APN-0046:
Configure CAN for SPAN®

Configure CAN for SPAN®

This application note provides general guidance on how to configure the Controller Area Network (CAN) interface for NovAtel's SPAN firmware.

SPAN Interface Configuration

Configuration of the CAN interface is through the USB/RS-232 interface via the CANCONFIG command, see *Page 3*. NovAtel SPAN systems can be configured to output GPS only or GPS/INS data. The following table shows the CAN port availability on NovAtel SPAN Systems:

System	CAN Ports
SPAN-CPT	CAN2
SPAN For OEMV	CAN1 CAN2
SPAN-SE and SPAN-MPPC	CAN1 CAN2

CANCONFIG Configure the CAN Interface for SPAN

Use the CANCONFIG command to configure the CAN interface for SPAN. All fields are mandatory (there are no optional fields).

Abbreviated ASCII Syntax: Message ID: 884

```
CANCONFIG port switch bit rate base tx mask source
```

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	header	-	-	This field contains the command name or the message header depending on whether the command is abbreviated ASCII, ASCII or binary, respectively	-	H	0
2	port	CAN1	1	Specify the CAN port	Enum	4	H
		CAN2	2				
3	switch	Disable	0	Enable/disable CAN configuration on the chosen port	Enum	4	H + 4
		Enable	1				
4	bit rate			CAN bit rate (Kbits/s), see Table 1: CAN Bit Rate (per second) *Page 3*	Enum	4	H + 8
5	base	0x0000 to 0xFFFF	0x0000 to 0xFFFF	Base address, see CAN Identifier Definition *Page 6*	Ulong	4	H + 12
6	tx mask	0x0000 to 0xFFFF	0x0000 to 0xFFFF	Transmit activation mask. Enables which messages are output, see Table 2: TX Mask, *Page 3*	Ulong	4	H + 16
7	source	INSGPS	0	CAN source from either the INS/GPS solution or the GPS only solution	Enum	4	H+20
		GPS	1				

Table 1: CAN Bit Rate (per second)

Binary	ASCII
0	10K
1	20K
2	50K
3	100K
4	125K
5	250K
6	500K
7	800K[1]
8	1M

Table 2: TX Mask

Set one or more bits to 1 in the mask to enable the corresponding message output.

Nibble #	Bit #	Mask	Message
N0	0	0x0001	0
	1	0x0002	1
	2	0x0004	Reserved
	3	0x0008	3
N1	4	0x0010	4
	5	0x0020	5
	6	0x0040	6
	7	0x0080	7
N2	8	0x0100	8
	9	0x0200	9
	10	0x0400	A

[1] If the CANCONFIG command is issued for 800 Kbps on a SPAN-SE or SPAN-MPPC, the hardware will run at 833 Kbps.

	11	0x0800	B
N3	12	0x1000	C
	13	0x2000	D
	14	0x4000	Reserved
	15	0x8000	F

CAN Message Format – Header

The following table displays the format of the CAN Header after the sync bits have been removed by the receiving CAN transceiver

Field	Description	Number Bits
Identifier	The ID for the CAN message	11 (base frame format) 29 (extended frame format)[2]
Data Bytes	The number of data bytes in the message	4

Basic CAN message format:

[CAN Header] [Data]

For example:
[11 bit Identifier (Base Address + Message Offset)] [4 bits (# of data bytes)] [8 bytes Data]
or…
[29 bit Identifier (Base Address + Message Offset)] [4 bits (# of data bytes)] [8 bytes Data]

CANCONFIGRATE Configure the CAN message output rate

CANCONFIGRATE command is available since 6.600 firmware version. Since 6.600 version, the CANCONFIGRATE command is added to configure the output rate of some of the CAN data messages when the CANCONFIG command mode is set to INSGPS; Any version prior than 6.600 has fixed output rate of 100Hz when in INSGPS mode; For both pre-6.600 and post-6.600 versions, the output rate of those messages when in GPS mode is locked at 20Hz.

[2] See Appendix 1 for more details on the different frame formats supported by NovAtel CAN.

Abbreviated ASCII Syntax **Message ID: 1816 (available in 6.600 and later firmware versions)**

```
CANCONFIGRATE  data rate
```

Field	Field Type	ASCII Value	Binary Value	Description	Binary Format	Binary Bytes	Binary Offset
1	Header	--	--	Log Header	--	H	0
2	Data Rate	50 100 200 IMURATE DEFAULT	1 2 3 4 0	Output rate of the selected CAN messages. Default is 100Hz.	ENUM	4	H

The following messages will be selectable when the receiver is operating in INSGPS mode:

- Message 0 (Mask 0x0001): Earth Rate Compensated Angular Rate
- Message 1 (Mask 0x0002): Gravity Compensated Acceleration
- Message 2 (Mask 0x0004): Acceleration including Gravity (Available in 6.600 and later firmware version)
- Message 3 (Mask 0x0008): Attitude
- Message 4 (Mask 0x0010): East Velocity
- Message 5 (Mask 0x0020): North Velocity
- Message 6 (Mask 0x0040): Up Velocity
- Message 7 (Mask 0x0080): Longitude
- Message 8 (Mask 0x0100): Latitude
- Message 9 (Mask 0x0200): Height

The rate-locked messages will be as follows (all at 1Hz):

- Message A (Mask 0x0400): Attitude Standard Deviations
- Message B (Mask 0x0800): Velocity Standard Deviations
- Message C (Mask 0x1000): Position Standard Deviations
- Message D (Mask 0x2000): Time Information

CAN Identifier Definition

A list of CAN identifiers begins on the next page and includes the activation bits, ID, bytes, description and default data rates for each. Note: All data types are in Little Endian Format (least significant byte first).

Message 0 Earth Rate Compensated Angular Rate

Mask: 0x0001 (BaseAddr + 0x0)

Rate: 100 Hz fixed for pre-6.600 firmware, selectable among 50 Hz, 100Hz, 200 Hz, IMURATE for 6.600 and later firmware.

Field #	Field Type	Data Description	Format	Binary Bytes	Binary Offset
1	Identifier	CAN Header Identifier (value = base address + 0x0)	Described in ***CAN Message Format - Header***		
2	Data Bytes	Number of Data Bytes to follow Nibble (value = 0x8)			
3	Time Stamp	Time Stamp, counter starts with last PPS (rolls over every second)	1 LSB = 0.1 ms	2	H
4	X Angular Rate	Angular rate, earth rate compensated	16-bit signed integer (default: 1 LSB = 0.01 degrees/s)	2	H + 2
5	Y Angular Rate	Angular rate, earth rate compensated	16-bit signed integer (default: 1 LSB = 0.01 degrees/s)	2	H + 4
6	Z Angular Rate	Angular rate, earth rate compensated	16-bit signed integer (1 LSB = 0.01 degrees/s)	2	H + 6

Message 1 Gravity Compensated Acceleration

Mask: 0x0002 (BaseAddr + 0x1)

Rate: 100 Hz fixed for pre-6.600 firmware, selectable among 50 Hz, 100Hz, 200 Hz, IMURATE for 6.600 and later firmware.

Field #	Field Type	Data Description	Format	Binary Bytes	Binary Offset
1	Identifier	CAN Header Identifier (value = base address + 0x1)	Described in ***CAN Message Format – Header***		
2	Data Bytes	Number of Data Bytes to follow Nibble (value = 0x8)			
3	Time Stamp	Time Stamp, counter starts with last PPS (rolls over every second)	1 LSB = 0.1 ms	2	H
4	X Acceleration	Acceleration with respect to body frame, gravity compensated	16-bit signed integer (1 LSB = 0.005 m/s/s)	2	H + 2
5	Y Acceleration	Acceleration with respect to body frame, gravity compensated	16-bit signed integer (1 LSB = 0.005 m/s/s)	2	H + 4
6	Z Acceleration	Acceleration with respect to body frame, gravity compensated	16-bit signed integer (1 LSB = 0.005 m/s/s)	2	H + 6

Message 2 Acceleration Including Gravity (Available in 6.600 and later firmware version)

Mask: 0x0004 (BaseAddr + 0x2)

Rate: selectable among 50 Hz, 100Hz, 200 Hz, IMURATE for 6.600 and later firmware; not available in pre-6.600 firmware.

Field #	Field Type	Data Description	Format	Binary Bytes	Binary Offset
1	Identifier	CAN Header Identifier (value = base address + 0x2)	Described in ***CAN Message Format - Header***		
2	Data Bytes	Number of Data Bytes to follow Nibble (value = 0x8)			
3	Time Stamp	Time Stamp, counter starts with last PPS (rolls over every second)	1 LSB = 0.1 ms	2	H
4	X Acceleration	X Acceleration from RAWIMU log	16-bit signed integer (1 LSB = 0.001 m/s/s)	2	H + 2
5	Y Acceleration	Y Acceleration from RAWIMU log	16-bit signed integer (1 LSB = 0.001 m/s/s)	2	H + 4
6	Z Acceleration	Z Acceleration from RAWIMU log	16-bit signed integer (1 LSB = 0.001 m/s/s)	2	H + 6

Message 3 Attitude

Mask: 0x0008 (BaseAddr + 0x3)

Rate: 100 Hz fixed for pre-6.600 firmware, selectable among 50 Hz, 100Hz, 200 Hz, IMURATE for 6.600 and later firmware.

Field #	Field Type	Data Description	Format	Binary Bytes	Binary Offset
1	Identifier	CAN Header Identifier (value = base address + 0x3)	Described in ***CAN Message Format - Header***		
2	Data Bytes	Number of Data Bytes to follow Nibble (value = 0x8)			
3	Time Stamp	Time Stamp, counter starts with last PPS (rolls over every second)	1 LSB = 0.1 ms	2	H
4	Roll	Roll from INSPVA log	16-bit signed integer (1 LSB = 0.01°)	2	H + 2
5	Pitch	Pitch from INSPVA log	16-bit signed integer (1 LSB = 0.01°)	2	H + 4
6	Yaw	Yaw from INSPVA log	16-bit un-signed integer (1 LSB = 0.01°)	2	H + 6

Message 4 East Velocity

Mask: 0x0010 (BaseAddr + 0x4)

Rate: 100 Hz fixed for pre-6.600 firmware, selectable among 50 Hz, 100Hz, 200 Hz, IMURATE for 6.600 and later firmware.

Field #	Field Type	Data Description	Format	Binary Bytes	Binary Offset
1	Identifier	CAN Header Identifier (value = base address + 0x4)	Described in ***CAN Message Format - Header***		
2	Data Bytes	Number of Data Bytes to follow Nibble (value = 0x6)			
3	Time Stamp	Time Stamp, counter starts with last PPS (rolls over every second)	1 LSB = 0.1 ms	2	H
4	East Velocity (INSGPS) Latitude Velocity (GPS)[3]	East/Latitude Velocity from INSPVA log	FLOAT (m/s)	4	H + 2

[3] Note: Velocity direction changes depending on whether INS/GPS or GPS is specified in the CANCONFIG command.

Message 5 North Velocity

Mask: 0x0020 (BaseAddr + 0x5)

Rate: 100 Hz fixed for pre-6.600 firmware, selectable among 50 Hz, 100Hz, 200 Hz, IMURATE for 6.600 and later firmware.

Field #	Field Type	Data Description	Format	Binary Bytes	Binary Offset
1	Identifier	CAN Header Identifier (value = base address + 0x5)	Described in ***CAN Message Format - Header***		
2	Data Bytes	Number of Data Bytes to follow Nibble (value = 0x6)			
3	Time Stamp	Time Stamp, counter starts with last PPS (rolls over every second)	1 LSB = 0.1 ms	2	H
4	North Velocity (INSGPS) Longitude Velocity (GPS)[4]	North/Longitude Velocity from INSPVA log	FLOAT (m/s)	4	H + 2

[4] Note: Velocity direction changes depending on whether INS/GPS or GPS is specified in the CANCONFIG command.

Message 6 Up Velocity

Mask: 0x0040 (BaseAddr + 0x6)

Rate: 100 Hz fixed for pre-6.600 firmware, selectable among 50 Hz, 100Hz, 200 Hz, IMURATE for 6.600 and later firmware.

Field #	Field Type	Data Description	Format	Binary Bytes	Binary Offset
1	Identifier	CAN Header Identifier (value = base address + 0x6)	Described in ***CAN Message Format - Header***		
2	Data Bytes	Number of Data Bytes to follow Nibble (value = 0x6)			
3	Time Stamp	Time Stamp, counter starts with last PPS (rolls over every second)	1 LSB = 0.1 ms	2	H
4	Up Velocity (INSGPS) Height Velocity (GPS)[5]	Up/Height Velocity from INSPVA log	FLOAT (m/s)	4	H + 2

[5] Note: Velocity direction changes depending on whether INS/GPS or GPS is specified in the CANCONFIG command.

Message 7 Longitude

Mask: 0x0080 (BaseAddr + 0x7)

Rate: 100 Hz fixed for pre-6.600 firmware, selectable among 50 Hz, 100Hz, 200 Hz, IMURATE for 6.600 and later firmware.

Field #	Field Type	Data Description	Format	Binary Bytes	Binary Offset
1	Identifier	CAN Header Identifier (value = base address + 0x7)	Described in ***CAN Message Format - Header***		
2	Data Bytes	Number of Data Bytes to follow Nibble (value = 0x8)			
3	Time Stamp	Time Stamp, counter starts with last PPS (rolls over every second)	1 LSB = 0.1 ms	2	H
4	Longitude	Longitude from INSPVAS log	6 byte signed integer (LSB = $180/2^{39}$ degrees)	6	H + 2

Message 8 Latitude

Mask: 0x0100 (BaseAddr + 0x8)

Rate: 100 Hz fixed for pre-6.600 firmware, selectable among 50 Hz, 100Hz, 200 Hz, IMURATE for 6.600 and later firmware.

Field #	Field Type	Data Description	Format	Binary Bytes	Binary Offset
1	Identifier	CAN Header Identifier (value = base address + 0x8)	Described in ***CAN Message Format - Header***		
2	Data Bytes	Number of Data Bytes to follow Nibble (value = 0x8)			
3	Time Stamp	Time Stamp, counter starts with last PPS (rolls over every second)	1 LSB = 0.1 ms	2	H
4	Latitude	Latitude from INSPVA log	6 byte signed integer (LSB = $180/2^{39}$ degrees)	6	H + 2

Message 9 Height

Mask: 0x0200 (BaseAddr + 0x9)

Rate: 100 Hz fixed for pre-6.600 firmware, selectable among 50 Hz, 100Hz, 200 Hz, IMURATE for 6.600 and later firmware.

Field #	Field Type	Data Description	Format	Binary Bytes	Binary Offset
1	Identifier	CAN Header Identifier (value = base address + 0x9)	Described in ***CAN Message Format - Header***		
2	Data Bytes	Number of Data Bytes to follow Nibble (value = 0x8)			
3	Time Stamp	Time Stamp, counter starts with last PPS (rolls over every second)	1 LSB = 0.1 ms	2	H
4	Height	Mean Sea Level Height from INSPVA log	FLOAT (meters)	4	H + 2

Message A Attitude Standard Deviations

Mask: 0x0400 (BaseAddr + 0xA)

Rate: 1 Hz

Field #	Field Type	Data Description	Format	Binary Bytes	Binary Offset
1	Identifier	CAN Header Identifier (value = base address + 0xA)	Described in ***CAN Message Format - Header***		
2	Data Bytes	Number of Data Bytes to follow Nibble (value = 0x8)			
3	Time Stamp	Time Stamp, counter starts with last PPS (rolls over every second)	1 LSB = 0.1 ms	2	H
4	Roll STD Dev	Roll Standard Deviation from INSCOV log	16-bit signed integer (LSB = 0.001 degrees)	2	H + 2
5	Pitch STD Dev	Roll Standard Deviation from INSCOV log	16-bit signed integer (LSB = 0.001 degrees)	2	H + 4
6	Yaw STD Dev	Roll Standard Deviation from INSCOV log	16-bit signed integer (LSB = 0.001 degrees)	2	H + 6

Message B Velocity Standard Deviation

Mask: 0x0800 (BaseAddr + 0xB)

Rate: 1 Hz

Field #	Field Type	Data Description	Format	Binary Bytes	Binary Offset
1	Identifier	CAN Header Identifier (value = base address + 0xB)	Described in ***CAN Message Format - Header***		
2	Data Bytes	Number of Data Bytes to follow Nibble (value = 0x8)			
3	Time Stamp	Time Stamp, counter starts with last PPS (rolls over every second)	1 LSB = 0.1 ms	2	H
4	North Vel STD Dev	North Velocity Standard Deviation from INSCOV log	16-bit signed integer (1 LSB = 0.001 m/s)	2	H + 2
5	East Vel STD Dev	East Velocity Standard Deviation from INSCOV log	16-bit signed integer (1 LSB = 0.001 m/s)	2	H + 4
6	Up Vel STD Dev	Up Velocity Standard Deviation from INSCOV log	16-bit signed integer (1 LSB = 0.001 m/s)	2	H + 6

Message C Position Standard Deviation

Mask: 0x1000 (BaseAddr + 0xC)

Rate: 1 Hz

Field #	Field Type	Data Description	Format	Binary Bytes	Binary Offset
1	Identifier	CAN Header Identifier (value = base address + 0xC)	Described in ***CAN Message Format - Header***		
2	Data Bytes	Number of Data Bytes to follow Nibble (value = 0x8)			
3	Time Stamp	Time Stamp, counter starts with last PPS (rolls over every second)	1 LSB = 0.1 ms	2	H
4	North STD Dev	North Standard Deviation of position	16-bit signed integer (1 LSB = 0.001 metres)	2	H + 2
5	East STD Dev	East Standard Deviation of position	16-bit signed integer (1 LSB = 0.001 metres)	2	H + 4
6	Height STD Dev	Height Standard Deviation of position	16-bit signed integer (1 LSB = 0.001 metres)	2	H + 6

Message D Time Information

Mask: 0x2000 (BaseAddr + 0xD)

Rate: 1 Hz

Field #	Field Type	Data Description	Format	Binary Bytes	Binary Offset
1	Identifier	CAN Header Identifier (value = base address + 0xD)	Described in ***CAN Message Format - Header***		
2	Data Bytes	Number of Data Bytes to follow Nibble (value = 0x8)			
3	Time Stamp	Time Stamp, counter starts with last PPS (rolls over every second)	1 LSB = 0.1 ms	2	H
4	UTC	Coordinated Universal Time (UTC) of the last PPS (seconds of the day)	3 byte integer (1 LSB = 1 s)	3	H + 2
5	Leap Seconds	UTC Leap Seconds	1 byte integer (1 LSB = 1 s)	1	H + 5
6	Differential Age	Differential age of GPS	1 byte integer (1 LSB = 1 s)	1	H + 6
7	Solution Age	Solution Age	1 byte integer (1 LSB = 1 s)	1	H + 7

Message E IMU Status (Available in 6.600 and later firmware version)

Mask: 0x4000 (BaseAddr + 0xE)

Rate: 1 Hz

Field #	Field Type	Data Description	Format	Binary Bytes	Binary Offset
1	Identifier	CAN Header Identifier (value = base address + 0xE)	Described in ***CAN Message Format - Header***		
2	Data Bytes	Number of Data Bytes to follow Nibble (value = 0x8)			
3	IMU Status	IMU status identical to the output from the RAWIMU log	4 byte unsigned integer	4	H
4	Reserved	Reserved field	4 byte unsigned integer	4	H+4

Message F System Status

Mask: 0x8000 (BaseAddr + 0xF)

Rate: 1 Hz

Field #	Field Type	Data Description	Format	Binary Bytes	Binary Offset
1	Identifier	CAN Header Identifier (value = base address + 0xF)	Described in ***CAN Message Format - Header***		
2	Data Bytes	Number of Data Bytes to follow Nibble (value = 0x8)			
3	INS Status	From *Inertial Solution Status* Table [6]	1 byte unsigned integer	1	H
4	SVs in View	Number of Satellites in View	1 byte unsigned integer	1	H + 1
5	Solution Status of OEMV	From *Solution Status* Table[6]	3 byte unsigned integer	3	H + 2
6	Position type of OEMV	From *Position Type* tableError: Reference source not found	3 byte unsigned integer	3	H + 5

[6] Refer to these tables in the *SPAN Technology for OEMV User Manual*, available on our website at http://www.novatel.com/assets/Documents/Manuals/om-20000104.pdf.

Appendix 1 CAN Bus Frame Format

Messages are transported on the CAN bus in "frames" and can be sent in two different formats. NovAtel CAN modules support both the "Base" and "Extended" frame formats. The main difference between formats is the length of the identifier: 11 bits for "base frame", 29 bits for "extended frame" format. The 29 bits include the 11-bit base identifier and an 18-bit extension.

Base Frame Format (CAN 2.0A)[7]:

Field #	Field Name	Length (bits)	Purpose
1	Start-of-frame	1	Denotes the start of frame transmission
2	Identifier	11	A (unique) identifier for the data which also represent the message priority
3	Remote transmission request (RTR)	1	Dominant (0) (see Remote Frame below)
4	Identifier extension bit (IDE)	1	Must be dominant (0) Optional
5	Reserved bit (r0)	1	Reserved bit (it must be set to dominant (0), but accepted as either dominant or recessive)
6	Data length code (DLC)*	4	Number of bytes of data (0-8 bytes)
7	Data field	0-8 bytes	Data to be transmitted (length dictated by DLC field)
8	CRC	15	Cyclic Redundancy Check
9	CRC delimiter	1	Must be recessive (1)
10	ACK slot	1	Transmitter sends recessive (1) and any receiver can assert a dominant (0)
11	ACK delimiter	1	Must be recessive (1)
12	End-of-frame (EOF)	7	Must be recessive (1)

[7] Reference Wikipedia: http://en.wikipedia.org/wiki/Controller_area_network .

Extended Frame Format (CAN 2.0B)[8]:

Field #	Field Name	Length (bits)	Purpose
1	Start-of-frame	1	Denotes the start of frame transmission
2	Identifier A	11	First part of the (unique) identifier for the data which also represent the message priority
3	Substitute remote request (SRR)	1	Must be recessive (1) Optional
4	Identifier extension bit (IDE)	1	Must be recessive (1) Optional
5	Identifier B	18	Second part of the (unique) identifier for the data which also represent the message priority
6	Remote transmission request (RTR)	1	Must be dominant (0)
7	Reserved bits (r0, r1)	2	Reserved bits (it must be set to dominant (0), but accepted as either dominant or recessive)
8	Data length code (DLC)*	4	Number of bytes of data (0-8 bytes)
9	Data field	0-8 bytes	Data to be transmitted (length dictated by DLC field)
10	CRC	15	Cyclic Redundancy Check
11	CRC delimiter	1	Must be recessive (1)
12	ACK slot	1	Transmitter sends recessive (1) and any receiver can assert a dominant (0)
13	ACK delimiter	1	Must be recessive (1)
14	End-of-frame (EOF)	7	Must be recessive (1)

[8] Reference Wikipedia: http://en.wikipedia.org/wiki/Controller_area_network

Appendix 2 Examples

EXAMPLE 1: Message 7 (INS/GPS position longitude from INSPVAS):

COMMAND: canconfig can2 enable 250k 110 80 insgps

The data rate is 250k, the base address is 0x110 and the mask is 0x0080.

OUTPUT: 1178001C19BFDBE7AEFF

1. The identifier is 117 (Base address of 0x0110 + message offset 0x07).

1178001C19BFDBE7AEFF

2. The number of data bytes to follow is 8

117***8***001C19BFDBE7AEFF

3. The time stamp is 001C.

1178***001C***19BFDBE7AEFF

It is in Little Endian format and must be flipped and converted to 0.1 ms as per the app note: 1C00 = 7168 x 10-1 ms = 716.8 ms. This time stamp rolls over every 1 second.

4. The longitude is 19BFDBE7AEFF .

1178001C***19BFDBE7AEFF***

The longitude is a 6 byte signed integer in Little Endian format. So that value is FFAEE7DBBF19 = -348297380071. Then according to the app note we must multiply by 180/(239): -348297380071 * 180/(239) = -114.038863854 degrees.

EXAMPLE 2: Message 0x0080 (INS/GPS position longitude) and 0x0100 (INS/GPS position latitude)

To get both the latitude and longitude messages coming out we must change the mask:

Longitude mask from above is 0x0080. Latitude mask from above is 0x0100. So the mask in the command will be 0x0080 + 0x0100 = 0x0180.

COMMAND: canconfig can2 enable 250k 110 180 insgps

OUTPUT: 1178F404F8A9DBE7AEFF

1188F4044A2A87592400

1. Notice that now we get both the longitude message 117 (Base address of 0x0110 + message offset 0x07) and latitude message 118 (Base address of 0x0110 + message offset of 0x08).

2. The time stamp for both messages is the same (since they both come from the same INSPVAS log).

1178***F404***F8A9DBE7AEFF

1188***F404***4A2A87592400

The time stamp is F404 = 126.8 ms.

3. The longitude is **F8A9DBE7AEFF**. The latitude is **4A2A875924000**.

Final Points

For further information regarding the topics covered within this application, please contact:

NovAtel Customer Service
1120 – 68 Ave. N.E.
Calgary, Alberta, Canada, T2E 8S5
Phone: 1-800-NOVATEL (in Canada or the U.S.) or +1-403-295-4500
Fax: 403-295-4501
E-mail: support@novatel.ca
Website: www.novatel.com