $V_{IL}$ Max (V) | $V_{IH}$ Min (V) | $V_{OL}$ Max (V) | $V_{OH}$ Min (V) | Drive (mA) | Description |
|---|---|---|---|---|---|---|---|---|---|
| 14 | UID | 3.3V CMOS | Input | - | - | - | - | - | USB Port Mode<br>Leave this pin floating to ensure the USB port is in Device mode.<br>Host mode is not currently supported on the OEM729.<br>Internal 10 kΩ pull up |
| 15 | VBUS | PWR | - | - | - | - | - | - | 5V output for hosted USB devices |
| 16 | GND | PWR | - | - | - | - | - | - | Ground reference |

## 4.3 OEM7700 Technical Specifications

**Table 110: OEM7700 Physical Description**

| | |
|---|---|
| Size | 46 mm x 71 mm x 8 mm |
| Weight | 31 grams |
| NovAtel Part Number | Generic assembly OEM7700 01019525 |

See the following sections for more information about the OEM7700:


- *OEM7700 Performance Specifications* on the next page
- *OEM7700 Mechanical Specifications* on page 216
- *OEM7700 Electrical and Environmental Specifications* on page 219
- *OEM7700 Data Communication Specifications* on page 221
- *OEM7700 Strobe Specifications* on page 223
- *OEM7700 Interface Connector* on page 225

### 4.3.1 OEM7700 Performance Specifications

All specifications subject to GNSS system characteristics.

**Table 111: OEM7700 Receiver Performance**

| | | |
|---|---|---|
| Position Accuracy[1] | Single point L1 | 1.5 m RMS |
| | Single point L1/L2 | 1.2 m RMS |
| | SBAS[2] | 60 cm RMS |
| | DGPS | 40 cm RMS |
| | TerraStar-L[3, 4] | 40 cm RMS |
| | TerraStar-C PRO[3, 4] | 2.5 cm RMS |
| | TerraStar-X[3, 4] | 2.0 cm RMS |
| | RTK | 1 cm + 1 ppm RMS |
| Signals Tracked | GPS | L1 C/A, L1C, L2C, L2P, L5 |
| | GLONASS[5] | L1 C/A, L2 C/A, L2P, L3, L5 |
| | BeiDou | B1I, B1C, B2I, B2a, B3I |
| | Galileo | E1, E5 AltBOC, E5a, E5b, E6 |
| | NavIC (IRNSS) | L5 |
| | QZSS | L1 C/A, L1C, L2C, L5, L6 |
| | SBAS | L1, L5 |
| | L-Band[6] | Up to 5 channels |
| Time to First Fix | Hot: <20 s (Almanac and recent ephemeris saved and approximate position and time entered)<br>Cold: <39 s (No almanac or ephemeris and no approximate position or time) | |

[1]Typical values. All position and velocity RMS values are based on Horizontal position accuracy. Performance specifications are subject to GNSS system characteristics, Signal-in-Space (SIS) operational degradation, ionospheric and tropospheric conditions, satellite geometry, baseline length, multipath effects and the presence of intentional or unintentional interference sources.

[2]GPS-only.

[3]Requires a TerraStar subscription which is available direct from NovAtel novatel.com/products/correction-services/terrastar-correction-services.

[4]Performance dependent on local observing conditions.

[5]Although hardware capable, GLONASS L5 is currently not available.

[6]Currently the receiver can track up to 3 L-Band channels.

| | | | | |
|---|---|---|---|---|
| Signal Reacquisition | <0.5 s L1 (typical)<br><1.0 s L2 and L5 (typical) | | | |
| Data Rates | Measurements | up to 100 Hz | | |
| | Position | up to 100 Hz | | |
| Time Accuracy[1] | 20 ns RMS | | | |
| Velocity Accuracy | <0.03 m/s RMS | | | |
| Measurement Precision | | | Code | Carrier |
| | GPS | L1 C/A | 4 cm | 0.5 mm |
| | | L2 P(Y) | 8 cm | 1.0 mm |
| | | L2C | 8 cm | 0.5 mm |
| | | L5 | 3 cm | 0.5 mm |
| | GLONASS | L1 C/A | 8 cm | 1.0 mm |
| | | L2 P | 8 cm | 1.0 mm |
| | | L2 C/A | 8 cm | 1.0 mm |
| | Galileo | E1 | 3 cm | 0.5 mm |
| | | E5a | 3 cm | 0.75 mm |
| | | E5b | 3 cm | 0.75 mm |
| | | E5 AltBOC | 3 cm | 0.75 mm |
| | | E6 | 3 cm | 0.75 mm |
| | BeiDou | B1I | 4 cm | 0.5 mm |
| | | B1C | 3 cm | 0.5 mm |
| | | B2I | 4 cm | 0.5 mm |
| | | B2a | 3 cm | 0.5 mm |
| | | B3I | 4 cm | 0.5 mm |
| Velocity Limit[2] | 515 m/s | | | |

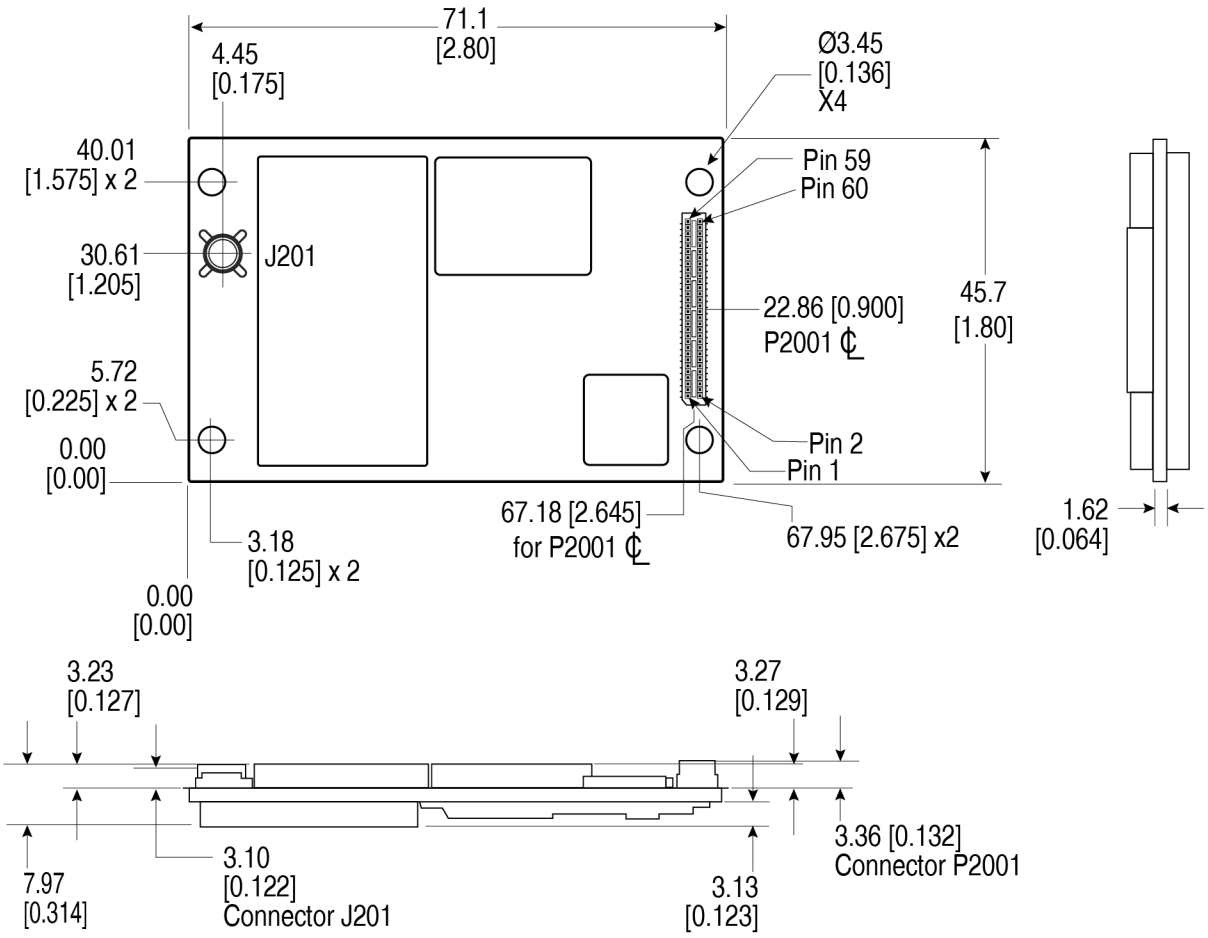
[1]Time accuracy does not include biases due to RF or antenna delay.

[2]Export licensing restricts operation to a maximum of 515 meters per second, message output impacted above 500 m/s.

### 4.3.2 OEM7700 Mechanical Specifications

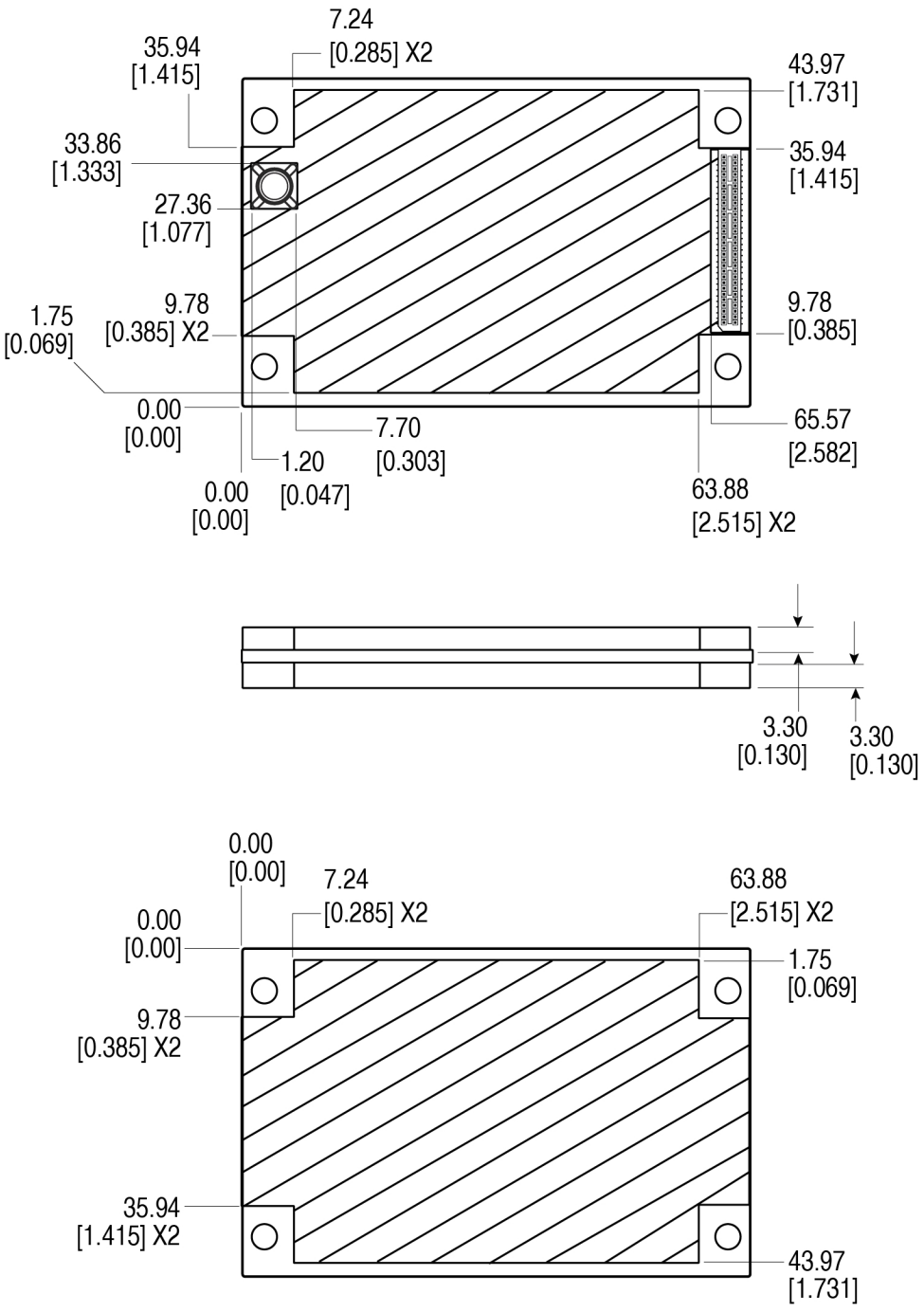
- *Figure 19: OEM7700 Dimensions* below
- *Figure 20: OEM7700 Keep-outs* on the next page
- *Figure 21: OEM7700 Mounting Surfaces* on page 218

**Figure 19: OEM7700 Dimensions**



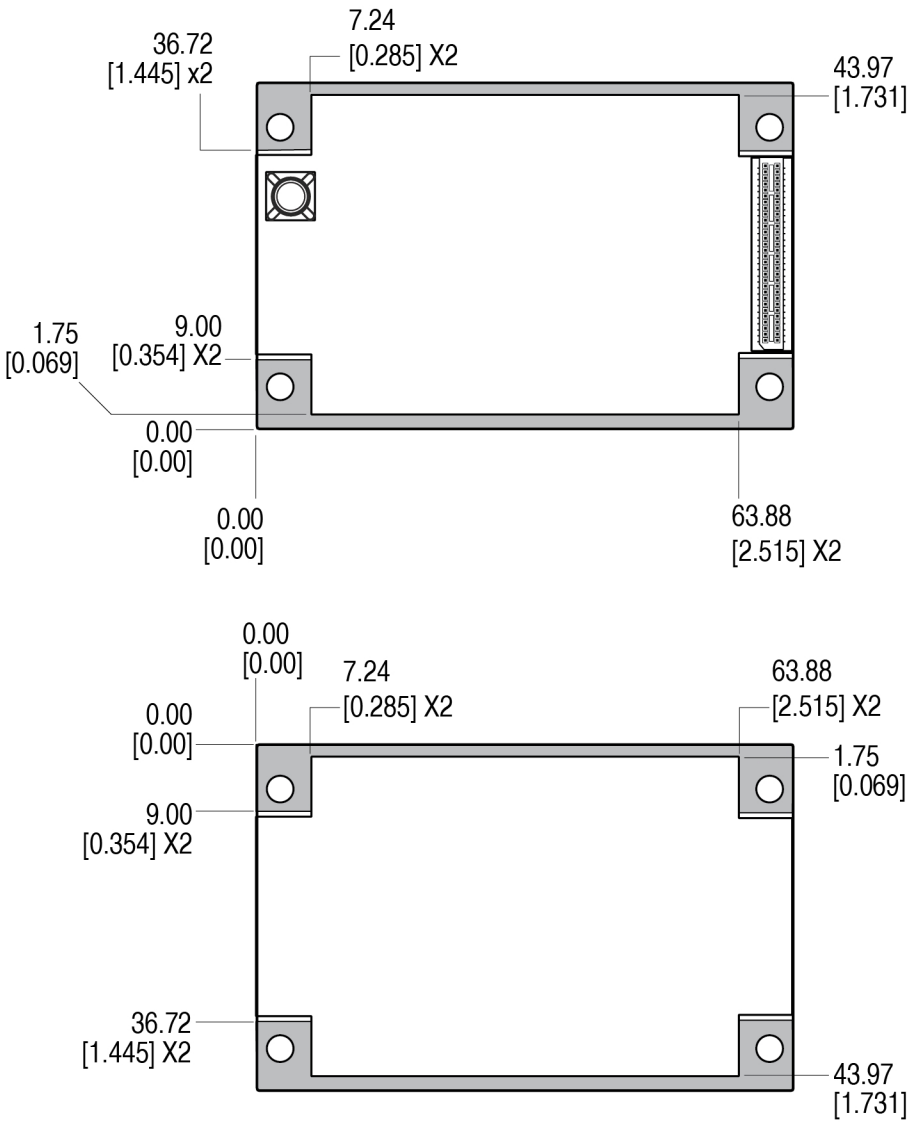

Notes: OEM7700
Dimensions are in millimetres [inches].
Connectors:
(a) J201 Huber + Shuhner 82_MMBX-S50-0-1/111 NE or 82_MMBX-0-1/111 NH or 82_MMBX-S50-0-1/111 NM
(b) P2001 Samtec ASP-189258-01

**Figure 20: OEM7700 Keep-outs**




Dimensions are in millimetres [inches].

**Figure 21: OEM7700 Mounting Surfaces**




Dimensions are in millimetres [inches].

The mounting surfaces are shown in gray.

These mounting surfaces are designed to work with mounting rails. For information about mounting the OEM7700, refer to Mounting the Printed Circuit Board.

Assembly tolerances must be considered when using mounting rail features.

### 4.3.3 OEM7700 Electrical and Environmental Specifications

**Table 112: OEM7700 Environmental Specifications**

| | |
|---|---|
| Operating Temperature | -40°C to +85°C |
| Storage Temperature | -55°C to +95°C |
| Humidity | 95% non-condensing |
| Random Vibration | MIL-STD-810G, Method 514.6, Category 24 (20 g RMS)[1] |
| Sinusoidal Vibration | IEC 60068-2-6 |
| Bump | ISO 9022-31-06 (25 g) |
| Shock<br>Operating<br>Non-operating | <br>MIL-STD-810G, Method 516.6 (40 g)<br>MIL-STD-810G, Method 516.6 (75 g) |
| Acceleration | MIL-STD-810G, Method 513.7 (16 g) |

**Table 113: OEM7700 Power Requirements**

| | |
|---|---|
| Voltage | +3.3 VDC ±5% |
| Allowable Input Voltage Ripple | 100 mV p-p maximum |
| Power Consumption | 0.9 W typical, GPS L1 only<br>1.3 W typical, GPS L1/L2, GLONASS L1/L2<br>1.8 W typical, all constellations, all frequencies, plus L-Band<br>These are typical values using serial ports without interference mitigation. These values can change with the number of satellites in view, firmware version, data logging rates and features in use. Use them as a guide for what you might expect but not as absolute values |
| Inrush Current | 2.0 A for less than 1.8 ms (typical) |

**Table 114: OEM7700 RF Input/LNA Power Output**

| | |
|---|---|
| Antenna Connector | MMBX female, 50 Ω nominal impedance |

[1] Requires mechanical mounting rails to meet 20 g; meets 7.7 g without rails.

| | | | |
|---|---|---|---|
| Cascaded antenna LNA gain (before receiver) | Firmware 7.04 and later | HDR disabled | +15 dB to +55 db, 26 dB typical |
| | | HDR enabled | +20 dB to +55 db, 30 dB typical |
| | Firmware before 7.04 | HDR disabled | +15 dB to +40 db, 26 dB typical |
| | | HDR enabled | +20 dB to +40 db, 30 dB typical |
| RF Input Frequencies | GPS L1:<br>GPS L2:<br>GPS L5: | 1575.42 MHz<br>1227.60 MHz<br>1176.45 MHz | |
| | GLONASS L1:<br>GLONASS L2:<br>GLONASS L3: | 1593-1610 MHz<br>1237-1254 MHz<br>1202.025 MHz | |
| | Galileo E1:<br>Galileo E5a:<br>Galileo E5b:<br>Galileo E5:<br>Galileo E6: | 1575.42 MHz<br>1176.45 MHz<br>1207.14 MHz<br>1191.795 MHz<br>1278.75 MHz | |
| | BeiDou B1I:<br>BeiDou B1C:<br>BeiDou B2I:<br>BeiDou B2a:<br>BeiDou B3I: | 1561.098 MHz<br>1575.42 MHz<br>1207.14 MHz<br>1176.45 MHz<br>1268.52 MHz | |
| | L-Band: | 1525 to 1560 MHz | |
| LNA Power | +5.0 VDC ±5%, 0 mA to 200 mA (supplied by card through center conductor of RF connector).<br>LNA Power is generated from the 3.3 V supply input for the OEM7700. | | |

### 4.3.4 OEM7700 Data Communication Specifications

**Table 115: Data Communications Interface**

| | |
|---|---|
| **COM1** | |
| Electrical format | LVCMOS |
| Data rates[1] | 2400, 4800, 9600 (default), 19200, 38400, 57600, 115200, 230400 or 460800 bit/s. |
| Signals supported | COM1_Tx, COM1_Rx, COM1_RTS, COM1_CTS |
| Electrostatic discharge protection | No |
| **COM2** | |
| Electrical format | LVCMOS |
| Data rates[1] | 2400, 4800, 9600 (default), 19200, 38400, 57600, 115200, 230400 or 460800 bit/s. |
| Signals supported | COM2_Tx, COM2_Rx, COM2_RTS, COM2_CTS |
| Electrostatic discharge protection | No |
| **COM3** | |
| Electrical format | LVCMOS |
| Data rates[1] | 2400, 4800, 9600 (default), 19200, 38400, 57600, 115200, 230400 or 460800 bit/s. |
| Signals supported | COM3_Tx, COM3_Rx |
| Electrostatic discharge protection | No |
| **COM4** | |
| Electrical format | LVCMOS |
| Data rates[1] | 2400, 4800, 9600 (default), 19200, 38400, 57600, 115200, 230400 or 460800 bit/s. |
| Signals supported | COM4_Tx, COM4_Rx |
| Electrostatic discharge protection | No |

[1]Data rates higher than 115200 bit/s are not supported by standard PC hardware. Special PC hardware may be required for higher rates, including 230400 bit/s and 460800 bit/s.

| COM5 | |
|---|---|
| Electrical format | LVCMOS[1] |
| Data rates[1] | 2400, 4800, 9600 (default), 19200, 38400, 57600, 115200, 230400 or 460800 bit/s. |
| Signals supported | COM5_Tx, COM5_Rx |
| Electrostatic discharge protection | No |
| **CAN Bus** | |
| Electrical Format | LVCMOS |
| Data rates | 1 Mbps maximum.<br>CAN Bus throughput is determined by slowest device on the bus |
| Signals supported | CAN0 and CAN1 |
| **USB** | |
| Electrical format | Conforms to USB 2.0 |
| Data rates | Hi-speed (480 Mb/s) |
| Signals supported | USB0 D+, USB0 D-<br>USB1 D+, USB1 D- |
| **ETHERNET** | |
| Physical layer | 10BASE-T/100BASE-TX |

[1]Upon power-up, COM5 is enabled by default. COM5 is multiplexed with COM2 hardware flow control.

### 4.3.5 OEM7700 Strobe Specifications

**Table 116: OEM7700 Strobes Description**

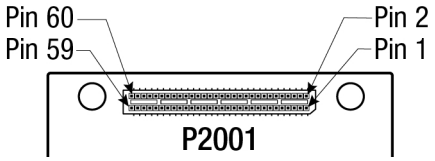
| Strobes | Input/Output | Factory Default | Comment |
|---|---|---|---|
| ERROR | Output | Active high | Normally low.<br>A high output on this pin indicates that the receiver is in an error state. For information about the cause of the error, log the **RXSTATUS** log. |
| EVENT_IN1<br>EVENT_IN2<br>EVENT_IN3<br>EVENT_IN4 | Input<br>Leading edge triggered | Active low | Input marks for which a pulse greater than 150 ns triggers certain logs to be generated. (Refer to the MARKxPOS and MARKxTIME logs and ONMARK trigger.) Polarity is configurable using the **EVENTINCONTROL** command. |
| EVENT_OUT1<br>EVENT_OUT2<br>EVENT_OUT3<br>EVENT_OUT4 | Output | Active low | Programmable variable frequency outputs ranging from 0 Hz to 50 MHz (refer to the **EVENTOUTCONTROL** command). |
| ME_RDY | Output | Active high | Indicates to the host system that the receiver is ready to receive commands.<br>This signal is not asserted while booting or during a reset.<br>**Note**: If an error is detected during receiver operation (for example, an over temperature condition), the ERROR signal is asserted and the ME_RDY signal will normally remain asserted: commands can still be sent to the receiver under these conditions. At the time ERROR is asserted, POS_VALID signal is changed to not asserted. In the rare case of a fatal, unrecoverable error that leads to a reboot of the receiver, ME_RDY, ERROR and POS_VALID are all not asserted while the receiver restarts. |
| POS_Valid (Position Valid) | Output | Active high | Indicates a valid GNSS position solution is available. |
| nRESET_IN | Input | Active low | Reset signal input from external system; active low.<br>This pin must be held low for >20 µs while stable power is already applied, to reset the OEM7700 card.<br>It is recommended to hold the nRESET_IN pin low for a >150 ms when initially applying power to the card, giving the power supply time to stabilize before the card starts to boot. |
| PPS | Output | Active low | A time synchronization output. This is a pulse where the leading edge is synchronized to receiver calculated GNSS Time. The polarity, period and pulse width can be configured using the **PPSCONTROL** command |

**Table 117: OEM7700 Strobe Electrical Specifications**

| Strobe | Sym | Min (V) | Max (V) | Current (mA) |
|---|---|---|---|---|
| ERROR | $V_{OL}$ | | 0.2 | 8 |
| | $V_{OH}$ | 3.1 | | |
| EVENT_IN1 (Mark )<br>EVENT_IN2 (Mark2)<br>EVENT_IN3 (Mark3)<br>EVENT_IN4 (Mark4) | $V_{IL}$ | | 0.8 | - |
| | $V_{IH}$ | 1.7 | | |
| EVENT_OUT1<br>EVENT_OUT2<br>EVENT_OUT3<br>EVENT_OUT4 | $V_{OL}$ | | 0.2 | 8 |
| | $V_{OH}$ | 3.1 | | |
| ME_RDY | $V_{OL}$ | | 0.2 | 4 |
| | $V_{OH}$ | 3.1 | | |
| POS_VALID | $V_{OL}$ | | 0.55 | 8 |
| | $V_{OH}$ | 2.3 | | |
| nRESET_IN | $V_{IL}$ | | 0.8 | - |
| | $V_{IH}$ | 2.55 | | |
| PPS | $V_{OL}$ | | 0.55 | 24 |
| | $V_{OH}$ | 2.3 | | |

### *4.3.6 OEM7700 Interface Connector*

**P2001 Main Connector 60-Pin Socket**




| Pin | Signal Name | Signal Type | Signal Direction | $V_{IL}$ Max (V) | $V_{IH}$ Min (V) | $V_{OL}$ Max (V) | $V_{OH}$ Min (V) | Drive (mA) | Description |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 3V3 | Power | Input | - | - | - | - | - | 3.3 V ±5% supply input |
| 2 | 3V3 | Power | Input | - | - | - | - | - | 3.3 V ±5% supply input |
| 3 | COM2_ TXD | 3.3V CMOS | Output | - | - | 0.2 | 3.1 | 4 | COM2 Transmit Data (UART)<br>For SPAN applications, this pin can be configured to output a timing signal periodically (generally 1PPS). |
| 4 | COM1_ TXD | 3.3V CMOS | Output | - | - | 0.55 | 2.3 | 24 | COM1 Transmit Data (UART)<br>For SPAN applications, this pin can be configured to output a timing signal periodically (generally 1PPS). |
| 5 | COM5_ TXD/ COM2_ RTS | 3.3V CMOS | Output | - | - | 0.2 | 3.1 | 4 | This pin is internally multiplexed. COM5_TXD is the default.<br>COM5_TXD: COM5 Transmit Data (UART).<br>COM2_RTS: COM2 Request to Send<br>Optional hardware flow control signal for the COM2 serial port. |
| 6 | COM1_ RTS | 3.3V CMOS | Output | - | - | 0.2 | 3.1 | 4 | COM1 Request to Send<br>Optional hardware flow control signal for the COM1 serial port. |
| 7 | GND | Ground | - | - | - | - | - | - | Ground reference |
| 8 | GND | Ground | - | - | - | - | - | - | Ground reference |
| 9 | COM2_ RXD | 3.3V CMOS | Input | 0.8 | 1.7 | - | - | - | COM2 Receive Data (UART) |
| 10 | COM1_ RXD | 3.3V CMOS | Input | 0.8 | 1.7 | - | - | - | COM1 Receive Data (UART) |

| Pin | Signal Name | Signal Type | Signal Direction | $V_{IL}$ Max (V) | $V_{IH}$ Min (V) | $V_{OL}$ Max (V) | $V_{OH}$ Min (V) | Drive (mA) | Description |
|---|---|---|---|---|---|---|---|---|---|
| 11 | COM5_ RXD/ COM2_ CTS | 3.3V CMOS | Input | 0.8 | 1.7 | - | - | - | This pin is internally multiplexed. COM5_RXD is the default.<br>COM5_RXD: COM5 Receive Data (UART).<br>COM2_CTS: COM2 Clear to Send Optional hardware flow control signal for the COM2 serial port. |
| 12 | COM1_ CTS | 3.3V CMOS | Input | 0.8 | 1.7 | - | - | - | COM1 Clear To Send Optional hardware flow control signals for the COM1 serial port. |
| 13 | COM4_ RXD | 3.3V CMOS | Input | 0.8 | 1.7 | - | - | - | COM4 Receive Data (UART) |
| 14 | COM3_ RXD | 3.3V CMOS | Input | 0.8 | 1.7 | - | - | - | COM3 Receive Data (UART) |
| 15 | STATUS_ GREEN | 3.3V CMOS | Output | - | - | 0.2 | 3.1 | 4 | Used to indicate the state of the receiver or provide error codes.<br>These outputs provide the same information as the Status Indicator LED (see Status LED).<br>STATUS_GREEN high and STATUS_RED low represents a **Green** LED.<br>STATUS_GREEN high and STATUS_RED high represents a **Yellow** LED.<br>STATUS_GREEN low and STATUS_RED high represents a **Red** LED. |
| 16 | STATUS_ RED | 3.3V CMOS | Output | - | - | 0.2 | 3.1 | 4 | |
| 17 | EVENT_ OUT1 | 3.3V CMOS | Output | - | - | 0.2 | 3.1 | 8 | EVENT1 (Mark1) Output Rising edge triggered.<br>Outputs a user-specified timing signal. Can be synchronized with PPS. Supports Variable Frequency Output function. |

| Pin | Signal Name | Signal Type | Signal Direction | $V_{IL}$ Max (V) | $V_{IH}$ Min (V) | $V_{OL}$ Max (V) | $V_{OH}$ Min (V) | Drive (mA) | Description |
|---|---|---|---|---|---|---|---|---|---|
| 18 | ME_RDY | 3.3V CMOS | Output | - | - | 0.2 | 3.1 | 4 | Receiver Ready (Active High)<br>Indicates to the host system that the receiver is ready to receive commands.<br>This signal is not asserted while booting or during a reset.<br>Internal 2.2 kΩ pull down. |
| 19 | COM4_ TXD | 3.3V CMOS | Output | - | - | 0.2 | 3.1 | 4 | COM4 Transmit Data (UART) |
| 20 | COM3_ TXD | 3.3V CMOS | Output | - | - | 0.2 | 3.1 | 4 | COM3 Transmit Data (UART) |
| 21 | ERROR | 3.3V CMOS | Output | - | - | 0.2 | 3.1 | 4 | Error Indicator (Active High)<br>Normally low. A high output on this pin indicates that the receiver is in an error state.<br>Internal 2.2 kΩ pull down. |
| 22 | POS_ VALID | 3.3V CMOS | Output | - | - | 0.2 | 3.1 | 4 | Position Valid Output (Active High)<br>A high output on this pin indicates that the receiver has computed a valid GNSS position.<br>Internal 10 kΩ pull down. |
| 23 | EVENT_ OUT3 | 3.3V CMOS | Output | - | - | 0.2 | 3.1 | 8 | EVENT3 (Mark3) Output Rising edge triggered.<br>Outputs a user-specified timing signal. Can be synchronized with PPS. Supports Variable Frequency Output function. |
| 24 | PPS | 3.3V CMOS | Output | - | - | 0.55 | 2.3 | 24 | Pulse Per Second output<br>This signal defaults to one pulse per second but may be altered across a wide range of frequencies using software commands. Edges can be synchronized to GNSS time reference. |

| Pin | Signal Name | Signal Type | Signal Direction | $V_{IL}$ Max (V) | $V_{IH}$ Min (V) | $V_{OL}$ Max (V) | $V_{OH}$ Min (V) | Drive (mA) | Description |
|---|---|---|---|---|---|---|---|---|---|
| 25 | EVENT_OUT4 | 3.3V CMOS | Output | - | - | 0.2 | 3.1 | 8 | EVENT4 (Mark4) Output<br>Rising edge triggered.<br>Outputs a user-specified timing signal. Can be synchronized with PPS. Supports Variable Frequency Output function. |
| 26 | EVENT_OUT2 | 3.3V CMOS | Output | - | - | 0.2 | 3.1 | 8 | EVENT2 (Mark2) Output<br>Rising edge triggered.<br>Outputs a user-specified timing signal. Can be synchronized with PPS. Supports Variable Frequency Output function. |
| 27 | GND | Ground | - | - | - | - | - | - | Ground reference |
| 28 | GND | Ground | - | - | - | - | - | - | Ground reference |
| 29 | EVENT_IN2 | 3.3V CMOS | Input | 0.8 | 1.7 | - | - | - | EVENT2 (Mark2) input<br>Rising or falling edge triggered. This input is used to provide a position or time data log based on an external trigger.<br>Internal 10 kΩ pull up. |
| 30 | EVENT_IN1 | 3.3V CMOS | Input | 0.8 | 1.7 | - | - | - | EVENT1 (Mark1) input<br>Rising or falling edge triggered. This input is used to provide a position or time data log based on an external trigger.<br>Internal 10 kΩ pull up. |
| 31 | EVENT_IN4 | 3.3V CMOS | Input | 0.8 | 1.7 | - | - | - | EVENT4 (Mark4) input<br>Rising or falling edge triggered. This input is used to provide a position or time data log based on an external trigger.<br>Internal 10 kΩ pull up. |

| Pin | Signal Name | Signal Type | Signal Direction | $V_{IL}$ Max (V) | $V_{IH}$ Min (V) | $V_{OL}$ Max (V) | $V_{OH}$ Min (V) | Drive (mA) | Description |
|---|---|---|---|---|---|---|---|---|---|
| 32 | EVENT_IN3 | 3.3V CMOS | Input | 0.8 | 1.7 | - | - | - | EVENT3 (Mark3) input<br>Rising or falling edge triggered. This input is used to provide a position or time data log based on an external trigger.<br>Internal 10 kΩ pull up. |
| 33 | GND | Ground | - | - | - | - | - | - | Ground reference |
| 34 | GND | Ground | - | - | - | - | - | - | Ground reference |
| 35 | CAN1_RXD | 3.3V CMOS | Input | 0.8 | 1.7 | - | - | - | CAN1 Receive Data<br>This is a CMOS-level signal, requiring an external CAN transceiver. |
| 36 | CAN0_TXD | 3.3V CMOS | Output | - | - | 0.2 | 3.1 | 4 | CAN0 Transmit Data<br>This is a CMOS-level signal, requiring an external CAN transceiver. |
| 37 | CAN1_TXD | 3.3V CMOS | Output | - | - | 0.2 | 3.1 | 4 | CAN1 Transmit Data<br>This is a CMOS-level signal, requiring an external CAN transceiver. |
| 38 | CAN0_RXD | 3.3V CMOS | Input | 0.8 | 1.7 | - | - | - | CAN0 Receive Data<br>This is a CMOS-level signal, requiring an external CAN transceiver. |
| 39 | I2C_SCA | 3.3V CMOS Open drain | I/O | 0.7 | 2.0 | 0.4 | 2.9 | - | I2C data line<br>Open drain signal with internal 2.2 kΩ pull up to 3.3 V |
| 40 | I2C_SCL | 3.3V CMOS Open drain | Output- | - | - | 0.4 | 3.1 | - | I2C clock line<br>Open drain signal with internal 2.2 kΩ pull up to 3.3 V |
| 41 | SPI_MOSI | 3.3V CMOS | Output | - | - | 0.2 | 3.1 | 4 | Serial Peripheral Interface Master Output/Slave Input |
| 42 | SPI_nCS | 3.3V CMOS | Output | - | - | 0.2 | 3.1 | 4 | Serial Peripheral Interface Chip Select<br>(Active Low) |

| Pin | Signal Name | Signal Type | Signal Direction | $V_{IL}$ Max (V) | $V_{IH}$ Min (V) | $V_{OL}$ Max (V) | $V_{OH}$ Min (V) | Drive (mA) | Description |
|---|---|---|---|---|---|---|---|---|---|
| 43 | SPI_ MISO | 3.3V CMOS | Input | 0.8 | 1.7 | - | - | - | Serial Peripheral Interface Master Input/Slave Output |
| 44 | SPI_ SCLK | 3.3V CMOS | Output | - | - | 0.2 | 3.1 | 4 | Serial Peripheral Interface Serial Clock |
| 45 | GND | Ground | - | - | - | - | - | - | Ground reference |
| 46 | GND | Ground | - | - | - | - | - | - | Ground reference |
| 47 | USB1_D- | Analog | I/O | - | - | - | - | - | USB1 signal.<br>This is one half of the USB1 differential pair. USB1_D+ and USB1_D- must be length-matched and routed as a 90 Ω differential pair. |
| 48 | USB0_D+ | Analog | I/O | - | - | - | - | - | USB0 signal.<br>This is one half of the USB0 differential pair. USB0_D+ and USB0_D- must be length-matched and routed as a 90 Ω differential pair. |
| 49 | USB1_D+ | Analog | I/O | | | | | | USB1 signal.<br>This is one half of the USB1 differential pair. USB1_D+ and USB1_D- must be length-matched and routed as a 90 Ω differential pair. |
| 50 | USB0_D- | Analog | I/O | | | | | | USB0 signal.<br>This is one half of the USB0 differential pair. USB0_D+ and USB0_D- must be length-matched and routed as a 90 Ω differential pair. |
| 51 | UID | 3.3V CMOS | Input | - | - | - | - | - | USB Port Mode Select.<br>Leave this pin floating to put USB0 into Device mode and USB1 into Host mode.<br>Tie this pin to GND to put USB0 into Host mode and USB1 into Device mode.<br>Internal 10 kΩ pull up |

| Pin | Signal Name | Signal Type | Signal Direction | $V_{IL}$ Max (V) | $V_{IH}$ Min (V) | $V_{OL}$ Max (V) | $V_{OH}$ Min (V) | Drive (mA) | Description |
|---|---|---|---|---|---|---|---|---|---|
| 52 | USB0_ VBUS | Power | I/O | 3.3 | 5.25 | - | - | - | When the USB port mode is set to Host, this pin is an output.<br>When the USB port mode is set to Device, this pin is an input.<br>Host or Device mode is set using the USB_Port_Mode_Select pin (Pin 51).<br>When an input, requires a voltage between 3.3 V and 5.25 V.<br>When an output, provides 5 V for hosted devices<br>USB0_VBUS is capable of providing up to 200 mA to a hosted USB device. Devices that require more than 200 mA must be powered separately. |
| 53 | nRESET_ IN | 3.3V CMOS | Input | 0.8 | 2.55 | - | - | - | Reset Input (Active Low)<br>Resets the OEM7700 receiver card. This pin must be held low for >20 µs while stable power is already applied, to reset the OEM7700 card.<br>It is recommended to hold the nRESET_IN pin low for a >150 ms when initially applying power to the card, giving the power supply time to stabilize before the card starts to boot.<br>Internal 10 kΩ pullup. |
| 54 | GND | Ground | - | - | - | - | - | - | Ground reference |
| 55 | ETH_ LINK_ ACT | 3.3V CMOS | Output | - | - | 0.2 | 3.1 | 8 | Ethernet Link and Activity LED indicator.<br>Polarity of the indicator signal is low. When there is an active link, the pin is low. When there is activity on the link, the pin outputs a blink signal.<br>**Caution**: Do not use ETH_BIAS to supply the LED. |

| Pin | Signal Name | Signal Type | Signal Direction | $V_{IL}$ Max (V) | $V_{IH}$ Min (V) | $V_{OL}$ Max (V) | $V_{OH}$ Min (V) | Drive (mA) | Description |
|---|---|---|---|---|---|---|---|---|---|
| 56 | ETH_ BIAS | Power | Output | - | - | - | - | - | DC Bias source for the Ethernet magnetics.<br>Do not use ETH_BIAS to supply any other circuitry. |
| 57 | ETH_ RX+ | Analog | Input | - | - | - | - | - | Ethernet Receive<br>One half of the Ethernet receive differential pair. ETH_RX+ and ETH_RX- must be routed as a 100 Ω differential pair. |
| 58 | ETH_TX+ | Analog | Output | - | - | - | - | - | Ethernet Transmit<br>One half of the Ethernet transmit differential pair. ETH_TX+ and ETH_TX- must be routed as a 100 Ω differential pair. |
| 59 | ETH_RX- | Analog | Input | - | - | - | - | - | Ethernet Receive<br>One half of the Ethernet receive differential pair. ETH_RX+ and ETH_RX- must be routed as a 100 Ω differential pair. |
| 60 | ETH_TX- | Analog | Output | - | - | - | - | - | Ethernet Transmit<br>One half of the Ethernet transmit differential pair. ETH_TX+ and ETH_TX- must be routed as a 100 Ω differential pair. |