Rovins

Interface Library

iXblue

Revision History

Edition	Date	Comments
A	07/2019	Creation
B	02/2020	Broadcast protocol added, protocol name updated, status updated, control commands added

Copyright

Disclaimer

iXblue specifically disclaims all warranties, either expressed or implied, including but not limited to implied warranties about merchantability and fitness for a particular purpose with respect to this product and documentation. iXblue reserves the right to revise or to make changes or improvements to this product or documentation at any time without notify any person of such revision or improvements.
In no event shall iXblue be liable for any consequential or incidental damages, including but not limited to loss of business profits or any commercial damages, arising out of the use of this product.

Trademarks

Microsoft, MS-DOS and Windows are registered trademarks of Microsoft Corporation.
Intel and Pentium are registered trademarks and Celeron is a trademark of Intel Corporation.

Export Control

This product is subject to export control regulations. Please refer to the commercial offer or contact iXblue for details about the applicable export control regulations and restrictions.

Rovins Documentation

The following documents give all the information you need in order to understand and to use your product.

- **Rovins Technical Description** *(ref.: MU-ROVINSTD-AN-001*)
  - > Rovins and iXblue technology presentation
  - > Technical specifications
  - > Certification and qualification, life cycle
  - > Mechanical, electrical and communication interface description
- **Rovins Installation & Setup Guide** *(ref.: MU-ROVINSISG-AN-001)*
  - > Conventions
  - > Physical and electrical installation
  - > Connecting to the Web-Based Graphical User Interface
  - > Setup the Rovins
  - > Contacting iXblue support
- **Rovins Operation Guide** *(ref.: MU-ROVINSOG-AN-001)*
  - > Introduction to the Inertial Navigation System
  - > Start-up Phases
  - > Web-Based Graphical User Interface description
  - > Configuring the navigation parameters & managing the external information
  - > Viewing the system information
  - > Recording data
- **Rovins Interface Library** *(ref.: MU-ROVINSIL-AN-001)*
  - > NMEA frames
  - > Digital input and output protocols
  - > Pulses interfaces specification
  - > Control commands
- **Rovins Quick Start Guide**
  - > Pack content verifying
  - > Installing and connecting Rovins
  - > Configuring and operating Rovins
- **SEACON 12 PIN TI 1M Pigtail Cable - Product Description** (Ref.: MU-PDCABLES-AN-001)
  - > cable and pinout of the SEACON 12 pins Pigtail Cable
- **SEACON 19 PIN TI 1M Pigtail Cable - Product Description** (Ref.: MU-PDCABLES-AN-002)
  - > cable and pinout of the SEACON 19 pins Pigtail Cable
- **SEACON 26 PIN TI 1M Pigtail Cable - Product Description** (Ref.: MU-PDCABLES-AN-003)
  - > cable and pinout of the SEACON 26 pins Pigtail Cable
- **Subsea Products - Illustrated Part Catalog** *ref.: MU-SUBSEADP-AN-001*
  - > Detailed part list
  - > Alphanumerical Index
- **Application Note - INS+DVL Calibration** *(ref.: MU-DVLINS-AN-001)*
  - > Configuring the calibration
  - > Calibrating the DVL+INS

- **Application Note - Installation and Configuration of AHRS and INS for Seabed Mapping Measurements** (*ref.: MU-HEAVAPN-AN-001*)
  - Using heave compensation on seabed mapping
  - Effect of vessel transient movements

Manual Overview

This document is the Interface Library for Rovins. It must be read and understood prior to using the product.
The manufacturer shall in no case be held liable for any application or use that does not comply with the stipulations in this manual.

The Rovins Interface Library is divided into several parts:

- **Part 1: Conventions**
  This part gives details of conventions used in this document.
- **Part 2: List of the Data provided by Rovins**
  This part lists the data provided by Rovins: navigation data, their standard deviations and external sensor data.
- **Part 3: Input NMEA Frame Definition**
  This part gives the definition of the different input NMEA frames.
- **Part 4: Output NMEA Frame Definition**
  This part gives the definition of the different output NMEA frames.
- **Part 5: Status Description**
  This part gives a detailed description of Rovins algorithm and user status words. These status words act as built-in test and control tools to check for Rovins operation.
- **Part 6: Digital Protocols**
  This part gives definition of the different input and output protocols.
- **Part 7: Pulses Interfaces Specification**
  This part gives a detailed description of the different pulse inputs and output protocols available.
- **Part 8: Control Command**
  This part details all the configuration and monitoring commands which can be used during operation.

Abbreviations and Acronyms

ACC	Accelerometer
EGNOS	European Geo-stationary Navigation Overlay Service
Galileo	European Satellite Navigation System
GLONASS	Global Navigation Satellite System (operated by the Russian Space Forces)
GNSS	Global Navigation Satellite System
GPS	Global Positioning System (operated by the United States Department of Defense)
GST	GPS Pseudo-range Noise (PRN) statistics (The GST message is used to support Receiver Autonomous Integrity Monitoring)
GUI	Graphical User Interface
HRP	Heading, Roll, Pitch
IMU	Inertial Measurement Unit
INS	Inertial Navigation System
IP	Internet Protocol
LVA	Lever Arms
MU	Measurement Unit
PPS	Pulse Per Second
PROM	Programmable Read Only Memory
RAM	Random Access Memory (volatile)
RTK	Real Time Kinematic
SD	Standard Deviation
Sec lat	Secant latitude = 1 / cosine latitude
UTC	International acronym for Coordinated Universal Time (CUT)
UTM	Universal Transverse Mercator
VTG	Vector Track and speed over Ground
WGS-84	World Geodetic System 1984
ZDA	Time and Date (The ZDA message identifies UTC time, day, month, and year, local zone number, and local zone minutes)
ZUPT	Zero velocity UPdaTe

Table of Contents

1 Conventions

For position data conventions, refer to Rovins Installation & Setup Guide.

1.1 Time Management

Rovins typical time management is to synchronize with the GNSS time.
If GNSS is not at all available or UTC not configured, Rovins starts per default on the 1st January 2006 and relies on its internal clock to maintain time.
Otherwise the date is maintained on the GNSS time as soon as the GNSS time is received. If GNSS is received and then lost for a period of time, Rovins will maintain time based on its internal clock and will jump to the GNSS time as soon as this one is recovered.
If Rovins is synchronized with the GNSS time, the data time tags in the output telegrams are the UTC; otherwise they are Rovins one (i.e., the time since power-up of the system).
To achieve this, you need to associate the UTC input to the GNSS input (refer to the Rovins Installation & Setup Guide).

Note: To improve time synchronization accuracy it is recommended to input into Rovins a PPS pulse from the GNSS.

1.2 Altitude Management

A dedicated mechanism is used to manage the altitude in the navigation algorithm:

- An altitude loop is in charge of estimating the current altitude based on measured accelerations, current attitude and external altitude control value
- This altitude estimation is provided to the Kalman filter together with altitude sensor input to estimate internal sensor biases and validate / reject the external altitude sensor input to determine altitude control value

The altitude loop computation is done by projecting accelerometers to local vertical axis using gyros, to determine the vertical acceleration which is used to compute vertical speed and altitude by first and double integration with respect to sampling time. This altitude is merged with altitude control value to provide best estimation of current altitude.
When no external altitude sensor is present for some time, the altitude loop switches to a “degraded mode” where the altitude output is stabilized using last altitude control value and associated standard deviation. After 600s without any altitude sensor, the loop switches into “safe mode” where the altitude output is set to last altitude output when the senor was lost. In this mode, the altitude will jump to the external sensor value when recovered.

Depending on selected mode, the algorithm will use the appropriate sensor to stabilize the altitude loop:

- In **Depth** mode, the altitude loop and Kalman filter will use depth sensor when available and USBL Z information otherwise. DVL vertical speed is also used in this case to stabilize the vertical speed in the Kalman filter.
- In **GNSS** mode, the altitude loop will use the most precise information coming from GNSS1, GNSS2 or manual GNSS altitude input. If a vertical speed is present (DVL, car model), it will also be used in the Kalman filter
- In **Hydro** mode, the altitude loop and the Kalman filter will use the most precise altitude information coming from GNSS1 or GNSS2 only in RTK mode. If no RTK is available, the altitude loop switches to “degraded mode”
- In **Stabilization** mode, the altitude loop will maintain the altitude around the initial altitude entered in initial position at boot time, or current altitude when this mode is selected during the mission

The altitude will be referenced either to Mean Sea Level (Geoïd), or to WGS84 Ellipsoid, depending on what was selected in the output protocol altitude reference parameter. The offset between Geoïd and Ellipsoid is set to 0 at boot time and is then updated with information coming from the GNSS. If no GNSS is ever received, this will stay to 0 (no Geoïd map embedded in the Rovins).

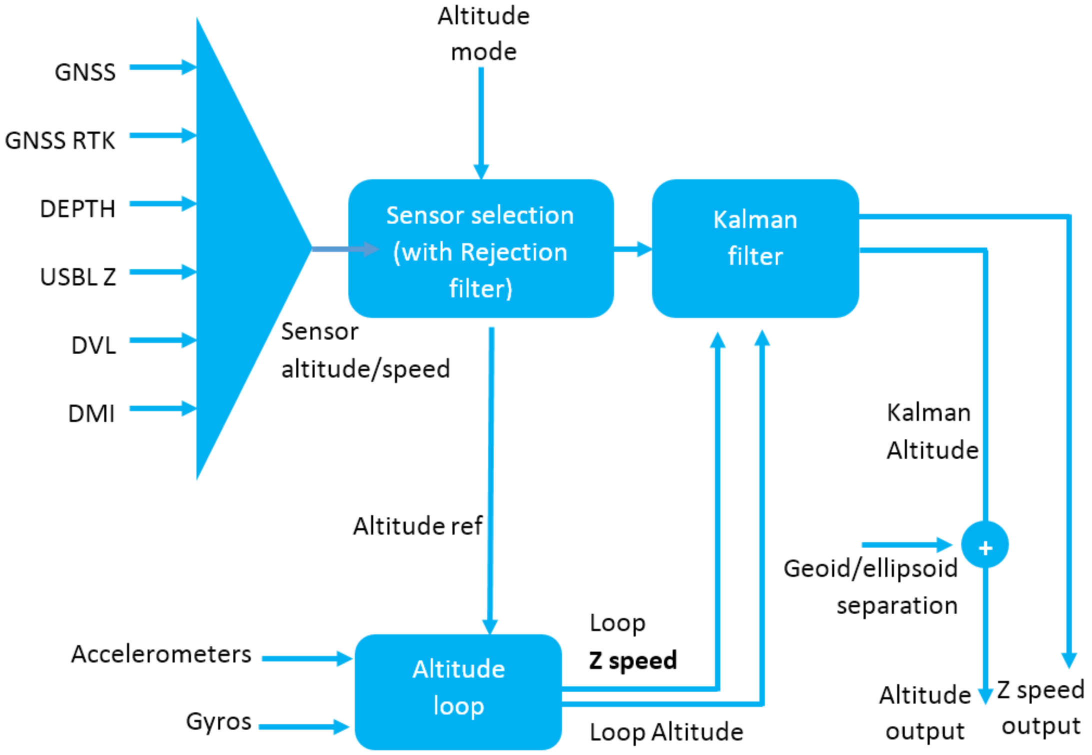


Figure 1 - Altitude loop diagram

1.3 Reference Frame Notations

The following notations will be found in the protocol descriptions and are explained hereafter:

- $\underline{X}_{V1}$, $\underline{X}_{V2}$, $\underline{X}_{V3}$: Subsea vehicle frame: $\underline{X}_{V1}$ (forward), $\underline{X}_{V2}$ (left), $\underline{X}_{V3}$ (up).
- $\underline{X}_{VH1}$, $\underline{X}_{VH2}$, $\underline{X}_{VH3}$: Subsea vehicle horizontal frame; Subsea vehicle frame compensated from roll and pitch.
- $\underline{X}_1$, $\underline{X}_2$, $\underline{X}_3$: Rovins body frame; $\underline{X}_1$ (forward), $\underline{X}_2$ (left), X_3 (up). Refer to the Rovins Installation & Setup Guide for convention description.
- $\underline{X}_{1IMU}$, $\underline{X}_{2IMU}$, $\underline{X}_{3IMU}$: IMU reference frame or internal sensor bloc frame; $\underline{X}_{1IMU}$ (forward), $\underline{X}_{2IMU}$ (left), $\underline{X}_{3IMU}$ (up).
- $\underline{X}_{North}$, $\underline{X}_{East}$, $\underline{X}_{Down}$: local geographical frame.
- $\underline{X}_{S1}$, $\underline{X}_{S2}$, $\underline{X}_{S3}$: external sensor body frame: $\underline{X}_{S1}$ (forward), $\underline{X}_{S2}$ (left), $\underline{X}_{S3}$ up). Sign convention is described in each protocol.
- **Rotation rates convention**: "$\underline{X}_{VI}$ (I=1, 2, 3) rotation rate" is the rotation rate in the inertial frame. The data is not compensated for earth rotation (15.04°/h) or craft rate. The rotation rate is positive when the rotation vector is pointing in $\underline{X}_{VI}$ (I=1, 2, 3) direction. Assuming that a right-handed corkscrew is oriented along the considered axis and that the screw is rotated to move in the positive direction, the positive direction of rotation is the same as the direction of rotation of the screw.. "
- **Acceleration convention**: "$\underline{X}_{VI}$ (I=1, 2, 3) acceleration" is positive when the acceleration vector is pointing in $\underline{X}_{VI}$ (I=1, 2, 3) direction. Accelerations are compensated from gravity unless specified in the protocol (i.e.: CONTROL protocol).

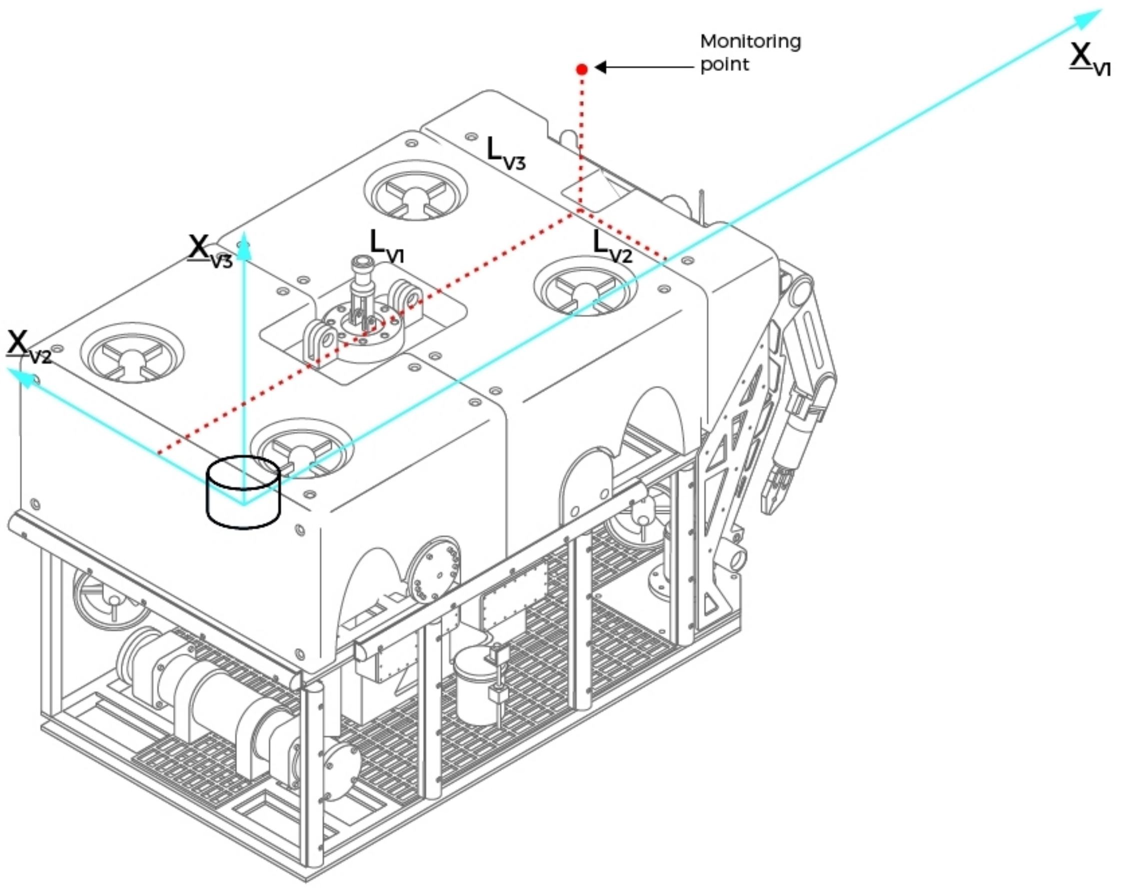


Figure 2 - Vehicle, System and Sensor reference frame

1.4 GNSS Quality Indicator Management at Input and Output of Rovins

For INPUT data:

When Rovins receives GGA only (no GST), Rovins will associate the standard deviation to the GNSS position according to each Q factor as per the following table. If GST is available, GST standard deviation will be used instead.

Q factor in GGA input telegram	Message in iXrepeater	Rovins corresponding SD attributed to GNSS position fix (m)
4	RTK	= 0.1
5	Float RTK	= 0.3
2	Differential	= 3
3	Military	=10
1	Natural	= 10
0 or 6	N/A	0 or > 6

For OUTPUT data:

Rovins does not copy the quality indicator received on GGA input to GGA output.

It estimates the quality factor corresponding to the standard deviation associated with the Rovins position data.

When GPS like telegram is output from Rovins the following correspondence table is applied.

INS calculated position SD (m)	Q factor in GPS like output protocol	Mode indicator
< 0.1	4	D
< 0.3	5	D
< 3	2	D
< 10	1	A
≥ 10	6	E

During initial alignment (at power-up or after a system restart) the quality factor is fixed to 6.

The Rovins SD is the horizontal dilution of precision (HDOP) calculated from Rovins SDLat and SDLong on position:

$$SD = HDOP = \sqrt{SDLat^2 + SDLong^2}$$

The mode indicator provides status information about the operation of the source device (such as positioning systems, velocity sensors, etc.) generating the sentence, and the validity of data being provided.

The possible indications are as follows:
A = Autonomous mode
D = Differential mode
E = Estimated (dead reckoning) mode
M = Manual input mode
S = Simulator mode
N = Data not valid
The mode indicator field should not be a null field.

1.5 Pressure to Depth Conversion Formula

Depth is calculated by using the following formula from the Unesco Technical Papers in *Marine Science n°44, Algorithms for computation of fundamental properties in seawater*.

$$Depth = \frac{(((-1.82E-15 * P + 2.279E-10) * P - 2.2512E-5) * P + 9.72659) * P}{Gravity}$$

Where pressure P is pressure compensated from atmospheric pressure (10.1325 dbar). It is calculated in decibars as follows:

$$P(dB) = 0.6894757 * x.x(PSI) - 10.1325$$

and the gravity (in m/s2) is calculated as follows:

$$X = \left[\sin\left(\frac{Latitude}{57.29578}\right)\right]^2$$

Where P is expressed in dbar and Latitude in degrees. PSI is the salinity unit. Latitude used in the formula is the one computed by Rovins.

1.6 Sound Velocity Conversion Formula

When only conductivity, pressure and temperature are available, conductivity ratio R needs to be converted to salinity to compute sound velocity using salinity and temperature (this for the Chen and Millero equation which expects pressure, temperature and salinity as inputs). Applied function are defined on pages 6, 7 and 8 of Unesco technical papers in marine science n° 44 - Algorithms for computation of fundamental properties in seawater, hereafter written:

$$S = a_0 + a_1 R_t^{1/2} + a_2 R_t + a_3 R_t^{3/2} + a_4 R_t^2 + a_5 R_t^{5/2} + \Delta S$$

Where :

$$\Delta S = \frac{(t-15)(b_0 + b_1 R_t^{1/2} + b_2 R_t + b_3 R_t^{3/2} + b_4 R_t^2 + b_5 R_t^{5/2})}{1 + 0.0162(t-15)}$$

$$R_t = \frac{R}{R_p . r_t} \qquad r_t = c_0 + c_1 t + c_2 t^2 + c_3 t^3 + c_4 t^4 \qquad R_p = \frac{p(e_1 + e_2 p + e_3 p^2)}{1 + d_1 t + d_2 t^2 + (d_3 + d_4 t)R} + 1$$

$$R = \frac{C(S,t,p)}{C(35,15,0)}$$ with C(35,15,0)= 4.29140 S/m (see AD2)

S = salinity in Practical Salinity Units (psu)

Table 1 - Table of Coefficients

Coefficients	Numerical values	Coefficients	Numerical values
a0	+ 0.0080	b0	+0.0005
a1	- 0.1692	b1	-0.0056
a2	+ 25.3851	b2	-0.0066
a3	+14.0941	b3	-0.0375
a4	- 7.0261	b4	+0.0636
a5	+2.7081	b5	-0.0144
c0	+0.6766097	d1	+3.426E-2
c1	+2.00564E-2	d2	+4.464E-4
c2	+1.104259E-4	d3	+4.215E-1
c3	-6.9698E-7	d4	-3.107E-3
c4	+1.0031E-9	e1	+2.070E-5
e2	-6.370E-10	e3	+ 3.989E-15

The speed of sound in seawater is then computed respect to Chen and Millero equation.

The UNESCO equation Chen and Millero is hereafter written:

c(S,T,P) =	Cw(T,P) + A(T,P)S + B(T,P)S3/2 + D(T,P)S2
Cw(T,P) =	(C00 + C01T + C02T2 + C03T3 + C04T4 + C05T5) +
	(C10 + C11T + C12T2 + C13T3 + C14T4)P +
	(C20 +C21T +C22T2 + C23T3 + C24T4)P2 +
	(C30 + C31T + C32T2)P3
A(T,P) =	(A00 + A01T + A02T2 + A03T3 + A04T4) +
	(A10 + A11T + A12T2 + A13T3 + A14T4)P +
	(A20 + A21T + A22T2 + A23T3)P2 +
	(A30 + A31T + A32T2)P3
B(T,P) =	B00 + B01T + (B10 + B11T)P
D(T,P) =	D00 + D10P

Table 2 - Table of coefficients

Coefficients	Numerical values	Coefficients	Numerical values
C00	+1402.388	A02	7.164E-5
C01	+5.03711	A03	2.006E-6
C02	-5.80852E-2	A04	-3.21E-8
C03	3.3420E-4	A10	9.4742E-5
C04	-1.47800E-6	A11	-1.2580E-5
C05	3.1464E-9	A12	-6.4885E-8
C10	0.153563	A13	1.0507E-8
C11	6.8982E-4	A14	-2.0122E-10
C12	-8.1788E-6	A20	-3.9064E-7
C13	1.3621E-7	A21	9.1041E-9
C14	-6.1185E-10	A22	-1.6002E-10
C20	3.1260E-5	A23	7.988E-12
C21	-1.7107E-6	A30	1.100E-10
C22	2.5974E-8	A31	6.649E-12
23	-2.5335E-10	A32	-3.389E-13

Coefficients	Numerical values	Coefficients	Numerical values
C24	1.0405E-12	B00	-1.922E-2
C30	-9.7729E-9	B01	-4.42E-5
C31	3.8504E-10	B10	7.3637E-5
C32	-2.3643E-12	B11	1.7945E-7
A00	1.389	D00	1.727E-3
A01	-1.262E-2	D10	-7.9836E-6

Some values are provided for checking the correct use of above equations by Rovins.
P = 10000 dbars ; T = 40 °C; R = 1.888091 --> SVEL = 1731.9957 m/s
P = 0 dbars ; T = 15 °C; R = 1.000000 --> SVEL = 1506.6633 m/s
P = 2000 dbars ; T = 20 °C; R = 1.200000 --> SVEL = 1557.2327 m/s
P = 1500 dbars ; T = 5 °C; R = 0.650000 --> SVEL = 1486.4762 m/s
T : temperature value(input field ttt.tttt)
P : pressure value (input field pppp.ppp)
R : conductivity ratio value (input field cc.ccccc)
SVEL is the value of the Sound Velocity computed and used by Rovins.

2 List of the Data Provided by Rovins

Table 3 - Navigation data provided by Rovins

NAVIGATION DATA				
Position Data	**Unit**	**Min value**	**Max value**	**Convention**
Latitude	Degree	-90	90	positive in northern hemisphere
Longitude		0	360	increasing eastwards
Altitude	Meter	NA	4,000	Altitude above sea level (Geoïd) or Ellipsoid WGS84
UTM North		0	10,000,000	WGS84
UTM East		0	1,000,000	
UTM zone	Integer	1	60	
UTM band	Letter	C	X	
Speed data	**Unit**	**Min value**	**Max value**	**Convention used**
Speed north	Meter/second	NA	41.667	Note: 41.667 m/s ~150 km/h~81 knot
Speed west				
Speed up				
Heading and attitude data	**Unit**	**Min value**	**Max value**	**Convention used**
Heading	Degree	0	360	To switch from Subsea vehicle to navigation frame, the order of rotations is roll then pitch then heading
Roll		-180	180	
Pitch		-90	90	
Rotation and acceleration data	**Unit**	**Min value**	**Max value**	Convention used
X1 rotation rate	Degree/second	-180 -750	180 750	Resolution is limited to 0.001°/s (3.6°/h) to comply with export regulations.
X2 rotation rate				
X3 rotation rate				
X1 linear acceleration	m/s²	-180 -150	180 -150	Resolution is limited to 9.81 10^{-3} m/s² (1mg) to comply with export regulations.
X2 linear acceleration				
X3 linear acceleration				
Motion sensing data	**Unit**	**Min value**	**Max value**	**Convention**
Heave	Meter	NA	NA	Heave measured along local vertical
Surge				
Sway				
Time	**Unit**	**Min value**	**Max value**	**Convention**
Time	hh/mm/ss	0	23/59/59.99	

Table 4 - Standard deviation data provided by Rovins

STANDARD DEVIATION DATA	
Position standard deviation	Unit
Latitude standard deviation	meter
Longitude standard deviation	
Altitude standard deviation	
Speed standard deviation	Unit
North speed standard deviation	meter/second
West speed standard deviation	
Up speed standard deviation	
Heading and attitude standard deviation	Unit
Heading standard deviation	degree
Roll standard deviation	
Pitch standard deviation	

Table 5 - External sensor data provided by Rovins

EXTERNAL SENSOR DATA		
GPS1 Data	**Unit**	**Convention**
Latitude	degree	positive in northern hemisphere
Longitude	degree	positive eastwards
Altitude	meter	Altitude above sea level
Last update	hour/minute/second	
GPS2 Data	**Unit**	**Convention**
Latitude	degree	positive in northern hemisphere
Longitude	degree	positive eastwards
Altitude	meter	Altitude above sea level
Last update	hour/minute/second	

Speed bottom track data	Unit	Convention used
Speed X	meter/second	Speed in Log Sensor frame X, Y, Z (body or beam)
Speed Y		
Speed Z		
Altitude	meter	
Sound velocity	meter/second	Sound velocity calculated by the DVL
Last update	hour/minute/second	
Speed water track data	**Unit**	**Convention used**
Speed X	meter/second	Speed with respect to water layer in Log Sensor frame X, Y, Z (body or beam)
Speed Y		
Speed Z		
Sound velocity sensor data	**Unit**	**Convention used**
Sound velocity	Meter/second	From external CTD or SVP sensor
Depth sensor data	Unit	Convention used
Depth	Meter	Depth from local sea level
USBL data	**Unit**	**Convention**
Latitude	degree	
Longitude	degree	
Altitude	meter	
Latency	second	Latency of USBL data (>0)
Last update	hour/minute/second	
LBL data	**Unit**	**Convention**
Beacon latitude	degree	
Beacon longitude	degree	
Beacon altitude	meter	
range	Meter	
Last update	hour/minute/second	
EM Log data	**Unit**	**Convention**
Speed module	Meters/second	

Speed of North current	Meters/second	
Speed of East current	Meters/second	
Last update	hour/minute/second	
CTD data	**Unit**	**Convention**
Sound speed	Meters/second	

3 Input NMEA Frame Definition

This section described all NMEA 0183 frames being common in several inputs protocols. Refer to the chapter 1 to know exactly which frames are decoded by the NMEA compatible protocols.

3.1 Standard NMEA Telegrams

3.1.1 $--GGA Frame

$--GGA,hhmmss.ss,llmm.mm,a, LLLmm.mm,b,q,ss,y.y,x.x,M,g.g,M,a.a,zzzz*hh<CR><LF>		
hhmmss.ss	UTC of position (**see chapter 1.1**)	(USED)
llmm.mm	Latitude in degrees (ll) and in minutes (mm.mm)	(USED)
a	Hemisphere N: North S: South	(USED)
LLLmm.mm	Longitude in degrees (LLL) and in minutes (mm.mm)	(USED)
b	Longitude sign E: East W: West	(USED)
q	GNSS quality indicator (**see chapter 1.4**) 0 and ≥ 6 fix invalid 1=GPS SPS Mode, fix valid 2=Differential GPS, SPS Mode, fix valid 3= GPS PPS Mode, fix valid 4= RTK. Satellite system used in RTK mode with fixed integers 5= Float RTK. Satellite system used in RTK mode with floating integers	(USED)
ss	Number of satellites. (**Note 1**)	(USED)
y.y	Horizontal dilution of precision. (**Note 1**)	(USED)
x.x	Antenna altitude	(USED)
M	Units of antenna altitude (meters)	(USED)
g.g	Geoidal separation.	(USED)
M	Units of geoidal separation (meters)	(NOT USED)
a.a	Age of differential GNSS data.	(NOT USED)
zzzz	Differential reference station ID.	(NOT USED)
hh	NMEA checksum	

Note 1: Data read and transmitted to certain output telegrams. The data is not used by the algorithm.

3.1.2 $--GLL FRAME

$--GLL,llmm.mm,a,LLLmm.mm,b,hhmmss.ss,S,M*hh<CR><LF>		
llmm.mm	Latitude in degrees (ll) and in minutes (mm.mm)	(USED)
a	ASCII 'N' for North, ASCII 'S' for South	(USED)
LLLmm.mm	Longitude in degrees (LLL) and in minutes (mm.mm)	(USED)
b	ASCII 'E' for East, ASCII 'W' for West	(USED)
hhmmss.ss	UTC time of position (**see chapter 1.1**)	(USED)
S	Status (**see chapter 1.4**): ASCII 'A' = data valid ASCII 'V' = data invalid Warning: shall not be a blank field	(USED)
M	Positioning system Mode indicator (**see chapter 1.4**): ASCII 'A' = Autonomous mode ASCII 'D' = Differential mode ASCII 'E' = Estimated (dead reckoning) mode ASCII 'M' = Manual input mode ASCII 'S' = Simulator N= Data not valid Warning: shall not be a blank field	(USED)
hh	NMEA checksum	

3.1.3 $--GST FRAME

$--GST,hhmmss.ss,x.x,. x.x,. x.x,. x.x,. x.x,. x.x,. x.x,*hh<CR><LF>		
hhmmss.ss	UTC time of the GGA fix associated with this sentence	(USED)
x.x	RMS value of the standard deviation on pseudo-ranges	(NOT USED)
x.x	Standard deviation of semi-major axis of error ellipse	(NOT USED)
x.x	Standard deviation of semi-minor axis of error ellipse	(NOT USED)
x.x	Orientation of semi-major axis of error ellipse	(NOT USED)
x.x	Standard deviation of latitude error, in meters	(USED)
x.x	Standard deviation of longitude error, in meters	(USED)
x.x	Standard deviation of altitude error, in meters	(USED)
hh	NMEA checksum	

3.1.4 $--RMC FRAME

$--RMC,hhmmss.ss,S,llmm.mm,a,LLLmm.mm,b,x.x,x.x,ddmmyy,x.x,S,M*hh<CR><LF>		
S	Status : ASCII 'A' = data valid ASCII 'V' = data invalid Warning: shall not be a blank field	(USED)
llmm.mm	Latitude in degrees (ll) and in minutes (mm.mm)	(USED)
a	ASCII 'N' for North, ASCII 'S' for South	(USED)
LLLmm.mm	Longitude in degrees (LLL) and in minutes (mm.mm)	(USED)
b	ASCII 'E' for East, ASCII 'W' for West	(USED)
x.x	Speed over ground in knots (**Note 1**)	(USED)
x.x	Course over ground in degrees	(NOT USED)
ddmmyy	Date	(NOT USED)
x.x	Magnetic variation	(NOT USED)
S	Mode indicator : ASCII 'E' = Easterly variation substracts from True course ASCII 'W' = Westerly variation adds to true course	(NOT USED)
M	Positioning system Mode indicator, used if present (**see chapter 1.4**) ASCII 'A' = Autonomous mode ASCII 'D' = Differential mode ASCII 'E' = Estimated (dead reckoning) mode ASCII 'M' = Manual input mode ASCII 'S' = Simulator N= Data not valid	(USED)
hh	NMEA checksum	

Note 1:Speed in RMC telegram is only used in INS algorithm.

3.1.5 $--VBW Frame

$--VBW,x.x,x.x,A,x.x,x.x,A,x.x,A,x.x,A*hh <CR><LF>		
x.x	Longitudinal (XV1) water speed, in knots, '-' = astern.	(USED)
x.x	Transverse (XV2) water speed, in knots, '-' = port.	(USED)
A	Status of water speed, A=Data valid. V= data invalid	(USED)
x.x	Longitudinal (XV1) ground speed, in knots, '-' = astern.	(USED)
x.x	Transverse (XV2) ground speed, in knots, '-' = port.	(USED)
A	Status of ground speed: A = data valid V = data invalid	(USED)
x.x	Stern transverse water speed in knots.	(NOT USED)
A	Status of stern water speed.	(NOT USED)
x.x	Stern transverse ground speed in knots.	(NOT USED)
A	Status of stern ground speed.	(NOT USED)
hh	NMEA checksum	

3.1.6 $--VHW Frame

$--VHW, x.x ,T, x.x ,M,x.x,N,x.x,K*hh<CR><LF>		
x.x	Heading, degrees true	(USED)
T	True	(USED)
x.x	Heading, degrees magnetic	(NOT USED)
M	Magnetic	(NOT USED)
x.x	Speed in knots	(USED)
N	Knots	(USED)
x.x	Speed in km/h	(NOT USED)
K	Kilometers/hour	(NOT USED)
hh	NMEA checksum	

3.1.7 $--ZDA Frame

$--ZDA,hhmmss.ss,,,,, *hh <CR><LF>		
hhmmss.ss	UTC of the last PPS	(USED)
dd	UTC day	(USED)
mm	UTC month	(USED)
yyyy	UTC year	(USED)
hh	Local zone hour (**Note 1**)	(USED)
mm	Local zone minutes (**Note 1**)	(USED)
hh	NMEA checksum	

Note 1: Data read and transmitted to certain output telegrams (i.e: GPS LIKE). The data is not used by the algorithm.

3.2 Non-Standard NMEA or ASCII Telegrams

3.2.1 UTC Frame

UTC yy.mm.dd hh:mm:ss ab <CR><LF>	
UTC	fixed text header
yy.mm.dd	year month and date (USED)
hh:mm:ss	UTC time not GNSS time. (USED)
a	Integer number representing the position-fix type: (**Note 1**) (USED) 1 = time only 2 = 1D & time 3 = currently unused 4 = 2D & time 5 = 3D & time ? = no time
b	Number of GNSS satellites being tracked. (NOT USED) Note – If the receiver is not tracking satellites, the time tag is based on the receiver clock. In this case, a and b are represented by "??". The time readings from the receiver clock are less accurate than time readings determined from the satellite signals.

Note 1: If a=?, telegram is invalid. if a=1, 2, 3, 4, or 5, telegram is valid.

4 Output NMEA Frame Definition

4.1 Standard Output NMEA Telegrams

4.1.1 $GPGGA Frame

$GPGGA,hhmmss.ss,LLII.IIIIIII,a,LLLII.IIIIIII,a,x,xx,x.xxx,x.xxx,M,x.xxx,M,x.xxx,xxxx*hh<CR><LF>		
hhmmss.ss	UTC time of position	**Note 2**
LLII.IIIIIII	Latitude in degrees (LL) and in minutes (II.IIIIIII)	**Note 2**
a	'N' for Northern hemisphere, 'S' for Southern hemisphere	**Note 2**
LLLII.IIIIIII	Longitude in deg (LLL) and in minutes (II.IIIIIII)	**Note 2**
a	'E' for East, 'W' for West	**Note 2**
x	GNSS quality indicator	**see chapter 1.4**
xx	Number of satellites in use	**Note 1**
x.xxx	Horizontal dilution of precision (HDOP)	**Note 2**
x.xxx	Antenna altitude above mean sea level (geoid) (meters)	**Note 2**
M	Unit of antenna altitude (fixed character = 'M' for meters)	
x.xxx	Geoidal separation	**Note 1**
M	Unit of Geoidal separation (fixed character = 'M' for meters)	
x.xxx	Age of the differential GNSS data	**Note 1**
xxxx	Differential reference station ID	**Note 1**
hh	NMEA checksum	

Note 1: Copy of last GNSS values received. When no GNSS has been received since power-up, these fields are null except for number of satellites in use set to 3 by default.
Note 2: Rovins calculated data. Magnetic course field is set to Rovins true course.

4.1.2 $GPGLL FRAME

$GPGLL,LLII.IIIIIIII,a,LLLII.IIIIIIII,a,hhmmss.ss,a,m*hh<CR><LF>		
LLII.IIIIIIII	Latitude in degrees (LL) and in minutes (II.IIIIIIII)	**Note 1**
a	‘N’ for Northern hemisphere, ‘S’ for Southern hemisphere	**Note 1**
LLLII.IIIIIIII	Longitude in deg (LLL) and in minutes (II.IIIIIIII)	**Note 1**
a	‘E’ for East, ‘W’ for West	**Note 1**
hhmmss.ss	UTC time of position	**Note 1**
a	Status ‘A’ for Data Valid, ‘V’ for Data Invalid	**Note 2**
m	Mode indicator ‘A’ or ‘D’ or ‘E'	**see chapter 1.4**
hh	NMEA checksum	**Note 1**

Note 1: Rovins calculated data. Magnetic course field is set to Rovins true course.
Note 2: Data invalid for initial alignment or speed saturation (i.e. INS User status ALIGNMENT OR INS User status SPEED_SATURATION bits set to 1).

4.1.3 $GPGST FRAME

$GPGST,hhmmss.ss,x.x,x.x,x.x,x.x,x.x,x.x,x.x*hh<CR><LF>		
hhmmss.ss	UTC Time	**Note 2**
x.x	RMS value of the standard deviation on pseudo-ranges	**Note 1**
x.x	Standard deviation of semi-major axis of error ellipse in meters	**Note 2**
x.x	Standard deviation of semi-minor axis of error ellipse in meters	**Note 2**
x.x	Orientation of semi-major axis of error ellipse	**Note 2**
x.x	Standard deviation of the error of Latitude in meters	**Note 2**
x.x	Standard deviation of the error of Longitude in meters	**Note 2**
x.x	Standard deviation of the error of Altitude	**Note 2**
hh	NMEA checksum	

Note 1: Copy of last GNSS values received. When no GNSS has been received since power-up, these fields are null except for number of satellites in use set to 3 by default.
Note 2: Rovins calculated data. Magnetic course field is set to Rovins true course.

4.1.4 $GPVTG FRAME

$GPVTG,x.xxx,T,x.xxx,M,x.xxx,N,x.xxx,K,a*hh<CR><LF>		
x.xxx	True course (deg)	**Note 1**
T	Fixed character = ‘T’	
x.xxx	Magnetic course (deg)	**Note 1**
M	Fixed character = ‘M’	
x.xxx	Speed (knots)	**Note 1**
N	Fixed character = ‘N’	
x.xxx	Speed (km/h)	**Note 1**
K	Fixed character=’K’	
a	Positioning system mode indicator ‘A’, ‘D’ or ‘E’	**see chapter 1.4**
hh	NMEA checksum	

Note 1: Rovins calculated data. Magnetic course field is set to Rovins true course.

4.1.5 $GPZDA AND $PHZDA FRAME

“—“ in the header is either GP or PH.

$--ZDA,hhmmss.ss,dd,mm,yyyy,hh,mm*hh<CR><LF>		
hhmmss.ss	UTC time of product	**Note 2**
dd	UTC day	**Note 2**
mm	UTC month	**Note 2**
yyyy	UTC year	**Note 2**
hh	Local zone hours	**Note 1**
mm	Local zone minutes	**Note 1**
hh	NMEA checksum	

Note 1: Copy of last GNSS values received. When no GNSS has been received since power-up, these fields are null except for number of satellites in use set to 3 by default.
Note 2: Rovins calculated data. Magnetic course field is set to Rovins true course.

4.1.6 $HEALF Frame

This frame is the new format for alert state reporting. It is sent only when the alert status changes or on alert request from ACN telegram. It complies with standard IEC 61924-2 (2012).

$HEALF,A,B,C,HHMMSS.SS,D,E,F,GGG,HHH,J,K,L,M--M*hh<CR><LF>		
A	Total number of ALF sentences	Fixed to 1
B	Sentence number	Fixed to 1
C	Sequential message identifier	Fixed to 1
HHMMSS.SS	Time	Time of last alert state change (UTC time if the system is synchronized in UTC, or internal time otherwise) in hours, minutes and seconds.
D	Alert category	Fixed to 'B' for a gyrocompass
E	Alert priority	Fixed to 'W' for a gyrocompass
F	Alert state	'V' for active - unacknowledged 'S' for active – silenced 'A' for active – acknowledged 'O' for active – responsibility transferred 'U' for rectified – unacknowledged 'N' for normal – no alert active
GGG	Manufacturer mnemonic	Empty (standardized code)
HHH	Alert identifier	Fixed to 240 for a gyrocompass
J	Alert instance	Fixed to 1
K	Revision counter	Starts at 1 and is incremented up to 99 after each change of content of any field of the alert (i.e: time, status). Resets to 1 after 99 is used.
L	Escalation counter	Starts at 0 and is incremented up to 9 each time the active-unacknowledged timer elapses. Resets to 1 after 9 is used.
M---M	Description text	Fixed to "System fault"
hh	NMEA checksum	

4.1.7 $HEHDT FRAME

$HEHDT,x.x,T*hh<CR><LF>		
x.x	True heading in degrees. Must be empty if HRP INVALID is set in user status	**Note 1**
T	Fixed ASCII character 'T'	
hh	NMEA checksum	

Note 1:

- 2 digits after the decimal point in default mode
- 5 digits after the decimal point in military mode
- Always 2 digits after the decimal point in case of UDP library protocol

4.1.8 $HETHS FRAME

$HETHS,x.x,a*hh<CR><LF>		
x.x	True heading in degrees	Note 1
a	Mode indicator character	see chapter 1.4
hh	NMEA checksum	

Note 1:

- 2 digits after the decimal point in default mode
- 5 digits after the decimal point in military mode
- Always 2 digits after the decimal point if Library protocol

4.2 Non-Standard NMEA or ASCII Telegrams

4.2.1 $PHCMP Frame

$PHCMP,llll.ll,a,x.xx,N*hh<CR><LF>		
llll.ll	Latitude in degrees (two first l) and in minutes (four last l)	
a	'N' for Northern hemisphere and 'S' for Southern hemisphere	
x.xx	Speed Norm in knots	
N	Fixed character = 'N'	
hh	NMEA checksum	

4.2.2 $PHGGA Frame

$PHGGA,hhmmss.ss,LLll.llllllll,a,LLLmm.mmmmmmm,a,x,xx,x.xxx,x.xxx,M,x.xxx,M,x.xxx,xxxx*hh<CR><LF>		
hhmmss.ss	is the UTC of position	
LLll.llllllll	is the latitude in degrees (LL) and in minutes (ll.llllllll)	8 digits after decimal point
a	is 'N' for Northern hemisphere, 'S' for Southern hemisphere	8 digits after decimal point
LLLll.llllllll	is the longitude in deg (LLL) and in minutes (ll.llllllll)	
a	is 'E' for East, 'W' for West	3 digits after decimal point
x	is the GNSS quality indicator	3 digits after decimal point
xx	is the number of satellites in use	
x.xxx	is the horizontal dilution of precision (HDOP) (*)	3 digits after decimal point
x.xxx	is the antenna altitude (meters) (here product altitude)	3 digits after decimal point
M	is the unit of antenna altitude (fixed character = 'M' for meters)	
x.xxx	is the Geoidal separation	
M	is the unit of Geoidal separation (fixed character = 'M' for meters)	
x.xxx	is the age of the differential GNSS data(seconds) (*)	
xxxx	is the differential reference station ID (*)	
hh	is the checksum	

(*) Last GNSS values received. When no GNSS has been received since power-up, these fields are null.

4.2.3 $PHHRP FRAME

$PHHRP,sxx,d,hhhhhhhh*hh<CR><LF>		
s	+ for Clockwise - for Counter Clockwise	
xx	Number of heading turn since the last reset	
d	'd' fix character	
hhhhhhhh	User Status	Hexadecimal value
hh	NMEA checksum	

4.2.4 $PHINF FRAME

$PHINF,hhhhhhhh*hh<CR><LF>		
hhhhhhhh	User Status	Hexadecimal value
hh	NMEA checksum	

4.2.5 $PHLIN FRAME

$PHLIN,x.xxx,y.yyy,z.zzz*hh<CR><LF>		
x.xxx	surge in meters (signed)	
y.yyy	sway in meters (signed)	
z.zzz	Heave in meters (signed)	
hh	NMEA checksum	

4.2.6 $PHPOS FRAME

$PHPOS,x.xxx,y.yyy,z.zzz,x.xxx,y.yyy,z.zzz *hh<CR><LF>		
x.xxx	Surge in m	With Selected Lever Arm
y.yyy	Sway in m	
z.zzz	Heave in m	
x.xxx	Surge in m	Without Lever Arm
y.yyy	Sway in m	
z.zzz	Heave in m	
hh	NMEA checksum	

4.2.7 $PHROT FRAME

$PHROT,x.xxx,y.yyy,z.zzz*hh<CR><LF>		
x.xxx	Roll speed in °/s (signed)	
y.yyy	Pitch speed in °/s (signed)	
z.zzz	Heading speed in °/s (signed)	
hh	NMEA checksum	

4.2.8 $PHSPD FRAME

$PHSPD,x.xxx,y.yyy,z.zzz*hh<CR><LF>		
x.xxx	Surge speed in m/s (signed)	
y.yyy	Sway speed in m/s (signed)	
z.zzz	Heave speed in m/s (signed	
hh	NMEA checksum	

4.2.9 $PHTRO FRAME

$PHTRO,x.xx,a,y.yy,b*hh<CR><LF>		
x.xx	Pitch in degrees	Pitch and Roll fields are empty if HRP_INVALID is set in user status
a	'M' for bow up and 'P' for bow down	
y.yy	Roll in degrees	
b	'B' for port down and 'T' for port up	
hh	NMEA checksum	

4.2.10 $PHVIT Frame

$PHVIT,x.xxx,y.yyy,z.zzz,x.xxx,y.yyy,z.zzz *hh<CR><LF>		
x.xxx	Surge speed in m/s	With Selected Lever Arm
y.yyy	Sway speed in m/s	
z.zzz	Heave speed in m/s	
x.xxx	Surge speed in m/s	Without Lever Arm
y.yyy	Sway speed in m/s	
z.zzz	Heave speed in m/s	
hh	NMEA checksum	

4.2.11 $PHVTG Frame

$PHVTG,x.xxx,T,x.xxx,M,x.xxx,N,x.xxx,K,a*hh<CR><LF>		
x.xxx	iis the True course (deg)	3 digits after decimal point
T	is a fixed character = ‘T’	
x.xxx	is the Magnetic course (deg): same value as the True	3 digits after decimal point
M	course	
x.xxx	is a fixed character = ‘M’	3 digits after decimal point
N	is the Horizontal speed (knots)	
x.xxx	is a fixed character = ‘N’	3 digits after decimal point
K	is the Horizontal speed (km/h)	
a	is a fixed character	
hh	is the positioning system mode indicator ‘A’, ‘D’ or ‘E’ (**)	
	is the checksum	

(*) Last GNSS values received. When no GNSS has been received since power-up, these fields are null.

(**) The quality indicator is managed as follows:

The INS does not copy the quality indicator received on GGA input to GGA output. The quality factor is set with respect to a correspondence table between INS calculated SD and Quality indicator in GGA telegram (refer to section 1.4).

4.2.12 $STALG Frame

$STALG,aaaaaaaa,bbbbbbbb*hh<CR><LF>		
aaaaaaaa	Algorithm status 1	Hexadecimal value
bbbbbbbb	Algorithm status 2	
hh	NMEA checksum	

4.2.13 $STSOR Frame

$STSOR,aaaaaaaa,bbbbbbbb*hh<CR><LF>		
aaaaaaaa	Sensor status 1	Hexadecimal value
bbbbbbbb	Sensor status 2	
hh	NMEA checksum	

4.2.14 $STSYS Frame

$STSYS,aaaaaaaa,bbbbbbbb*hh<CR><LF>		
aaaaaaaa	System status 1	Hexadecimal value
bbbbbbbb	System status 2	
hh	NMEA checksum	

4.2.15 $TIME_ Frame

$TIME_,hhmmss.sss*hh<CR><LF>		
hh	Hours	System Time or UTC Time if time-synchronized
mm	Minutes	
ss.sss	Seconds	
hh	NMEA checksum	

5 Status Description

The System status, Algorithm status, Sensor status and User status are updated in real-time as a built-in test tool. They are coded as bits, assembled into one or two 16 hexadecimal characters long words.

The User status is a synthetic fusion of the System status and Algorithm status. It also incorporates information on the FOG gyrometers and accelerometers status. The whole status is available through some ASCII-NMEA compliant protocols. Some binary protocols also use specific status bits for alarm and error detection. ASCII NMEA compliant protocols require a checksum to be sent at the end of each line.

When a status bit is set to 1, the corresponding message is displayed in the System Status area of the Web-Based Graphical User Interface with different colors:

- **Message in blue**: information message
- **Message in orange**: warning message
- **Message in red**: error message

5.1 Status Overview

The inertial product has built-in tests at a low-level of the hardware. These tests convert information translated into status information. The status information will help user to automatically check, in real-time, for any malfunctioning, failure or degradation of the system.
A warning status persists for 3 seconds. An error status persists for 10 seconds. This remains true whatever the protocol frequency.
The Web-Based Graphical User Interface display associated with the status follows in real-time the status.

There are different status information:

- Sensor Status:
  - Sensor Status 1: It corresponds to a low-level sensor status word. Each flag is linked to sensors state (optical source board, accelerometers and temperature sensors).
  - Sensor Status 2: It corresponds to a high-level sensor status. Each flag is a combination of specific sensor status flag
- Algorithm Status:
  - Each flag of the algorithm status is set/unset by navigation algorithm (algorithm state, external sensors, errors…).
- System Status:
  - Each flag of the system status is linked to system state (input/output activity, sensor detection, system errors…)
- User Status:
  - Each flag of the user status is a combination of flags from sensor, algorithm and/or system status.

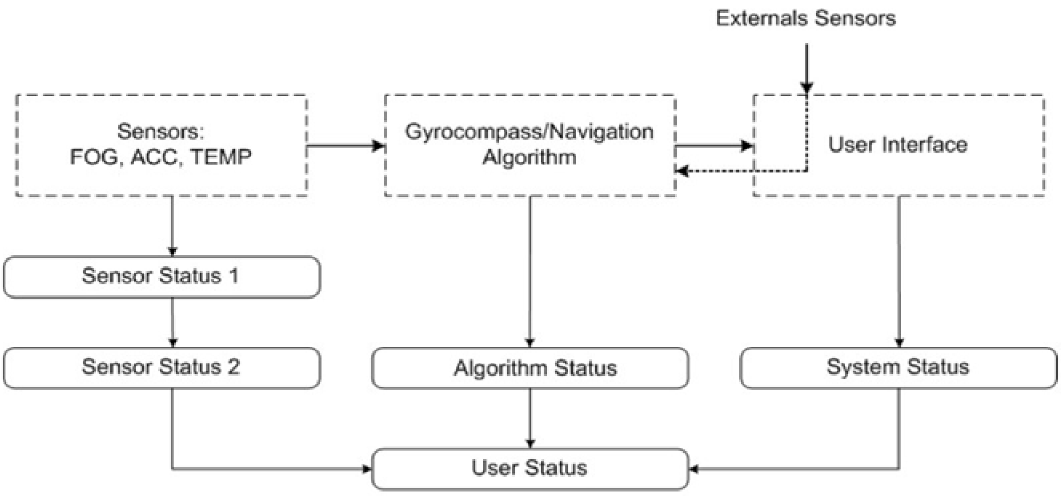


Figure 3 - Status tree description

5.2 Sensor Status

Sensor status 1 corresponds to a low-level sensor status.

Each of sensor status 1 flag can trigger "Degraded_Mode" or "Failure_Mode" flags, visible in sensor status 2.

Triggering thresholds corresponds to a number of successive occurrences (counted at sensor raw data frequency of fe=1/Te=400Hz). If an error occurs the total error count is incremented and if an error does not occur total error count is decremented as depicted below. If threshold is 100 and error is permanent it will take 100 * Te= 0.25 s to declare the error as “degraded mode” or “failure mode” in sensor status 2. This also means that we can tolerate 99 consecutive errors without declaring “degraded mode” or “failure mode” in sensor status 2. Each of sensor status 1 flag will be taken into account in sensor status 2 if “sensor status 2 threshold” is equal or greater to the specified value. For example, you need above the threshold count of 100 “FOG_n_SATURATION” errors, in sensor status 1 to flag “FOG_n_ERROR” in sensor status 2.

5.2.1 Low-level Sensor Status 1 Description

Bit	Name	Appearance conditions	Status value	Log inside Event Viewer	Log inside Error Viewer*
0	DATA_READY_ERR	Loss of sensors raw data (FOG or ACC)	0x00000001	No	No
1	SOURCE_POWER_CONTROL_ERR	FOG Optical source control power failure. Set when measured optical power < 50% of set point (triggered by FOG Optical source board: bits 7 and 0 source status)	0x00000002	No	No
2	SOURCE_DIODE_ERR	FOG Optical source diode off (bit 11 source status)	0x00000004	No	No
3	SOURCE_MODE_ERR	FOG Optical source not in power control mode (bit 8 source status)	0x00000008	No	No
4	ACC_X_SATURATION_ERR	Accelerometer saturation (ADC saturation: OVF: bit 2 of ACC_n (n=X,Y,Z) status)	0x00000010	No	No
5	ACC_Y_SATURATION_ERR		0x00000020	No	No
6	ACC_Z_SATURATION_ERR		0x00000040	No	No
7	ACC_X_ACQ_ERR	Accelerometer acquisition error. (ADC incorrect frame: Ee: bit 3 ACC_n (n=X,Y,Z) status)	0x00000080	No	No
8	ACC_Y_ACQ_ERR		0x00000100	No	No
9	ACC_Z_ACQ_ERR		0x00000200	No	No
10	FOG_X_SATURATION_ERR	FOG saturation. (ADC saturation: OVF_CAN: bit 0 FOG_n (n=X,Y,Z) status)	0x00000400	No	No
11	FOG_Y_SATURATION_ERR		0x00000800	No	No
12	FOG_Z_SATURATION_ERR		0x00001000	No	No
13	FOG_X_VPI_ERR	VPI voltage control error. (VPI used to convert FOG phase shift to voltage and rotation: ERROR_VPI: bit 1 FOG_n (n=X,Y,Z) status)	0x00002000	No	No
14	FOG_Y_VPI_ERR		0x00004000	No	No
15	FOG_Z_VPI_ERR		0x00008000	No	No
16	*reserved*	-	0x00010000	No	No
17	*reserved*	-	0x00020000	No	No
18	*reserved*	-	0x00040000	No	No
19	FOG_X_ACQ_ERR	FOG frame acquisition error. (sensor board level: Framing FOG_n (n=X,Y,Z) byte)	0x00080000	No	No
20	FOG_Y_ACQ_ERR		0x00100000	No	No
21	FOG_Z_ACQ_ERR		0x00200000	No	No

Bit	Name	Appearance conditions	Status value	Log inside Event Viewer	Log inside Error Viewer*
22	FOG_X_CRC_ERR	FOG frame CRC error. (DSP board level: Check of CRC FOG_n (n=X,Y,Z) Data)	0x00400000	No	No
23	FOG_Y_CRC_ERR		0x00800000	No	No
24	FOG_Z_CRC_ERR		0x01000000	No	No
25	TEMP_ACQ_ERR	Temperature acquisition error (Optical source, ACC and FOG temperatures on sensor board)	0x02000000	No	No
26	TEMP_THRESHOLD_ERR	Set to 1, if any of the measured temperature (TACCx,y,z; TFOG x,y,z, Tsource board) is greater than 90° or Tsensor board is greater than 115°C. In practice the Tsensor board will trigger the alarm first since temperature sensor is on an electronic component	0x04000000	No	No
27	DTEMP_THRESHOLD_ERR	DTEMP_THRESHOLD_ERR will be set to 1 if any temperature variation is greater than 5°/minute.	0x08000000	No	No
28	SENSOR_DATA_FIFO_WARNING	Sensor raw data FIFO half full	0x10000000	No	No
29	SENSOR_DATA_FIFO_ERR	Sensor raw data FIFO full	0x20000000	No	No
30	SOURCE_POWER_ERR	Difference between measured source power and requested source power greater than 10%	0x40000000	No	No
31	SOURCE_RECEPTION_ERR	Source data reception error (loss of source data, computed at DSP level)	0x80000000	No	No

* Error viewer corresponds to the Statistics in the Maintenance page of the Web-Based Graphical User Interface.

5.2.2 High-level Sensor Status 2 Description

Sensor status 2 corresponds to a high-level sensor status.

Each of the sensor status 2 flag is a combination ('or') of specific sensor status flag. "Degraded_Mode" or "Failure_Mode" flags are set if one the sensor status 1 flag count respectively reaches degraded mode or failure mode threshold.

Bit	Name	Status label displayed on the GUI	Appearance conditions	Status value	Log inside Event Viewer	Log inside Error Viewer*
0	FOG_X_ERR	**FOG X Error**	FOG_X_SATURATION_ERR (bit 10 Sensor Status 1) or FOG_X_ACQ_ERR (bit 19 Sensor Status 1) or FOG_X_CRC_ERR (bit 22 Sensor Status 1) or FOG_X_VPI_ERR (bit 13 Sensor Status 1) or FOG_X_LOW POWER (bit 16 Sensor Status 1)	0x00000001	Yes	Yes
1	FOG_Y_ERR	**FOG Y Error**	All Sensor Status 1 FOG_Y abnormalities (like FOG_X_ERR): (bits 11, 20, 23, 14, 17 and 5 of Sensor Status 1).	0x00000002	Yes	Yes
2	FOG_Z_ERR	**FOG Z Error**	All Sensor Status 1 FOG_Z abnormalities (like FOG_X_ERR): (bits 12, 21, 24, 15, 18 and 6 of Sensor Status 1).	0x00000004	Yes	Yes
3	SOURCE_ERR	**Source error**	SOURCE_POWER_CONTROL_ERR (bit 1 Sensor Status 1)	0x00000008	Yes	Yes
4	ACC_X_ERR	**Acc X Error**	ACC_X_SATURATION_ERR (bit 4 Sensor Status 1) or ACC_X_ACQ_ERR (bit 7 Sensor Status 1)	0x00000010	Yes	Yes
5	ACC_Y_ERR	**Acc Y Error**	All Sensor Status 1 ACC_Y abnormalities (like ACC_X_ERR): (bits 5 and 8 of Sensor Status 1).	0x00000020	Yes	Yes
6	ACC_Z_ERR	**Acc Z Error**	All Sensor Status 1 ACC_Z abnormalities (like ACC_X_ERR):	0x00000040	Yes	Yes

Bit	Name	Status label displayed on the GUI	Appearance conditions	Status value	Log inside Event Viewer	Log inside Error Viewer*
			(bits 6 and 9 of Sensor Status 1).			
7	TEMP_ERR	**Temperature Alarm**	TEMP_ACQ_ERR (bit 25 Sensor Status 1) or TEMP_THRESHOLD_ERR (bit 26 Sensor Status 1) or DTEMP_THRESHOLD_ERR (bit 27 Sensor Status 1)	0x00000080	Yes	Yes
8	DSP_OVERLOAD	**DSP overload**	DATA_READY_ERR (bit 0 Sensor Status 1) SENSOR_DATA_FIFO_WARNING (bit 28 Sensor Status 1) or SENSOR_DATA_FIFO_ERR (bit 29 Sensor Status 1)	0x00000100	Yes	No
9	ERR_INIT_CAN_ACC_X	-	CAN_ACC_X Initialization Error	0x00000200	No	Yes
10	ERR_INIT_CAN_ACC_Y	-	CAN_ACC_Y Initialization Error	0x00000400	No	Yes
11	ERR_INIT_CAN_ACC_Z	-	CAN_ACC_Z Initialization Error	0x00000800	No	Yes
12	MODELISATION_ERROR	**Modelisation Error**	Modelisation version not managed	0x00001000	No	Yes
13 - 29	*reserved*	-	-	-	No	Yes
30	DEGRADED_MODE	**Degraded Mode**	Fixed numbers trigger the Degraded Mode.	0x40000000	Yes	Yes
31	*reserved*	**Failure Mode**	Fixed numbers trigger the Failure Mode.	0x80000000	Yes	Yes

* Error viewer corresponds to the Statistics in the Maintenance page of the Web-Based Graphical User Interface.

5.3 Algorithm Status

5.3.1 Algorithm Status 1 Description

This Algorithm status is updated in real time and monitored through dedicated flags. Each flag is a bit which is set to "1" when flag is ON and set to "0" when flag is OFF.

Bit	Name	Status label displayed on the GUI	Conditions	Value	Log inside Event Viewer	Log inside Error Viewer
0	NAVIGATION	**Navigation mode**	Navigation phase Kalman filter enable	0x00000001	Yes	No
1	ALIGNMENT	**Alignment**	Alignment phase (5 minutes from power)	0x00000002	Yes	No
2	FINE ALIGNMENT	-	Fine alignment is set after the coarse alignment.	0x00000004	Yes	No
3	DVL_DMI_ CALIBRATION	**DVL calibration**	DVL Calibration	0x00000008	Yes	No
4	GPS_ALTITUDE	**GNSS altitude**	GNSS altitude used	0x00000010	Yes	No
5	DEPTHSENSOR_ALTITUDE	**Depth altitude**	Altitude from depth sensor used	0x00000020	Yes	No
6	ZERO_ALTITUDE	**Stabilization**	Altitude stabilized around zero	0x00000040	Yes	No
7	HYDRO_ALTITUDE	**Altitude Hydro**	Altitude = tide + heave	0x00000080	Yes	No
8	LOG_RECEIVED	**DVL BT reception**	Receiving sensor data	0x00000100	No	No
9	LOG_VALID	**DVL BT valid**	Information used for navigation	0x00000200	Yes	No
10	LOG_WAITING	**DVL BT waiting**	Sensor configured but no data received	0x00000400	No	No
11	LOG_REJECTED	**DVL BT rejected**	Information rejected by navigation algorithm	0x00000800	Yes	No
12	GPS_RECEIVED	**GNSS reception**	Receiving sensor data	0x00001000	No	No
13	GPS_VALID	**GNSS valid**	Information used for navigation	0x00002000	Yes	No
14	GPS_WAITING	**GNSS waiting**	Sensor configured but no data received	0x00004000	No	No
15	GPS_REJECTED	**GNSS rejected**	Information rejected by navigation algorithm	0x00008000	Yes	No
16	USBL_RECEIVED	**USBL reception**	Receiving sensor data	0x00010000	No	No
17	USBL_VALID	**USBL valid**	Information used for navigation	0x00020000	Yes	No
18	USBL_WAITING	**USBL waiting**	Sensor configured but no data received	0x00040000	No	No
19	USBL_REJECTED	**USBL rejected**	Information rejected by navigation algorithm	0x00080000	Yes	No
20	DEPTH_RECEIVED	**Depth reception**	Receiving sensor data	0x00100000	No	No
21	DEPTH_VALID	**Depth valid**	Information used for navigation	0x00200000	Yes	No
22	DEPTH_WAITING	**Depth waiting**	Sensor configured but no data received	0x00400000	No	No
23	DEPTH_REJECTED	**Depth rejected**	Information rejected by navigation algorithm	0x00800000	Yes	No

Bit	Name	Status label displayed on the GUI	Conditions	Value	Log inside Event Viewer	Log inside Error Viewer
24	LBL_RECEIVED	**LBL reception**	Receiving sensor data	0x01000000	No	No
25	LBL_VALID	**LBL valid**	Information used for navigation	0x02000000	Yes	No
26	LBL_WAITING	**LBL waiting**	Sensor configured but no data received	0x04000000	No	No
27	LBL_REJECTED	**LBL rejected**	Information rejected by navigation algorithm	0x08000000	Yes	No
28	ALTITUDE_SATURATION	**Altitude exceeded**	Altitude greater than maximum accepted value: 4000 m	0x10000000	Yes	No
29	SPEED_SATURATION	**Speed exceeded**	Speed greater than maximum accepted value: 41.15 m/s	0x20000000	Yes	No
30	INTERPOLATION_MISSED	**-**	Acceleration greater than maximal accepted value. Rotation rate greater than maximal accepted value	0x40000000	Yes	No
31	HEAVE_INITIALIZATION	**Heave init**	Heave filter initialization. Starts at power on and lasts in 6 minutes after power on	0x80000000	Yes	No

5.3.2 Algorithm Status 2 Description

Bit	Name	Status label displayed on the GUI	Conditions	Value	Log inside Event Viewer	Log inside Error Viewer
0	WATERTRACK_RECEIVED	**DVL WT reception**	Receiving sensor data	0x00000001	No	No
1	WATERTRACK_VALID	**DVL WT valid**	Information used for navigation	0x00000002	Yes	No
2	WATERTRACK _WAITING	**DVL WT waiting**	Sensor configured but no data received	0x00000004	No	No
3	WATERTRACK_REJECTED	**DVL WT rejected**	Information rejected by navigation algorithm	0x00000008	Yes	No
4	GPS2_RECEIVED	**GNSS2 reception**	Receiving sensor data	0x00000010	No	No
5	GPS2_VALID	**GNSS2 valid**	Information used for navigation	0x00000020	Yes	No
6	GPS2_WAITING	**GNSS2 waiting**	Sensor configured but no data received	0x00000040	No	No
7	GPS2_REJECTED	**GNSS2 rejected**	Information rejected by navigation algorithm	0x00000080	Yes	No
8	*reserved*	-	*reserved*	0x00000100	No	No
9	*reserved*	-	*reserved*	0x00000200	No	No
10	*reserved*	-	*reserved*	0x00000400	No	No
11	*reserved*	-	*reserved*	0x00000800	No	No
12	ALTITUDE_RECEIVED	**Altitude reception**	Receiving sensor data	0x00001000	No	No
13	ALTITUDE _VALID	**Altitude valid**	Information used for navigation	0x00002000	Yes	No
14	ALTITUDE _WAITING	**Altitude waiting**	Sensor configured but no data received	0x00004000	No	No
15	ALTITUDE _REJECTED	**Altitude rejected**	Information rejected by navigation algorithm	0x00008000	Yes	No
16	ZUPT_MODE_ACTIVATED	**ZUPT enabled**	ZUPT mode activated	0x00010000	Yes	No
17	ZUPT_MODE_VALID	**ZUPT valid**	ZUPT mode valid	0x00020000	Yes	No

Bit	Name	Status label displayed on the GUI	Conditions	Value	Log inside Event Viewer	Log inside Error Viewer
18	RO_ZUPT_MODE	**Rot ZUPT enabled**	AutoStatic Bench ZUPT mode activated	0x00040000	Yes	No
19	RO_ZUPT_VALID	**Rot ZUPT valid**	AutoStatic Bench ZUPT mode valid	0x00080000	Yes	No
20	*Reserved*	**-**	*Reserved*	0x00100000	No	No
21	*Reserved*	-	*Reserved*	0x00200000	No	No
22	*Reserved*	**-**	*Reserved*	0x00400000	Yes	No
23	EMULATION_MODE	**Emulation mode**	Simulation of DSP input sensor data (FOG,ACC)	0x00800000	Yes	No
24	EMLOG_RECEIVED	**EM LOG reception**	Receiving information	0x01000000	No	No
25	EMLOG_VALID	**EM LOG valid**	Information used for navigation	0x02000000	Yes	No
26	EMLOG_WAITING	**EM LOG waiting**	Sensor configured but no data received	0x04000000	No	No
27	EMLOG_REJECTED	**EM LOG rejected**	Information rejected by navigation algorithm	0x08000000	Yes	No
28	MANUALGPS_RECEIVED	**Manual position reception**	Receiving information	0x10000000	No	No
29	MANUALGPS_VALID	**Manual position valid**	Information used for navigation	0x20000000	Yes	No
30	MANUALGPS_WAITING	**Manual position waiting**	Sensor configured but no data received	0x40000000	No	No
31	MANUALGPS_REJECTED	**Manual position rejected**	Information rejected by navigation algorithm	0x80000000	Yes	No

5.3.3 Algorithm Status 3 Description

Bit	Name	Status label displayed on the GUI	Conditions	Value	Log inside Event Viewer	Log inside Error Viewer
0	*reserved*	-	*reserved*	0x00000001	No	No
1	*reserved*	-	*reserved*	0x00000002	No	No
2	*reserved*	-	*reserved*	0x00000004	No	No
3	*reserved*	-	*reserved*	0x00000008	No	No
4	*reserved*	-	*reserved*	0x00000010	No	No
5	*reserved*	-	*reserved*	0x00000020	Np	No
6	*reserved*	-	*reserved*	0x00000040	No	No
7	*reserved*	-	*reserved*	0x00000080	No	No
8	USBL2_RECEIVED	**USBL 2 reception**	Receiving information	0x00000100	No	No

Bit	Name	Status label displayed on the GUI	Conditions	Value	Log inside Event Viewer	Log inside Error Viewer
9	USBL2_VALID	USBL 2 valid	Information used for navigation	0x00000200	Yes	No
10	USBL2_WAITING	USBL 2 waiting	Sensor configured but no data received	0x00000400	No	No
11	USBL2_REJECTED	USBL 2 rejected	Information rejected by navigation algorithm	0x00000800	Yes	No
12	USBL3_RECEIVED	USBL 3 reception	Receiving information	0x00001000	No	No
13	USBL3_VALID	USBL 3 valid	Information used for navigation	0x00002000	Yes	No
14	USBL3_WAITING	USBL 3 waiting	Sensor configured but no data received	0x00004000	No	No
15	USBL3_REJECTED	USBL 3 rejected	Information rejected by navigation algorithm	0x00008000	Yes	No
16	*reserved*	-	-	0x00010000	No	No
17	CALCHK	Calibration check	DVL calibration check On/Off	0x00020000	Yes	No
18	*reserved*	-	-	0x00040000	Yes	No
19	REL_SPD_ZUP_ ACTIVATED	Rel. Spd ZUPT Mode enabled	Relative Speed ZUP activated	0x00080000	Yes	No
20	*REL_SPD_ZUP_VALID*	Rel. Spd ZUPT Mode valid	Relative Speed ZUP valid	0x00100000	Yes	No
21	EXT_SENSOR_ OUTDATED	Timing issue	One of the External Sensor received Is outdated	0x00200000	Yes	No
22	SENSOR_USED_ BEFORE_CALIB	Sensor used before calibration	Sensor used before calibration	0x00400000	No	No
23	*reserved*	-	-	0x00800000	Yes	No
24	*reserved*	-	*reserved*	0x01000000	No	No
25	*reserved*	-	*reserved*	0x02000000	No	No
26	*reserved*	-	*reserved*	0x04000000	No	No

Bit	Name	Status label displayed on the GUI	Conditions	Value	Log inside Event Viewer	Log inside Error Viewer
27	*reserved*	-	*reserved*	0x08000000	No	No
28	*reserved*	-	*reserved*	0x10000000	No	No
29	*reserved*	-	*reserved*	0x20000000	No	No
30	OPTIMAL_ALIGNMENT	**Optimal Alignment**	Optimal alignment flag is set only when heading standard deviation < 0.04° seclat Flag persistence time: 1 s	0x40000000	Yes	No
31	*reserved*	-	*reserved*	0x80000000	No	No

5.3.4 Algorithm Status 4 Description

Bit	Name	Status label displayed on the GUI	Conditions	Value	Log inside Event Viewer	Log inside Error Viewer
0	LOG2_RECEIVED	**DVL2 BT reception**	Receiving sensor data	0x00000001	No	No
1	LOG2_VALID	**DVL2 BT valid**	Information used for navigation	0x00000002	Yes	No
2	LOG2 _WAITING	**DVL2 BT waiting**	Sensor configured but no data received	0x00000004	No	No
3	LOG2_REJECTED	**DV2L BT rejected**	Information rejected by navigation algorithm	0x00000008	Yes	No
4	WATERTRACK2_RECEIVED	**DVL2 WT reception**	Receiving sensor data	0x00000010	No	No
5	WATERTRACK2_VALID	**DVL2 WT valid**	Information used for navigation	0x00000020	Yes	No
6	WATERTRACK2 _WAITING	**DVL2 WT waiting**	Sensor configured but no data received	0x00000040	No	No
7	WATERTRACK2_REJECTED	**DV2L WT rejected**	Information rejected by navigation algorithm	0x00000080	Yes	No
8	DVL_DIST_TRAVELED_VALID	-	Algorithm DVL calibration status	0x00000100	Yes	No
9	DVL_CALIBRATION_NONE	-		0x00000200	Yes	No
10	DVL_ROUGH_CALIBRATION	**Rough Calibration**		0x00000400	Yes	No
11	DVL_FINE_CALIBRATION	**Fine Calibration**		0x00000800	Yes	No
12	DVL_CHECK_CALIBRATION	**Verification in progress**		0x00001000	Yes	No
13	*reserved*	-	*reserved*	0x00002000	No	No
14	*reserved*	-	*reserved*	0x00004000	No	No
15	*reserved*	-	*reserved*	0x00008000	No	No
16	*reserved*	-	*reserved*	0x00010000	No	No
17	*reserved*	-	*reserved*	0x00020000	No	No
18	*reserved*	-	*reserved*	0x00040000	No	No
19	*reserved*	-	*reserved*	0x00080000	No	No

Bit	Name	Status label displayed on the GUI	Conditions	Value	Log inside Event Viewer	Log inside Error Viewer
20	*reserved*	-	*reserved*	0x00100000	No	No
21	*reserved*	-	*reserved*	0x00200000	No	No
22	*reserved*	-	*reserved*	0x00400000	No	No
23	*reserved*	-	*reserved*	0x00800000	No	No
24	*reserved*	-	*reserved*	0x01000000	No	No
25	*reserved*	-	*reserved*	0x02000000	No	No
26	*reserved*	-	*reserved*	0x04000000	No	No
27	*reserved*	-	*reserved*	0x08000000	No	No
28	CINT_FLASH_MEMORY_WRERR	**CINT Write Error**	Memory writing error after 3 retries	0x10000000	Yes	Yes
29	CINT_FLASH_MEMORY_ERERR	**CINT Erase Error**	Memory erasing error after 3 retries	0x20000000	Yes	Yes
30	CALIBRATION_COMPLETED	-	Calibration criteria has been reached	0x40000000	No	No
31	INCONSISTENT_SENSOR_DATA (1)	-	A forced sensor data has been rejected	0x80000000	Yes	No

(1) If the flag is raised, this means that the REJECTED flag of the concerned sensor has been raised also.

5.3.5 Algorithm Status 5 Description

Bit	Name	Status label displayed on the GUI	Conditions	Value	Log inside Event Viewer	Log inside Error Viewer
0-4	reserved	-	reserved	0x00000001	No	No
5-18	*reserved*	-	reserved for future use	0x00000020	Yes	No
19	*reserved*	-	*reserved*	0x00080000	No	No
20	TR_ZUPT_ACTIVATED	**Transv. ZUPT enabled**	Transverse ZUPT activated	0x00100000	Yes	No
21	*TR_ZUPT_VALID*	**Transv. ZUPT valid**	*Transverse ZUPT valid*	0x00200000	Yes	No
22	*PO_ZUPT_ACTIVATED*	**Pos. ZUPT enabled**	*Position ZUPT activated*	0x00400000	Yes	No
23	PO_ZUPT_VALID	**Pos. ZUPT valid**	Position ZUPT valid	0x00800000	Yes	No
24	*reserved*	-	*reserved*	0x01000000	Yes	No
25	*reserved*	-	*reserved*	0x02000000	Yes	No
26	CDSP_FLASH_MEMORY_WRERR	**CDSP Write Error**	Memory writing error after 3 retries	0x04000000	Yes	No
27	CDSP_FLASH_MEMORY_ERERR	**CDSP Erase Error**	Memory erasing error after 3 retries	0x08000000	Yes	No
28	CINT_FLASH_MEMORY_VIRGIN	**CINT Memory Virgin**	Memory is virgin on both configuration sectors (CRC and validity counter =0xffffffff)	0x10000000	Yes	Yes

Bit	Name	Status label displayed on the GUI	Conditions	Value	Log inside Event Viewer	Log inside Error Viewer
29	CDSP_FLASH_MEMORY_ERROR	**CINT Memory Error**	Both sectors used to save configuration are corrupted	0x20000000	Yes	Yes
30	CINT_FLASH_MEMORY_Backup	**Backup Memory Used**	Previous configuration not been saved correctly	0x40000000	Yes	No
31	reserved	-	reserved	0x80000000	No	No

5.3.6 Algorithm Status 6 Description

Bit	Name	Status label displayed on the GUI	Conditions	Value	Log inside Event Viewer	Log inside Error Viewer
0	*reserved*	-	*reserved*	0x00000001	No	No
1	*reserved*	-	*reserved*	0x00000002	No	No
2-31	*reserved*	-	*reserved*	0x00000004	No	No

5.4 System Status

This System status is updated in real time and monitored through dedicated flags. It is dedicated to getting information on the status of serial input and output lines together with Ethernet activity. It also controls the detection of external sensors and system malfunction. Each flag is a bit which is set to "1" when flag is ON and set to "0" when flag is OFF.

5.4.1 System Status 1

This System status is updated in real time and monitored through dedicated flags. It is dedicated to get information on status of serial input and output lines together with Ethernet activity. It also controls the detection of external sensors and system malfunction. Each flag is a bit which is set to "1" when flag is ON and set to "0" when flag is OFF.

Bit	Name	Status label displayed on the GUI	Conditions	Value	Log Inside Event Viewer	Log inside Error Viewer
0	SERIAL_IN_R_ERR	**Input R error**	Serial input framing error	0x00000001	Yes	No
1	INT_GPS_INPUT_ERR	**GNSS Input error**	GNSS Input port framing error only when using internal GNSS	0x00000002	Yes	No
2	INT_GPS_RAW_INPUT_ERR	**GNSS PP Input error**	GNSS PP Input port framing error only when using internal GNSS	0x00000004	Yes	No
3	reserved	-	-	0x00000008	No	No
4	INPUT_A_ERR	**Input A error**	Input port framing error	0x00000010	Yes	No
5	INPUT_B_ERR	**Input B error**	Input port framing error	0x00000020	Yes	No
6	reserved	-	-	0x00000040	No	No
7	reserved	-	-	0x00000080	No	No
8	INPUT_R_ACTIVITY	**Input R**	Input port activity	0x00000100	No	No
9	INT_GPS_ACTIVITY	**Internal GNSS**	Input port activity only when using internal GNSS	0x00000200	No	No
10	INT_GPS_RAW_ACTIVITY	**Internal GNSS PP**	Input port activity only when using internal GNSS	0x00000400	No	No
11	reserved	-	-	0x00000800	No	No
12	INPUT_A_ACTIVITY	**Input A**	Input port activity	0x00001000	No	No
13	INPUT_B_ACTIVITY	**Input B**	Input port activity	0x00002000	No	No
14	reserved	-	-	0x00004000	No	No
15	reserved	-	-	0x00008000	No	No
16	OUTPUT_R_FULL	**Output R full**	Output port full	0x00010000	Yes	No
17	OUTPUT_A_FULL	**Output A full**	Output port full	0x00020000	Yes	No
18	OUTPUT_B_FULL	**Output B full**	Output port full	0x00040000	Yes	No
19	reserved	-	-	0x00080000	Yes	No
20	OUTPUT_GPS_FULL	**GNSS Output full**	Output port full	0x00100000	Yes	No

Bit	Name	Status label displayed on the GUI	Conditions	Value	Log Inside Event Viewer	Log inside Error Viewer
21	OUTPUT_GPS_RAW_FULL	**GNSS PP Output full**	Output port full	0x00200000	Yes	No
22	ETHERNET_PORT_FULL	**Ethernet output full**	Ethernet output full (common for all Ethernet output ports)	0x00400000	Yes	No
23	*reserved*	-	*reserved*	0x00800000	No	No
24	INTERNAL_TIME_USED	**Internal time**	Internal time used	0x01000000	No	No
25	*EVENT_MARKER_ERR*	-	Event Marker process error (LANDINS) (input pulse frequency too high)	0x02000000	No	No
26	ETHERNET_PORT_ACTIVITY	-	Ethernet link reception (common for all Ethernet input ports)	0x04000000	No	No
27	PULSE_IN_A_ACTIVITY	**Pulse in A**	Input Pulse activity	0x08000000	No	No
28	PULSE_IN_B_ACTIVITY	**Pulse in B**	Input Pulse activity	0x10000000	No	No
29	PULSE_IN_C_ACTIVITY	**Pulse in C**	Input Pulse activity	0x20000000	No	No
30	PULSE_IN_D_ACTIVITY	**Pulse in D**	Input Pulse activity	0x40000000	No	No
31	*reserved*	-	*reserved*	0x80000000	No	No

5.4.2 System Status 2

Bit	Name	Status label displayed on the GUI	Conditions	Value	Log inside Event Viewer	Log inside Error Viewer
0	DVL_BT_DETECTED	**DVL BT reception**	DVL Bottom Track data detected	0x00000001	No	No
1	DVL_WT_DETECTED	**DVL WT reception**	DVL Water Track data detected	0x00000002	No	No
2	GPS_DETECTED	**GPS reception**	GPS data detected	0x00000004	No	No
3	GPS2_DETECTED	**GPS2 reception**	GPS2 data detected	0x00000008	No	No
4	USBL_DETECTED	**USBL reception**	USBL data detected	0x00000010	No	No
5	LBL_DETECTED	**LBL reception**	LBL data detected	0x00000020	No	No
6	DEPTH_DETECTED	**DEPT reception**	GPS data detected	0x00000040	No	No
7	EMLOG_DETECTED	**EM LOG reception**	GPS data detected	0x00000080	No	No
8	*reserved*	-	*reserved*	0x00000100	No	No
9	UTC_DETECTED	**UTC synchro**	UTC data detected (with synchronization ZDA or ZDA+PPS)	0x00000200	No	No
10	ALTITUDE_DETECTED	**Altitude reception**	Altitude data detected	0x00000400	No	No
11	PPS_DETECTED	**Synchro PPS1**	PPS signal detected	0x00000800	No	No

Bit	Name	Status label displayed on the GUI	Conditions	Value	Log inside Event Viewer	Log inside Error Viewer
12	ZUP_MODE_ACTIVATED	**ZUPT Mode enabled**	ZUPT Mode activated	0x00001000	No	No
13	*reserved*	-	*reserved*	0x00002000	No	No
14	MANUAL_GPS_DETECTED	**Manual position reception**	Manual GPS data detected	0x00004000	No	No
15	CTD_DETECTED	**Sound velocity reception**	CTD data detected	0x00008000	No	No
16	SIMULATION MODE	**Simulation mode**	Simulation mode active	0x00010000	Yes	No
17	*reserved*	-	*reserved*	0x00020000	No	No
18	DSP_ INCOMPATIBILITY	**Incompatible DSP firmware**	DSP firmware and CINT firmware not compatible	0x00040000	Yes	No
19	HEADING_ALERT	**Heading accuracy**	Heading standard deviation above user configurable value	0x00080000	Yes	No
20	POSITION_ALERT	**Position accuracy**	Position standard deviation above user configurable value	0x00100000	Yes	No
21	WAIT_FOR_POSITION	**Waiting for position**	Algorithm waiting for position to start	0x00200000	Yes	No
22	*reserved*	-	*reserved*	0x00400000	No	No
23	POLAR_MODE	**Polar Mode**	Polar mode	0x00800000	No	No
24	INTERNAL_LOG	**Internal logging**	Internal Log in progress	0x01000000	No	No
25	*reserved*	-	*reserved*	0x02000000	No	No
26	DOV_CORR_DETECTED	**Gravity corr. reception**	Vertical Deflection Correction Received	0x04000000	No	No
27	MPC_OVERLOAD	**Latency Warning**	Interface board CPU overload (too much output protocols with too high output rates)	0x08000000	Yes	No
28	POWER_SUPPLY_FAILURE	**Power failure**	Unit power failure	0x10000000	Yes	No
29	RD_MODE	**Usine Mode**	Unit in R&D mode (Internal use only)	0x20000000	No	No
30	CONFIGURATION_SAVED	**Configuration saved**	Configuration saved after having been changed	0x40000000	Yes	No
31	INPUT_PARAM_ERROR	**Input Param Error**	Input parameters error injected by production tools	0x80000000	No	No

5.5 User Status

User status is a synthetic fusion of System status, Algorithm Status and Sensor status. This User status is updated in real time and monitored through dedicated flags. Each flag is a bit which is set to "1" when flag is ON and set to "0" when flag is OFF. This status is used by several output protocols.

Bit	Name	Conditions	Value
0	DVL_RECEIVED_VALID	LOG_RECEIVED (bit 8 Algo Status 1) when LOG Rejection mode = Always true or LOG_VALID (bit 9 Algo Status 1) or WATERTRACK_RECEIVED (bit 0 Algo Status 2) when WaterTrack Rejection mode = Always true or WATERTRACK_VALID(bit 1 Algo Status 2)	0x00000001
1	GPS_RECEIVED_VALID	OR function on GPS status	0x00000002
2	DEPTH_RECEIVED_VALID	DEPTH_RECEIVED (bit 20 Algo Status 1) when DEPTH Rejection mode = Always true or DEPTH_VALID (bit 21 Algo Status 1)	0x00000004
3	USBL_RECEIVED_VALID	USBL_RECEIVED (bit 16 Algo Status 1) when USBL Rejection mode = Always true or USBL_VALID (bit 17 Algo Status 1)	0x00000008
4	LBL_RECEIVED_VALID	LBL_RECEIVED (bit 24 Algo Status 1) when LBL Rejection mode = Always true or LBL_VALID (bit 25 Algo Status 1)	0x00000010
5	GPS2_ RECEIVED_VALID	GPS2_RECEIVED (bit 4 Algo Status 2) when GPS2 Rejection mode = Always true or GPS2_VALID (bit 5 Algo Status 2)	0x00000020
6	EMLOG_RECEIVED_VALID	EMLOG_RECEIVED (bit 24 Algo Status 2) when EMLOG Rejection mode = Always true or EMLOG_VALID (bit 25 Algo Status 2)	0x00000040
7	MANUAL_GPS_ RECEIVED_VALID	OR function on manual fix status	0x00000080
8	TIME_RECEIVED_VALID	UTC detected	0x00000100
9	FOG_ANOMALY	FOG_X_ERR (bit 0 Sensor Status 2) or FOG_Y_ERR (bit 1 Sensor Status 2) or FOG_Z_ERR (bit 2 Sensor Status 2) or SOURCE_ERR (bit 3 Sensor Status 2)	0x00000200
10	ACC_ANOMALY	ACC_X_ERR (bit 4 Sensor Status 2) or ACC_Y_ERR (bit 5 Sensor Status 2) or ACC_Z_ERR (bit 6 Sensor Status 2)	0x00000400
11	TEMPERATURE_ERR	TEMP_ERR (bit 7 Sensor Status 2)	0x00000800
12	CPU_OVERLOAD	DSP_OVERLOAD (bit 8 Sensor Status 2) or MPC_OVERLOAD (bit 27 System Status 2)	0x00001000
13	DYNAMIC_EXCEDEED	INTERPOLATION_MISSED (bit 30 Algo Status 1)	0x00002000
14	SPEED_SATURATION	SPEED_SATURATION (bit 29 Algo Status 1)	0x00004000
15	ALTITUDE_SATURATION	ALTITUDE_SATURATION (bit 28 Algo Status 1)	0x00008000
16	INPUT_A_ERR	INPUT_A_ERR (bit 1 System Status 1)	0x00010000
17	INPUT_B_ERR	INPUT_B_ERR (bit 2 System Status 1)	0x00020000
18	INPUT_C_ERR	INPUT_C_ERR (bit 3 System Status 1)	0x00040000
19	INPUT_D_ERR	INPUT_D_ERR (bit 4 System Status 1)	0x00080000
20	INPUT_E_ERR	INPUT_E_ERR (bit 5 System Status 1)	0x00100000
21	OUTPUT_A_ERR	OUTPUT_A_FULL (bit 16 System Status 2)	0x00200000

Bit	Name	Conditions	Value
22	OUTPUT_B_ERR	OUTPUT_B_FULL (bit 17 System Status 2)	0x00400000
23	OUTPUT_C_ERR	OUTPUT_C_FULL (bit 18 System Status 2)	0x00800000
24	OUTPUT_D_ERR	OUTPUT_D_FULL (bit 19 System Status 2)	0x01000000
25	OUTPUT_E_ERR	OUTPUT_E_FULL (bit 20 System Status 2)	0x02000000
26	HRP_INVALID	ALIGNEMENT (bit 1 Algo Status 1) or FOG_ANOMALY (bit 9 User Status) or ACC_ANOMALY (bit 10 User Status) or SPEED_SATURATION (bit 29 Algo Status 1)	0x04000000
27	ALIGNEMENT	ALIGNEMENT (bit 1 Algo Status 1)	0x08000000
28	FINE_ALIGNMENT	FINE_ALIGNEMENT (bit 2 Algo Status 1)	0x10000000
29	NAVIGATION	NAVIGATION (bit 0 Algo Status 1)	0x20000000
30	DEGRADED_MODE	DEGRADED_MODE (bit 30 Sensor Status 2) or INTERPOLATION_MISSED (bit 30 Algo Status 1) or ALIGNEMENT (bit 1 Algo Status 1)	0x40000000
31	FAILURE_MODE	FAILURE_MODE (bit 31 Sensor Status 2) or ALTITUDE_SATURATION (bit 28 Algo Status 1) or SPEED_SATURATION (bit 29 Algo Status 1) or RESTORE_ATTITUDE_FAILED (bit 23 Algo Status 3)	0x80000000

5.6 High-level Status

Each flag of the high-level Status is a combination of flags from sensor, algorithm and/or system status. Unlike other status, some status bits are set when others are not set.

Bit	Name	Set when following bits are returned	Value
0	SYSTEM OK	SYSTEM INIT (bit 1 HighLevel Status) NOT SET and SYSTEM ERROR (bit 2 HighLevel Status) NOT SET and SYSTEM WARNING (bit 3 HighLevel Status) NOT SET	0x00000001
1	SYSTEM INIT	ALIGNMENT (bit 1 Algo Status 1) SET	0x00000002
2	SYSTEM ERROR	ALTITUDE SATURATION (bit 28 Algo Status 1) SET or SPEED SATURATION (bit 29 Algo Status 1) SET or ELECTRONIC ERROR (bit 8 high-level Status) SET or SERIAL OUT ERROR (bit 7 high-level Status) SET or SERIAL IN ERROR (bit 5 high-level Status) SET or FAILURE MODE (bit 31 Sensor Status 2) SET or FOG ANOMALY (bit 9 User Status) or ACC ANOMALY (bit 10 User Status)	0x00000004
3	SYSTEM WARNING	INTERPOLATION MISSED (bit 30 Algo Status 1) SET or DEGRADED MODE (bit 30 Sensor Status 2) SET	0x00000008
4	SERIAL IN OK	SERIAL IN ERROR (bit 5 HighLevel Status) NOT SET	0x00000010
5	SERIAL IN ERROR	INPUT R,A,B,C,D or E ERR (bits 0 to 5 System Status 1) SET	0x00000020
6	SERIAL OUT OK	SERIAL OUT ERROR (bit 7 HighLevel Status) NOT SET	0x00000040
7	SERIAL OUT ERROR	OUTPUT R,A,B,C,D or E FULL (bits 16 to 20 System Status 1) SET	0x00000080
8	ELECTRONIC OK	ELECTRONIC ERROR (bit 9 HighLevel Status) NOT SET	0x00000100
9	ELECTRONIC ERROR	TEMPERATURE ERROR (bit 16 HighLevel Status) SET or CPU ERROR (bit 15 HighLevel Status) SET or ACC ERROR (bit 13 HighLevel Status) SET or FOG ERROR (bit 11 HighLevel Status) SET	0x00000200
10	FOG OK	FOG ERROR (bit 11 HighLevel Status) NOT SET	0x00000400
11	FOG ERROR	FOG X,Y,or Z ERROR (bits 0,1,2 Sensor Status 2) SET	0x00000800
12	ACC OK	ACC ERROR (bit 13 HighLevel Status) NOT SET	0x00001000
13	ACC ERROR	OR function on accelerometer error	0x00002000
14	CPU OK	CPU ERROR (bit 15 HighLevel Status) NOT SET	0x00004000
15	CPU ERROR	DSP OVERLOPAD (bit 8 Sensor Status 2) SET or CINT OVERLOAD (bit 27 System Status 2) SET	0x00008000
16	TEMPERATURE OK	TEMP ERROR (bit 17 HighLevel Status) NOT SET	0x00010000

Bit	Name	Set when following bits are returned	Value
17	TEMPERATURE ERROR	TEMP ERROR (bit 7 Sensor Status 2) SET	0x00020000
18	GPS1	GPS DETECTED (bit 2 System Status 2) NOT SET	0x00040000
20	MANUAL GPS	MANUAL GPS DETECTED (bit 14 System Status 2) NOT SET	0x00100000
21	DVL BOTTOM TRACK	DVL BT DETECTED (bit 0 System Status 2) NOT SET	0x00200000
22	DVL WATER TRACK	DVL WT DETECTED (bit 1 System Status 2) NOT SET	0x00400000
23	EM LOG	EM LOG DETECTED (bit 7 System Status 2) NOT SET	0x00800000
24	DEPTH	DEPTH DETECTED (bit 6 System Status 2) NOT SET	0x01000000
25	USBL	USBL DETECTED (bit 4 System Status 2) NOT SET	0x02000000
26	LBL	LBL DETECTED (bit 5 System Status 2) NOT SET	0x04000000
27	ALTITUDE	ALTITUDE DETECTED (bit 10 System Status 2) NOT SET	0x08000000
28	UTC SYNC	UTC DETECTED (bit 9 System Status 2) NOT SET	0x10000000
29	PPS SYNC	PPS DETECTED (bit 11 System Status 2) NOT SET	0x20000000
30	CTD	CTD DETECTED (bit 15 System Status 2) NOT SET	0x40000000
31	ZUPT	ZUPT MODE ACTIVATED (bit 16 Algo Status 2) NOT SET and ZUPT MODE VALID (bit 17 Algo Status 2) NOT SET and AUTOSTATICBENCH ZUPT MODE (bit 18 Algo Status 2) NOT SET and AUTOSTATICBENCH ZUPT VALID (bit 19 Algo Status 2) NOT SET	0x80000000

6 Digital Protocols

Input digital interfaces are user-configurable with different protocols (or formats) to be selected from the digital interface library. Digital protocols can handle input data with different format: binary, ASCII and NMEA compliant.

An excel sheet is provided on the Rovins flash drive with the input and output protocols in order to choose the protocol depending on the parameters.

6.1 Detailed Specification of Input Protocols

APOS PSIMLBP

Standard: ASCII proprietary (compatible HIPAP APOS USBL/LBL data format).
Data received: Time, Position, depth, STD on position and depth.
Data frame: ASCII frame contains a header, 12 fields, and a checksum.
This ASCII sentence contains the beacon absolute positions given by an LBL system. The operator can define various parameters. In order to be properly taken into account by Rovins algorithm some fields must be properly set otherwise data are rejected.

The format is:

$PSIMLBP,hhmmss.ss,Tpc,aa,S,C,x.x,y.y,d.d,M.M,m.m,e.e,t.t[,D,E,…]*hh<CR><LF>		
hhmmss.ss	Real time of the measurement. Rovins only considers UTC time	USED
	Warning: Shall not be blank field.	
	Transponder code **Note 1**	
TpC	Up to three transponders can be taken into account	USED
	Warning: Shall not be blank field. At most, 3 characters expected.	
	Type of the item positioned	
	Status **Note 2**	
aa	Warning: Shall not be blank field.	NOT USED
	Only 1 character expected.	
S	ASCII ‘A’ = data valid	USED
	ASCII ‘V’ = data invalid	
	Coordinate system **Note 3**	
	Warning: Shall not be blank field. Only 1 character expected.	
	ASCII ‘C’ = Cartesian ASCII ‘P’ = Polar ASCII ‘U’ = UTM ASCII	
C	‘R’ = Radians	USED
	X_coordinate (latitude or UTM Northing) **Note 3**	
	Y_coordinate (longitude or UTM Easting) **Note 3**	
	Depth of the position in meters	
	Major axis of the error ellipse **Note 4**	
x.x	Minor axis of the error ellipse **Note 4**	USED
	Direction of the major axis in the error ellipse (in degrees) **Note 4**	
y.y	r.m.s. value of the normalized residuals	USED
	Potential additional fields	
d.d	is the NMEA checksum	USED
M.M		USED
m.m		USED
e.e		USED
t.t		NOT USED

$PSIMLBP,hhmmss.ss,Tpc,aa,S,C,x.x,y.y,d.d,M.M,m.m,e.e,t.t[,D,E,…]*hh<CR><LF>		
D,E,…		NOT USED
***hh**		USED

Note 1: The Tp code field contains a string characters code of the transponder for which the sentence contains a measurement. The characters are the same as the ones used on the HPR display and in the HPR operator manual. Up to three transponders can be taken into account by Rovins. The TP code and associated level arms shall be set with iXBlue Repeater software.

Note 2: The status field is ‘A’ when position is OK, and ‘V’ when the position is not OK or missing.

Note 3: Rovins uses either UTM or Radian coordinates of the transponder. Cartesian or Polar coordinates are not taken into account by Rovins. The Northing (X_coordinate) and the Eastings (Y_coordinate) are the UTM coordinates of the transponder.The Latitude (X_ coordinate) and the Longitude (Y_coordinate) are the geographic position in Radians. Positive latitude is north. Positive longitude is east. The Latitude and Longitude are in radians with 9 digits after the decimal point, giving a resolution of 0.01m.

Note 4: The “major axis of the error ellipse”, the “minor axis of the error ellipse” and “direction of the major axis in the error ellipse” are used to calculated the X,Y standard deviations set for Rovins algorithm.

$$\sigma_{lat} = \sqrt{M^2 \times (\cos e)^2 + m^2 \times (\sin e)^2}$$

$$\sigma_{long} = \sqrt{M^2 \times (\sin e)^2 + m^2 \times (\cos e)^2}$$

$$\mathrm{cov}_{LatLong} = -\cos e \times \sin e \times (m^2 - M^2)$$

M = major axis of the error ellipse
m = minor axis of the error ellipse
e = direction of the major axis in the error ellipse (in radians here)
The vertical depth standard deviation is fixed at 10 m.
Depending of sensor set-up, Rovins rejects the USBL data if USBL position is too far from the Rovins computed position: automatic rejection mode (recommended mode).

APOS PSIMSSB

Standard: ASCII proprietary. Input custom protocol.
Data received: Fix position, Time.
Data frame: NMEA ASCII frame.
This ASCII sentence contains the position of a USBL transponder. The operator may define various parameters. In order to be properly taken into account by INS algorithms, some fields must be properly set otherwise the data are rejected.

The format is:

$PSIMSSB,hhmmss.ss,TpC,s,eee,C,O,F,x.x,y.y,d.d,a.a,I,A.A,B.B[,D,E,…]*hh<CR><LF>		
$PSIMSSB	NMEA message header	USED
hhmmss.ss	Real time of the measurement. The INS only considers UTC time Warning: Shall not be blank field.	USED
TpC	Transponder Tp code **Note 1** Warning: Shall not be blank field. At most, 3 characters expected.	USED
s	Status **Note 2** Warning: Shall not be blank field. Only 1 character expected. ASCII ‘A’ = data valid ASCII ‘V’ = data invalid	USED
eee	Error code **Note 3** INS will not reject the USBL frame if this field is empty Warning: Blank field or 3 characters expected.	USED
C	Coordinate system Note 4 Note 1 Warning: Shall not be blank field. Only 1 character expected. ASCII ‘C’ = Cartesian ASCII ‘P’ = Polar ASCII ‘U’ = UTM ASCII ‘R’ = Radians	USED
O	Orientation **Note 5** Warning: Shall not be blank field. Only 1 character expected. ASCII ‘H’ = vessel Head up ASCII ‘ E’ = East ASCII ‘N’ = North	USED
F	Filter **Note 6** ASCII ‘M’ = Measured data ASCII ‘F’ = Filt6red data ASCII ‘P’ = Predicted data	NOT USED
x.x	X_coordinate (latitude or UTM Northing) **Note 4**	USED
y.y	Y_coordinate (longitude or UTM Easting) **Note 4**	USED
d.d	Depth in meters	USED
a.a	Expected accuracy of the position **Note 7**	USED
I	Additional information **Note 8**	USED

$PSIMSSB,hhmmss.ss,TpC,s,eee,C,O,F,x.x,y.y,d.d,a.a,I,A.A,B.B[,D,E,…]*hh<CR><LF>		
	Warning: Shall not be blank field. Only 1 character expected. ASCII 'N' = None ASCII 'C' = Compass ASCII 'I' = Inclinometer ASCII 'D' = Depth ASCII 'T' = Time from transponder to transducer ASCII 'V' = Velocity	
A.A	First additional value depending on additional info **Note 8** Warning: Blank field or first additional value	USED
B.B	Second additional value depending on additional info	NOT USED
D,E, …	Potential additional fields ignored	NOT USED
hh	NMEA checksum Warning: Shall not be blank field	USED

Note 1
The Tp code field contains a string characters code of the transponder for which the sentence contains a measurement. The characters are the same as the ones used on the HPR display and in the HPR operator manual. Up to three transponders can be taken into account by INS.

Note 2
The status field is 'A' when position is OK, and 'V' when the position is not OK or missing.
The Error code field contains in both case further description. INS rejects all data flagged 'V'.

Note 3
INS will not reject the USBL frame if this field is empty. If it is not the case, the error codes that make USBL frame rejected by INS are :
'NRy' : No reply received
'AmX': Error in X direction
'AmY' : Error in Y direction
'VRU': VRU error
'GYR': Gyro error
'ATT': Attitude sensor error
'ExM': External depth wanted but not received.
'???': System Unknown error

Note 4
INS uses either UTM or Radian coordinates of the transponder. Cartesian or Polar coordinates are not taken into account by Rovins.
The Northing (X_coordinate) and the Eastings (Y_coordinate) are the UTM coordinates of the transponder.
The Latitude (X_coordinate) and the Longitude (Y_coordinate) are the geographic position in Radians. Positive latitude is north. Positive longitude is east. The Latitude and Longitude are in radians with 9 digits after the decimal point, giving a resolution of 0.01m.

Note 5
INS takes into account only North Orientation.

Note 6

INS doesn’t check this field. It is recommended to use only measured data.

Note 7

Depending of sensor set-up, INS rejects the USBL data If USBL position is too far from the INS computed position: automatic rejection mode (recommended mode). The horizontal (X,Y) “expected accuracy of position” in the protocol is set for INS X,Y position standard deviation. The vertical depth standard deviation and the correlated latlong standard deviation is fixed at 10 m for INS.

Note 8

INS uses only Time delay “Time from transponder to transducer” when available and if INS is not UTC synchronized with GNSS time. Unit in seconds. If INS is UTC synchronized it will use the time stamp in the telegram (first field).

EM LOG VBW

Standard: Input NMEA 0183.

Data received:

- EM LOG: Water referenced longitudinal speed.
- DVL: Water track or Bottom track speeds.

Data frame:

$--VBW,x.x,x.x,S,x.x,x.x,s*hh <CR><LF>		
x.x	XV1 longitudinal water speed, in knots, '-' for astern.	USED (*) (**) (****)
x.x	XV2 transverse water speed, in knots, '-' for port.	USED (*) (**)
S	is the status of the water speed 'A' = data valid 'V' = data invalid	USED
x.x	XV1 longitudinal ground speed, in knots, '-' for astern.	USED (*) (***)
x.x	XV2 transverse ground speed, in knots, '-' for port.	USED (*) (***)
s	is the status of the ground speed A' = data valid 'V' = data invalid	USED
hh	Checksum	

The default standard deviation on speed taken into account by Rovins is 0.5 m/s.

(*): Shall not be blank field if data valid (Status='A').

(**): Used as DVL Water track speed if DVL is configured.

(***): Used as DVL Bottom track speed is DVL is configured.

(****): Used as EM Log speed is EM Log is configured.

EM LOG VHW

Standard: Input NMEA 0183.

Data received: EM LOG: Water referenced longitudinal speed.

Data frame:

$--VHW,x.x,T,x.x,M,x.x,N,x.x,K*hh <CR><LF>		
x.x	is the true heading in degrees.	NOT USED
T	T = True	NOT USED
x.x	is the magnetic heading in degrees.	NOT USED
M	M = magnetic.	NOT USED
x.x	is the XV1 longitudinal water speed, in knots. Warning : Shall not be a blank field	USED
N	N = knots.	NOT USED
x.x	is the speed, in km/h.	NOT USED
K	K = km/h	NOT USED
hh	Is the hexadecimal checksum.	USED

The default standard deviation on speed taken into account by Rovins is 0.5 m/s.

EXTERNAL SENSOR BIN

Standard: Binary, Input custom protocol.
Data received: DVL speeds, Fix position, Depth.
Data frame: 11 fields – 32 bytes – MSB first.
*Rovins will take account input DVL speeds and Fix Position if DVL and Fix position status are valid.

Message <F0><F1>…< F10>			
Field 0	Byte 0	Synchronization byte	'$'
Field 1	Bytes 1 to 4	DVL Longitudinal ground speed (XV1 speed)	IEEE floating point format, m/s (positive towards the bow)
Field 2	Bytes 5 to 8	DVL Transverse ground speed (XV2 speed)	IEEE floating point format, m/s (positive towards starboard) Warning: Opposite sign of INS usual convention
Field 3	Bytes 9 to12	DVL Vertical ground speed (XV3 speed)	IEEE floating point format, m/s (positive towards up side)
Field 4	Byte 13	DVL Status*	1 when valid, 0 when not valid
Field 5	Byte 14	Fix position Status*	1 when valid, 0 when not valid
Field 6	Bytes 15 to18	Latitude	Signed 32 bits +/-231 = +/-Pi
Field 7	Bytes 19 to 22	Longitude	Signed 32 bits +/-231 = +/-Pi
Field 8	Bytes 23 to 26	Fix position Age	IEEE floating point format, seconds
Field 9	Bytes 27 to 30	Depth	IEEE floating point format, meters
Field 10	Byte 31	Checksum	Addition of all the bytes from 0 to 30

*Rovins will take account input DVL speeds and Fix Position if DVL and Fix position status are valid.

GAPS

Standard: Input ASCII.
Data received: USBL beacon position.
Data frame: This protocol is meant to interface GAPS output to INS input.

$PTSAG,#NNNNN,hhmss.s,jj,mm,aaaa,BBB,DDMM.M,H,DDDMM.M,D,A,M.M,A,M.M*hh<CR><LF>		
#NNNNN	Accoustic recurrence n°	NOT USED
hhmss.s	Time validity of the data	USED
jj	Day= 0 to 31	NOT USED
mm	Month= 1 to 12	NOT USED
aaaa	Year	NOT USED
BBB	Is the beacon or ship ID 0: Ship 1 to 128: beacon	USED
DDMM.M	Latitude in degree (DD) minutes (MM.MMMM)	USED
H	'N' for Northern hemisphere, 'S' for Southern hemisphere	USED
DDDMM.M	Longitude in degree (DDD) minutes (MM.MMMM)	USED
D	'E' for East, 'W' for West	USED
A	Lat/Long Beacon position validity. If A=0xF position is valid else position is false.	USED
M.M	Calculated beacon depth in meters (*)	USED
A	Beacon depth validity. A= 0 : no depth A= 1: Calculated A= 2: Sensor	NOT USED
M.M	Sensor depth in meters If Beacon is not equipped with depth sensor this field is 9999.	NOT USED
hh	Checksum	USED

(*) Beacon can be equipped with a depth sensor. Calculated beacon depth is the fusion between sensor depth and calculated USBL depth. If Ship position is sent, the last 3 fields are set to 0.

(**) The USBL fix SD on position is calculated taking into account the GAPS SD on position and internal INS SD on position. Hence we use, X,Y,Depth of $PTSAX telegram to calculate slant range SD and SDlat, SDlong, SDaltitude of the INS $PIXSE,STDPOS telegram as follow:

$$SD_{range} = 2\% \cdot \sqrt{X^2 + Y^2 + Depth^2}$$

$$SD_{latitude} = \sqrt{SD_{lat}^{\ 2} + SD_{range}^{\ 2}} ; SD_{longitude} = \sqrt{SD_{lat}^{\ 2} + SD_{range}^{\ 2}} ;$$

$$SD_{altitude} = \sqrt{SD_{altitude}^{\ 2} + SD_{range}^{\ 2}}$$

$PIXSE,STDPOS,x.x,y.y,z.z*hh<CR><LF> (*)		
x.x	Latitude standard deviation in meters	USED
y.y	Longitude standard deviation in meters	USED
z.z	Altitude standard deviation in meters	USED
hh	Checksum	USED

(*) This telegram is INS SD on position output by INS.

$PTSAX,#NNNNN,hhmss.s,jj,mm,aaaa,BBB,X.X,Y.Y,A,P.P,A,C.C*hh<CR><LF>		
#NNNNN	Accoustic recurrence n°	NOT USED
hhmss.s	Time validity of the data	USED
jj	Day= 0 to 31	NOT USED
mm	Month= 1 to 12	NOT USED
aaaa	Year	NOT USED
BBB	Is the beacon ID 1 to 128: beacon	USED
X.X	X forward position in meters (*)	USED
Y.Y	Y starboard position in meters (*)	USED
A	XY position validity If A=0xF position is valid else position is false.	USED
P.P	Calculated beacon depth in meters (**)	USED
A	Beacon depth validity. A= 0 : no depth A= 1: Calculated A= 2: Sensor	NOT USED
C.C	Sensor depth in meters If Beacon is not equipped with depth sensor this field is 9999.	NOT USED
hh	Checksum	USED

(*) This is relative position in GAPS reference frame.

(**) Beacon can be equipped with a depth sensor. Calculated beacon depth is the fusion between sensor depth and calculated USBL depth.

(***) PTSAX is only used to calculate the standard deviation on GAPS position given by the PTSAG telegram.

$PIXSE,GPSIN,x.x,y.y,z.z,hhmmss.s*hh<CR><LF> (*)		
x.x	Latitude in degrees	USED
y.y	Longitude in degrees	USED
z.z	Altitude in meters	USED
hhmss.s	Time validity of the data	USED
Q	Quality factor (**)	USED
hh	Checksum	USED

(*) Last valid GNSS fix received by GAPS

(**) Refer to section 1.4.

$PIXSE,UTCIN_, hhmss.sss*hh<CR><LF> (*)		
hhmss.sss	Time validity of the data	USED
hh	Checksum	USED

(*) This is a GAPS time. We recommend to use time and PPS pulse from GNSS when available to get the best time synchronization accuracy.

GPS

Standard: Input ASCII.

Data received: Time, latitude, longitude, altitude, hemisphere, quality factor, number of satellites, HDOP, depth, geoïdal separation, checksum NMEA

Data frame is composed of:

- NMEA $--GGA Frame see section 3.1.1
- NMEA $--GLL Frame see section 3.1.2
- NMEA $--GST Frame see section 3.1.3
- NMEA $--RMC Frame see section 3.1.4
- UTC Frame see section 3.2.1
- NMEA $--ZDA Frame see section 3.1.7

HALLIBURTON SAS

Standard: ASCII, NMEA 0183.

Data received: Time stamp, USBL Beacon position in Lat/Long, USBL position standard deviations, age of data, Beacon depth, Delta latitude, Delta longitude, Standard deviations, GNSS Latitude, Longitude, GNSS quality, GNSS time.

Data frame:

$PUSBA,hhmmss.s,llmm.m,a,LLLmm.m,b,±c.c,±c.c,±c.c,±c.c,r.r,t.t,s.s,a.a, llmm.m,a,LLLmm.m,b,d.d,c<CR><LF>

hhmmss.s	UTC time stamp of position in hours (hh) minutes (mm) and second (ss.ss) (taken into account if INS is UTC time synchronized otherwise age of data is taken into account)	ss.s : 40 bit float (*)
llmm.m	is the latitude in degrees (ll) and in minutes (mm.m)	mm.m : 40 bit float
a	is the hemisphere N: North S: South	char
LLLmm.m	is the longitude in degrees (LLL) and in minutes (mm.m)	mm.m : 40 bit float
b	is longitude sign E: East W: West	char
±c.c	is estimated errors in latitude, covar(Lat), in meters^2	40 bit float
±c.c	is estimated error in longitude, covar(Long), in meters^2	40 bit float
±c.c	is estimated correlated lat/long error, covar(LatLong), in meters^2	40 bit float
±c.c	is estimated error in depth, covar(depth) in meters^2	40 bit float
r.r	range standard deviation error in meters (NOT USED)	40 bit float
t.t	angular observation standard deviation in degrees (NOT USED)	40 bit float
s.s	is the slant range in meters (NOT USED)	40 bit float
a.a	is the age of the data in seconds (taken into account if INS is not UTC time synchronized)	40 bit float
llmm.m	is the vessel latitude in degrees (ll) and in minutes (mm.m) (NOT USED)	mm.m: 40 bit float
a	is the hemisphere N: North S: South	char (NOT USED)
LLLmm.m	is the vessel longitude in degrees (LLL) and in minutes (mm.m)	mm.m : 40 bit float
b	is longitude sign E: East W: West	char
d.d	is the beacon depth	40 bit float
c	is the computed mode (NOT USED)	

(*) xx.x : 2 characters before "."

$PUSBR,hhmmss.ss,l.l,y.y,±ccc.c,±ccc.c,±ccc.c,±ccc.c,r.r,t.tt,ssss.s,aa.a,<CR><LF>

where:	hhmmss.ss	UTC time stamp of position (taken into account if INS is UTC time synchronized otherwise age of data is taken into account)[1] (NOT USED)
	l.l	is the correction of latitude in meters (NOT USED)
	y.y	is the correction of longitude in meters (NOT USED)
	±ccc.c	are estimated errors in position, covar(Lat), covar(Long), covar(LatLong), covar(deltaT) in meters (NOT USED)
	r.r	is the standard error of range observation in meters (NOT USED)
	t.tt	is the standard error of angular observation in degrees (NOT USED)
	s.s	is the slant range in meters (NOT USED)
	a.a	is the age of the data in seconds (NOT USED)

$PLBL,hhmmss.ss,r.r,d.d,e.e,llm.m,a,LLLm.m,b,a.a,<CR><LF>

where:	hhmmss.ss	UTC time stamp of position (taken into account if INS is UTC time synchronized otherwise delay of data is taken into account)[1]
	r.r	is the range in meters
	d.d	is the delay in seconds
	e.e	is the range standard error in meters
	llm.m	is the seabed beacon latitude in deg (ll) and in min (m.m)
		is the hemisphere
	a	N: North
		S: South
		is the seabed beacon longitude in deg (LLL) and in min (m.m)
	LLLm.m	is longitude sign
	b	E: East
		W: West
	a.a	is the seabed beacon depth in meters

$--GGA Frame, see chapter 3.1.1

$GPZDA,hhmmss.ss,,,,, *hh <CR><LF>		
where :	hhmmss.ss	is the UTC of the last PPS received
	hh	is the checksum

IXBLUE STD BIN V2, V3

For the description of this protocol, refer to the section "IXBLUE STD BIN V2 V3", page 172 in the Digital Output protocols chapter.

IXSEA AUV

Standard: ASCII.

Data frame:

Conventions

Telegram format

The telegram is a combination of sensor blocks. Each block contains header, telegram identification and checksum. Any combination of sensor blocks can be sent at input of INS. Multiple sensor blocks of a kind can be sent (i.e: multiple USBL or LBL beacon positions).

The checksum is the sum of signed bytes of the telegram (telegram length – 2 checksum bytes).

All identification values (telegram identification, system type, rejection mode…) are expressed in decimal value otherwise specified.

Time

INS time can be synchronized with GNSS UTC time when UTC time block is sent to INS at regular intervals (i.e.: every second). To improve accuracy it is recommended to input 1 PPS pulse at pulse input of INS. If INS is not time synchronized INS will use the data latency information in sensor data blocks to evaluate age of data. Time in sensor data blocks is reset to 0 every 24 hour.

Data types

Each telegram description uses following convention:

Type name	Description
Byte	Unsigned 8 bit integer
Character	Signed 8 bit integer
Word	Unsigned 16 bit integer
Short	Signed 16 bit integer
DWord	Unsigned 32 bit integer
Long	Signed 32 bit integer
Float	IEEE Float 32 bits
Double	IEEE Float 64 bits

All 16 and 32 bits integers are represented in Big endian convention (MSB sent first).

Frames

Refer to section 1.3 for the reference frame.

Altitude convention

Altitude is referenced to Geoïdal model (mean sea level) if managed by the GNSS. In this case, geoïdal separation field is a valid IEEE float that contains distance between geoid and ellipsoid at current position. Otherwise if GNSS does not manage mean sea level altitude, the altitude field if referenced from ellipsoid and geoidal separation field contains NaN value 0x7FC00000.

DVL speed compensation in INS using sound speed

The speed of sound sent in the telegram is the value that was used internally by the DVL to compute velocity. INS will use both the speed of sound internally used by the DVL (calculated using temperature, depth, salinity or fixed set value) expressed as CDVL and the speed of sound measured by an external sensor (CTD, SVP) expressed as CEXT. The corrected DVL speed will be calculated using the following formula:

$$V_{corrected} = \frac{C_{EXT}}{C_{DVL}} \cdot V_{DVL}$$

If water track or bottom track data is sent from another sensor (i.e: EM LOG, Speed correlation sensor) that does not use sound velocity, this field should be sent as NaN value 0x7FC00000. In this case no compensation is made on speed.

Protocol description

GNSS input telegram (Id=1,2,3; version 0x01)

Data	Format	Units
Header	Byte	Value : '$'
Telegram size	Byte	48 Bytes
Telegram identification	Byte	1 for GNSS1 2 for GNSS2 3 for Manual GNSS
Data bloc version	Byte	0x01
Data validity time	Dword	GNSS time tag in steps of 100µs
GNSS quality	Byte	Refer to section 1.4
Latitude	Long	+/- 231= +/-Pi radian
Longitude	Long	+/- 231= +/-Pi radian
Altitude	Float	meters
Latitude standard deviation	Float	Meters
Longitude standard deviation	Float	Meters
Altitude standard deviation	Float	meters
Validity flag	Byte	1= Valid; 0= Invalid

Data	Format	Units
Latitude/Longitude covariance	Float	$meters^2$
Real time latency of data	Float	seconds (refer to Time convention of this protocol)
Geoidal separation	Float	meters (refer to Altitude convention of this protocol)
Checksum	Word	unsigned sum of all fields except checksum

UTC time (Id= 4; version 0x01)

Data	Format	Units
Header	Byte	Valeur : '$'
Telegram size	Byte	14 Bytes
Telegram identification	Byte	4
Data bloc version	Byte	0x01
UTC Time	Dword	UTC time in steps of 100us
Day	Byte	Day (1-31), 0 if unavailable
Month	Byte	Month (1-12), 0 if unavailable
Year	Word	Year (>2008), 0 if unavailable
Checksum	Word	Unsigned sum of all fields except checksum

USBL telegram (Id=5; version 0x01)

Data	Format	Units
Header	Byte	Value : '$'
Telegram size	Byte	49 Bytes
Telegram identification	Byte	5
Data bloc version	Byte	0x01
Data validity time	Dword	USBL time tag in steps of 100µs
Beacon ID	6 Byte	6 ASCII characters (1)
Latitude	Long	+/- 231= +/-Pi Radian
Longitude	Long	+/- 231= +/-Pi Radian
Real time latency of data	Float	Seconds (refer to Time convention in this protocol)
Altitude	Float	Meters
Latitude standard deviation	Float	Meters
Longitude standard deviation	Float	Meters
Latitude/Longitude covariance	Float	Meters^2
Altitude standard deviation	Float	Meters
Validity flag	Byte	1=Valid; 0=Invalid
Checksum	Word	Unsigned sum of all fields except checksum

(1) If beacon ID length is less than 6 bytes, it must be padded with null (\0) ASCII characters at the end.

Depth telegram (Id= 6; version 0x01)

Data	Format	Units
Header	Byte	Value : '$'
Telegram size	Byte	23 Bytes
Telegram identification	Byte	6
Data bloc version	Byte	0x01
Data validity time	Dword	Depth time tag in steps of 100us
Depth	Float	Meters
Depth standard deviation	Float	Meters
Validity flag	Byte	1=Valid; 0=Invalid
Real time latency of data	Float	Seconds (Refer to Time convention in this protocol)
Checksum	Word	Unsigned sum of all fields except checksum

LBL telegram (Id= 7; version 0x01)

Data	Format	Units
Header	Byte	Value : '$'
Telegram size	Byte	41 Bytes
Telegram identification	Byte	7
Data bloc version	Byte	0x01
Data validity time	Dword	LBL Time tag by steps of 100us
Beacon ID	6 Byte	6 ASCII characters (1)
Beacon latitude	Long	+/- 231= +/-Pi Radian
Beacon longitude	Long	+/- 231= +/-Pi Radian
Beacon altitude	Float	meters
Range	Float	meters
Real time latency of data	Float	seconds (refer to Time convention in this protocol)
Range standard deviation	Float	meters
Validity flag	Byte	1=Valid; 0=Invalid
Checksum	Word	unsigned sum of all fields except checksum

(1) If beacon ID length is less than 6 bytes, it must be padded with null (\0) ASCII characters at the end.

Ground speed (Id= 8; version 0x01)

Data	Format	Units
Header	Byte	Value : '$'
Telegram size	Byte	51 Bytes
Telegram identification	Byte	8
Data bloc version	Byte	0x01
Data validity time	Dword	Ground speed time tag in steps of 100us
XS1 speed	Float	meters/second (refer to section 1.3)
XS2 speed	Float	meters/second
XS3 speed	Float	meters/second
DVL speed of sound or	Float	meters/second (refer to DVL convention in this protocol)
External sensor speed of sound	Float	meters/second (refer to DVL convention in this protocol)
DVL altitude (bottom range)	Float	Meter
XS1 speed standard deviation	Float	meters/second
XS2 speed standard deviation	Float	meters/second
XS3 speed standard deviation	Float	meters/second
Validity flag	Byte	1=Valid; 0=Invalid
Real time latency of data	Float	Seconds (refer to Time convention in this protocol)
Checksum	Word	Unsigned sum of all fields except checksum

Water speed (Id= 9; version 0x01)

Data	Format	Units
Header	Byte	Valeur : '$'
Telegram size	Byte	51 Bytes
Telegram identification	Byte	9
Data bloc version	Byte	0x01
Data validity time	Dword	Water speed time tag by steps of 100us
XS1 speed	Float	meters/second (refer to section 1.3)
XS2 speed	Float	meters/second
XS3 speed	Float	meters/second
DVL speed of sound or	Float	meters/second (refer to DVL convention in this protocol)
External sensor speed of sound	Float	meters/second (refer to DVL convention in this protocol)
DVL altitude (bottom range)	Float	meter
XS1 standard deviation	Float	meters/second
XS2 standard deviation	Float	meters/second
XS3 standard deviation	Float	meters/second
Validity flag	Byte	1=Valid; 0=Invalid
Real time latency of data	Float	seconds (refer to Time convention in this protocol)
Checksum	Word	unsigned sum of all fields except checksum

IXSEA USBL INS1

Standard: Input ASCII.
Data received: Latitude, Longitude, Immersion, time stamp, STD on position.
21 bytes expected. LSB first. Dedicated to receive USBL data from GAPS USBL system
Data frame:

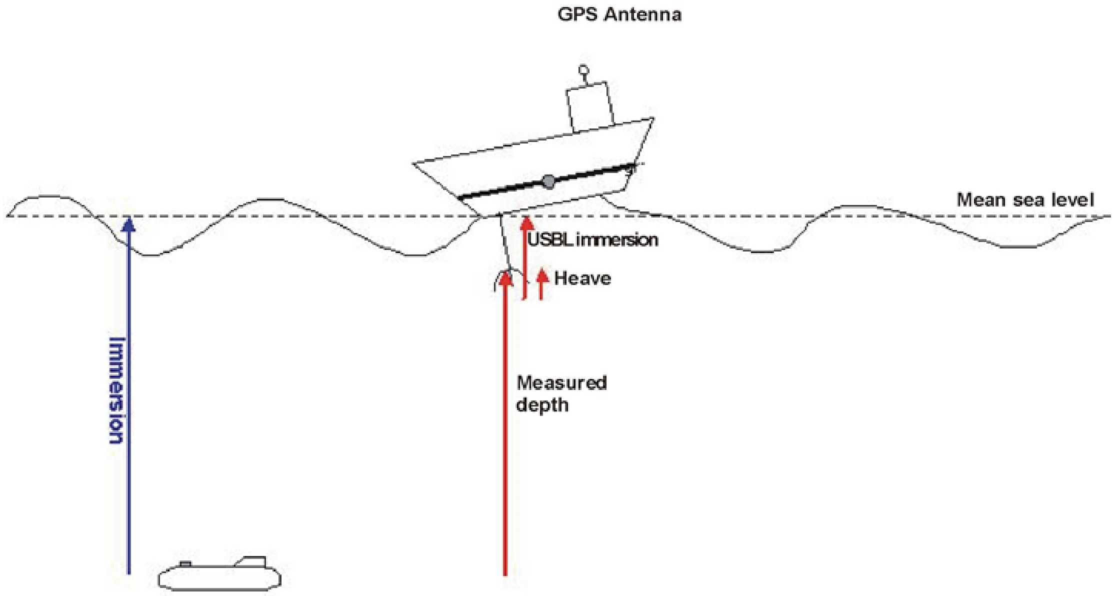
Byte Nb	Field	Nb of bits	definition	Value LSB	Min Value	Max Value
1	GGhex	8	header	55hex		
2	Bits 2 to 0	3	Message number	001bin		
	Bits 7 to 3	5	Transponder ID	NA	0	31
3, 4, 5, 6	LLLLLLLLhex	32	Latitude (deg) (Two-complement coded)	180/232	- 90 °	90x(1-232) °
7, 8, 9, 10	NNNNNNNNhex	32	Longitude (deg)	360/232	0 °	360.(1-232) °
11 to12 and bit 0 to 3 of byte 13	ZZZZZhex	20	Z (m), immersion	0,01 m	0 m (00000hex)	+10485,75 m (FFFFFhex)
13	Bit 4	1	Reserved			
	Bits 7 to 5	3	Standard deviation of semi-major axis of error ellipse in meters: 000 : 0 m < standard deviation < 0,5 m 001 : 0,5 m < standard deviation < 1,5 m 010 : 1,5 m < standard deviation < 3 m 011 : 3 m < standard deviation < 6 m 100 : 6 m < standard deviation < 12 m 101 : 12 m < standard deviation < 25 m 110 : 25 m < standard deviation < 50 m 111 : standard deviation > 50 m 000 to 110 : STD1 is set to max value (i.e : 50 m for 110) 111 : STD1 is set to 500 m			

Byte Nb	Field	Nb of bits	definition	Value LSB	Min Value	Max Value
14	Bit 0	1	Reserved			
	Bits 3 to 1	3	Standard deviation of semi-minor axis of error ellipse in meters: 000 : 0 m < standard deviation < 0,5 m 001 : 0,5 m < standard deviation < 1,5 m 010 : 1,5 m < standard deviation < 3 m 011 : 3 m < standard deviation < 6 m 100 : 6 m < standard deviation < 12 m 101 : 12 m < standard deviation < 25 m 110 : 25 m < standard deviation < 50 m 111 : standard deviation > 50 m 000 to 110 : STD1 is set to max value (i.e : 50 m for 110) 111 : STD1 is set to 500 m			
	Bits 7 to 4	4	Angle between the North and the major axe of the XY error ellipse l	180/24	0 °	15/16 x180 °
15	Bit 0	1	Reserved			
	Bits 3 to 1	3	Standard deviation of the Z(m) position 000 : 0 m < standard deviation < 0,5 m 001 : 0,5 m < standard deviation < 1,5 m 010 : 1,5 m < standard deviation < 3 m 011 : 3 m < standard deviation < 6 m 100 : 6 m < standard deviation < 12 m 101 : 12 m < standard deviation < 25 m 110 : 25 m < standard deviation < 50 m 111 : standard deviation > 50 m 000 to 110 : STD1 is set to max value (i.e : 50 m for 110) 111 : STD1 is set to 500 m			
	Bits 7 to 4	4	Reserved			
16	PPhex	8	Reserved			
17, 18 19	rrrrrrrrrrrrrrrrrrrrrrbin (8+8+6 bits of: - byte 17, - byte 18, - bits 5 to 0 of byte 19)	22	UTC time stamp of position in ms.	1 ms	0 ms	(3 599 999 ms, i.e. 1h - 1ms The largest values of this field do not have any meaning

Byte Nb	Field	Nb of bits	definition	Value LSB	Min Value	Max Value
	Bits 7 and 6 of byte19	2	Reserved			
20	KKhex	8	Status byte Bits 0 to 8 - Reserved			
21	SShex	8	Checksum exclusive OR by byte for the whole 20 first bytes thus header included.			

Immersion definition

The immersion corresponds to the mobile depth with respect to the mean sea level (heave corrected). This data is coherent with the one that would be given by a depth sensor on the underwater mobile. But it does not allow to deduce the absolute height as there is no compensation for tide.



ROV immersion calculated by GAPS =
transponder depth (by the GAPS USBL acoustic array) +
+ USBL antenna immersion
- USBL antenna heave

MICRO SVT_P

Standard: ASCII, Input CTD.

Data received: Time Stamps, Sound Velocity, Pressure, Temperature.

Data frame:

yy/mm/dd<SP>hh:mm:s.s<SP>p.p<SP>s.s<SP>t.t<CR><LF>		
yy	Year (Unsigned decimal integer – 2 digits)	NOT USED
/	‘ / ’ character	
mm	Month (Unsigned decimal integer – 2 digits)	
/	‘ / ’ character	
dd	Day (Unsigned decimal integer – 2 digits)	
<SP>	Space character	
hh	Hour (Unsigned decimal integer – 2 digits)	NOT USED
:	‘ : ’ character	
mm	Minutes (Unsigned decimal integer – 2 digits)	
:	‘ : ’ character	
s.s	Seconds (Unsigned decimal floating point)	
<SP>	Space character	
p.p	Pressure in dbar (Decimal floating point)	USED
<SP>	Space character	
s.s	Speed of sound in m/s (Decimal floating point)	USED
<SP>	Space character	
t.t	Temperature in °C (Decimal floating point)	NOT USED
<CR>	Carriage Return character	
<LF>	Line Feed character	

Depth is calculated by using the formula described in section 1.5. For this protocol, we expect that a Tare has been applied to the pressure sensor. Hence, we do not compensate atmospheric pressure Pa (10.1325 dbar) before pressure to depth conversion.

MINISVS

Standard: ASCII, Input CTD.

Data frame: This protocol applies to SVP sensors:

- Format 1 and 2 are compatible with the Valeport miniSVS.
- Format 2 is compatible with Applied Micro systems Micro SV-X
- Format 3 is compatible with the Valeport miniSVS with optional pressure sensor.

Both pressure sensor input and sound velocity can be used by Rovins.

Applicable document :

- Valeport miniSVS Operating manual 0650808i.doc page 7 and 8 for format 1, 2 and 3.
- Applied Microsystems Micro SV-X Micro SV User Manual V1.21 page 11 for format 2

Format 1: <F1><F2><CR><LF>				
<F1>	<space>	Header		
<F2>	SSSSSSS	Speed of sound in mm/sec (i.e : 1234567 mm/sec)	Integer	USED

Format 2: <F1><F2><F3><CR><LF>				
<F1>	<space>	Header		
<F2>	SSSS.SSS	Speed of sound in m/sec (i.e : 1234.567 m/sec)	Float	USED

Format 3: <space><F1><space><F2><CR><LF>				
<F1>	PPPP.P (i.e. 1234.5 dbar) (*) PPP.PP (i.e. 123.45 dbar) PP.PPP (i.e. 12.345 dbar)	Pressure value is a fixed length string depending on the range of pressure sensor.	Float	USED
<F2>	SSSSSSS	Speed of sound in mm/sec (i.e : 1234567 mm/sec)	Integer	USED

Depth is output from Valeport SVX-2 in meters only if a Tare has been applied and the latitude has been supplied to the instrument. We are expecting pressure in dbar.

(*) Depth is calculated by using the formula described in section 1.5. For this protocol, we expect that a Tare has been applied to the pressure sensor. Hence, we do not compensate atmospheric pressure Pa (10.1325 dbar) before pressure to depth conversion.

NMEA Standard

Standard: Input ASCII.

Data received: Alert message, position, position SD, time, ground speed.

Data frame is composed of:

- NMEA $--ACK Frame, see section 1
- NMEA $--ACN Frame, see section 1
- NMEA $--GGA Frame, see section 3.1.1
- NMEA $--GLL Frame, see section 3.1.2
- NMEA $--GST Frame, see section 3.1.3
- NMEA $--RMC Frame, see section 3.1.4
- NMEA $--UTC Frame, see section3.2.1
- NMEA $--VBW Frame, see section 3.1.5
- NMEA $--VHW Frame, see section 3.1.6
- NMEA $--ZDA Frame, see section 3.1.7

NORTEK DF21/DF22

Standard: ASCII. NORTEK proprietary protocol.

Data received: Pressure, Temperature, Internal Speed of Sound, Range to Bottom, Figure of Merit, SpeedX, SpeedY, SpeedZ, Delay X, Delay Y, Delay Z.

Data frame:

Unique definition message used for two types of DVL data (ID message) :

- 0x1B: Speed data in Bottom track
- 0x1D: Speed data in Water track

Little Indian (LSB sent first)

Header

Field	Index	Type	Designation	Comment
0	0	UINT 8	Header ID	Fixed value = 0xA5
1	1	UINT 8	Header Size	(NOT USED)
2	2	UINT 8	ID Message. See (1)	0x1B: DVL Bottom Track OR 0x1D: DVL Water Track.
3	3	UINT 8	Family	(NOT USED)
4	4 to 5	UINT 16	Data Size of following record	(NOT USED)
5	6 to 7	UINT 16	Data Checksum	(NOT USED)
6	8 to 9	UINT 16	Header Checksum	(NOT USED)

Ref: Nortek Doc - 8.1 Header definition - Page 39.

Errata: Nortek documentation tag a lot of things as 'signed' on array (8.1) but samples code (8.1.1 & 8.1.2) on the same doc declare it as 'unsigned', use 'unsigned' on our implementation.

UINT 8: Integer 8 bits unsigned / UINT 16: Integer 16 bits unsigned.

DF21 / DF22 Record Data

Field	Index	Type	Designation	Comment
0	10	UINT 8	Version	Fixed value = 1
1	11	UINT 8	Offset of Data	Number of bytes from start of record to start of data
2	12 to 15	UINT 32	Serial number	(NOT USED)
3a	16	UINT 8	Year	(NOT USED)
3b	17	UINT 8	Month	(NOT USED)
4	18	UINT 8	Day. See (2)	(NOT USED)
5	19	UINT 8	Hour. See (2)	LSB = 1 Hour (For NTP mechanism)
6	20	UINT 8	Minute. See (2)	LSB = 1 Minute (For NTP mechanism)
7	21	UINT 8	Seconds. See (2)	LSB = 1 Second (For NTP mechanism)
8	22 to 23	UINT 16	Microsec100	LSB = 100 µsec
9	24 to 25	UINT 16	#Beams. See (3)	Number of Beams 4: All Beams used

Field	Index	Type	Designation	Comment
10	26 to 29	UINT 32	Error	(NOT USED)
11	30 to 33	UINT 32	Status	(NOT USED)
12	34 to 37	UINT 32	Sound of Speed	Speed of Sound in [m/s] Float 32 bits (IEE754)
13	38 to 41	UINT 32	Temperature	(NOT USED)
14	42 to 45	UINT 32	Pressure	Pressure in [Bar] Float 32 bits (IEE754)
15	46 to 61	FLOAT 32	Velocity Beam [0,1,2,3]	(NOT USED)
16	62 to 65	FLOAT 32	Distance Beam 0.	See (4)
17	66 to 69	FLOAT 32	Distance Beam 1.	Range to bottom altitude in [m] Float 32 bits (IEE754) used only on BT mode Range 0 to 65535m Invalid data set to 0.0
18	70 to 73	FLOAT 32	Distance Beam 2.	
19	74 to 77	FLOAT 32	Distance Beam 3.	
20	78 to 81	FLOAT 32	FOM Beam 0	(NOT USED)
	82 to 85	FLOAT 32	FOM Beam 1	(NOT USED)
	86 to 89	FLOAT 32	FOM Beam 2	(NOT USED)
	90 to 93	FLOAT 32	FOM Beam 3	(NOT USED)
21	94 to 97	FLOAT 32	DT1 Beam 0	(NOT USED)
	98 to 101	FLOAT 32	DT1 Beam 1	(NOT USED)
	102 to 105	FLOAT 32	DT1 Beam 2	(NOT USED)
	106 to 109	FLOAT 32	DT1 Beam 3	(NOT USED)
22	110 to 113	FLOAT 32	DT2 Beam 0	(NOT USED)
	114 to 117	FLOAT 32	DT2 Beam 1	(NOT USED)
	118 to 121	FLOAT 32	DT2 Beam 2	(NOT USED)
	122 to 125	FLOAT 32	DT2 Beam 3	(NOT USED)
23	126 to 129	FLOAT 32	Time Vel Est Beam 0	(NOT USED)
	130 to 133	FLOAT 32	Time Vel Est Beam 1	(NOT USED)
	134 to 137	FLOAT 32	Time Vel Est Beam 2	(NOT USED)
	128 to 141	FLOAT 32	Time Vel Est Beam 3	(NOT USED)
24	142 to 145	FLOAT 32	Velocity X. See (5)	Data velocity in Ship coordinate in [m/s] Float 32 bits (IEE754) used on BT & WT mode Invalid data set to -32.768 m/s
25	146 to 149	FLOAT 32	Velocity Y. See (5)	
26	150 to 153	FLOAT 32	Velocity Z1. See (5)	
27	154 to 157	FLOAT 32	Velocity Z2. See (5)	(NOT USED)
28	158 to 161	FLOAT 32	FOM X	Figure of Merit in [m/s] Float 32 bits (IEE754) used on BT & WT mode Range 0 to 10m/s Invalid data set to 10.0
29	162 to 165	FLOAT 32	FOM Y	
30	166 to 169	FLOAT 32	FOM Z1	
31	170 to 173	FLOAT 32	FOM Z2	(NOT USED)

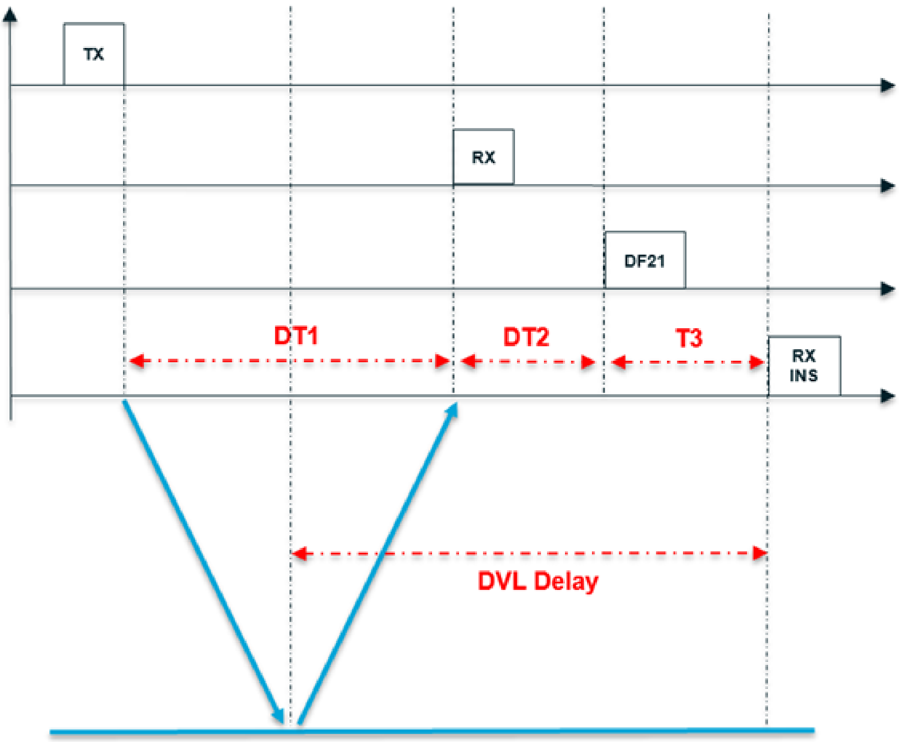
Field	Index	Type	Designation	Comment
32	174 to 177	FLOAT 32	DT1 X. See (6)	Beam data latency in Ship coordinate in [s] Float 32 bits (IEE754) used only in BT mode Range 0.0 to 5.0s Invalid data set to 0.0
33	178 to 181	FLOAT 32	DT1 Y. See (6)	
34	182 to 185	FLOAT 32	DT1 Z1. See (6)	
35	186 to 189	FLOAT 32	DT1 Z2. See (6)	
36	190 to 193	FLOAT 32	DT2 X. See (6)	Computing data latency in Ship coordinate in [s] Float 32 bits (IEE754) used only in BT mode Range 0.0 to 5.0s Invalid data set to 0.0
37	194 to 197	FLOAT 32	DT2 Y. See (6)	
38	198 to 201	FLOAT 32	DT2 Z1. See (6)	
39	202 to 205	FLOAT 32	DT2 Z2. See (6)	
40	206 to 221	FLOAT 32	Time Vel Est Axis	(NOT USED)

Ref: Nortek Doc - 8.3 DF21/DF22 - DVL Bottom Track & Water Track Data Record Definitions - Page 49 to 51.
UINT8: Integer 8 bits unsigned / UINT 16: Integer 16 bits unsigned
UINT32: Integer 32 bits unsigned / FLOAT 32: Float 32 bits using IEEE754 format.

Bit #	Description
0	Beam 1 Velocity Valid
1	Beam 2 Velocity Valid
2	Beam 3 Velocity Valid
3	Beam 4 Velocity Valid
4	Beam 1 Distance Valid
5	Beam 2 Distance Valid
6	Beam 3 Distance Valid
7	Beam 4 Distance Valid
8	Beam 1 Figure of Merit Valid
9	Beam 2 Figure of Merit Valid
10	Beam 3 Figure of Merit Valid
11	Beam 4 Figure of Merit Valid
12	X Velocity Valid - Warning this flag can be set with an out of range value [-32;+32]
13	Y Velocity Valid - Warning there a bug on the vendor documentation. - Warning this flag can be set with an out of range value [-32;+32]
14	Z1 Velocity Valid - Warning this flag can be set with an out of range value [-32;+32]
15	Z2 Velocity Valid - Warning this flag can be set with an out of range value [-32;+32]
16	X Figure of Merit Valid
17	Y Figure of Merit Valid
18	Z1 Figure of Merit Valid
19	Z2 Figure of Merit Valid
20	Less than 3% processing capacity left – Do not monitor reserved for vendors debug.
21	Less than 6% processing capacity left – Do not monitor reserved for vendors debug.
22	Less than 12% processing capacity left – Do not monitor reserved for vendors debug.
28 to 31	Wakeup State:

Bit #	Description
	0010 = break 0011 = RTC alarm 0000 = bad power 0001 = power applied Do not monitor reserved for vendors debug.

- The ID value is used to define if the data is in the Bottom track or Water track mode.
- The Absolute time is applied when the NTP synchronization is set to ON into IHM DVL parameters.
- The DVL shall be configured to ALL beam used. By default: #Beams is set to 4.
- Range to bottom is computed as an average of 4 ranges to bottom given (excluding zero field meaning bad detection).
- iXblue/Nortek Ship Coordinates convention:
  - > INS Forward = XDVL * 0.7071068 + YDVL * 0.7071068
  - > INS Left = XDVL * 0.7071068 - YDVL * 0.7071068
  - > INS Up = - ZDVL.
- The DVL delay is estimated as below:



DVL time = INS time + DT1 / 2 + DT2 + T3

T3 delay is estimated automatically by the INS depend on the communication layer (ethernet or serial, with specific baud rate etc.…)

PAROSCIENTIFIC

Standard: ASCII, Paroscientific pressure sensor.

Data received: Depth calculated using sensor pressure.

Data frame:

***0001x.x<CR><LF>**	
x.x	is the pressure measured in PSI unit (i.e : *000114.5 is 14.5 PSI)

Depth is calculated by using the formula described in section 1.5.

POSIDIONA

Standard: Binary. POSIDIONA

Data received: Transponder number, Vessel Latitude, Vessel Longitude, Vessel Altitude, Latitude standard deviation, Longitude standard deviation, Altitude standard deviation, Delay.

Data frame: The frame contains 10 fields – 32 bytes – MSB are received first.

Message <F0><F1>…			
Field 0	Byte 0	First synchronization byte	Fixed value = 0x24
Field 1	Byte 1	Transponder number	Is a number in the range [0, 255] The firs received transponder number is NOT USED as a second synchronization byte.
Field 2	Bytes 2 to 5	USBL Latitude	Signed 32 bits +/-231 = +/-Pi
Field 3	Bytes 6 to 9	USBL longitude	Signed 32 bits +/-231 = +/-Pi
Field 4	Bytes 10 to 13	USBL Depth	IEEE floating point format, meters
Field 5	Bytes 14 to 17	USBL Latitude std. deviation	IEEE floating point format, meters
Field 6	Bytes 18 to 21	USBL Longitude std. deviation	IEEE floating point format, meters
Field 7	Bytes 22 to 25	USBL Depth std. deviation	IEEE floating point format, meters
Field 8	Bytes 26 to 29	USBL Position delay	IEEE floating point format, seconds
Field 9	Bytes 30 to 31	Checksum	2 characters Addition of all the bytes from 1 to 29.

PRESSURE SENSOR

Standard: ASCII, Comma separated ASCII input protocol with NMEA compliant checksum.

Data received: Pressure, pressure status.

Data frame: This input frame must contain 3 commas delimiters and if some fields are not set properly, the frame is rejected (i.e.: If the pressure field or the status field is a blank field).

$PS,p.p,ss,t.t*hh<CR><LF>		
$PS	Start of sentence : HEX 24 50 53	
,	Data field delimiter : HEX 2C	
p.p	Absolute pressure in bar. This field is parsed as a floating point with at least 1 digit after the comma	USED **Warning: Shall not be a blank field**
,	Data field delimiter : HEX 2C	
ss	Status of the pressure sensor 11 or 10 when pressure valid 01 or 01 when pressure invalid	USED **Warning: Shall not be a blank field**
,	Data field delimiter : HEX 2C	
t.t	Is the temperature in °C	NOT USED **Warning: Blank field allowed**
*	Checksum field delimiter : HEX 2A	
hh	Checksum field (hexadecimal value of the XOR of each character in the sentence, between, but excluding “$” and “*”)	
<CR><LF>	End of sentence : HEX 0D 0A	

RAMSES POSTPRO

Standard: Binary. RAMSES POSTPRO
Data received: LBL, Pressure, Speed of sound, LBL postprocessing
Data frame:

Conventions

Telegram format

The telegram is a combination of sensor blocks. Each block contains header, telegram identification and checksum. Any combination of sensor blocks can be sent at input of INS. Multiple sensor blocks of a kind can be sent (i.e: multiple USBL or LBL beacon positions).
The checksum is the sum of signed bytes of the telegram (telegram length – 2 checksum bytes).All identification values (telegram identification, system type, rejection mode…) are expressed in decimal value otherwise specified.

Time

INS time can be synchronized with GNSS UTC time when UTC time block is sent to INS at regular intervals (i.e: every second). To improve accuracy it is recommended to input a 1 PPS pulse at pulse input of INS. If INS is not time synchronized INS will use the data latency information in sensor data blocks to evaluate age of data. Time in sensor data blocks is reset to 0 every 24 hour.

Data types

Each telegram description uses following convention:

Type name	Description
byte	Unsigned 8 bit integer
character	Signed 8 bit integer
word / ushort	Unsigned 16 bit integer
short	Signed 16 bit integer
dword / ulong	Unsigned 32 bit integer
long	Signed 32 bit integer
float	IEEE Float 32 bits
double	IEEE Float 64 bits

All 16 and 32 bits integers are represented in Big endian convention (MSB sent first). NaN is defined by the following value 0x7FC00000.

Data blocs used by INS

Beacon slam position

Message <F0><F1>…….<F15>				
Field 0	Byte 0	byte	Synchronisation	0x24
Field 1	Byte 1	byte	Telegram size	44
Field 2	Byte 2	byte	Telegram identification	52
Field 3	Byte 3	byte	Data bloc version	0x01
Field 4	Bytes 4 to 7	dword	Data validity time	Time since 01/01/1970 in seconds
Field 5	Bytes 8 to 11	dword	Data validity time	µs
Field 6	Byte 12	byte	Beacon ID Note 1	0 to N (0 for Ramses)
Field 7	Bytes 13 to 16	long	Latitude	+/- 231= +/-180°
Field 8	Bytes 17 to 20	long	Longitude	+/- 231= +/-180°
Field 9	Bytes 21 to 24	float	Immersion	m
Field 10	Byte 25	byte	Position validity Note 2	0 : invalid 1 : in calibration process 2: valid for SLAM but not for INS aiding 3 : valid for SLAM and INS aiding
Field 11	Bytes 26 to 29	float	Latitude std dev.	m
Field 12	Bytes 30 to 33	float	Longitude std dev.	m
Field 13	Bytes 34 to 37	float	Altitude std dev.	m
Field 14	Bytes 38 to 41	float	North/East covariance	m - NOT USED
Field 15	Bytes 42 to 43	word	CRC	unsigned sum of all fields except checksum

Note 1 : Only RAMSES position is used by INS.

Note 2 : INS should only use position when position validity is 3.

Beacon slant range

Message <F0><F1>…….<F15>				
Field 0	Byte 0	byte	Synchronization	0x24
Field 1	Byte 1	byte	Telegran size	42
Field 2	Byte 2	byte	Telegram identification	53
Field 3	Byte 3	byte	Data block version	0x01
Field 4	Bytes 4 to 7	dword	Data validity time	Time since 01/01/1970 in seconds
Field 5	Bytes 8 to 11	dword	Data validity time	µs
Field 6	Byte 12	byte	Beacon ID	1 to N
Field 7	Bytes 13 to 14	word	Acoustic mode	NOT USED
Field 8	Bytes 15 to 18	long	Latitude	+/- 231= +/-180°

Message <F0><F1>…….<F15>				
Field 9	Bytes 19 to 22	long	Longitude	+/- 231= +/-180°
Field 10	Bytes 23 to 26	float	Immersion	m
Field 11	Bytes 27 to 30	float	Slant range	m
Field 12	Bytes 31 to 34	float	Range std dev	m
Field 13	Bytes 35 to 38	float	Measure age	s
Field 14	Byte 39	byte	Measurement validity according to Ramses algorithms **Note 3**	0 : invalid Slam and invalid rough filter 1 : invalid Slam 2 : invalid rough filter 3 : valid distance but invalid for INS 4 : valid distance and valid for INS
Field 15	Bytes 40 to 41	word	CRC	unsigned sum of all fields except checksum

Note 3 : INS should only use beacon ranges when position validity is 4.

Sound velocity and pressure

Message <F0><F1>…….<F11>				
Field 0	Byte 0	byte	Synchronisation	0x24
Field 1	Byte 1	byte	Telegram size	31
Field 2	Byte 2	byte	Telegram identification	55
Field 3	Byte 3	byte	Data bloc version	0x01
Field 4	Bytes 4 to 7	dword	Data validity time	Time since 01/01/1970 in seconds
Field 5	Bytes 8 to 11	dword	Data validity time	µs
Field 6	Byte 12	byte	Device number	(0-5) - NOT USED
Field 7	Bytes 13 to 16	float	Conductivity	mS/cm - NOT USED
Field 8	Bytes 17 to 20	float	Pressure **Note 5**	dbar (NaN if data not available)
Field 9	Bytes 21 to 24	float	Salinity	PSU - NOT USED
Field 10	Bytes 25 to 28	float	Sound velocity **Note 5**	Range]1300 m/s, 1700 m/s [(NaN if data not available)
Field 11	Bytes 29 to 30	word	CRC	unsigned sum of all fields except checksum

Note 5 : If a parameter is not available the field is defined by default to NaN (i.e : only depth sensor connected to RAMSES).

When valid, the pressure is used to compute the depth, and the INS can take it into account if configured to use it.

For pressure to depth conversion refer to section 1.5.

Data blocs broadcasted for post-processing

The blocs which ID decimal value are in the range [50,79] are all broadcasted into the post-processing output protocol.
Only the ones which ID are 52, 53 and 55, are taken into account by the INS.

RDI PD0

Standard: Binary.

Data received: Bottom Track velocities (Transverse velocity, Longitudinal velocity, Vertical velocity), Water Track velocities (Transverse velocity, Longitudinal velocity, Vertical velocity), Range to Bottom (given for each of 4 beams), Speed of Sound, Depth (calculated with Pressure).

Data frame:

PD0 Standard Output Data Buffer Format

Always Output	**HEADER** (6 bytes +[2 bytes * Nb of DATA TYPES])
	FIXED LEADER DATA (53 or 59 bytes)
	VARIABLE LEADER DATA (65 bytes)
WD-command **WP-command**	**VELOCITY** (2 bytes + 8 bytes per Depth cell)
	CORRELATION MAGNITUDE (2 bytes + 4 bytes per Depth cell)
	ECHO INTENSITY (2 bytes + 4 bytes per Depth cell)
	PERCENT GOOD (2 bytes + 4 bytes per Depth cell)
BP-command	**BOTTOM TRACK DATA** (85 bytes)
Always Output	**RESERVED** (2 bytes)
	CHECKSUM (2 bytes)

Header

Message <F0><F1>….<F10><F11>			
Field 0	Byte 0	Header ID	Fixed value = 0x7F
Field 1	Byte 1	Data Source ID	Fixed value = 0x7F
Field 2	Bytes 2 to 3	No. of Bytes in ensemble excluding 2 bytes checksum (N)	(NOT USED)
Field 3	Byte 4	Spare	(NOT USED)
Field 4	Byte 5	Number of Data Type	8 bits unsigned
Field 4+1	Bytes 6 to 7	Offset For Data Type #1	16 bits unsigned Offset of each Data Type in Global Frame
Field 4+2	Bytes 8 to 9	Offset For Data Type #2	
…	…	…	
Field 4+N	Bytes 4+2*N to 5+2*N	Offset For Data Type #N	

Data Type corresponds to Fixed Leader, Variable Leader, Velocity, Correlation Magnitude, Echo Intensity, Percent Good and Bottom Track Data.

Fixed Leader Data

Message <F0><F1>….<F10><F11>			
Field 0	Bytes 0 to 1	Fixed Leader ID	Fixed value = 0x00 0x00
Field 1	Bytes 2 to 24	(NOT USED)	(NOT USED)
Field 2	Byte 25	Coordinate Transform	xxx00xxx = Beam coordinates xxx01xxx = Instrument Coordinate xxx10xxx = Ship coordinate (*)
Field 2	Bytes 26 to 52/58	(NOT USED)	(NOT USED)

(*) DVL must be set in beam, instrument or ship coordinates when used with Rovins.

Variable Leader Data

Message <F0><F1>….<F10><F11>			
Field 0	Bytes 0 to 1	Variable Leader ID	Fixed value = 0x80 0x00
Field 1	Bytes 2 to 13	**Note 1**	(NOT USED)
Field 2	Bytes 14 to 15	Sound of Speed	LSB = 1m/s range 1400 to 1600 m/s
Field 3	Bytes 16 to 17	Depth of Transducer	(NOT USED)
Field 4	Bytes 18 to 47	**Note 1**	(NOT USED)
Field 5	Bytes 48 to 51	Pressure: Water pressure relative to one atmosphere. See section 1.5.	LSB = 1 deca-pascal, range 0 to 4294967295 deca-pascal
Field 6	Bytes 52 to 55	Pressure variance	LSB = 1 deca-pascal, range 0 to 4294967295 deca-pascal
Field 7	Bytes 56 to 64	**Note 1**	(NOT USED)

Note 1: Data may be added to an existing data type only by adding the bytes to the end of the data format.
As an example, the variable leader data contains information on ensemble number, time, heading, pitch, roll, temperature, pressure, etc.
The format for the bytes 0-52 is now specified by changes added in support to the Navigator DVL.
If additional sensor data is to be added to the variable leader data then it must be added to the end of the data string (bytes 53-x as an example).

Velocity Data Format

Message <F0><F1>….<F10><F11>			
Field 0	Bytes 0 to 1	Velocity ID	Fixed value = 0x00 0x01
Field 1	Bytes 2 to 1+8*DepthCellNb	See section 1.5	(NOT USED)

Correlation Magnitude, Echo Intensity and Percent Good Data

Message <F0><F1>….<F10><F11>			
Field 0	Bytes 0 to 1	Correlation Magnitude ID OR Echo Intensity ID OR Percent-Good ID	Fixed value = 0x00 0x02 OR 0x00 0x03 OR 0x00 0x04
Field 1	Bytes 2 to 1+4*DepthCellNb	See section 1.5	(NOT USED)

Bottom Track Data

Message <F0><F1>….<F10><F11>			
Field 0	Bytes 0 to 1	Bottom Track ID	Fixed value = 0x00 0x06
Field 1	Bytes 2 to 15	See section 1.1.5	(NOT USED)
Field 2	Bytes 16 to 17	BEAM#1 Range To Bottom.	Unsigned integer LSB = 1cm, range 0 to 65535cm (0 when bottom detection is bad) (**)
Field 3	Bytes 18 to 19	BEAM#2 Range To Bottom.	
Field 4	Bytes 20 to 21	BEAM#3 Range To Bottom.	
Field 5	Bytes 22 to 23	BEAM#4 Range To Bottom.	
Field 6	Bytes 24 to 25	BEAM#1 Bottom Track Velocity	Instrument or ship coordinates: X,Y and Z-axis BT velocities Beam coordinates: Beams 1 to 4 BT velocities (***) Signed integer: LSB = 1mm/s, Range -32766 to 32767 mm/s 0x8000 when not received
Field 15	Bytes 58 to 84	See section 1.5	(NOT USED)

(**) Range to bottom used is an average of the 4 ranges to bottom given for each beam, excluding zero fields meaning bad detection.

(***) The meaning of the velocity depends on the EX (coordinate system) command setting. The four velocities are as follows:
a) Instrument Coordinates: 1->2, 4->3, toward face, error
b) Ship Coordinates: Starboard, Fwd, Upward, Error
c) Beam Coordinates: Beam1, Beam2, Beam3, Beam4
The PD0 speed sign is opposite to PD6 protocol so the sign is inverted before sending data to the INS algorithm. So the sign convention of speed described by the above figure applies.

(****) Depth is calculated by using the formula described in section 1.5. For this protocol, we expect that a Tare has been applied to the pressure sensor. Hence, we do not compensate atmospheric pressure Pa (10.1325 dbar) before pressure to depth conversion.

Checksum

Message <F0><F1>….<F10><F11>			
Field 0	Bytes 0 to 1	Checksum Data	Sum of all frame bytes excluding checksum (modulo 65535)

RDI PD3 AND RDI PD3 RT

Standard: Binary. DVL Input Data.
Data received: Bottom Track velocities (Transverse , longitudinal and vertical velocity), Water Track velocities (Transverse , longitudinal and vertical velocity), Range to Bottom.
Data frame: The frame contains at most 22 fields – 57 bytes LSB First.

Message <F0><F1><F2>… <F21>			
Field 0	Byte 0	DVL Data ID	Fixed value = 0x7E
Field 1	Byte 1	Data to Follow status Bit #0 System Coordinates Bit #1 Vertical velocities Bit #2 Water Reference velocities Bit #3 Range To Bottom (4 beams) Bit #4 Range To Bottom (average) Bit #5 Not used Bit #6 Not used Bit #7 Sensor/Other Data	If Bit #0 of “Data to Follow” status is set to 1, then Earth coordinates are used. If Bit #0 of “Data to Follow” status is set to 0, then Ship coordinates are used. The INS will reject the message if Earth coordinates are used i.e. Bit#0 set to 1. The DVL must be set to send data in the Ship coordinates. For the RDI_PD3 input, the DVL Roll and Pitch compensations must be disabled (EX100xx DVL command) For the RDI_PD3_RT input, the DVL Roll and Pitch compensations must be enabled (EX101xx DVL command)
Field 2	Bytes 2-3	X Bottom Track transverse **Note 1**	This field is always received 16 bits signed integer LSB = 1mm/s. + for ship motion to Starboard. USED
Field 3	Bytes 4-5	Y Bottom Track longitudinal velocity **Note 1**	This field is always received 16 bits signed integer LSB = 1mm/s. + for ship motion to Forward. USED
Field 4	Bytes 6-7	Z Bottom Track vertical velocity **Note 1**	This field is received if Bit #1 of “Data to Follow” status is set. If this field is not received, the INS will take into account the value of 0 m/s on the vertical velocity. 16 bits signed integer LSB = 1 mm/s. USED
Field 5	Bytes 8-9	X Water Track transverse velocity **Note 1**	This field is received if Bit #2 of “Data to Follow” status is set. 16 bits signed integer LSB = 1 mm/s. USED
Field 6	Bytes 10-11	Y Water Track longitudinal velocity Note 1	This field is received if Bit #2 of “Data to Follow” status is set. 16 bits signed integer LSB = 1 mm/s. USED

Message <F0><F1><F2>… <F21>			
Field 7	Bytes 12-13	Z WaterTrack vertical velocity **Note 1**	This field is received if Bit #1 and Bit#2 of “Data to Follow” status are set If this field is not received, the INS will take into account the value of 0 m/s on the vertical velocity. 16 bits signed integer LSB = 1 mm/s. USED

Note 1: Positive values indicate vessel motion to (X) Starboard/East, (Y) Forward/North, (Z) Upward.

If the value is –32768 (0x8000), the value is not valid:
i.e : When X,Y or Z Bottom Track velocity value = 0x8000, the INS will ignore X,Y and Z Bottom Track input velocities.

When X,Y or Z Water Track velocity value = 0x8000, the INS will ignore X,Y and Z Water Track input velocities.

Message <F0><F1><F2>… <F21>			
Field 8	Bytes 14-15	Beam 1 Range to Bottom	These fields are received if Bit #3 of “Data to Follow” status is set. 16 bits unsigned integer LSB = 1 cm; Range = 0 to 65535 cm When a bottom detection is bad, the field is set 0. USED only if Average Range to Bottom (next field) is not received
Field 9	Bytes 16-17	Beam 2 Range to Bottom	
Field 10	Bytes 18-19	Beam 3 Range to Bottom	
Field 11	Bytes 20-21	Beam 4 Range to Bottom	
Field 12	Bytes 22-23	Average Range to Bottom	This field is received if Bit #4 of “Data to Follow” status is set. It contains the average vertical range to bottom as determined by each beam. 16 bits unsigned integer LSB = 1 cm; Range = 0 to 65535 cm When a bottom detection is bad, the field is set 0. USED
Field 13	Bytes 24-39	16 bytes spare	NOT USED
Field 14	Byte 40	Sensor/Other Data status Bit #0 Time Bit #1 Heading Bit #2 Pitch Bit #3 Roll Bit #4 Temperature Bit #5 Active Built-In-Test Bit #6 Not used Bit #7 Not used	This field is received if Bit #7 of “Data to Follow” status is set NOT USED

Message <F0><F1><F2>… <F21>			
Field 15	Byte 41	Hours	PING Time. These fields are received if Bit #0 of “Sensor/Other Data” status is set. NOT USED
	Byte 42	Minutes	
	Byte 43	Seconds	
	Byte 44	Hundredth of seconds	
Field 16	Bytes 45-46	Heading	These fields are received if Bit #1 of “Sensor/Other Data” status is set. NOT USED
Field 17	Bytes 47-48	Pitch	These fields are received if Bit #2 of “Sensor/Other Data” status is set. NOT USED
Field 18	Bytes 49-50	Roll	These fields are received if Bit #3 of “Sensor/Other Data” status is set. NOT USED
Field 19	Bytes 51-52	Temperature	These fields are received if Bit #4 of “Sensor/Other Data” status is set. NOT USED
Field 20	Bytes 53-54	Built-In-Test result	These fields are received if Bit #5 of “Sensor/Other Data” status is set. NOT USED
Field 21	Bytes 55-56	Checksum	Sum of all the bytes excluding checksum (modulo 65535)

RDI PD4

Standard: Binary.

Data received: Bottom Track velocities (Transverse velocity, Longitudinal velocity, Vertical velocity) Water Track velocities (Transverse velocity, Longitudinal velocity, Vertical velocity) Range to Bottom (given for each of 4 beams) Speed of Sound.

Data frame: 12 fields – 47 bytes - LSB received first.

Message <F0><F1>….<F10><F11>			
Field 0	Byte 0	Input identification	Fixed value = 0x7D
Field 1	Byte 1	PD4 input frame	Fixed value = 0x00
Field 2	Bytes 2 to 3	No. of Bytes	(NOT USED)
Field 3	Byte 4	System configuration (***)	010xxxxx (instrument coordinate velocities) Or 100xxxxx (ship coordinate velocities) The INS is expecting bit 6 = 0 (no tilt sensor used) and any frame excepted earth coordinates mode. Otherwise, DVL frame is rejected.
Field 4	Bytes 5 to 6	XDVL Bottom track transverse velocity (*)	16 bits signed integer; LSB = 1mm/s
Field 5	Bytes 7 to 8	YDVL Bottom track longitudinal velocity (*)	16 bits signed integer; LSB = 1mm/s
Field 6	Bytes 9 to 10	ZDVL Bottom track vertical velocity (*)	16 bits signed integer; LSB = 1mm/s
Field 7	Bytes 11 to 12	Not used	(NOT USED)
Field 8	Bytes 13 to 14	Beam1 range to bottom(**)	For each measure: 16 bits unsigned integer Range 0 to 65535 cm (LSB=1cm) (bad detection: field set to 0)
Field 9	Bytes 15 to 16	Beam2 range to bottom(**)	
Field 10	Bytes 17 to 18	Beam3 range to bottom(**)	
Field 11	Bytes 19 to 20	Beam4 range to bottom(**)	
Field 12	Byte 21	Not used	(NOT USED)
Field 13	Bytes 22 to 23	XDVL Water track transverse velocity (*)	16 bits signed integer; LSB = 1mm/s
Field 14	Bytes 24 to 25	YDVL Water track longitudinal velocity (*)	16 bits signed integer; LSB = 1mm/s
Field 15	Bytes 26 to 27	ZDVL Water track vertical velocity (*)	16 bits signed integer; LSB = 1mm/s
Field 16	Bytes 28 to 40	Not used	(NOT USED)

Message <F0><F1>….<F10><F11>			
Field 17	Bytes 41 to 42	Speed of Sound	Manual or calculated speed of sound 16 bits unsigned integer, LSB = 1m/s Range from 1300 to 1700m/s
Field 18	Bytes 43 to 46	Not used	(NOT USED)

(*) If one of those velocity value is –32768 (0x8000), the input frame is not valid and rejected by the Rovins.

Depending on DVL settings, XDVL, YDVL and ZDVL may be output in instrument frame or ship frame.

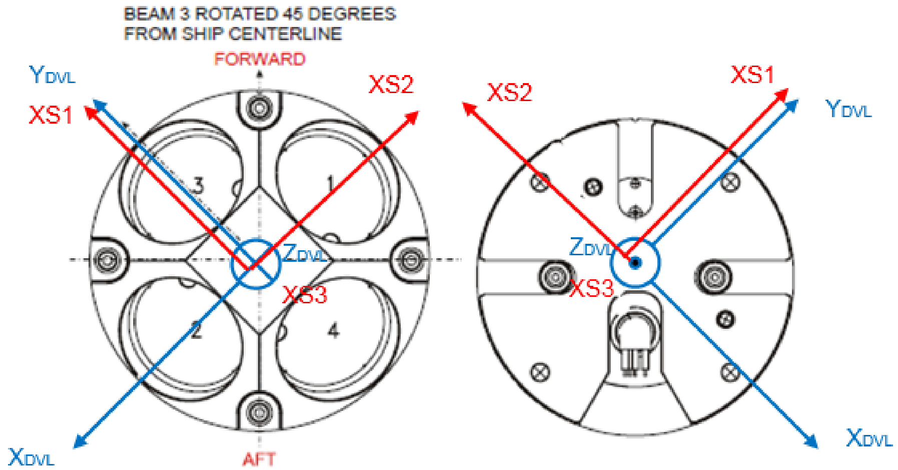
In both cases, Rovins will rotate the DVL input by 90°, applying following rule:

XS1 = YDVL

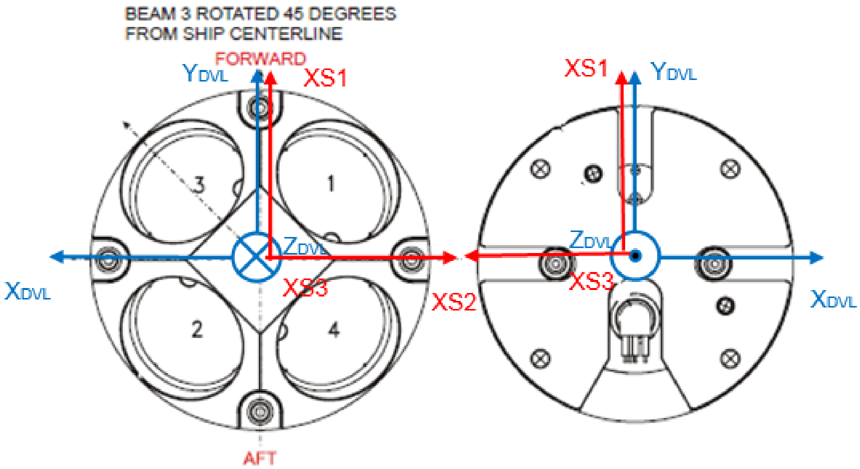
XS2 = -XDVL

XS3 = ZDVL

The diagram below illustrates DVL raw speed and Rovins rotated speed when DVL is set in instrument frame, before Rovins transformation (blue) and after Rovins rotation (red):



The diagram below illustrates DVL raw speed and Rovins rotated speed when DVL is set in ship frame in case the DVL EA field is set to 45° to get aligned with the Subsea vehicle, before Rovins transformation (blue) and after Rovins rotation (red):



(**) Range to bottom used is an average of the 4 ranges to bottom given for each beams, excluding zero fields meaning bad detection

(***) DVL must be set to send data in instrument body frame or ship frame. Misalignment calibration between Subsea vehicle and DVL body frames must be performed for optimal results. DVL Tilt sensor must not be used to compensate from attitude: this will be performed by Subsea vehicle.

The default standard deviation is set to 0.2 m/s and can be customized through the advanced filtering page from the Web-Based Graphical User Interface. Refer to the Rovins Installation & Setup Guide to get information about the advanced filtering page.

RDI PD6

Standard: ASCII.

Data received: DVL bottom track and water track speeds, DVL sound velocity, DVL altitude, DVL range to bottom, DVL depth (when pressure sensor. option).

An additional time stamp telegram can be received (:UT) and decoded. This latter “UT” telegram is not in RDI PD6 telegram but can be supplied by client application.

Data frame:

Time Stamping Of Bottom Track Speed

:UT,hhmmss.sss<CR><LF>	
hhmmss.sss	UTC of the DVL bottom track measurement (:BI telegram)

System Attitude Data (Not Used)

:SA,±PP.PP,±RR.RR,HH.HH<CR><LF>	
PP.PP	is the pitch in degrees. (NOT USED)
RR.RR	is the roll in degrees. (NOT USED)
HH.HH	is the heading in degrees. (NOT USED)

Timing And Scaling Data

This frame is used to set the date of BI and WI frame info.

:TS,YYMMDDHHmmsshh,ss.s,+TT.T,DDDD.D,CCCC.C,BBB<CR><LF>	
YYMMDDHHmmsshh	Is the date: year, month, day, hour, minute, second, hundredths of seconds. (NOT USED)
ss.s	Is the salinity in parts per thousand [ppt]. (NOT USED)
+TT.T	Is the temperature in °C. (NOT USED)
DDDD.D	Is the depth of transducer face in meters.
CCCC.C	Is the DVL manual or calculated sound velocity. (NOT USED)
BBB	Is the Built-In test (BIT) result code. (NOT USED)

Water-Mass, Instrument-Referenced Velocity Data

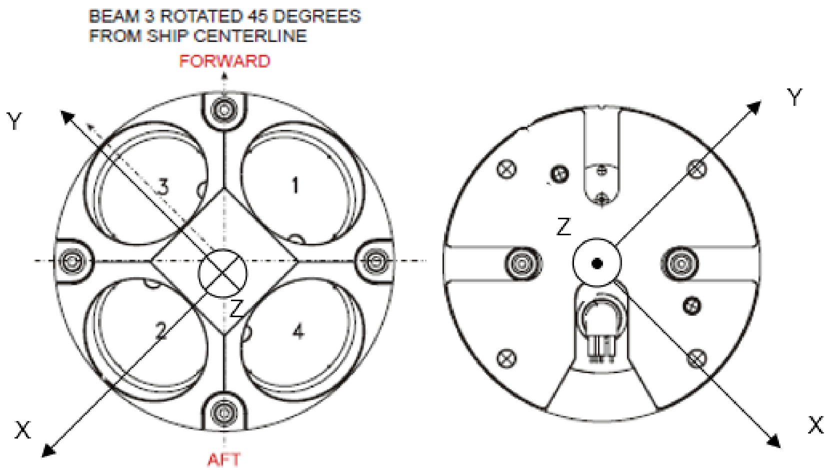
:WI,±TTTTT,±LLLLL,±NNNNN,±MMMMM,S<CR><LF>	
TTTTT	is the X transverse velocity relative to current, data in mm/s (*)
LLLLL	is the Y longitudinal velocity relative to current, data in mm/s (*)
NNNNN	is the Z normal velocity relative to current, data in mm/s (*)
MMMMM	Error velocity data in mm/s (NOT USED)
S	is the status 'A': Valid (**)

Bottom-Track, Instrument-Referenced Velocity Data

:BI,±TTTTT,±LLLLL,±NNNNN,±MMMMM,S<CR><LF>	
TTTTT	is the X transverse velocity data in mm/s (*)
LLLLL	is the Y longitudinal velocity data in mm/s (*)
NNNNN	is the Z normal velocity data in mm/s (*)
MMMMM	Error velocity data in mm/s (NOT USED)
S	is the status, 'A': Valid. (**)

Bottom-Track, Earth-Referenced Distance Data

: BD,±EEEEEEEE.EE,±NNNNNNNN.NN,±UUUUUUUU.UU,DDDD.DD,TTT.TT <CR><LF>	
EEEEEEEE.EE	is the East distance data in meters. (NOT USED)
NNNNNNNN.NN	is the North distance data in meters. (NOT USED)
UUUUUUUU.UU	is the upward distance data in meters. (NOT USED)
DDDD.DD	Range to bottom in meters
TTT.TT	is the time since the last good-velocity estimate in seconds. (NOT USED)



(*) DVL reference frame in instruments coordinate before Rovins coordinate transformation (X=Y; Y=-X, Z=Z):

±XXXXX = X-axis velocity data in mm/s (+ = Bm1 Bm2 xdcr movement relative to bottom)

±YYYYY = Y-axis velocity data in mm/s (+ = Bm4 Bm3 xdcr movement relative to bottom)

±ZZZZZ = Z-axis velocity data in mm/s (+ = transducer movement away from bottom)

(**) A value different than 'A' indicates an invalid data transmitted by sensor. In such case, the data is considered as being invalid by Rovins.
(***) DVL must be set to send data in instrument body frame. Misalignment calibration between Rovins and DVL body frames must be performed for optimal results.
Contact support@ixblue.com to retrieve the proper calibration procedure.

SBE 37SI

Standard: ASCII.

Data received: Only pressure converted to depth in INS and Sound Velocity is taken into account..

Data frame: The following describes both protocol formats chosen to interface to a Seabird SBE 37-SI MicroCAT CTD probe.

Note: psu= salinity unit. 1 g of Na+Cl- per 1 kg of sea water.

Format 1: **ttt.tttt,cc.ccccc,(pppp.ppp,)dddd.ddd,sss.ssss,vvvv.vvv,rrr.rrrr,dd mmm yyyy,hh:mm:ss**		
ttt.tttt	Temperature (°C, UTS-90)	(NOT USED)
cc.ccccc	Conductivity (S/m)	(NOT USED)
pppp.ppp	Pressure (dbars)	See Note above
dddd.ddd	Depth (m)	CTD setting must be OutputDepth=Y (NOT USED)
sss.ssss	Salinity (psu)	CTD setting must be OutputSal=Y (NOT USED)
vvvv.vvv	Sound Velocity (m/s)	CTD Setting must be OutputSV=Y
rrr.rrrr	Density sigma (kg/m3)	CTD setting must be OutputDensity=Y (NOT USED)
dd mmm yyyy	Date	CTD setting must be OutputDate=Y (NOT USED)
hh:mm:ss	Time	CTD setting must be OutputTime =Y (NOT USED)

Format 2: **ttt.tttt,cc.ccccc, (pppp.ppp,) dddd.ddd, sss.ssss,vvvv.vvv, rrr.rrrr, mm-dd-yyyy, hh:mm:ss**		
ttt.tttt	Temperature (°C, UTS-90)	(NOT USED)
cc.ccccc	Conductivity (S/m)	(NOT USED)
pppp.ppp	Pressure (dbars)	See Note above
dddd.ddd	Depth (m)	CTD setting must be OutputDepth=Y (NOT USED)
sss.ssss	Salinity (psu)	CTD setting must be OutputSal=Y (NOT USED)
vvvv.vvv	Sound Velocity (m/s)	CTD Setting must be OutputSV=Y
rrr.rrrr	Density sigma (kg/m3)	CTD setting must be OutputDensity=Y (NOT USED)
mm-dd-yyyy	Date	CTD setting must be OutputDate=Y (NOT USED)
hh:mm:ss	Time	CTD setting must be OutputTime =Y (NOT USED)

Alternate supported formats

Format 3: **ttt.tttt,cc.ccccc, (pppp.ppp,) dddd.ddd, sss.ssss,vvvv.vvv,N,N,N**		
ttt.tttt	Temperature (°C, UTS-90)	(NOT USED)
cc.ccccc	Conductivity (S/m)	(NOT USED)
pppp.ppp	Pressure (dbars)	See Note 1
dddd.ddd	Depth (m)	CTD setting must be OutputDepth=Y (NOT USED)
sss.ssss	Salinity (psu)	CTD setting must be OutputSal=Y (NOT USED)
vvvv.vvv	Sound Velocity (m/s)	CTD Setting must be OutputSV=Y
N	Density sigma (kg/m3)	CTD setting must be OutputDensity=N (NOT USED)
N	Date	CTD setting must be OutputDate=N (NOT USED)
N	Time	CTD setting must be OutputTime =N (NOT USED)

Format 4: **ttt.tttt,cc.ccccc,pppp.ppp**		
ttt.tttt	Temperature (°C, UTS-90)	See Note 2
cc.ccccc	Conductivity (S/m)	
pppp.ppp	Pressure (dbars)	

Note 1: When there is no pressure sensor installed, the pressure field may be not a part of the input sentence or may be a blank field. When valid, the pressure is used to compute the depth, and Rovins can take it into account if configured to use it.
Depth is calculated by using the formula described in section 1.5.

Note 2: When only conductivity, pressure and temperature are available, conductivity ratio R needs to be converted to salinity to compute sound velocity using salinity and temperature (this for the Chen and Millero equation which expects pressure, temperature and salinity as inputs).
Sound velocity is calculated by using the formula described in 1.6.

SBE 49

Standard: ASCII.
Data frame: The following describes format to interface to a CTD Sensor SBE49.

ttt.tttt,<space>cc.ccccc,<space>pppp.ppp,<space>sss.ssss,<space>vvvv.vvv<CR><LF>

where:	ttt.tttt	Temperature (°C, ITS-90)	NOT USED
	cc.ccccc	Conductivity (S/m)	NOT USED
	pppp.ppp	Pressure (dbars)	USED. See NOTE 1
	sss.ssss	Salinity (psu)	NOT USED. CTD setting must be OutputSal=Y
	vvvv.vvv	Sound Velocity (m/s)	See NOTE 2. CTD Setting must be OutputSV=Y

Data is expected in the order listed, with a comma followed by a space between each parameter. Shown with each parameter are the number of digits and the placement of the decimal point. Leading zeros are suppressed, except for one zero to the left of the decimal point.
Example of FastCAT output with OutputSal=Y, OutputSV=Y:
23.7658, 0.00019, 0.062, 0.0125, 0.456

- Temperature = 23.7658 (NOT USED by the INS)
- Conductivity = 0.00019 (NOT USED by the INS)
- Pressure = 0.062
- Salinity = 0.0125 (NOT USED by the INS)
- Sound velocity = 0.456

NOTE 1: Depth is calculated by using the formula described in section 1.5.
NOTE 2: Sound velocity is calculated by using the formula described in 1.6.

SEAKING 700

Standard: ASCII, Tritech seaking700 Free Run specific protocol.
Data received: Sound velocity, Depth.
Data frame: See Tritech system documentation for full details on this frame.

Rovins will decode a message in the following format:
%D<SlotReplyHdr><Bathymetric System Reply Data><CR><LF> where SlotReplyHdr describes the data format of the Bathymetric System Reply Data. The data protocol is Winson raw or Winson processed data format.

SEAKING 700 message

%D<SlotReplyHdr><Bathymetric System Reply Data><CR><LF>

<SlotReplyHdr> item

<F0><F1><F2><F3><F4>			
Field	ASCII Hex Format	Data Description	
0	0000 to FFFF	Total Number of Bytes in Message (including header and terminators)	NOT USED
1	01 to 0C	Slot Number (not used)	NOT USED
2	00 to 63	Generic Device Type (not used)	NOT USED
3	0	Data Reply Mode is ASCII (0=ASCII Decimal, 1=ASCII Hex, 2=Binary, 3=CSV*) Only ASCII Decimal or CSV mode are processed by the INS.	USED
4	1	Data sent in Raw or processed data format (SeaKing Long=3, SeaKing Short=2, Raw data=1, Processed Data=0) Only Raw and processed data format are processed by the INS.	USED

* Comma Separated ASCII Values (at most 14 commas) :
Data from SCU can be sent in ASCII mode with each field separated by comma delimiters. Numeric data is represented in Decimal ASCII format although not following the exact number of characters as defined by the DATA TYPE for ASCIIText mode.

For example, in ASCIIText mode, the Integer value –128 will be represented as '–00128', as defined by the DATA TYPE. In CSV mode this field would read as 'xx,-128,xx' (shown as part of comma delimited string). Only the required number of characters that will represent the ASCII value are used in each case.

<Bathymetric System Reply Data> item

(identical structure for Raw or Processed mode, only Altimeter data differs)

when field 3 in <SlotReplyHdr> item is set to ASCII Decimal mode: ±aaaaabbbbbbbbbb±c-ccccddddddddddeeeeeeeeee±fffffggggg±hhhhhiiiiijjjjj±kkkkkkkkkklll ±mmmmmmmmmmnnnnnnnn
when field 3 in <SlotReplyHdr> item is set to CSV mode:
,±aaaaa,bbbbbbbbbb,±ccccc,dddddddddd,eeeeeeeeee,±fffff,ggggg,±hhhhh,iiiii,jjjjj, ±kkkkkkkkkk,lll,±mmmmmmmmmm,nnnnnnnn

Field	Data Description	Data Range	
±aaaaa	Internal temperature in tenths of a degree centigrade	-00200 to +00500	NOT USED
bbbbbbbbbb	Digiquartz pressure in 100,000ths of a PSIa	0000000000 to 100000000	NOT USED
±ccccc	Digiquartz temperature in 1/100ths of a degree centigrade	-05400 to +10700	NOT USED
dddddddddd	Raw digiquartz pressure reading is the number of 8MHz counts for 10,000 digiquartz pulses	000000000 to 0010000000	NOT USED
eeeeeeeeee	Raw digiquartz temperature reading is the number of 8MHz counts for 40,000 digiquartz pulses	000000000 to 0010000000	NOT USED
±fffff	Local oscillator calibration coefficient in Hz	-00500 to +00500	NOT USED
ggggg	Conductivity in µSiemens per centimetre	00000 to 65000	NOT USED
±hhhhh	Conductivity probe temperature in hundredths of a degree centigrade	-01000 to +05000	NOT USED
iiiii	Salinity in parts per 1,000,000 calculated from Conductivity readings	00000 to 65535	NOT USED
jjjjj	Velocity of Sound in metres per second * 10 calculated from Conductivity readings (or 'Manual' VOS if no CT probe)	14000 to 15500	USED
±kkkkkkkkkk	Altimeter (return path) reading in clicks of 200nsecs (This value does not include 'Altimeter Position offset'. For raw mode and includes 'Altimeter Position offset' for processed mode)	+0000000000 to +0000203390	NOT USED
lll	Bathymetric system devices Bit 0 = 1 = Digiquartz valid Bit 1 = 1 = Conductivity valid	000 to 063 INS takes into account:	USED

when field 3 in <SlotReplyHdr> item is set to ASCII Decimal mode: ±aaaaabbbbbbbbbb±c-ccccddddddddddeeeeeeeeee±fffffggggg±hhhhhiiiiijjjjj±kkkkkkkkkklll ±mmmmmmmmmmnnnnnnnn when field 3 in <SlotReplyHdr> item is set to CSV mode: ,±aaaaa,bbbbbbbbbb,±ccccc,dddddddddd,eeeeeeeeee,±fffff,ggggg,±hhhhh,iiiii,jjjjj, ±kkkkkkkkkk,lll,±mmmmmmmmmm,nnnnnnnn		
	Bit 2 = 1 = Altimeter valid Bit 3 = 1 = Internal temperature valid (only installed in SK701 Bathy) Bit 4 = 1 = Velocity of sound calculation valid Bit 5 = 1 = Salinity calculation valid Ex. : Digiquartz valid lll = 001 Digiquartz & Conductivity valid lll = 003 Digiquartz & Altimeter valid lll = 005 Digiquartz, Conductivity & Altimeter valid lll = 007	the sound velocity when bit 4 is set to 1 the depth when bit 0 is set to 1
±mmmmmmmmmm	Depth in millimetres (This value does not include 'Bathy Position Offset' and 'Bathy Zero Offset' for raw mode and includes these corrections for processed mode)	+0000000000 to +0000700000 USED
nnnnnnnn	Time at Start of Scan	00000000 to 23595999 USED

Example:
Internal temperature = 5 degrees = 50
Digiquartz pressure = 200 PSIa = 20000000
Digiquartz temperature = 5 degrees = 500
Raw digiquartz pressure reading = 2135648 = 2135648
Raw digiquartz temperature reading = 1986497 = 1986497
Local oscillator calibration = -10 Hz = -10
Conductivity = 40 mS/cm = 40000
Conductivity temperature = 5 degrees = 500
Conductivity Salinity = 3.4 pts/1000 = 3400
Velocity of Sound = 1475 metres per second = 14750
Altimeter reading = 24 metres = 162710 (return path)
Bathymetric system devices = SK704 (CTDA) = 55
Depth in millimetres = 136.921 metres = 136921
Time in HHMMSSCC = 09:45:33:74 = 09453374
ASCIIText = "+000500020000000+00500000213564800019864970-0001040000

+005000340014750+0000162710055+000013692109453374”

Depth is calculated by using the formula described in section 1.5.

SVP 70

Standard: ASCII, Protocol output by the SVP 70 Reson sensor.
Data received: Only Sound velocity, Pressure will be used by Rovins.
Data frame:

$SVP70,MMMMMM,SSSS.SSS,tt.t,PPP.P,aaa,n,f,v<CR><LF>	
MMMMMM	Time in milliseconds since the switch on (NOT USED)
SSSS.SSS	Speed of sound in m/s
tt.t	Temperature in Celsius degree (NOT USED)
PPP.P	Pressure in tenth of bar
aaa	Signal strength (NOT USED)
n	Approximate signal noise (NOT USED)
f	Filter type (NOT USED)
v	Last sample validity, 1 is OK , 0 is NOK

(*) Depth is calculated by using the formula described in section 1.5.
For this protocol, we expect that a Tare has been applied to the pressure sensor. Hence, we do not compensate atmospheric pressure Pa (10.1325 dbar) before pressure to depth conversion.

SVX2

Standard: ASCII, Protocol output by Valeport MIDAS SVX2 SVP sensor.
Data received: Sound velocity and Pressure will be used by Rovins.
Data frame: Format 1 and 2 are compatible with Valeport MIDAS SVX2. Both pressure sensor input and sound velocity can be used by Rovins.

Format 1: <F1><TAB><F2><TAB><<F3><TAB><F4><TAB><F5><TAB><F6><TAB><F7><TAB><F8><TAB> <CR><LF>			
<F1>	Sound velocity	Float	**(USED)**
<F2>	Units = “M/SEC” or “F/S”	String	**(USED)**
<F3>	Depth	Float	**(USED)**
<F4>	Units= “M” or “DBAR” or “F” (*) (**)	String	**(USED)**
<F5>	Temperature	Float	**(NOTUSED)**
<F6>	Units=”C”	Sting	**(NOT USED)**
<F7>	Conductivity	Float	**(NOT USED)**
<F8>	Units=”MS/CM”	String	**(NOT USED)**

Format 1 string sample :
1483.576 M/SEC 0010.225 DBAR 0020.215 C -000.002 MS/CM

Format 2: <F1><TAB><F2><TAB><<F3><TAB><F4><TAB><F5><TAB><F6><TAB><F7><TAB><F8><TAB> <F9><TAB>><F10><TAB><CR><LF>			
<F1>	Sound velocity	Float	(USED)
<F2>	Units = “M/SEC” or “F/S”	String	(USED)
<F3>	Depth	Float	(USED)
<F4>	Units= “M” or “DBAR” or “F” (*)	String	(USED)
<F5>	(**)	Float	(NOTUSED)
<F6>	Temperature	Sting	(NOT USED)
<F7>	Units=”C”	Float	(NOT USED)
<F8>	Conductivity	String	(NOT USED)
<F9>	Units=”MS/CM”	Float	(NOT USED)
<F10>	Salinity	String	(NOT USED)
	Units=”PSU”		

Format 2 string sample :
1483.576 M/SEC 0010.225 DBAR 0020.200 C 0000.000 MS/CM 0000.000 PSU
(*) Deph is output from Valeport SVX2 in meters only if a Tare has been applied and the latitude has been supplied to the instrument. If depth is sent in “DBAR” we expect that a Tare has been applied.

(**) Depth is calculated by using the formula described in section 1.5. For this protocol, we expect that a Tare has been applied to the pressure sensor. Hence, we do not compensate atmospheric pressure Pa (10.1325 dbar) before pressure to depth conversion.

TERPS8000

Standard: ASCII, DPS8xxB pressure sensor (see TERPS User Manual, GE Druck)
Data received: Depth.
Data frame:

<F1><space><F2><CR><LF>				
<F1>	PPPP.PP (i.e 1234.56) P.PPPPP (i.e 1.23456)	Pressure value is a fixed length string depending on the unity selected for pressure sensor. The position of the decimal point is variable	Float	USED
<F2>	CCC CCCC	Unity: Bar or mBar	Character	USED

Depth is calculated by using a formula, refer to section 1.5.

USBL BOX POSTPRO

Standard: Binary. USBL BOX POSTPRO.
Data received: USBL, USBL Postprocessing
Data frame:

Conventions

Telegram format

The telegram is a combination of sensor blocks. Each block contains header, telegram identification and checksum. Any combination of sensor blocks can be sent at input of INS. Multiple sensor blocks of a kind can be sent (i.e: multiple USBL or LBL beacon positions).
The checksum is the sum of signed bytes of the telegram (telegram length – 2 checksum bytes).
All identification values (telegram identification, system type, rejection mode…) are expressed in decimal value otherwise specified.

Time

INS time can be synchronized with GNSS UTC time when UTC time block is sent to INS at regular intervals (i.e: every second). To improve accuracy it is recommended to input a 1 PPS pulse at pulse input of INS. If INS is not time synchronized INS will use the data latency information in sensor data blocks to evaluate age of data. Time in sensor data blocks is reset to 0 every 24 hour.

Data types

Each telegram description uses following convention:

Type name	Description
byte	Unsigned 8 bit integer
character	Signed 8 bit integer
word / ushort	Unsigned 16 bit integer
short	Signed 16 bit integer
dword / ulong	Unsigned 32 bit integer
long	Signed 32 bit integer
float	IEEE Float 32 bits
double	IEEE Float 64 bits

All 16 and 32 bits integers are represented in Big endian convention (MSB sent first). NaN is defined by the following value 0x7FC00000.

Data blocs used by INS

Beacon position telegram

Message <F0><F1>…				
Field 0	Byte 0	byte	Synchronisation	0x24
Field 1	Byte 1	byte	Telegram size	56
Field 2	Byte 2	byte	Telegram identification	0x87
Field 3	Byte 3	byte	Data bloc version	0x01
Field 4	Bytes 4 to 5	short	Beacon Identifier	Is a number in the range : [0, 65535]
Field 5	Bytes 6 to 9	dword	Position Age	µs
Field 6	Bytes 10 to 14	dword	Data validity time	Time since 01/01/1970 in seconds
Field 7	Bytes 14 to 17	dword	Data validity time	µs
Field 8	Bytes 18 to 21	long	Latitude	+/- 231= +/-180°
Field 9	Bytes 22 to 25	long	Longitude	+/- 231= +/-180°
Field 10	Bytes 26 to 29	float	Depth	meters
Field 11	Bytes 30 to 33	float	Covariance North/North	meters^2 (Latitude SD)^2
Field 12	Bytes 31 to 37	float	Covariance North/East	meters^2 (Latitude Longitude SD)^2
Field 13	Bytes 38 to 41	float	Covariance North/Depth	meters^2 (NOT USED)
Field 14	Bytes 42 to 45	float	Covariance East/East	meters^2 (Longitude SD)^2
Field 15	Bytes 46 to 49	float	Covariance East/Depth	meters^2 (NOT USED)
Field 16	Bytes 50 to 53	float	Covariance Depth/Depth	meters^2 (Altitude SD)^2
Field 17	Bytes 54 to 55	word	CRC	unsigned sum of all fields except checksum

Data blocs broadcasted for post-processing

The blocs which ID hexadecimal value are in the range [0x80, 0x8F] are all broad-casted into the post-processing output protocol.

Only the one which ID is 0x87 and specified above, is taken into account by the INS.

USBL LBL CTD

Standard: ASCII. BLUEFIN proprietary protocol.

Data received: SBL Fix Latitude, Longitude, Depth, Standard deviations, LBL Latitude, Longitude, Depth, Beacon ID, Range, Range standard deviation, CTD Conductivity, Temperature, Pressure, Time and date, Salinity

Data frame:

Remark:

- x.x data format specification means that INS expects a float value (no matter the number of digits before or after the decimal point) .
- ±x.x data format specification means that INS expects the sign character before the float value (here again, no matter the number of digits before or after the decimal point).

$BFUSBL,LLmm.mmmm,a,LLLmm.mmmm,b,±x.x,x.x,x.x,x.x,x.x,x.x*hh<CR><LF>

LLmm.mmmm	is the latitude in degrees (LL) and in minutes (mm.mmmm)	USED
a	is ‘N’ for Northern hemisphere, ‘S’ for Southern hemisphere	USED
LLLmm.mmmmmm	is the longitude in degrees (LLL) and in minutes(mm.mmm)	USED
b	is ‘E’ for East, ‘W’ for West	USED
±x.x	is the depth in meters	USED
x.x	is the latitude standard deviation in meters	USED
x.x	is the longitude standard deviation in meters	USED
x.x	is the latitude-longitude covariance in meters2	USED
x.x	is the depth standard deviation in meters	USED
x.x	is the age of the data in seconds	USED
hh	is the checksum	USED

$BFLBL,llmm.mmmm,a,LLLmm.mmmm,b,x.x,i,x.x,x.x,x.x*hh<CR><LF>

LLmm.mmmm	is the latitude in degrees (LL) and in minutes (mm.mmmm)	USED
a	is ‘N’ for Northern hemisphere, ‘S’ for Southern hemisphere	USED
LLLmm.mmmmmm	is the longitude in degrees (LLL) and in minutes(mm.mmm)	USED
b	is ‘E’ for East, ‘W’ for West	USED
x.x	is the beacon depth in meters	USED
i	is the beacon identifier	* USED
x.x	is the range	USED
x.x	is the range standard deviation	USED
x.x	is the age of the data in seconds	USED
hh	is the checksum	USED

$BFCTD,±x.x,±x.x,±x.x,hh:mm:ss mm-dd-yy,±x.x,±x.x*hh<CR><LF>

±x.x	is the conductivity in mS:cm	**

±x.x	is the temperature in °C	**
±x.x	is the pressure in decibar (1)	** USED
hh:mm:ss mm-dd-yy	is the time and the date	NOT USED
±x.x	is salinity in PSU	NOT USED
±x.x	is sound velocity in m/s	** USED
hh	is the checksum	USED

* The beacon identifier value received in the $BFLBL data frame will be issued in the output Sensor RD.

** Conductivity, temperature, pressure and sound velocity values received in the $BFCTD data frame will be issued in the output Navigation & CTD.

(1) Depth is calculated by using the formula described in section 1.5.

VBW

Standard: Input NMEA 0183.

Data received: EM LOG: Water referenced longitudinal speed. DVL: Water track or Bottom track speeds.

.**Data frame:**

$--VBW,x.x,x.x,S,x.x,x.x,s*hh <CR><LF>		
x.x	XV1 longitudinal water speed, in knots, '-' for astern.	USED (*) (**)(****)
x.x	XV2 transverse water speed, in knots, '-' for port.	USED (*) (**)
S	is the status of the water speed	USED
	'A' = data valid 'V' = data invalid	
	XV1 longitudinal ground speed, in knots, '-' for astern.	
x.x	XV2 transverse ground speed, in knots, '-' for port.	USED (*) (***)
x.x	is the status of the ground speed	USED (*) (***)
s	A' = data valid 'V' = data invalid	USED
x.x	Stern transverse water speed in knots.	NOT USED
s	is the status of stern water speed.	NOT USED
x.x	Stern transverse ground speed in knots.	NOT USED
s	is the status of stern ground speed	NOT USED
hh	Checksum	

The default standard deviation on speed taken into account by Rovins is 0.5 m/s.

(*): Shall not be blank field if data valid (Status='A').

(**): Used as DVL Water track speed if DVL is configured.

(***): Used as DVL Bottom track speed is DVL is configured.

(****): Used as Emlog speed is Emlog is configured.

6.2 Detailed Specifications of Output Protocols

AIPOV

Standard: output NMEA 0183.
Data sent: UTC Time, Heading, Roll, Pitch, Rotation Rates, Linear Accelerations, Position, Speed, True course, User Status.
Data frame: ASCII frame.

$AIPOV, hhmmss.ssss, h.hhh, r.rrr, p.ppp, x.xxx, y.yyy, z.zzz, e.ee,f.ff,g.gg, LL.LLLLLLLL, ll.lllllll, a.aaa, i.iii, j.jjj,k.kkk, m.mmm, n.nnn, o.ooo, c.ccc,hhhhhhhh*hh<CR><LF>

Parameter	ASCII model	Unit	Resolution	Range	Exact definition of parameter	Sign convention
Header	$AIPOV				ASCII header	
UTC time	hhmmss.ssss	second	10-4 s		UTC time in hour, minutes, seconds	
Heading	h.hhh	Decimal degree	10-3 deg	0-360	Heading of aircraft	0 : North 90 : East
Roll	+/-r.rrr	Decimal degree	10-3 deg	+/- 180	Roll of aircraft	> 0 when left wing goes up
Pitch	+/-p.ppp	Decimal degree	10-3 deg	+/-90	Pitch of aircraft	> 0 when nose up (*)
Rotation Rate XV1	+/-x.xxx	Degrees per second	10-3 deg/s	+/- 750	Angular rate around along-aircraft-axis	> 0 when left wing goes up
Rotation Rate XV2	+/-y.yyy	Degrees per second	10-3 deg/s	+/- 750	Angular rate around across aircraft axis	> 0 when nose up (*)
Rotation Rate XV3	+/-z.zzz	Degrees per second	10-3 deg/s	+/- 750	Angular rate around third aircraft axis	> 0 when turning counter clockwise (*)
Linear Acceleration XV1	+/-e.ee	m/s^2	10-2 m/s^2	+/- 147.15 (15g)	Acceleration on along aircraft axis. Gravity not included	> 0 when acceleration towards front of vehicle
Linear acceleration XV2	+/-f.ff	m/s^2	10-2 m/s^2	+/- 147.15 (15g)	Acceleration on across aircraft axis Gravity not included	> 0 when acceleration towards right wing (*)

Parameter	ASCII model	Unit	Resolution	Range	Exact definition of parameter	Sign convention
Linear Acceleration XV3	+/-g.gg	m/s²	10-2 m/s²	+/- 147.15 (15g)	Acceleration on third aircraft axis Gravity not included	> 0 when acceleration goes down (*)
Latitude	+/-LL.LLLLLLLL	Decimal degree	10-8 deg	+/-90	Latitude of aircraft position	> 0 when North latitude
Longitude	+/-ll.lllllll	Decimal degree	10-8 deg	+/-180	Longitude of aircraft position	>0 when East
Altitude	a.aaa	meters	10-3 m	15000	Altitude of aircraft	
North Velocity	+/-I.iii	m/s	10-3 m/s	+/- 250	Velocity of aircraft along North axis	> 0 when going North
East Velocity	+/-j.jjj	m/s	10-3 m/s	+/- 250	Velocity of aircraft along East axis	> 0 when going East (*)
Vertical Velocity	+/-k.kkk	m/s	10-3 m/s	+/- 250	Velocity of aircraft along vertical axis	> 0 when going down
Along Velocity (XV1)	+/-m.mmm	m/s	10-3 m/s	+/- 250	Velocity on along aircraft axis.	> 0 when velocity towards front of vehicle
Across Velocity XV2	+/-n.nnn	m/s	10-3 m/s	+/- 250	Velocity on across aircraft axis	> 0 when velocity towards right wing (*)
Down Velocity XV3	+/-o.ooo	m/s	10-3 m/s	+/- 250	Velocity along third aircraft axis	> 0 when velocity goes down (*)
True course	c.ccc	Decimal degrees	10-3 deg	0-360	Direction of the aircraft horizontal velocity	

Parameter	ASCII model	Unit	Resolution	Range	Exact definition of parameter	Sign convention
User Status	hhhhhhhh	hexadecimal			AIRINS integrity status (32 bits)	
Checksum	hh	hexadecimal				

ANSCHUTZ STD20

Standard: Binary 18 bytes.
Data sent: Heading and heading rate.
Data frame: 8 fields – 18 bytes. Big Endian (MSB sent first).

<F0><F1><F2><B3><B4><B5><B6><F7>			
Field 0	Byte 0	Header	set to 0x02
Field 1	Bytes 1 to 2	Heading	If binary heading value is h15h14Bh13h12h11h10h9h8h7h6h5h4h3h2h1h0, the heading is coded on 12 bits only, and h3h2h1h0 LSB bits are not provided. Heading Byte 1 = 01h15h14h13h12h11h10 Heading Byte 2 = 01h9h8h7h6h5h4 LSB 180/2P15P = 0.00549°
Field 2	Bytes 3 to 7	Fixed fields	All the bytes are set to 0x40
Field 3	Bytes 8 to 10	Heading rate	If binary heading rate value is h15h14Bh13h12h11h10h9h8h7h6h5h4h3h2h1h0, the heading rate is 2 complement binary coded on 3 bytes. Heading rate Byte 8 = 01s0h15h14Bh13h12, where s is the sign bit set to 0 if heading rate positive, otherwise set to 1. Heading rate Byte 9 = 01 h11h10h9Bh8Bh7h6 Heading rate Byte 10 = 01h5h4h3h2h1h0 LSB = 2636.718/2P15 = 0.08046 °/min Opposite sign of the ANSCHUTZ STD20 heading derivative Warning: Opposite sign of the INS usual convention
Field 4	Bytes 11 to 12	Status word	See Note 1 below
Field 5	Bytes 13 to 15	Fixed fields	All the bytes are set to 0x40
Field 6	Byte 16	Checksum	Checksum is computed as follow ((0xFF xor[Bytes 1 to 15]) and 0x3F) or 0x40
Field 7	Byte 17	End of frame	set to 0x03

Note 1: Status word

MSB: 0|1|a|b|0|0|0|f LSB: 0|1|g|1|0|1|0|0	Description	Involved INS status bits
a=0 and b=0	System OK	N/A
a=1 and b=1	System Error	When one of those INS User status bit is set to 1: FOG_ANOMALY ACC_ANOMALY TEMPERATURE_ERR CPU_OVERLOAD DEGRADED_MODE DYNAMIC_EXCEEDED FAILURE_MODE FINE ALIGNMENT INPUT_x_ERR OUTPUT_x_ERR
f=1 and g=1	Default status	N/A
f=0 and g=1	Settling phase	When the INS User status bit ALIGNMENT is set to 1
f=0 and g=0	Heading not valid	When the INS User status bit HRP_INVALID is set to 1

BHO_GRAVI

Standard: ASCII.

Data frame: BHO_GRAVI output protocol is an ASCII output protocol made of 2 telegrams as defined below:

- $GPGLL output frame, refer to section 4.1.2.
- $GPVTG output frame, refer to section 4.1.4.

BROADCAST FROM PORT X

Standard: That of the input port X

Data sent: Used to broadcast all the data coming from an input port* (port X) to the selected output port.

*The broadcast coming from A to E binary input does not work.

BUC

Standard: Binary protocol.
Data sent: Roll, Pitch, Heave and Heading.
Data frame: 6 fields - 10 bytes.
Except the heading, each data sent is two complemented coded. LSB are sent first.

Message <F0><F1><F2>.....<F5>			
Field 0	Byte 0	Sensor status	Fixed value = 0x9A
Field 1	Byte 1	Synchronization byte	0x90
Field 2	Bytes 2 to 3	Roll	+/-180° ; LSB = 0.01° Sign “+” when port up
Field 3	Bytes 4 to 5	Pitch	+/-180° ; LSB = 0.01° Sign “+” when bow up Warning: Opposite sign of Rovins usual convention.
Field 4	Bytes 6 to 7	Heave	LSB = 0.01 m Sign +” when Rovins goes up
Field 5	Bytes 8 to 9	Heading	0° to 360° ; LSB = 0.01°

CONTROL

Standard:Binary protocol. CONTROL output.
Data sent: Rotation rates and accelerations. Acceleration are not compensated for g vector.
Data frame:

Message <$><F1><F2>…..<F6>			
Field 0	Byte 0	Header	'$' : integer 8 bits
Field 1	Bytes 1 to 4	Acceleration XV1 *	Signed 32 bits LSB = 30g / 231 = 0.01397µg
Field 2	Bytes 5 to 8	Acceleration XV2 *	Signed 32 bits LSB = 30g / 231 = 0.01397µg
Field 3	Bytes 9 to 12	Acceleration XV3 *	Signed 32 bits LSB = 30g / 231 = 0.01397µg
Field 4	Bytes 13 to 16	Rotation rates XV1 **	Signed 32 bits LSB = 100°/s / 231 = 46.566.10-9 °/s
Field 5	Bytes 17 to 20	Rotation rates XV2 **	Signed 32 bits LSB = 100°/s / 231 = 46.566.10-9 °/s
Field 6	Bytes 21 to 24	Rotation rates XV3 **	Signed 32 bits LSB = 100°/s / 231 = 46.566.10-9 °/s

* in non-military mode: acceleration quantification is 1mg.
** in non-military mode: rotation rate quantification with 3.6°/h.

CONTROL NO G

Standard: Binary protocol. CONTROL output.

Data sent: Rotation rates and accelerations. Acceleration are compensated for g vector.

Data frame:

Message <$><F1><F2>…..<F6>			
Field 0	Byte 0	Header	'$' : integer 8 bits
Field 1	Bytes 1 to 4	Acceleration XV1 *	+/- 231 = +/- 30g Signed 32 bits with saturation
Field 2	Bytes 5 to 8	Acceleration XV2 *	+/- 231 = +/- 30g Signed 32 bits with saturation
Field 3	Bytes 9 to 12	Acceleration XV3 *	+/- 231 = +/- 30g Signed 32 bits with saturation
Field 4	Bytes 13 to 16	Rotation rates XV1 **	+/- 231 = +/- 100°/s Signed 32 bits with saturation
Field 5	Bytes 17 to 20	Rotation rates XV2 **	+/- 231 = +/- 100°/s Signed 32 bits with saturation
Field 6	Bytes 21 to 24	Rotation rates XV3 **	+/- 231 = +/- 100°/s Signed 32 bits with saturation

* in non-military mode: acceleration quantification is 1mg.

** in non-military mode: rotation rate quantification with 3.6°/h

DOLOG HRP

Standard: Binary protocol. Output Dolog custom protocol.
Data sent: Status, Heading, Roll, Pitch, Heading Rate, Roll Rate, Pitch Rate
Data frame: The frame contains 10 fields - 16 bytes. MSB are sent first.

Message <F0><F1><F2>.....<F9>			
Field 0	Byte 0	0x02	Start of sentence
Field 1	Byte 1	Status	Bit 7, 6, 3, 2 and 1 not used Bit 5 : Alarm bit = 1 if ACC or FOG error Bit 4 : Alignment bit = 1 for Alignment Bit 0 : Data valid bit = 1 for HRP data valid
Field 2	Bytes 2 to 3	Heading	Unsigned 16 bits integer 215 = 180° 0 to 359.99°
Field 3	Bytes 4 to 5	Roll	Signed 16 bits integer +/-215 = +/-90° -90° to 89.99° (Positive when port side up)
Field 4	Bytes 6 to 7	Pitch	Signed 16 bits integer +/-215 = +/-90° -90° to 89.99° (Positive when bow up) Warning: Opposite sign of the INS usual convention
Field 5	Bytes 8 to 9	Heading rate*	Signed 16 bits integer +/-215 = +/-45°/s -45 to 44.99°/s (Positive when the INS heading angle decreases) Warning: Opposite sign of the INS usual convention
Field 6	Bytes 10 to 11	Roll rate*	Signed 16 bits integer +/-215 = +/-45°/s -45 to 44.99°/s (Positive when the INS roll angle increases)
Field 7	Bytes 12 to 13	Pitch rate*	Signed 16 bits integer +/-215 = +/-45°/s -45 to 44.99°/s (Positive when the INS pitch angle decreases) Warning: Opposite sign of the INS usual convention
Field 8	Byte 14	Checksum	Negative sum of all the bytes before checksum byte (ignoring overflow)
Field 9	Byte 15	0x03	End of sentence

* The precision of rotation rate data is limited to 3,6 deg/h to comply with export regulation.

DORADO

Standard: Binary protocol. Output Dorado custom protocol.

Data sent: Position, Heading, Heading, Roll, Pitch, Rotation rates, speed coordinates in geographical frame, INS lat/long position, Log misalignment.

Data frame:

<F0><F1>…F16><F17>			
Field 0	Byte 0	Header	integer 8 bits : ‘$’
Field 1	Byte 1	Status	Integer 8 bits :0x01 => Initial alignment 0x06 => Invalid attitude 0x08 => Invalid heading
Field 2	Bytes 2 to 5	Heading	Floating IEEE 32 bits *in Radians
Field 3	Bytes 6 to 9	Roll	Floating IEEE 32 bits * in Radians (+ if port up)
Field 4	Bytes 10 to 13	Pitch	Floating IEEE 32 bits * in Radians (+ if bow down)
Field 5	Bytes 14 to 17	Rotation rate XV3	Floating IEEE 32 bits ** in Radians /sec
Field 6	Bytes 18 to 21	Rotation rate XV1	Floating IEEE 32 bits ** in Radians /sec
Field 7	Bytes 22 to 25	Rotation rate -XV2	Flotting IEEE 32 bits ** in Radians /sec Warning : sign opposite with the conventional sign
Field 8	Bytes 26 to 29	Depth	Floating IEEE 32 bits in Meters
Field 9	Bytes 30 to 33	Down speed	Floating IEEE 32 bits in m/s (positive downwards)
Field 10	Bytes 34 to 37	East speed	Floating IEEE 32 bits in Meters/sec
Field 11	Bytes 38 to 41	South speed	Floating IEEE 32 bits in Meters/sec
Field 12	Bytes 42 to 45	Latitude	Integer 32 bits signed in Radians (+/- 231= +/- 180°)
Field 13	Bytes 46 to 49	Longitude	Integer 32 bits signed in Radians (+/- 231= +/- 180°)
Field 14	Bytes 50 to 53	Kalman log misalignment estimation	Floating IEEE 32 bits in Radians

<F0><F1>…F16><F17>			
Field 15	Bytes 54 to 57	Word reserved	Floating IEEE 32 bits : Value forced with 0.0
Field 16	Bytes 58 to 59	Counter of sequences	Integer 16 bits from 0 to 65535
Field 17	Byte 60	Checksum	Integer 8 bits : Summon of all the bytes of the frame, heading included

* In non-military mode: Heading, roll, pitch quantification with 0.001°.
** In non-military mode: rotation rate quantization is limited to 3.6°/h.

EMT SDV GGS

Standard: Binary protocol.
Data sent: Status, Roll, Pitch, Depth, Heading.
Data frame: 7 fields - 11 bytes.
Except the heading, each sent data is two complemented coded. LSB are sent first.

Message <F0><F1><F2>…..<F6>			
Field 0	Byte 0	Sensor status	0x90 if ok 0x9A if alignment
Field 1	Byte 1	Synchronization byte	0x90
Field 2	Bytes 2 to 3	Roll*	+/-180° ; LSB = 0.01° Sign "+" when port up
Field 3	Bytes 4 to 5	Pitch*	+/-180° ; LSB = 0.01° Sign "+" when bow up Warning: Opposite sign of INS usual convention.
Field 4	Bytes 6 to 7	Depth	LSB = 1 mm
Field 5	Bytes 8 to 9	Heading	0° to 360° ; LSB = 0.01°
Field 6	Byte 10	Checksum	XOR on bytes 0 to 9

* The attitude angles are computed with respect to TSS convention.
Roll and Pitch are referenced to the local vertical acceleration.
The formula calculation with respect to INS standard convention (Euler Angle or Tate Bryant) is given hereafter:

$$Roll_{TSS} = Sin^{-1} (Sin (Roll_{TB}) \times Cos (Pitch_{TB})) \text{ and } Pitch_{TSS} = Pitch_{TB}$$

EVENT MARKER

Standard: Custom.
Data sent: Input pulse reception time (i.e., event time).
Data frame: ASCII frame.

x<TAB> y.yyyyyy<TAB>c<CR><LF>

x	Input pulse: 0 for A 1 for B 2 for C 3 for D
y.yyyyyy	Pulse reception time (i.e: event time) in seconds
c	Pulse counter (integer)

Example:
In this example Rovins received two Event Markers, one on the Input Pulse A and the second on the Input Pulse D at the same frequency:
0 11354.374484 1
3 11354.374524 1
0 11354.379561 2
3 11354.379608 2
0 11354.384646 3
3 11354.384692 3
0 11354.389714 4
3 11354.389780 4

EXT SENSOR BIN

Standard: Binary protocol. Output SOC custom protocol.
Data sent : Status, Heading, Attitude, Rotation rates, Depth, Speeds, Position, Log misalignment.
Data frame: The frame contains 18 fields - 61 byte. MSB are sent first.

Message <F0><F1><F2>…..<F17>			
Field 0	Byte 0	'$'	Synchronization byte
Field 1	Byte 1	Status	1 if Alignment
Field 2	Bytes 2 to 5	Heading	Radians IEEE floating point format
Field 3	Bytes 6 to 9	Roll	Radians IEEE floating point format Sign "+" when port up
Field 4	Bytes 10 to 13	Pitch	Radians IEEE floating point format Sign "+" when bow down
Field 5	Bytes 14 to 17	XV3 rotation rate*	Rad/s IEEE floating point format
Field 6	Bytes 18 to 21	Xv1 rotation rate*	Rad/s IEEE floating point format
Field 7	Bytes 22 to 25	-XV2 rotation rate*	Rad/s IEEE floating point format Warning: Opposite sign of INS usual convention
Field 8	Bytes 26 to 29	Depth	Meters IEEE floating point format
Field 9	Bytes 30 to 33	Down speed	M/s IEEE floating point format
Field 10	Bytes 34 to 37	East speed	M/s IEEE floating point format
Field 11	Bytes 38 to 41	South speed	M/s IEEE floating point format
Field 12	Bytes 42 to 45	Latitude	+/-231 = +/-Pi Signed 32 bits
Field 13	Bytes 46 to 49	Longitude	+/-231 = +/-Pi Signed 32 bits
Field 14	Bytes 50 to 53	Log misalignment	Radians IEEE floating point format
Field 15	Bytes 54 to 57	Spare fields	4 bytes
Field 16	Bytes 58 to 59	Counter	Incremented by 1 Unsigned 16 bits
Field 17	Byte 60	Checksum	Addition of all the bytes for 0 to 59

* The resolution of rotation rate data is limited to 3.6°/h to comply with export regulation.

GAPS BIN

Standard: Binary protocol. This protocol is used by the INS II GAPS 3.

Data sent: Heading, Attitude, Position, speeds, heave, attitude, heading and position standard deviations, status.

Data frame: On INS III, interface status is mapped to INS II equivalent status. All fields are transmitted MSB first.

Data	Format	Units
Header	8 bit unsigned integer	Value : '$'
Time tag of data	64 bit unsigned integer	See Note 1
INS interface status 1	32 bit unsigned integer	See Note 3
INS algorithm status 2	32 bit unsigned integer	See Note 4
Heading	32 bit IEEE Float	Rad
Roll	32 bit IEEE Float	Rad
Pitch	32 bit IEEE Float	Rad
Latitude	32 bit signed integer +/- 231= +/- 90°	Degrees
Longitude	32 bit signed integer +/- 231= +/- 180°	Degrees
Altitude (positive up)	32 bit IEEE Float	m
North speed	32 bit IEEE Float	m/s
West speed	32 bit IEEE Float	m/s
Vertical speed (positive up)	32 bit IEEE Float	m/s
Heave on selected lever arm (positive up)	32 bit IEEE Float	m
Latitude standard deviation	32 bit IEEE Float	m
Longitude standard deviation	32 bit IEEE Float	m
Altitude standard deviation	32 bit IEEE Float	m
Heading standard deviation	32 bit IEEE Float	Rad
Roll standard deviation	32 bit IEEE Float	Rad
Pitch standard deviation	32 bit IEEE Float	Rad
CRC	16 bit unsigned integer	Note
End of frame	8 bit unsigned integer	Value: '#'

Note 1:

The 64 bits time tag is described hereafter :

Bit [63..56] spare

Bit [55..52] x 10 days (0 to 3)	Bit [51..48] days (0 to 9)
Bit [47..44] x 10 hour (0 to 2)	Bit [43..40] hours (0 to 9)
Bit [39..35] x 10 minutes (0 to 5)	Bit [34..31] minutes (0 to 9)
Bit [31..28] x 10 seconds (0 to 5)	Bit [27..24] seconds (0 to 9)
Bit [23..20] x 1/10 seconds (0 to 9)	Bit [19..16] x 1/100 seconds (0 to 9)
Bit [15..12] x 1000 μ seconds (0 to 9)	Bit [11..8] x 100 μ seconds (0 to 9)
Bit [7..4] x 10 μ seconds (0 to 9)	Bit [3..0] μ seconds (0 to 9)

Note 2: Crc computation is performed using XOR with polynom = X15+X10+X3, initialized to 0xFFFF.

CRC code:

```
unsigned short blkcrc(unsigned char* bufptr, unsigned len)
{
    unsigned char i;
    unsigned short data;
    unsigned short crc = 0xffff;

    if (len == 0)
        return ~crc;

    do
    {
        for (i = 0, data = (unsigned short)(0xff & *bufptr++);
            i < 8;
            i++, data >>= 1)
        {
            if ((crc & 0x0001) ^ (data & 0x0001))
            {
                crc = (crc >> 1) ^ 0x8408;
            }
            else
            {
                crc >>= 1;
            }
        }
    } while (--len);

    crc = ~crc;
    data = crc;
    crc = (crc << 8) | ((data >> 8) & 0xff);

    return crc;
}
```

Note 3: INS interface status 1 (mapped from INS II System status 1)

Hexadecimal Value	Bit number	Description	Hexadecimal Value	Bit number	Description
hhhhhhhH	0	Log received	hhhHhhhh	16	Serial input A error
	1	GNSS received		17	Serial input B error
	2	Depth received		18	Serial input C error
	3	USBL received		19	FIFO Full
hhhhhhHh	4	LBL received	hhHhhhhh	20	Serial output A full
	5	GNSS2 received		21	Serial output B full
	6	Log EM received		22	Serial output C full
	7	Settings Saved		23	Serial output D full
hhhhhHhh	8	FOG X1 Error	hHhhhhhh	24	Serial IN A activity
	9	FOG X2 Error		25	Serial IN B activity
	10	FOG X3 Error		26	Serial IN C activity
	11	source error		27	CPU overload
hhhhHhhh	12	Acc. X1 error	Hhhhhhhh	28	Heading not valid
	13	Acc. X2 error		29	Attitude not valid
	14	Acc. X3 error		30	Altitude received
	15	Temperature error		31	Reserved

Note 4: INS interface status 2 (mapped from INS Algorithm status 1)

Hexadecimal Value	Bit number	Description	Hexadecimal Value	Bit number	Description
hhhhhhhH	0	Navigation mode	hhhHhhhh	16	USBL received
	1	Alignment		17	USBL data valid
	2	Fine alignment		18	Waiting for USBL data
	3	Dead reckoning mode		19	USBL data rejected
hhhhhhHh	4	Altitude calculated using GNSS	hhHhhhhh	20	Depth sensor received
	5	Altitude calculated using Depth sensor		21	Depth sensor data valid
	6	Altitude stabilized		22	Waiting for Depth sensor data
	7	Altitude Hydro		23	Depth sensor data rejected
hhhhhHhh	8	Log received	hHhhhhhh	24	LBL received
	9	Log data valid		25	LBL data valid
	10	Waiting for Log data		26	Waiting for LBL data
	11	Log data rejected		27	LBL data rejected
hhhhHhhh	12	GNSS received	Hhhhhhhh	28	Saturation of altitude
	13	GNSS data valid		29	Saturation of speed
	14	Waiting for GNSS data		30	Acc or Rotation speed dynamic exceeded
	15	GNSS data rejected		31	Heave initialization

GPS LIKE

Standard: Output NMEA 0183 compatible.

Data sent: This protocol outputs Rovins computed position, speed, time, standard deviations values in a “GPS like” format.

Time management: see chapter 1.1.

Data frame:

$GPZDA,hhmmss.ss,dd,mm,yyyy,hh,mm*hh<CR><LF> **

hhmmss.ss	UTC time of Rovins	**
dd	UTC day	*
mm	UTC month	*
yyyy	UTC year	*
hh	Local zone hours	*
mm	Local zone minutes	*
hh	Checksum	**

$GPGGA,hhmmss.ss,LLII.IIIIIII,a,LLLII.IIIIIII,a,x,xx,x.xxx,x.xxx,M,x.xxx,M,x.xxx,xxxx*hh<CR><LF>

hhmmss.ss	UTC time of position	**
LLII.IIIIIII	Latitude in degrees (LL) and in minutes (II.IIIIIII)	**
a	‘N’ for Northern hemisphere, ‘S’ for Southern hemisphere	**
LLLII.IIIIIII	Longitude in deg (LLL) and in minutes (II.IIIIIII)	**
a	‘E’ for East, ‘W’ for West	**
x	GNSS quality indicator	**
xx	Number of satellites in use	*
x.xxx	Horizontal dilution of precision (HDOP)	**
x.xxx	Antenna altitude above mean sea level (geoid) (meters)	**
M	Unit of antenna altitude (fixed character = ‘M’ for meters)	**
x.xxx	Geoidal separation	*
M	Unit of Geoidal separation (fixed character = ‘M’ for meters)	*
x.xxx	Age of the differential GNSS data	*
xxxx	Differential reference station ID	*
hh	Checksum	**

$GPGST,hhmmss.ss,x.x,x.x,x.x,x.x,x.x,x.x,x.x*hh<CR><LF>

hhmmss.ss	UTC Time	**
x.x	RMS value of the standard deviation on pseudo-ranges	*
x.x	Standard deviation of semi-major axis of error ellipse in meters	**
x.x	Standard deviation of semi-minor axis of error ellipse in meters	**
x.x	Orientation of semi-major axis of error ellipse	**
x.x	Standard deviation of the error of Latitude in meters	**
x.x	Standard deviation of the error of Longitude in meters	**
x.x	Standard deviation of the error of Altitude	**
hh	Checksum	**

$GPVTG,x.xxx,T,x.xxx,M,x.xxx,N,x.xxx,K,a*hh<CR><LF>

x.xxx	True course (deg)	**
T	Fixed character = ‘T’	**
x.xxx	Magnetic course (deg)	** identical to true course
M	Fixed character = ‘M’	**
x.xxx	Speed (knots)	**
N	Fixed character = ‘N’	**
x.xxx	Speed (km/h)	**
K	Fixed character=’K’	**
a	Positioning system mode indicator ‘A’, ‘D’ or ‘E’	***
hh	Checksum	**

$GPGLL,LLll.llllllll,a,LLLll.llllllll,a,hhmmss.ss,a,m*hh<CR><LF>

LLll.llllllll	Latitude in degrees (LL) and in minutes (ll.llllllll)	**
a	‘N’ for Northern hemisphere, ‘S’ for Southern hemisphere	**
LLLll.llllllll	Longitude in deg (LLL) and in minutes (ll.llllllll)	**
a	‘E’ for East, ‘W’ for West	**
hhmmss.ss	UTC time of position	**
a	Status ‘A’ for Data Valid, ‘V’ for Data Invalid	Data invalid for initial alignment and speed saturation (i.e User status ALIGNMENT AND User status SPEED_ SATURATION bits set to 1)
m	Mode indicator ‘A’, ‘D’ or ‘E’	***
hh	Checksum	**

* Copy of last GNSS values received. When no GNSS has been received since power-up, these fields are null.
** Rovins calculated data.
*** see chapter 1.4

Some empty fields are allowed in –GGA and –VTG data frames. See samples hereafter.
$--GGA,161229.487,3723.2475,N,12158.3416,W,1,07,1.0,9.0,M,,,,0000*18

```
$--GGA,064036.289,4836.5375,N,00740.9373,E,1,04,3.2,200.2,M,,,,0000*0E
$--VTG,309.62,T,,M,0.13,N,0,2,K*6E
```

**** The ZDA sentence is always sent at 1 Hz whatever the chosen refresh rate.

GPS LIKE SHORT

Standard: Output NMEA 0183 compatible.

Data sent: This protocol outputs Rovins computed position, speed, time, standard deviations values in an “GPS like” format.

Time management: If Rovins has never received GNSS, date starts on 1st January 2006. Otherwise date is maintained on GNSS time as soon as GNSS time is received. If GNSS is lost, Rovins will maintain time with its internal clock. If Rovins is synchronized with GNSS time, the time tags are UTC time, otherwise they are Rovins time (time since power-up of the system).

Data frame:

$GPGGA,hhmmss.ss,LLII.IIIIIII,a,LLLII.IIIIIII,a,x,xx,x.xxx,x.xxx,M,x.xxx,M,x.xxx,xxxx*hh<CR><LF>

hhmmss.ss	UTC time validity of position	**
LLII.IIIIIII	Latitude in degrees (LL) and in minutes (II.IIIIIII)	**
a	‘N’ for Northern hemisphere, ‘S’ for Southern hemisphere	**
LLLII.IIIIIII	Longitude in deg (LLL) and in minutes (II.IIIIIII)	**
a	‘E’ for East, ‘W’ for West	**
x	GNSS quality indicator	***
xx	Number of satellites in use	*
x.xxx	Horizontal dilution of precision (HDOP)	*
x.xxx	Antenna altitude (meters) (here Rovins altitude)	**
M	Unit of antenna altitude (fixed character = ‘M’ for meters)	**
x.xxx	Geoidal separation	*
M	Unit of Geoidal separation (fixed character = ‘M’ for meters)	*
x.xxx	Age of the differential GNSS data	*
xxxx	Differential reference station ID	*
hh	Checksum	

$GPGST,hhmmss.ss,x.x,x.x,x.x,x.x,x.x,x.x,x.x*hh<CR><LF>

hhmmss.ss	UTC Time validity of data	**
x.x	Standard deviation of the range inputs to the navigation process	*
x.x	Standard deviation of the half main roads of the ellipse of error in meters	**
x.x	Standard deviation of the small half centers ellipse of error in meters	**
x.x	Angle of orientation of the ellipse of error in meters	**
x.x	Standard deviation of the error of Latitude in meters	**
x.x	Standard deviation of the error of Longitude in meters	**
x.x	Standard deviation of the error of Altitude	**
hh	Checksum	

* Copy of last GNSS values received. When no GNSS has been received since power-up, these fields are null.

** Rovins calculated data.

*** The quality indicator is managed as follows:
Rovins does not copy the quality indicator received on GGA input to GGA output.
The quality factor is set with respect to a correspondence table between Rovins calculated SD and Quality indicator in GGA telegram (refer to section 1.4).

Some empty fields are allowed in --GGA and –VTG data frames. See samples hereafter :
$--GGA,161229.487,3723.2475,N,12158.3416,W,1,07,1.0,9.0,M,,,,0000*18
$--GGA,064036.289,4836.5375,N,00740.9373,E,1,04,3.2,200.2,M,,,,0000*0E
$--VTG,309.62,T,,M,0.13,N,0,2,K*6E

**** The ZDA sentence is always sent at 1 Hz whatever the chosen refresh rate.

GYROCOMPASS

Standard: Output NMEA 0183 compatible.
Data sent: Heading, Roll, Pitch, Status.
Data frame:

$HEHDT,x.xxx,T*hh<CR><LF>		
x.xxx **T** **hh**	is the true heading in degrees is a fixed character = 'T' is the checksum	3 digits after the decimal point (*)

$PIXSE,ATITUD,x.xxx,y.yyy*hh<CR><LF>		
x.xxx **y.yyy** **hh**	is the roll in degrees is the pitch in degrees is the checksum	3 digits after the decimal point (*) 3 digits after the decimal point (*)

$PIXSE,STATUS,hhhhhhhh,llllllll *hh<CR><LF>		
hhhhhhhh **llllllll** **hh**	is the hexadecimal value of the 32 LSB bits of the Rovins System status is the hexadecimal value of the 32 MSB bits of the Rovins System status is the checksum	Refer to section 1

(*) 5 digits after the decimal point in Military mode.

GYROCOMPASS 2

Standard: ASCII. Output NMEA 0183 compatible.
Data sent: Heading, Roll, Pitch, Heave, Status.
Data frame:

$HEHDT,x.x,T*hh<CR><LF>		
x.x	True heading in degrees. Must be empty if HRP INVALID is set in user status	**Note 1**
T	Fixed ASCII character ‘T’	
hh	NMEA checksum	

$PHTRH,x.xx,a,y.yy,b,z.zz,c*hh<CR><LF>		
x.xx	Pitch in degrees	
a	‘M’ for bow up and ‘P’ for bow down	
y.yy	Roll in degrees	
b	‘B’ for port down and ‘T’ for port up	
z.zz	Heave absolute value in meters	
c	‘U’ if Rovins goes up and ‘O’ if Rovins goes down	
hh	NMEA checksum	

$PIXSE,STATUS,hhhhhhhh,llllllll *hh<CR><LF>		
hhhhhhhh **llllllll** **hh**	is the hexadecimal value of the 32 LSB bits of the INS System status is the hexadecimal value of the 32 MSB bits of the INS System status is the checksum	Refer to section 1.

HALLIBURTON SAS

Standard: Output NMEA 0183 compatible.

Data sent: Position time stamp, Lat/Long position, Depth, DVL range to bottom, Lat/Long standard deviation. Depth standard deviation, UTM position, DVL estimation of course misalignment and scale factor error. Speed of sound, heading, roll, pitch, heave, rotation rates, course over ground, horizontal speed, speed in mobile frame, heading, roll, pitch standard deviation. North, East, Vertical speed standard deviation.

Sensor and system status.

Data frame:

$PIXSE,HSPOS_,hhmmss.ss,LLll.llllll,H,LLLmm.mmmmmm,D,d.dd,a.aa,x.xx,y.yy,z.zz,d.dd ,nn,c,e.e,n.n,m.mmmm,s.ssss,v.v*hh<CR><LF>

hhmmss.ss	UTC time stamp if system is UTC synchronized, or in system time if not	2 digits after decimal point
LLll.llllll	Latitude in degress (LL), lminutes (ll) and decimals of minutes (llllll)	6 digits after decimal point
H	Hemisphere N or S	
LLLmm.mmmmmm	Longitude integer degrees (LLL), Minutes (mm) and decimals of minutes (mmmmmm)	6 digits after decimal point
D	Hemisphere E or W	
d.dd	Depth in meters	Float, 2 digits after decimal point
a.aa	Altitude of the DVL in meters	Float, 2 digits after decimal point
x.xx	Latitude standard deviation error in meters	Float, 2 digits after decimal point
y.yy	Longitude standard deviation error in meters	Float, 2 digits after decimal point
z.zz	Latitude/Longitude Covariance error in meters	Float, 2 digits after decimal point
d.dd	Depth standard deviation error in meters	Float, 2 digits after decimal point
nn	UTM Zone Longitude integer	
c	UTM Zone Latitude character	
e.e	UTM Position East in meters	Float, 1 digit after decimal point
n.n	UTM Position North in meters	Float, 1 digit after decimal point
m.mmmm	Estimate of DVL course misalignment in degrees	Float, 4 digit after decimal point
s.ssss	Estimate of DVL scale factor correction in %	Float, 4 digit after decimal point
v.v	Speed of sound in meters/sec	Float, 1 digit after decimal point
hh	Checksum NMEA	

$PIXSE,HSATIT,h.hhh,r.rrr,p.ppp,h.hhh,a.aaa,b.bbb,c.ccc,d.ddd,e.eee,f.fff,g.ggg,h.hhh,i.ii,j.jj,k.kk,l.ll,m.mm,n.nn*hh<CR><LF>

h.hhh*	Heading in degrees	Float, 3 digits after decimal point
r.rrr*	Roll in degrees (+ if port up)	Float, 3 digits after decimal point
p.ppp*	Pitch in degrees (+ bow down)	Float, 3 digits after decimal point
h.hhh	Heave in meters	Float, 3 digits after decimal point
**a.aaa ** **	Rotation rate XV3 in degrees/sec	Float, 3 digits after decimal point
**b.bbb ** **	Rotation rate XV1 in degrees/sec	Float, 3 digits after decimal point
**c.ccc ** **	Rotation rate XV2 in degrees/sec	Float, 3 digits after decimal point
d.ddd	Horizontal speed course in degrees	Float, 3 digits after decimal point
e.eee	Horizontal speed in meters /sec	Float, 3 digits after decimal point
f.fff	Speed XV1 in Meters/sec	Float, 3 digits after decimal point
g.ggg	Speed XV2 in Meters/sec	Float, 3 digits after decimal point
h.hhh	Speed XV3 in Meters/sec	Float, 3 digits after decimal point
i.ii	Heading standard deviation error in degrees	Float, 2 digits after decimal point
j.jj	Roll standard deviation error in degrees	Float, 2 digits after decimal point
k.kk	Pitch standard deviation error in degrees	Float, 2 digits after decimal point
l.ll	North speed standard deviation error in meters/sec	Float, 2 digits after decimal point
m.mm	East speed standard deviation error in meters/sec	Float, 2 digits after decimal point
n.nn	Vertical speed standard deviation error in meters/sec	Float, 2 digits after decimal point
hh	Checksum NMEA	

* In Military mode 5 digits after decimal point.

** In non-military mode: rotation rate quantification is 3.6°/h

$PIXSE,HSSTAT,FFAAVVQQ*hh<CR><LF>

FF	Statut 1 : Bit 0 (0x01) : FOG or Source error Bit 1 (0x02) : Accelerometer error Bit 2 (0x04) : Serial input or Ethernet port A error Bit 3 (0x08) : Serial input or Ethernet port B error Bit 4 (0x10) : Serial input or Ethernet port C error Bit 5 (0x20) : Serial input or Ethernet port D error Bit 6 (0x40) : Serial input or Ethernet port E error
AA	Statut 2 : Bit 0 (0x01) GNSS valid Bit 1 (0x02) : DVL Bottom track valid Bit 2 (0x04) : USBL valid Bit 3 (0x08) : Dead Reckoning
VV	Statut 3 : Bit 0 (0x01) : GNSS1 detected Bit 1 (0x02) : DVL Bottom or Water track detected Bit 2 (0x04) : USBL detected Bit 3 (0x08) : Depth detected Bit 4 (0x10) : Activity on serial input or Ethernet port A Bit 5 (0x20) : Activity on serial input or Ethernet port B Bit 6 (0x40) : Activity on serial input or Ethernet port C Bit 7 (0x80) : Activity on serial input or Ethernet port D
QQ	Statut 4 : Bit 0 (0x01) : Alignment Bit 1 (0x02) : Fine Alignement Bit 2 (0x04) : Navigation
hh	Checksum NMEA

HDMS

Standard: Binary protocol.
Data sent: Roll, Pitch, Heave and Heading.
Data frame: 6 fields - 10 bytes.
Except the heading, each data sent is two complemented coded. LSB are sent first.

Message <F0><F1><F2>.....<F5>			
Field 0	Byte 0	Sensor status	Fixed value = 0x90
Field 1	Byte 1	Synchronization byte	Fixed value = 0x90
Field 2	Bytes 2 to 3	Roll	+/-180° ; LSB = 0.01° Sign "+" when port up
Field 3	Bytes 4 to 5	Pitch	+/-180° ; LSB = 0.01° Sign "+" when bow up Warning: Opposite sign of Rovins usual convention.
Field 4	Bytes 6 to 7	Heave (*)	LSB = 0.01 m Sign +" when Rovins goes up
Field 5	Bytes 8 to 9	Heading	0° to 360° ; LSB = 0.01°

(*)The heave corresponds to the lever arm set on the output port.

HEAVE POSTPRO

Standard: Binary protocol.
Data sent: Heave.
Data frame:

Conventions

Time

Rovins clock gives time starting when system is turned on. Time is reset to 0 every 24 hour. Early block versions used IEEE 32 floats to store time, which lacks accuracy when hours increase. Thus this floating point representation was replaced by a fixed point datation of 100 µs granularity in new bloc versions.

Data types

Each telegram description uses following convention:

Type name	Description
Byte	Unsigned 8 bit integer
Character	Signed 8 bit integer
Word	Unsigned 16 bit integer
Short	Signed 16 bit integer
DWord	Unsigned 32 bit integer
Long	Signed 32 bit integer
Float	IEEE Float 32 bits
Double	IEEE Float 64 bits

All 16 and 32 bits integers are represented in Big endian convention (MSB sent first).

Encryption

Only the data fields of the ACC are enciphered. A new encrypting key is sent every 6000 frame that enables the decoding of the 6000 previous frames (hence every 60 second at 100 Hz).
Every 32 bits field is encoded as follow: Byte1 XOR KeyH, Byte2 XOR KeyL, Byte3 XOR KeyH and Byte4 XOR KeyL.

Protocol description

Heave Telegram Version 0x01

Data	Format	Units
Header	Byte	Value : '$'
Telegram size	Byte	NN bytes
Telegram identification	Byte	30
Data bloc version	Byte	0x10
Data validity time	Dword	Rovins time tag in steps of 100 ms
Telegram counter	Word	N/A
Decoding key	Word	KeyH then KeyL
ACC X on COG	Float	meters/second2 in terrestrial frame (enciphered)
ACC Y on COG	Float	meters/second2 in terrestrial frame (enciphered)
ACC Z on COG	Float	meters/second2 in terrestrial frame (enciphered)
Real Time Surge	Float	meters (INS level)
Real Time Sway	Float	meters (INS level)
Real Time Heave	Float	meters (INS level)
Smart Heave Delay	Dword	in steps of 100us
Smart Heave*	Float	Delayed heave in meters (Rovins level)
Heave Status	Byte	0x01 if Heave Ok
Real Time Heave Mode	Byte	Real Time Heave Filter mode: 1-Slight, 2-Moderate, 3-Rough, 4-Automatic
Heave Period	Float	Estimated Heave Period
Heave Amplitude	Float	Estimated Heave Amplitude

* The smart heave is delayed by 100 s.

HEHDT

Standard: Output NMEA 0183.

Data sent: Heading.

Data frame:

NMEA **$HEHDT** Frame refer to section 4.1.7.

HEHDT FIXED

Standard: Output ASCII.
Data sent: Heading.
Data frame:

$HEHDT,xxx.x,T<CR><LF>

where:	xxx.x	is the true heading in degrees	3 digits before and 1 digit after the decimal point
	T	is a fixed character = 'T'	

HEHDT HEROT

Standard: Output NMEA 0183.
Data sent: Heading and heading rotation rate.
Data frame:
This protocol is a standard NMEA protocol "$__HDT" and "$__ROT".
The output is respectively heading in degrees and heading rotation rate in °/min.
Rovins usual sign convention is used.

$HEHDT,x.x,T*hh<CR><LF>		
x.x	is the true heading in degrees	3 digits after the decimal point (*)
T	is a fixed character = 'T'	
hh	is the checksum	

$HEROT,x.x,A*hh<CR><LF>		
x.x	Heading rate of turn , °/min '-'= bow turns to port	2 digits after the decimal point in "Default" dual- use mode (**)
A	A= data valid , V = data invalid	
hh	Is the checksum	

(*) 5 digits after the decimal point in Military mode.
(**) 4 digits after the decimal point in Military mode.
Refer to export regulation for the resolution of rotation rate.

HETHS HEROT

Standard: Output NMEA 0183.
Data sent: Heading and heading rotation rate.
Data frame:

NMEA $HETHS Frame, refer to section 4.1.8.

$HEROT,x.x,A*hh<CR><LF>		
x.x	Heading rate of turn , °/min '-'= bow turns to port	2 digits after the decimal point in "Default" dual- use mode (**)
A	A= data valid , V = data invalid	
hh	Is the checksum	

(*) 5 digits after the decimal point in Military mode.
(**) 4 digits after the decimal point in Military mode.
The resolution of rotation rate data is limited to 3.6°/h to comply with export regulation.

HYDROGRAPHY

Standard: ASCII. Output NMEA 0183 compatible.
Data sent: Heading, Roll, Attitude, Position, Heave.
Data frame:

$HYDRO,a.aaa,b.bbb,c.ccc,x.xxxxxxx,y.yyyyyyy,z.zzz,w.www*hh<CR><LF>		
a.aaa	is the heading in degrees	3 digits after the decimal point
b.bbb	is the roll in degrees	3 digits after the decimal point
c.ccc	is the pitch in degrees	3 digits after the decimal point
x.xxxxxxx	is the latitude in degrees	7 digits after the decimal point
y.yyyyyyy	is the longitude in degrees	7 digits after the decimal point
z.zzz	is the altitude in meters	3 digits after the decimal point
w.www	is the heave in meters	3 digits after the decimal point
hh	is the checksum	

INDYN

Standard: Output NMEA compatible.

Data sent: Position, Heading, Heading, Roll, Pitch, Heading rate, Roll rate, Pitch rate, Longitudinal speed.

Data frame: ASCII frame.

$INDYN,x.xxxxxxxx,y.yyyyyyyy,z.zzz,h.hhh,r.rrr,p.ppp,a.aaa,b.bbb,c.ccc,s.sss*hh<CR><LF>

x.xxxxxxxx	is the latitude in degrees	Float, 8 digits after decimal point
y.yyyyyyyy	is the longitude in degrees	Float, 8 digits after decimal point
z.zzz	is the altitude in meters	Float, 3 digits after decimal point
h.hhh	is the heading in degrees	Float, 3 digits after decimal point
r.rrr	is the roll in degrees (positive for port up)	Float, 3 digits after decimal point
p.ppp	is the pitch in degrees (positive when bow down)	Float, 3 digits after decimal point
a.aaa	is the heading rate in °/s	Float, 3 digits after decimal point
b.bbb	is the roll rate in °/s (positive when roll increase)	Float, 3 digits after decimal point
c.ccc	is the pitch rate in °/s (positive when pitch increase)	Float, 3 digits after decimal point
s.sss	Speed XV1 in m/s (positive towards the bow)	Float, 3 digits after decimal point
hh	is the checksum	

INHDT

Standard: Output NMEA compliant.
Data sent: Heading in conventional or polar mode.
Data frame: ASCII frame

$INHDT,x.x,T*hh<CR><LF>		
x.x	Heading in degrees relative to True North or Polar North, as selected by the operator (**Note 1**)	Fixed point decimal with a variable number of digits of degrees and 1 fixed digit for decimal fraction of degrees. Range [0.0°; 359.9°]; Granularity = 0.1°
T	/	Always set to ASCII "T"
hh	NMEA checksum	

Note 1: The Operator selects whether True or Polar North should be output using the Web-Based Graphical User Interface.

IXBLUE STD BIN V2 V3

Standard: Binary protocol.

Important

The value of the heave without lever arm is set to 0 when selecting Smart heave in the web-based user interface with this protocol for the output settings.
The smart heave is delayed by 100 s.

Conventions

Validity time fields

Output mode

If the protocol is used as output, the validity time fields contain internal system time that can be synchronized with GNSS UTC time when UTC time is sent to system at regular intervals (i.e.: every second), see chapter 1.1.

Input mode

If the protocol is used as input, the validity time field is used to provide external sensor validity time, formatted in steps of 100 µs. Three cases can be distinguished:

- Timestamp shall be positive or null to send timestamp ([0:863999999] steps of 100 µs). In this case, used timestamp corresponds to the transmitted timestamp.
- Timestamp shall be negative to indicate sensor delay. In this case, used timestamp correspond to reception time minus transmitted delay.
- Timestamp shall be 0x7FC0000 if no time information has to be sent. In this case, external sensor timestamp corresponds to reception timestamp
- Data types

Each telegram description uses the following convention:

Type name	Description
Byte	Unsigned 8 bit integer
Character	Signed 8 bit integer
Word	Unsigned 16 bit integer
Short	Signed 16 bit integer
DWord	Unsigned 32 bit integer
Long	Signed 32 bit integer
Float	IEEE Float 32 bits
Double	IEEE Float 64 bits

All 16 and 32 bits integers are represented in Big Endian convention (MSB sent first).

Direction

Given that the protocol can be used both as output or input, the following conventions is used to indicate if a data is used in output mode, input mode or both (mainly apply on bit masks):

Type name	Description
I	Filled/Used in input mode only
O	Filled/Used in output mode only
I/O	Filled/Used in both input and output mode

Frames

Different frames are used to express data, see chapter 1.3

Altitude convention

Altitude can be referenced to Geoïdal model (mean sea level) if managed by the GNSS input. In this case, geoïdal separation field is a valid IEEE float that contains distance between geoid and ellipsoid at the current position. Otherwise if GNSS does not manage mean sea level altitude, the altitude field if referenced from ellipsoid and geoidal separation field contains NaN value 0x7FC00000.

DVL speed compensation in INS using sound speed

When available, INS will use both the sound speed calculated by the DVL (using temperature, depth, salinity) expressed as CDVL and the speed of sound measured by an external sensor (CTD, SVP) expressed as CEXT.
The corrected DVL speed will be calculated using the following formula:

$$V_{corrected} = \frac{C_{EXT}}{C_{DVL}} \cdot V_{DVL}$$

Protocol description

The iXblue Std Bin protocol can be used both as output or input protocol. The output telegram is used to get navigation data and received external sensors from the system. The input telegram is used to send external sensors data to the system.

Protocol structure

The structure of the protocol is the same whether it's used as input or output. Each protocol frame contains a header message, followed the message body that can contain a combination of different data blocks.
When used as output, the message body contains both navigation data blocks and external sensors data blocks (external sensors received by the system). When used as input, the message body only contains external sensors data blocks (to send to the system).
Additional command frames can be sent to the system, using a specific command header, to send parameters to the product. The system will reply to these command frames with answer frames starting with answer header as detailed in the next chapters.
Each message is terminated by a checksum to validate all the content of the telegram. This checksum is a DWord unsigned sum of all frame bytes, including the header and excluding the checksum itself.

Header	Body	Checksum

Protocol Use

As described in the following paragraphs, the protocol header is slightly different in input and output modes. However, the input mode manages both the input telegram coded as specified hereafter (including input header format) and the output telegram coming from another INS (including output header format). This enables to loopback one system to another using iXblue Std Bin protocol.

Message Headers

There are four types of headers in this protocol:

- Command header, used to send commands to the system
- Answer header, used to contain responses to a command sent to the system
- Output navigation data header, used to contains system navigation data and external sensor copies
- Input sensor data header, used to send external sensors to the system

The headers hold the protocol version, and for navigation messages, the bit masks that indicate which data block is present, the telegram size and other information depending on protocol direction.

Input message header for commands

Input messages that contain commands to be sent to the system are formatted using following header (available from version 3 only):

Data	Format	Units
Header 1	Byte	Value: ‘C’
Header 2	Byte	Value: ‘M’
Protocol version	Byte	0x03
Total telegramsize	Word	Number of bytes of the message including 7 byte header

The body part of each command message contains a single command formatted as detailed in section 1. They are used to send parameters to the system.

Output message header for answers to commands

Output messages that contain answers to commands that were sent to the system are formatted using following header (available from version 3 only):

Data	Format	Units
Header 1	Byte	Value: ‘A’
Header 2	Byte	Value: ‘N’
Protocol version	Byte	0x03
Total telegram size	Word	Number of bytes of the message including 7 byte header

Output message header for navigation data

Version 0x2 header

Data	Format	Units
Header 1	Byte	Value: 'I'
Header 2	Byte	Value: 'X'
Protocol version	Byte	0x02
Navigation data blocks bit mask	DWord	Bloc combination identification bit mask Bit is set to 1 when the bloc is output, 0 otherwise.
External data bit mask	DWord	Bloc combination identification bit mask Bit is set to 1 when the bloc is output, 0 otherwise.
Total telegram size	Word	Number of bytes of the message including 21 byte header
Navigation data validity time	Dword	Validity time in steps of 100 ms
Counter	DWord	Cycling counter inside [0:232-1]

Version 0x3 header

Navigation data output telegram header version 3 is formatted as follows:

Data	Format	Units
Header 1	Byte	Value: 'I'
Header 2	Byte	Value: 'X'
Protocol version	Byte	0x03
Navigation data blocks bit mask	DWord	Bloc combination identification bit mask Bit is set to 1 when the bloc is output, 0 otherwise.
Extended Navigation data block bit mask	DWord	Bloc combination identification bit mask Bit is set to 1 when the bloc is output, 0 otherwise.
External data bit mask	DWord	Bloc combination identification bit mask Bit is set to 1 when the bloc is output, 0 otherwise.
Total telegram size	Word	Number of bytes of the message including 25 byte header
Navigation data validity time	Dword	Validity time in steps of 100 µs
Counter	DWord	Cycling counter inside [0:2^{32}-1]

Navigation data input message header

Navigation data input messages are used to send external sensor data to the system using the same structure as output messages.

Version 0x2 header

On protocol version 2, navigation data messages are formatted using following header:

Data	Format	Units
Header 1	Byte	Value: 'I'

Data	Format	Units
Header 2	Byte	Value: 'X'
Protocol version	Byte	0x02
Navigation data blocks bit mask	DWord	Unused in input mode. Shall be set to 0x00000000
External data bit mask	DWord	Bloc combination identification bit mask Bit is set to 1 when the bloc is output, 0 otherwise.
Total telegram size	Word	Total telegram bytes number: 21 bytes header + X bytes message + 4 bytes checksum
External sensors time stamp reference	Byte	External sensors timestamp: 0x00: UTC time 0x01: INS System Time: Internal use
RFU	7 Bytes	Shall be set to 0x00.

Version 0x3 header

On protocol version 3, navigation data messages are formatted using following header:

Data	Format	Units
Header 1	Byte	Value: 'I'
Header 2	Byte	Value: 'X'
Protocol version	Byte	0x03 for version 0x03
Navigation data blocks bit mask	DWord	Unused in input mode. Shall be set to 0x00000000
Extended navigation data blocks bit mask	DWord	Unused in input mode. Shall be set to 0x00000000
External data bit mask	DWord	Bloc combination identification bit mask Bit is set to 1 when the bloc is output, 0 otherwise.
Total telegram size	Word	Total telegram bytes number: 25 bytes header + X bytes message + 4 bytes checksum
External sensors time stamp reference	Byte	External sensors timestamp: 0x00: UTC time 0x01: INS System Time: Internal use
RFU	7 Bytes	Shall be set to 0x00.

Bit Masks description for navigation data messages

Navigation data blocks

Following table describes the navigation data blocks and their corresponding bit number:

Bit n°	Internal Block description	Dir.
0	Attitude & Heading	I/O
1	Attitude & Heading standard deviation	O
2	Real time Heave/Surge/Sway	O
3	Smart HeaveTM	O
4	Heading/Roll/Pitch Rate	O
5	Rotation rate in vessel frame, compensated from Earth rotation	I/O
6	Acceleration in vessel frame, compensated from gravity	O
7	Position	I/O
8	Position standard deviation	O
9	Speed in geographic frame	I/O
10	Speed std dev in geographic frame	O
11	Current in geographic frame	O
12	Current std dev in geographic frame	O
13	System date	O
14	INS sensor status	O
15	INS algorithm status	O

Bit n°	Internal Block description	Dir.
16	INS system status	O
17	INS user status	O
21	Heave, surge and sway speeds	O
22	Speed in vessel frame	O
23	Acceleration in geographic frame not compensated from gravity	I/O
24	Course and speed over ground	O
25	Temperatures (ACC/FOG/ANA)	O
26	Attitude quaternion data	O
27	Attitude quaternion standard deviation	O
28	Raw accelerations in vessel frame not compensated from gravity	O
29	Accelerations standard deviation vessel frame	O
30	Rotation rate standard deviation (vessel frame)	O

Extended navigation data blocks

Following table describes the extended navigation data blocks and their corresponding bit number. This section is only available from protocol version 3:

Bit n°	Internal Block description	Dir.
0	Rotation accelerations in vessel frame	O
1	Rotation acceleration standard deviation	O
2	Raw rotation rate in vessel frame, not compensated from Earth rotation	I/O

External sensor data blocks

Following table details the external sensor and status blocks and their corresponding bit number:

Bit n°	Block description	Dir.
0	UTC data	I/O
1	GNSS1 data	I/O
2	GNSS2 data	I/O
3	manual GNSS data	I/O
4	EMLOG1 data	I/O
6, 7, 8	USBL1 - 3 data	I/O
9	Depth data	I/O
10	DVL Ground Speed	I/O
11	DVL Water Speed	I/O
12	Sound Velocity	I/O
14 - 17	LBL data (LBL1 to LBL4)	I/O

Bit n°	Block description	Dir.
21	DVL2 Ground Speed	I/O
22	DVL2 Water Speed	I/O

Navigation data blocks

The navigation data blocks are used in output mode only, except for Attitude and Heading data block (bit n°0), body rotation rates data block (bit n°5), position data block (bit n°7), speed in geographical frame data block (bit n°9) and acceleration in geographical frame data block (bit N°23) from V3.

Attitude & Heading data block (bit n°0)

Attitude and Heading relate to Subsea vehicle reference frame.

Data	Format	Units
Heading	Float	In degrees, inside [0°:360°[
Roll	Float	In degrees, positive when port up, inside [-180°:180°[
Pitch	Float	In degrees, positive bow down, inside [-90°:90°]

Attitude & Heading standard deviation data block (bit n°1)

Data	Format	Units
Heading Standard deviation	Float	In degrees, inside [0°:360°[
Roll Standard deviation	Float	In degrees, inside [0°:360°[
Pitch Standard deviation	Float	In degrees, inside [0°:360°[

Heave/Surge/ Sway data block (bit n°2)

Data	Format	Units
Real time heave without lever arm(*)	Float	In meters, positive up (XV3H direction)
Real time heave at selected lever arm	Float	In meters, positive up (XV3H direction)
Real time surge at selected lever arm	Float	In meters, positive forward in XV1H
Real time sway at selected lever arm	Float	In meters, positive port side in XV2H

(*) if the user specified the COG lever arm, this field corresponds to COG heave. Otherwise, if COG lever arm is not specified, this is the heave at the system position (i.e. center of body frame).

Smart Heave™ data block (bit n°3)

This block contains Smart Heave™ data and related time.

Data	Format	Units
Smart Heave Time(*)	Dword	Smart heave validity time (steps of 100 µs)
Smart Heave™ At primary lever arm (**)	Float	In meters, positive up (XV3H direction)

(*) The smart heave is delayed by 100s. This field contains the time of validity for the Smart Heave.
(**) The smart heave is only available on primary lever arm.

Heading/Roll/Pitch rate data block (bit n°4)

This block contains attitude rates data (derivative of heading, roll and pitch):

Data	Format	Units
Heading rotation rate	Float	In °/s, positive when heading increases
Roll rotation rate	Float	In °/s, positive when port going up
Pitch rotation rate	Float	In °/s, positive when bow going down

Body rotation rates data block in Subsea vehicle frame (bit n°5)

This block contains rotation rates in vessel frame XV1, XV2, XV3, compensated from earth rotation, gyros biases and scale factors:

Data	Format	Units
Rotation rate XV1	Float	In °/s, positive when port is going up
Rotation rate XV2	Float	In °/s, positive when bow is going down
Rotation rate XV3	Float	In °/s, positive counter clock wise around XV3

Accelerations data block in Subsea vehicle frame (bit n°6)

This block contains accelerations at primary lever arm in vessel frame XV1, XV2, XV3, compensated from gravity:

Data	Format	Units
Acceleration XV1	Float	In m/s^2, positive forward
Acceleration XV2	Float	In m/s^2, positive toward port
Acceleration XV3	Float	In m/s^2, positive up

Linear acceleration expressed in Subsea vehicle frame, noted:

$$\underline{a}_v = \frac{d}{dt}(\underline{v}_v)$$

It is computed by compensating gravity and Coriolis acceleration from raw measured accelerations as body frame is not Galilean.

Position data block (bit n°7)

This block contains position data in WGS84 frame at selected lever arm:

Data	Format	Units
Latitude	Double	In degree, positive North, inside [-90°:90°]
Longitude	Double	In degree, increasing toward East, inside [0°:360°[
Altitude reference	Byte	0: Geoid (Mean Sea Level), 1: Ellipsoid
Altitude	Float	In meters, positive up

Position standard deviation data block (bit n°8)

This block contains position standard deviation data:

Data	Format	Units
North standard deviation	Float	In meters
East standard deviation	Float	In meters
North/East correlation	Float	No unit
Altitude standard deviation	Float	In meters

Speed data block in geographic frame (bit n°9)

This block contains speed data at primary lever arm in geographical frame:

Data	Format	Units
North velocity	Float	In m/s, positive north
East velocity	Float	In m/s, positive east
Up velocity	Float	In m/s, positive up

Speed standard deviation data block in geographic frame (bit n°10)

This block contains speed standard deviation data:

Data	Format	Units
North velocity std. deviation	Float	In m/s
East velocity std. deviation	Float	In m/s
Up velocity std. deviation	Float	In m/s

Current data block in geographic frame (bit n°11)

This block contains estimated current data:

Data	Format	Units
North current std. deviation	Float	In m/s
East current std. deviation	Float	In m/s

System date data block (bit n°13)

This is INS internal date if system is not date synchronized to GNSS otherwise it is ZDA date.

Data	Format	Units
Day	Byte	Day inside [0:31], 0 if unavailable
Month	Byte	Month inside [0:12], 0 if unavailable
Year	Word	Year inside [0:65535], 0 if unavailable

INS Sensor Status (bit n°14)

This block contains sensor statuses detailed in section 5:

Data	Format	Units
Sensor Status 1	DWord	N/A
Sensor Status 2	DWord	N/A

INS Algorithm Status (bit n°15)

This block contains INS algorithm statuses detailed in section 5:

Data	Format	Units
Algorithm Status 1	DWord	N/A
Algorithm Status 2	DWord	N/A
Algorithm Status 3	DWord	N/A
Algorithm Status 4	DWord	N/A

INS System Status (bit n°16)

This block contains INS system status detailed in section 5:

Data	Format	Units
System Status 1	DWord	N/A
System Status 2	DWord	N/A
System Status 3	DWord	N/A

INS User Status (bit n°17)

This block contains INS user status detailed in section 5:

Data	Format	Units
User Status	DWord	N/A

Heave/Surge/Sway speed data block (bit n°21)

Data	Format	Units
Real time heave speed (at selected lever arm)	Float	In m/s, positive up in XV3H direction
Surge speed (at selected lever arm)	Float	In m/s, positive forward in XV1H
Sway speed (at selected lever arm)	Float	In m/s, positive port side in XV2H

Speed data block in vessel frame (bit n°22)

This block contains speed data at primary lever arm in Subsea vehicle frame:

Data	Format	Units
XV1 velocity	Float	In m/s, positive along XV1
XV2 velocity	Float	In m/s, positive along XV2
XV3 velocity	Float	In m/s, positive along XV3

Acceleration data block in geographic frame (bit n°23)

This block contains accelerations at primary lever arm in geographical frame, not compensated from gravity:

Data	Format	Units
North acceleration	Float	In m/s^2, positive North
East acceleration	Float	In m/s^2, positive East
Vertical acceleration	Float	In m/s^2, positive up

Course and speed over ground (bit n°24)

This block contains course and speed over ground data, at primary lever arm:

Data	Format	Units
Course over ground	Float	In degrees, inside [0°:360°[
Speed over ground	Float	In m/s

Temperatures (bit n°25)

This block contains the average temperature data:

Data	Format	Units
Mean temperature FOG: Average(TEMP_FOG_X, TEMP_FOG_Y, TEMP_FOG_Z)	Float	In °Celsius
Mean temperature ACC: Average(TEMP_ACC_X, TEMP_ACC_Y, TEMP_ACC_Z)	Float	In °Celsius
Temperature [TEMP_ANA] (Sensor board temperature)	Float	In °Celsius

Attitude quaternion (bit n°26)

This block contains the attitude quaternion representation:

Data	Format	Units
Attitude quaternion component q0	Float	N/A
Attitude quaternion component q1	Float	N/A
Attitude quaternion component q2	Float	N/A
Attitude quaternion component q3	Float	N/A

Refer to the Appendix for details on quaternions definitions, operations and relationship with Euler angles.

Attitude quaternion standard deviation (bit n°27)

This block contains the standard deviations of attitude quaternion:

Data	Format	Units
Attitude quaternion std dev component	Float	N/A
Attitude quaternion std dev component	Float	N/A
Attitude quaternion std dev component	Float	N/A

Refer to the end of this chapter for details on quaternions definitions, operations and relationship with Euler angles.

Raw acceleration in Subsea vehicle frame (bit 28)

This block contains the acceleration in Subsea vehicle frame, at primary lever arm, not compensated from gravity:

Data	Format	Units
Acceleration XV1	Float	In m/s^2, positive forward
Acceleration XV2	Float	In m/s^2, positive toward port
Acceleration XV3	Float	In m/s^2, positive up

Acceleration standard deviation in Subsea vehicle frame (bit 29)

This block contains the acceleration standard deviation in Subsea vehicle frame:

Data	Format	Units
Acceleration XV1 std dev	Float	In m/s^2
Acceleration XV2 std dev	Float	In m/s^2
Acceleration XV3 std dev	Float	In m/s^2

Rotation rate standard deviation in Subsea vehicle frame (bit 30)

This block contains the rotation rate standard deviation in Subsea vehicle frame:

Data	Format	Units
Rotation rate XV1 std dev	Float	In °/s
Rotation rate XV2 std dev	Float	In °/s
Rotation rate XV3 std dev	Float	In °/s

Extended navigation data blocks

Rotation accelerations in Subsea vehicle frame (bit 0)

This block contains the rotation accelerations (derivative of compensated rotation rates) in Subsea vehicle frame:

Data	Format	Units
Raw rotation acceleration XV1	Float	In $°/s^2$
Raw rotation acceleration XV2	Float	In $°/s^2$
Raw rotation acceleration XV3	Float	In $°/s^2$

Rotation acceleration standard deviation in Subsea vehicle frame (bit 1)

This block contains the rotation acceleration standard deviation in Subsea vehicle frame:

Data	Format	Units
Raw rotation acceleration std dev XV1	Float	In $°/s^2$
Raw rotation acceleration std dev XV2	Float	In $°/s^2$
Raw rotation acceleration std dev XV3	Float	In $°/s^2$

Raw rotation rate in Subsea vehicle frame (bit 2)

This block contains raw rotation rates in Subsea vehicle frame, not compensated from Earth rotation:

Data	Format	Units
Raw rotation rate XV1	Float	In °/s
Raw rotation rate XV2	Float	In °/s
Raw rotation rate XV3	Float	In °/s

External sensors data blocks

The external sensors data blocks are used in both output and input mode.

- When used as an output, the sensor blocks are sent only when the corresponding data is received by the system. The data validity time corresponds to the internal time or the UTC time if the system is UTC synchronized.
- When used as input, the sensor blocks are sent only to sent external sensor data to the system. The time tag data corresponds to the sensor validity time.

UTC data block (bit n°0)

Last UTC data block received or UTC data block to send:

Data	Format	Units
Data validity time	Dword	Time tag in steps of 100µs
UTC Source	Byte	0: UTC1, 1: UTC2

GNSS1, GNSS2 and Manual GNSS data blocks (bits n°1,2,3)

Last GNSS1, GNSS2 or Manual GNSS data received or GNSS data to send:

Data	Format	Units
Data validity time	Long	Time tag in steps of 100 µs
GNSS identification	Byte	0: GNSS1 1: GNSS2 2: Manual GNSS
GNSS quality	Byte	Refer to Table 1 Fixed to 1 for manual GNSS
Latitude	Double	In degrees, positive North, inside [-90° :90°]
Longitude	Double	In degrees, positive East, inside [0° :360°[
Altitude	Float	In meters, positive up, in geoid reference
Latitude standard deviation	Float	In meters
Longitude standard deviation	Float	In meters
Altitude standard deviation	Float	In meters
Latitude/Longitude covariance	Float	In meters^2 Fixed to 0 for manual GPS
Geoidal separation	Float	Meters

Table 6 - INS interpretation of GNSS quality

Quality indicator	Corresponding SD attributed to GNSS position fix if no GST is received	Positioning system mode indicator
0	Data invalid	N/A
1	10 m	Natural
2	3 m	Differential
3	10 m	Military
4	0.1 m	RTK
5	0.3 m	Float RTK
6 - 255	Data invalid	Other mode

EMLOG1 and EMLOG2 data blocks (bits n°4,5)

Last EMLOG1 or EMLOG2 data received or EMLOG data to send:

Data	Format	Units
Data validity time	Long	Time tag in steps of 100 µs
EMLOG ID	Byte	0: EMLOG1 1: EMLOG2
XV1 longitudinal water speed	Float	In m/s, positive forward
XV1 speed standard deviation	Float	In m/s

USBL1, USBL2 and USBL3 data blocks (bits n°6,7,8)

Last USBL1, USBL2 or USBL3 data received or USBL data to send:
(*) If beacon ID length is less than 8 bytes, it is padded with null (\0) ASCII characters at the end.

Depth data block (bit n°9)

Last DEPTH data received or DEPTH data to send:

Data	Format	Units
Data validity time	Long	Time tag in steps of 100 µs
Depth	Float	In Meters
Depth standard deviation	Float	In Meters

DVL1, DVL2 Ground speed data block (bits 10,21)

Last DVL1 or DVL2 data received or DVL data to send:

Data	Format	Units
Data validity time	Long	Time tag in steps of 100 µs
DVL identification	Byte	0: DVL 1

Data	Format	Units
		1: DVL 2
XV1 longitudinal ground speed*	Float	In m/s
XV2 transverse ground speed*	Float	In m/s
XV3 vertical ground speed*	Float	In m/s
DVL speed of sound	Float	In m/s
DVL altitude (bottom range)	Float	In Meter
XV1 speed standard deviation	Float	In m/s
XV2 speed standard deviation	Float	In m/s
XV3 speed standard deviation	Float	In m/s

XV1, XV2 XV3 speeds are corrected with external Sound velocity sensor measurement.

DVL1, DVL2 Water speed data block (bits 11, 22)

Last DVL1 or DVL2 data received or DVL data to send:

Data	Format	Units
Data validity time	Long	Time tag in steps of 100 µs
DVL identification	Byte	0: DVL 1 1: DVL 2
XV1 longitudinal water speed	Float	In m/s
XV2 transverse water speed	Float	In m/s
XV3 vertical water speed	Float	In m/s
DVL speed of sound	Float	In m/s
XV1 speed standard deviation	Float	In m/s
XV2 speed standard deviation	Float	In m/s
XV3 speed standard deviation	Float	In m/s

External Sound velocity data block (bit n°12)

Last external sound velocity received from a sound velocity probe, or sound velocity to send:

Data	Format	Units
Data validity time	Long	Time tag in steps of 100 µs
External sensor speed of sound	Float	In m/s

LBL data blocks (bit n°14)

Last received LBL data (up to 10 LBL datablocks in STD BIN frame), or LBL data to send:
(*) If beacon ID length is less than 8 bytes, it must be padded with null (\0) ASCII characters at the end.

IXSEA ICCB1

Standard: Binary protocol. Custom to interface to Interface and Control Cabinet Box (ICCB).
Data sent: Status, Heading, Roll, Pitch, Horizontal speed over ground, Checksum.
Data frame: The frame contains 7 fields - 15 bytes, MSB are sent first.

Message <F0><F1><F2>…..<F6>			
Field 0	Byte 0	0x02	Start of sentence
Field 1	Byte 1	Status	System status byte (see table below)
Field 2	Bytes 2 to 4	Heading	Unsigned 24 bits integer 224 = 360°
Field 3	Bytes 5 to 7	Roll	Signed 24 bits integer +/-223 = +/-180° Sign “+” when port side up
Field 4	Bytes 8 to 10	Pitch	Signed 24 bits integer +/-223 = +/-180° Warning: Opposite sign of INS usual convention Sign “+” when bow up
Field 5	Bytes 11 to 12	Horizontal speed norm over ground	Unsigned 16 bits integer LSB= 1 cm/s
Field 6	Bytes 13 to 14	Checksum CRC16-Modbus	Computed on bytes 1 to 12

The table below details status byte bit definition according to user status byte table:

Status byte bit index	**Description**	**Corresponding user status bits**
0	‘1’ = Data is invalid ‘0’ = Data is valid	27 or 9 or 10 or 11 or 13 or 14 or 15 or 30 or 31
1	‘1’ = Initial alignment ‘0’ = End of initial alignment	27
2	‘1’ = Fine alignment in progress ‘0’ = End of fine alignment	28
3	‘1’ = Internal sensor error ‘0’ = Internal sensor OK	9 or 10 or 11
4	‘1’ = Algorithm error ‘0’ = Algorithm OK	13 or 14 or 15
5	‘1’ = Input stream error ‘0’ = Input stream OK	16 or 17 or 18 or 19 or 20
6	‘1’ = Output stream error ‘0’ = Output stream OK	21 or 22 or 23 or 24 or 25
7	‘1’ = External sensor valid ‘0’ = External sensor lost	0 or 1 or 3 or 4 or 5 or 6 or 7

IXSEA TAH

Standard: Output NMEA 0183 compatible.
Data sent: Time, Roll, Pitch, Heading, heading rotation rate, Attitude and heading angles, Heave, surge, sway movement.
Data frame:

$PHOCT,01,hhmmss.sss,G,AA,HHH.HHH,N,eRRR.RRR,L,ePP.PPP,K,eFF.FFF,
M,eHH.HHH,eSS.SSS,eWW.WWW,eZZ.ZZZ,eYY.YYY,eXX.XXX,eQQQ.QQ*hh

$PHOCT	is the header
01	is the protocol version identifier
hhmmss.sss	is the UTC valid time of observations hh is the local zone hours (from 00 to 23 hours) mm is the local zone minutes (from 00 to +59) ss.sss is the local zone seconds (from 00.000 to +59.999)
G	is UTC Time status: T=Valid E=Invalid (*)
AA	is the latency for heading, roll, pitch AA= 03 (data latency in ms to be used if UTC time of observations is not used) (**)
HHH.HHH	is the true heading in degree (from 000.000 to 359.999)
N	is the True Heading status: T=Valid E=Invalid I=Initializing
eRRR.RRR	is the Roll in degree (e= +/-, positive if port side up) from -180.000 to +180.000
L	is the Roll status: T=Valid E=Invalid I=Initializing
ePP.PPP	is the Pitch degree (e= +/-, positive if bow down), from –90.000 to +90.000
K	is the Pitch status: T=Valid E=Invalid I=Initializing
eFF.FFF	is the Heave without Primary Lever arms applied in meters (e= +/-, positive up) from -99.999 to +99.999
M	is the Heave status (also used for surge, sway & speed) : T=Valid E=Invalid I=Initializing
eHH.HHH	is the Heave with Chosen lever arms applied in meters (e= +/-, positive up) from –99.999 to +99.999
eSS.SSS	is the Surge with Lever arms applied in meters, (e= +/-) from –99.999 to +99.999
eWW.WWW	is the Sway with Lever arms applied in meters, (e= +/-) from –99.999 to +99.999
eZZ.ZZZ	is the Heave speed with Lever arms in meters/s, (e= +/-) from –99.999 to +99.999
eYY.YYY	is the Surge speed with Lever arms in meters/s, (e= +/-) from –99.999 to +99.999
eXX.XXX	is the Sway speed with Lever arms in meters/s, (e= +/-) from –99.999 to +99.999

eQQQQ.QQ	is the heading rate of turns in degree/minute, (e= +/-) from –9999.99 to +9999.99 (****)
hh	is the NMEA checksum: hexadecimal encoded XOR of all bytes excluding the starting character '$' and the stop one '*

(*) UTC time is valid (G=T) if both PPS pulse and &ZDA telegram are received and valid. If either PPS or ZDA telegram are not received or valid, UTC time flag is invalid: G=E.
(**) The attitude and heading latency for Rovins data is as follow:

		Output latency
Serial ports	All	2.35 ms
Ethernet ports (in priority order)	First	2.95 ms
	Second	3.45 ms
	Third	3.95 ms
	Fourth	4.45 ms
	Fifth	4.95 ms

Be aware that latency on Ethernet data is guaranteed only for point to point link.
If INS is time synchronized, time stamp is used and latency is then not relevant.
(***) For INS Heave with Primary lever arm and Heave with chosen lever arm are output (Primary, A, B, C).
(****) Rotation rate resolution is limited to 3.6°/h

KINETIC SCIENTIFIC

Standard: Binary protocol.

Data frame: 12 fields, 24 bytes. For data coded on several bytes, the bytes are sent MSB first.

Message <F0><F1><F2>…..<F16>				
Field	**Byte(s)**	**Data**	**Description**	**Note(s)**
0	0	Header	Fixed value = 0xA5	
1	1	Status	0x00: Full Accuracy 0x01: Fine Alignment (bit 2 of Algorithm status 1) 0x02: Alignment (bit 1 of Algorithm status 1) 0x03: ACC/FOG anomaly (bit 9 or 10 of User status) 0x04: Failure Mode (bit 31 of User status)	
2	2-3	Sequence	Incremental 16 bits integer Counter: 0 to 65535	
3	4-5	Heading	Unsigned 16 bits integer Range = [0,360°], LSB=0.01°	
4	6-7	Roll	Signed 16 bits integer Range = [-180°,+180°], LSB=0.01° Positive for left-port side up.	Note 1
5	8-9	Pitch	Signed 16 bits integer Range = [-90°,+90°], LSB=0.01° Positive for front side-bow down.	Note 1
6	10-11	Heading rate	Signed 16 bits integer Range = [-256°/s,+256°/s], LSB=2^{-7}°/s	Note 1
7	12-13	Roll rate		
8	14-15	Pitch rate		
9	16-17	XV1 Acceleration	Not compensated from g Signed 16 bits integer Range = [-16g,16g], LSB=2^{-11}g	Note 1
10	18-19	XV2 Acceleration		
11	20-21	XV3 Acceleration		
12	22-23	Checksum	Fletcher Checksum	Note 2

Note 1: Saturation values. When maximum or minimum value is reached, the output is set to maximum or minimum value.

Note 2: Fletcher Checksum Algorithm

Following pseudo-code implementation is of a modified Fletcher Checksum.

Compute performs the computation of the Fletcher Sum over a vector of bytes and returns a 16-bit value.

Append is passed a vector of bytes and a packet length. It presumes that the last two bytes of the packet are to hold the 16-bit checksum. It calls compute on the first Length-2 bytes of the packet and then fills in the last two bytes with the value returned by Compute.

Check is passed a vector of bytes, the last two of which are presumed to be the Fletcher Sum of the preceeding bytes. It uses Compute to determine the correctness of the packet and returns a Boolean with this result.

```
type Packet is array of byte;

function Check (The_Packet : Packet;
                The_Length : Integer) return Boolean is
begin
   return Compute (The_Packet, The_Length) = 0;
end Check;


procedure Append (The_Packet : Packet;
                  The_Length : Integer) is
   Sum : Integer;
begin
   Sum := Compute (The_Packet, The_Length - 2;
   The_Packet [The_Length - 2] := Sum and 255;
   The_Packet [The_Length - 1] := Sum >> 8;
end Append;


function Compute (The_Packet : Packet;
                  The_Length : Integer) return Integer is
   N : Integer;
   Sum1 : Integer;
   Sum2 : Integer;
   R_Upper : Integer
   R_Lower : Integer

begin
   Sum1 := 255;
   Sum2 := 255;
   for N in 0 .. The_Length - 1 loop
      Sum1 := Sum1 + The_Packet [N];
      if Sum1 > 255 then
         Sum1 := Sum1 - 255;
      end if;
      Sum2 := Sum2 + Sum1;
      if Sum2 > 255 then
         Sum2 := Sum2 - 255;
      end if;
   end loop;
   R_Lower := Sum1 + Sum2;
   R_Lower := 255 - (R_Lower and 255) + (R_Lower >> 8);
   R_Upper := Sum1 + R_Lower;
   R_Upper := 255 - (R_Upper and 255) + (R_Upper >> 8);
   return (R_Upper << 8) + R_Lower;
end Compute;
```

An alternate version of the loop in Compute is:

```
for N in 0 .. The_Length - 1 loop
   Sum1 := (Sum1 + The_Packet [N]) mod 255;
   Sum2 := (Sum2 + Sum1) mod 255;
end loop;
```

Note that the Fletcher Checksum computed over the 24 ordered bytes (Values followed by the Checksum) has a value of (0x0000).

Hereafter are some Fletcher Checksum examples:

```
Example 1
Buffer 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
Checksum 00 00

Example 2
Buffer 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 10 11 12 13 14 15
Checksum 26 F1

Example 3
Buffer 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 10 11 12 13 14 15 16
Checksum 12 EF

Example 4
Buffer FF FE FD FC FB FA F9 F8 F7 F6 F5 F4 F3 F2 F1 F0 EF EE ED EC EB EA
Checksum D9 0E

Example 5
Buffer 11 1C 14 1A 10 1D 15 DB 0D 08 0E CA 11 1C 14 4A 19 12 1E D7 04 07
Checksum 87 59
```

KVH EXTENDED

Standard: ASCII telegram.
Data sent: Roll, Pitch, Heading and Heading Rate.
Data frame:

%P,R,H,Y<CR><LF>	
%	is the preamble character
P **Note 1**	is the pitch angle value X 10 in degrees Signed integer; range (-900° to 900°) Positive for front-side bow up Warning: Opposite sign of INS usual convention
R **Note 1**	is the roll angle value X 10 in degrees Signed integer; range (-1800° to 1800°) Positive for left-port side up
H	is the heading angle value X 10 in degrees Signed integer; range (0°to 3600°)
Y	is the heading rate value X 100 in degrees/s Signed integer; range (-9900 degrees/s ,9900 degrees/s) Positive when heading angle increases
<CR><LF>	is the end of the frame : 0x0D 0x0A (NO CHECKSUM)

Note 1: The KVH EXTENDED sign convention is ISO Convention.

	ISO Convention	Inverse ISO Convention	INS Convention
Roll > 0	for left-port side up	for left-port side down	for left-port side up
Pitch > 0	for front-side bow up	for front side-bow down	for front side-bow down

KM BINARY

Standard: Binary protocol.
Data frame: 132 bytes, octets, defined by a header and without terminating by checksum. The data are encoded in little endian. The floated data follows the IEEE-754 norm.
The table below describes the whole fields of KM Binary frame. The bracketed text means that the iXblue name of an protocol element is different from the KM Binary sensor input format.

Table 7 - Protocol structure

Element	Unit if applicable	Format	Size in bytes	Comments
Start ID		Byte x 4	4	This field is fixed. The tag of the frame start is #KMB
Size (Dgm Length)		Uint16	2	Frame size = 132
Version (Dgm version)		Uint16	2	Fixed value = 1
UTC seconds	s	Uint32	4	Integer part of time from the 1rst January 1970 in seconds
UTC nanoseconds	Ns	Uint32	4	Integer part of time from the 1rst January 1970 in nanoseconds
Status		Uint32	4	See table 2
Latitude	deg	Double	8	Positive in the northern hemisphere
Longitude	deg	Double	8	Positive in the eastern hemisphere
Ellipsoïd height	m	Float	4	
Roll	deg	Float	4	Positive port side up
Pitch	deg	Float	4	Positive bow up
Heading	deg	Float	4	Clockwise
Heave	m	Float	4	Positive downward
Roll rate	deg / s	Float	4	Positive if port side up movement
Pitch rate	deg / s	Float	4	Positive if bow up movement
Heading rate (Yaw rate)	deg / s	Float	4	
North speed (North velocity)	m / s	Float	4	Positive speed toward North
East speed (East velocity)	m / s	Float	4	Positive speed toward East

Element	Unit if applicable	Format	Size in bytes	Comments
Vertical Speed (Down velocity)	m /s	Float	4	Positive speed downward (diving)
Northing Standard deviation (latitude error)	m	Float	4	-1 if unavailable
Easting Standard deviation (longitude error)	m	Float	4	-1 if unavailable
Height Standard deviation (height error)	m	Float	4	-1 if unavailable
Roll Standard deviation (heading error)	deg	Float	4	-1 if unavailable
Pitch Standard deviation (heading error)	deg	Float	4	-1 if unavailable
Heading Standard deviation (heading error)	deg	Float	4	-1 if unavailable
Heave Standard deviation (heading error)	m	Float	4	-1 if unavailable
North acceleration	m / s^2	Float	4	Positive acceleration toward North, offset by the gravity
East acceleration	m / s^2	Float	4	Positive acceleration toward East, offset by the gravity
Vertical acceleration (Down acceleration)	m / s^2	Float	4	Positive acceleration downard (diving), offset by the gravity
Delayed heave UTC seconds	s	Uint32	4	Integer part of time from the 1rst January 1970 in seconds minus the integer part of the delay of the heave
Delayed heave UTC nanoseconds	ns	Uint32	4	Decimal part of the time from the 1rst January 1970 in nanoseconds minus the decimal part of the delay of the heave
Delayed heave	m	Float	4	Positive downward

Table 8 - Status word definition

Bit N°	Status	Condition
0	Invalid position and horizontal speed	ALG1_INS_ALIGNEMENT = 1
1	Invalid roll and pitch	(ALG1_INS_ALIGNEMENT = 1) OR (USR_INS_HRP_UNVALID = 1)
2	Invalid heading	
3	Invalid heave and vertical speed	ALG1_INS_ALIGNEMENT = 1
4	Invalid accelerations	
5	Invalid delayed heave	(ALG1_INS_ALIGNEMENT = 1) - 100s
6-15	Reserved	
16	Position or horizontal speed in degraded mode	(ALG1_ INS_ KALMAN = 1) AND (SYS3_SYSTEM_READY = 0)
17	Roll and pitch in degraded mode	(ALG1_INS_KALMAN = 1) AND (Roll standard deviation > 0.01° RMS)
18	Heading in degraded mode	(ALG1_ INS_ KALMAN = 1) AND (SYS3_SYSTEM_READY = 0)
19	Heave and vertical speed in degraded mode	
20	Acceeration in degraded mode	
21	Delayed heave in degraded mode	(SYS3_SYSTEM_READY = 0) - 100s

LONG BIN NAV HR

Standard : Binary protocol. Derived from the LONG BINARY NAV protocol with 3 supplementary bytes to increase resolution on HRP and on time.

Data sent: Time, Latitude, Longitude, Altitude, Heave, North Velocity , East Velocity, Down Velocity, Roll, Pitch, Heading, Heading rate, Roll rate, Pitch rate, Status and standard deviations for Latitude, Longitude, North Velocity, East Velocity, Down Velocity, Roll, Pitch and Heading data.

Data frame: The frame contains 26 fields - 61 bytes.

Message <F0><F1><F2>…..<F25> (Note 1)			
Field 0	Byte 0	'q' (0x71)	Synchronization byte
Field 1	Bytes 1 to 4	Data time tag	32 bits integer (seconds). Time of day (0-24 h)
Field 2	Bytes 5 to 6	Data time tag	Unsigned 16 bits integer (fraction of seconds). LSB = 15.26 µs.
Field 3	Bytes 7 to 10	Latitude (WGS84) '+': North of equator	Signed 32 bits integer $+/-2^{31} = +/- 180°$
Field 4	Bytes 11 to 14	Longitude (WGS84) '+': East of	Signed 32 bits integer $+/-2^{31} = +/- 180°$
Field 5	Bytes 15 to 18	Altitude** (WGS84) "+" Up	Signed 32 bits integer. LSB=1 mm
Field 6	Bytes 19 to 20	Heave "+" when down	Signed 16 bits integer. LSB= 1 mm Warning: Opposite sign of INS usual convention
Field 7	Bytes 21 to 22	North velocity	Signed 16 bits integer. LSB= 1 mm/s
Field 8	Bytes 23 to 24	East velocity	Signed 16 bits integer LSB= 1 mm/s
Field 9	Bytes 25 to 26	Down velocity	Signed 16 bits integer LSB= 1 mm/s
Field 10	Bytes 27 to 29	Roll Sign "+" when port side up	Signed 24 bits integer $+/-2^{23} = +/- 180°$
Field 11	Bytes 30 to 32	Pitch Sign "+" when bow up	Signed 24 bits integer $+/-2^{23} = +/- 180°$ Warning: Opposite sign of INS usual convention
Field 12	Bytes 33 to 35	Heading	Unsigned 24 bits integer $2^{23} = 180°$
Field 13	Bytes 36 to 38	Heading rotation rate****	Signed 24 bits integer $+/-2^{23} = +/- 180°/s$
Field 14	Bytes 39 to 41	Roll rotation rate****	Signed 24 bits integer $+/-2^{23} = +/- 180°/s$

Message <F0><F1><F2>.....<F25> (Note 1)			
Field 15	Bytes 42 to 44	Pitch rotation rate****	Signed 24 bits integer +/-2^{23} = +/- 180°/s Warning: Opposite sign of INS usual convention
Field 16	Bytes 45 to 48	User Status	4 Bytes INS User status
Field 17	Bytes 49 to 50	Latitude std. deviation	Unsigned 16 bits integer. LSB= 1 cm ; [0, 655 m]
Field 18	Bytes 51 to 52	Longitude std. deviation	Unsigned 16 bits integer. LSB= 1 cm ; [0, 655 m]
Field 19	Bytes 53 to 54	North velocity std. deviation	Unsigned 16 bits integer. LSB= 1 mm/s
Field 20	Bytes 55 to 56	East velocity std. deviation	Unsigned 16 bits integer. LSB= 1 mm/s
Field 21	Bytes 57 to 58	Down velocity std. deviation	Unsigned 16 bits integer. LSB= 1 mm/s
Field 22	Bytes 59 to 60	Roll std. deviation	Unsigned 16 bits integer 2^{15} = 90°
Field 23	Bytes 61 to 62	Pitch std. deviation	Unsigned 16 bits integer 2^{15} = 90°
Field 24	Bytes 63 to 64	Heading std. deviation	Unsigned 16 bits integer 2^{15} = 90°
Field 25	Bytes 65 to 66	Checksum (CRC)***	Computed on bytes 1 to 64

Note 1: MSB is sent first then LSB (big-endian convention).

*The precision of rotation rate data is limited to 3.6 deg/h to comply with export regulation.
** refer to section 1.2 for altitude management.
***CRC computation is given hereafter:

```
unsigned short blkcrc(unsigned char* bufptr, unsigned len)
{
unsigned char i;
unsigned short data;
unsigned short crc = 0xffff;
    if (len == 0)
        return ~crc;
    do
    {
        for (i = 0, data = (unsigned short)(0xff & *bufptr++); i < 8; i++, data >>= 1)
        {
        if ((crc & 0x0001) ^ (data & 0x0001))
        {
        crc = (crc >> 1) ^ 0x8408;
        }
        else
        {
        crc >>= 1;
        }
```

```
        }
        } while (--len);
        crc = ~crc;
        data = crc;
        crc = (crc << 8) | ((data >> 8) & 0xff);
        return crc;
    }
```

**** HRP rate values are averaged over the last sampling period for output rates from 0.1 to 50 Hz; for output rates faster than 50 Hz, instantaneous values are used.

LONG BIN NAV HR2

Standard: Binary protocol. Derived from the LONG BINARY NAV HR protocol with higher resolution on position data and User status replaced by Sensor, Algorithm and System Status.
Data sent: Time, Latitude, Longitude, Altitude, Heave, North Velocity , East Velocity, Down Velocity, Roll, Pitch, Heading, Heading rate, Roll rate, Pitch rate, Sensor/Algorithm and System Status, standard deviations for Latitude, Longitude, North Velocity, East Velocity, Down Velocity, Roll, Pitch and Heading data.
Data frame: The frame contains 21 fields - 95 bytes.

Message <F0><F1><F2>…..<F25> (Note 1)			
Field 0	Byte 0	‘q’ (0x71)	Synchronization byte
Field 1	Bytes 1 to 4	Data time tag	32 bits integer (seconds). Time of day (0-24 h)
Field 2	Bytes 5 to 6	Data time tag	Unsigned 16 bits integer (fraction of seconds). LSB = 15.26 µs.
Field 3	Bytes 7 to 14	Latitude (WGS84) ‘+’: North of equator	Signed 64 bits integer +/-263 = +/- 180°
Field 4	Bytes 15 to 22	Longitude (WGS84) ‘+’: East of	Signed 64 bits integer +/-263 = +/- 180°
Field 5	Bytes 23 to 26	Altitude** (WGS84) “+” Up	Signed 32 bits integer. LSB=1 mm
Field 6	Bytes 27 to 28	Heave “+” when down	Signed 16 bits integer. LSB= 1 mm Warning: Opposite sign of Rovins usual convention
Field 7	Bytes 29 to 30	North velocity	Signed 16 bits integer. LSB= 1 mm/s
Field 8	Bytes 31 to 32	East velocity	Signed 16 bits integer LSB= 1 mm/s
Field 9	Bytes 33 to 34	Down velocity	Signed 16 bits integer LSB= 1 mm/s
Field 10	Bytes 35 to 37	Roll Sign “+” when port side up	Signed 24 bits integer +/-223 = +/- 180°
Field 11	Bytes 38 to 40	Pitch Sign “+” when bow up	Signed 24 bits integer +/-223 = +/- 180° Warning: Opposite sign of Rovins usual convention
Field 12	Bytes 41 to 43	Heading	Unsigned 24 bits integer 223 = 180°
Field 13	Bytes 44 to 46	Heading rotation rate****	Signed 24 bits integer +/-223 =

Message <F0><F1><F2>.....<F25> (Note 1)			
			+/- 180°/s
Field 14	Bytes 47 to 49	Roll rotation rate****	Signed 24 bits integer +/-223 = +/- 180°/s
Field 15	Bytes 50 to 52	Pitch rotation rate****	Signed 24 bits integer +/-223 = +/- 180°/s
Field 16	Bytes 53 to 56	System Status 1	4 Bytes Rovins System Status 1 (see section 5)
Field 17	Bytes 57 to 60	System Status 2	4 Bytes Rovins System Status 2 (see section 5)
Field 18	Bytes 61 to 64	Algorithm Status 1	4 Bytes RovinsAlgo Status 1 (see section 5)
Field 19	Bytes 65 to 68	Algorithm Status 2	4 Bytes RovinsAlgo Status 2 (see section 5)
Field 20	Bytes 69 to 72	Sensor Status 1	4 Bytes Rovins Sensor Status 1 (see section 5)
Field 21	Bytes 73 to 76	Sensor Status 2	4 Bytes Rovins Sensor Status 2 (see section 5)
Field 22	Bytes 77 to 78	Latitude std. deviation	Unsigned 16 bits integer. LSB= 1 cm
Field 23	Bytes 79 to 80	Longitude std. deviation	Unsigned 16 bits integer. LSB= 1 cm
Field 24	Bytes 81 to 82	North velocity std. deviation	Unsigned 16 bits integer. LSB= 1 mm/s
Field 25	Bytes 83 to 84	East velocity std. deviation	Unsigned 16 bits integer. LSB= 1 mm/s
Field 26	Bytes 85 to 86	Down velocity std. deviation	Unsigned 16 bits integer. LSB= 1 mm/s
Field 27	Bytes 87 to 88	Roll std. deviation	Unsigned 16 bits integer 215 = 90°
Field 28	Bytes 89 to 90	Pitch std. deviation	Unsigned 16 bits integer 215 = 90°
Field 29	Bytes 91 to 92	Heading std. deviation	Unsigned 16 bits integer 215 = 90°
Field 30	Bytes 93 to 94	Checksum (CRC)***	Computed on bytes 1 to 92

Note 1 : MSB is sent first then LSB (big-endian convention).

*The precision of rotation rate data is limited to 3.6 deg/h to comply with export regulation.
** refer to section 1.2 for altitude management.
***CRC computation is given hereafter:

```
unsigned short blkcrc(unsigned char* bufptr, unsigned len)
{
unsigned char i;
unsigned short data;
unsigned short crc = 0xffff;
    if (len == 0)
        return ~crc;
    do
    {
        for (i = 0, data = (unsigned short)(0xff & *bufptr++); i < 8; i++, data >>= 1)
        {
        if ((crc & 0x0001) ^ (data & 0x0001))
        {
        crc = (crc >> 1) ^ 0x8408;
        }
        else
        {
        crc >>= 1;
        }
        }
        } while (--len);
        crc = ~crc;
        data = crc;
        crc = (crc << 8) | ((data >> 8) & 0xff);
        return crc;
    }
```

**** HRP rate values are averaged over the last sampling period for output rates from 0.1 to 50 Hz; for output rates faster than 50 Hz, instantaneous values are used.

LONG BINARY NAV

Standard: Binary protocol. Custom protocol.

Data sent: Time, Latitude, Longitude, Altitude, Heave, North Velocity, East Velocity, Down Velocity, Roll, Pitch, Heading, X1 Rotation Rate, X2 Rotation Rate, X3 Rotation Rate, Status and standard, deviations for Latitude, Longitude, North Velocity, East Velocity, Down Velocity, Roll, Pitch and Heading data.

Data frame: The frame contains 26 fields – 61 bytes.

Message <F0><F1><F2>…..<F25> (Note 1)			
Field 0	Byte 0	'q'	Synchronization byte
Field 1	Bytes 1 to 4	Data time tag	UTC_TIME ; 32 bits integer (seconds)
Field 2	Bytes 5 to 6	Data time tag	Unsigned 16 bits integer 0.0001 seconds (0 to 10000)
Field 3	Bytes 7 to 10	Latitude '+': North of equator	Signed 32 bits integer +/-2^{31} = +/- 180° (+/- 180 degrees)
Field 4	Bytes 11 to 14	Longitude '+': East of Greenwich	Signed 32 bits integer +/-2^{31} = +/- 180° (+/- 180 degrees)
Field 5	Bytes 15 to 18	Altitude**	Signed 32 bits integer (centimeters)
Field 6	Bytes 19 to 20	Heave "+" when down	Signed 16 bits integer (centimeters) Warning: Opposite sign of PHINS usual convention
Field 7	Bytes 21 to 22	North velocity	Signed 16 bits integer (centimeters/second)
Field 8	Bytes 23 to 24	East velocity	Signed 16 bits integer (centimeters/second)
Field 9	Bytes 25 to 26	Vertical velocity "+" when down	Signed 16 bits integer (centimeters/second) Warning: Opposite sign of INS usual convention
Field 10	Bytes 27 to 28	Roll Sign "+" when port side up	Signed 16 bits integer +/-2^{15} = +/- 180° (+/- 180 degrees)
Field 11	Bytes 29 to 30	Pitch Sign "+" when bow up	Signed 16 bits integer +/-2^{15} = +/- 180° (+/- 180 degrees) Warning: Opposite sign of INS usual convention
Field 12	Bytes 31 to 32	Heading	Unsigned 16 bits integer 2^{15} = 180° (0 to 360 degrees)

Note 1: MSB is sent first then LSB (big-endian convention).

*The precision of rotation rate data is limited to 3.6 deg/h to comply with export regulation.

** refer to section 1.2 for altitude management.

***CRC computation is given hereafter:

```
unsigned short blkcrc(unsigned char* bufptr, unsigned len)
{
unsigned char i;
unsigned short data;
```

```
unsigned short crc = 0xffff;
    if (len == 0)
        return ~crc;
    do
    {
        for (i = 0, data = (unsigned short)(0xff & *bufptr++); i < 8; i++, data >>= 1)
        {
        if ((crc & 0x0001) ^ (data & 0x0001))
        {
        crc = (crc >> 1) ^ 0x8408;
        }
        else
        {
        crc >>= 1;
        }
        }
        } while (--len);
        crc = ~crc;
        data = crc;
        crc = (crc << 8) | ((data >> 8) & 0xff);
        return crc;
    }
```

LONG BIN NAV SM

Standard: Binary protocol.

Data sent: Time, Latitude, Longitude, Heave, North Velocity, East Velocity, Heave speed, Roll, Pitch, Heading, Heading rate, Roll rate, Pitch rate, Status and standard deviations for Latitude, Longitude, North Velocity, East Velocity, Heave speed, Roll, Pitch and Heading data.

For multi-byte fields, the MSB is sent first (big-endian convention).

Data frame: The frame contains 25 fields – 63 bytes.

Message <F0><F1><F2>…..<F24>			
Field 0	Byte 0	Synchronization byte	Fixed character 's' (0x73)
Field 1	Bytes 1 to 4	Data validity time	Unsigned 32 bits integer. Epoch time in second (number of second since 1st January 1970)
Field 2	Bytes 5 to 6	Residual data validity time	Unsigned 16 bits integer (fraction of seconds from 0 to 65535). LSB = 1/65536 (about 15.26 µs).
Field 3	Bytes 7 to 10	Latitude (WGS84)	Signed 32 bits integer $+/-2^{31}$ = +/- 180° Positive toward North
Field 4	Bytes 11 to 14	Longitude (WGS84)	Signed 32 bits integer $+/-2^{31}$ = +/- 180° Positive toward East of Greenwich meridian
Field 5	Bytes 15 to 16	Heave	Signed 16 bits integer. LSB= 1 mm Positive upwards
Field 6	Bytes 17 to 18	North velocity	Signed 16 bits integer. LSB= 1 mm/s Positive toward North
Field 7	Bytes 19 to 20	East velocity	Signed 16 bits integer LSB= 1 mm/s Positive toward East
Field 8	Bytes 21 to 22	Heave speed	Signed 16 bits integer LSB= 1 mm/s Positive upwards
Field 9	Bytes 23 to 25	Roll	Signed 24 bits integer $+/-2^{23}$ = +/- 180° Positive when port side up
Field 10	Bytes 26 to 28	Pitch	Signed 24 bits integer $+/-2^{23}$ = +/- 180° Positive when bow up (warning: inverse of INS standard convention)
Field 11	Bytes 29 to 31	Heading	Unsigned 24 bits integer 2^{23} = 180° Increases clockwise.
Field 12	Bytes 32 to 34	Heading rotation rate	Signed 24 bits integer $+/-2^{23}$ = +/- 180°/s Positive clockwise (See Note 1)
Field 13	Bytes 35 to 37	Roll rotation rate	**Signed 24 bits integer +/-223 = +/- 180°/s Positive when port moves upwards (See Note 1)**
Field 14	Bytes 38 to 40	Pitch rotation rate	**Signed 24 bits integer +/-223 = +/- 180°/s**

Message <F0><F1><F2>…..<F24>			
			Positive when bow moves upwards (warning: inverse of INS standard convention) (See Note 1)
Field 15	Bytes 41 to 44	User Status	4 Bytes INS User status (see M/SP/POP/157)
Field 16	Bytes 45 to 46	Latitude std. deviation	Unsigned 16 bits integer. LSB= 1cm; [0, 655 m]
Field 17	Bytes 47 to 48	Longitude std. deviation	Unsigned 16 bits integer. LSB= 1cm; [0, 655 m]
Field 18	Bytes 49 to 50	North velocity std. deviation	Unsigned 16 bits integer. LSB= 1mm/s
Field 19	Bytes 51 to 52	East velocity std. deviation	Unsigned 16 bits integer. LSB= 1mm/s
Field 20	Bytes 53 to 54	Down velocity std. deviation	Unsigned 16 bits integer. LSB= 1mm/s
Field 21	Bytes 55 to 56	Roll std. deviation	Unsigned 16 bits integer 215 = 90°
Field 22	Bytes 57 to 58	Pitch std. deviation	Unsigned 16 bits integer 215 = 90°
Field 23	Bytes 59 to 60	Heading std. deviation	Unsigned 16 bits integer 215 = 90°
Field 24	Bytes 61 to 62	CRC	Computed on bytes 1 to 60 (see Note 2)

Note 1: Heading, Roll and Pitch rates are averaged over the last sampling period for output rates from 0.1 to 50Hz; for output rates faster than 50Hz, instantaneous values are used.

Note 2 : CRC computation is described below:

```
unsigned short blkcrc(unsigned char* bufptr, unsigned len)
{
unsigned char i;
unsigned short data;
unsigned short crc = 0xffff;
    if (len == 0)
        return ~crc;
    do
    {
        for (i = 0, data = (unsigned short)(0xff & *bufptr++); i < 8; i++, data >>= 1)
        {
        if ((crc & 0x0001) ^ (data & 0x0001))
        {
        crc = (crc >> 1) ^ 0x8408;
        }
        else
        {
        crc >>= 1;
        }
        }
        } while (–len);
        crc = ~crc;
        data = crc;
        crc = (crc << 8) | ((data >> 8) & 0xff);
        return crc;
    }
```

LRS10 78 IC - LRS10 78 IIC

Standard: Binary protocol. IC=ISO Convention, IIC= Inverse ISO Convention.

Data sent: Status, date, time ref GNSS, heading, roll, pitch, heading rate, roll rate, pitch rate, latitude, longitude, depth, latitude accuracy, longitude accuracy, position correlation, GNSS latitude, GNSS longitude, North velocity, East velocity, down velocity, Log speed, course made good, speed over ground, direction of the current, speed of the current.

Data frame: 78 bytes – For data coded on several bytes, the bytes are sent MSB byte first. All signed integer are coded as two's complement. This protocol telegram is assumed to be sent at the rate of 10 Hz.

Message <F0><F1><F2>…..<F33>			
Field 0	Byte 0 Byte 1	Header	0x5A 0xA5
Field 1	Byte 2	NUMDATA	0x48
Field 2	Byte 3	IDENT	0x02
Field 3	Byte 4	Status 1	**Table 1**
Field 4	Byte 5	Status 2	**Table 2**
Field 5	Byte 6	BITE Status	**Table 3**
Field 6	Bytes 7-8	Date **Note 1**	Unsigned 16 bits; LSB= 1 day, [1 to 366 days]
Field 7	Bytes 9-11	Time Ref GNSS	Unsigned 24 bits; LSB= 0.01 s, [0 to 86400 s]
Field 8	Bytes 12-13	Spare	Set to 0
Field 9	Bytes 14-15	Heading	Unsigned 16 bits; LSB= $180/2^{15}$ deg, [0 to 360 deg] Positive and increasing when bow turns to starboard.
Field 10	Bytes 16-17	Roll	Signed 16 bits; LSB= 90/215 deg, ±90 deg
		For LRS10_78_IIC	Positive for left-port side down. Warning: Opposite sign of INS usual convention Inverse ISO Convention **Note 9**
		For LRS10_78_IC	Positive for left-port side up. ISO Convention **Note 9**
Field 11	Bytes 18-19	Pitch	Signed 16 bits; LSB= $90/2^{15}$ deg, ±90 deg
		For LRS10_78_IIC	Positive for front side- bow down. Inverse ISO Convention **Note 9**
		For LRS10_78_IC	Positive for front side-bow up. Warning: Opposite sign of INS usual convention ISO Convention **Note 9**
Field 12	Bytes 20-21	Heading rate **Note 2**	Signed 16 bits; LSB= $1/2^{15}$ rd/s, ±1 rd/s Positive when heading angle increases
Field 13	Bytes 22-23	Roll rate **Note 2**	Signed 16 bits; LSB= 1/215 rd/s, ±1 rd/s Positive when roll angle increases
Field 14	Bytes 24-25	Pitch rate **Note 2**	Signed 16 bits; LSB= $1/2^{15}$ rd/s, ±1 rd/s Positive when pitch angle increases

Message <F0><F1><F2>…..<F33>			
Field 15	Bytes 26-29	Latitude	Signed 32 bits; LSB= $90/2^{31}$ deg, ±90 deg Positive in the northern hemisphere.
Field 16	Bytes 30-33	Longitude	Signed 32 bits; LSB= $180/2^{31}$ deg, ±180 deg Positive East of Greenwich meridian, up to 180 deg.
Field 17	Bytes 34-35	Depth **Note 3**	Signed 16 bits; LSB= $655.36/2^{15}$, ±655.36 m
Field 18	Bytes 36-39	Latitude accuracy **Note 4**	Unsigned 32 bits; LSB= $10800/2^{31}$ Nm, [0 to 10800 Nm]
Field 19	Bytes 40-43	Longitude accuracy Note 4	Unsigned 32 bits; LSB= $10800/2^{31}$ Nm, [0 to 10800 Nm]
Field 20	Bytes 44-45	Position correlation **Note 4**	Signed 16 bits; LSB= $1/2^{15}$, ±1 V^2
Field 21	Bytes 46-49	GNSS Latitude	Signed 32 bits; LSB= $90/2^{31}$ deg, ±90 deg Positive in the northern hemisphere.
Field 22	Bytes 50-53	GNSS Longitude	Signed 32 bits; LSB= $180/2^{31}$ deg, ±180 deg Positive East of Greenwich meridian, up to 180 deg.
Field 23	Bytes 54-55	North Velocity	Signed 16 bits; LSB= $65.536/2^{15}$, ±65.536 m/s
Field 24	Bytes 56-57	East Velocity	Signed 16 bits; LSB= $65.536/2^{15}$, ±65.536 m/s
Field 25	Bytes 58-59	Down Velocity	Signed 16 bits; LSB= $65.536/2^{15}$, ±65.536 m/s
Field 26	Bytes 60-61	Log Speed	Signed 16 bits; LSB= $65.536/2^{15}$, ±65.536 m/s
		Navigation Data	
Field 27	Bytes 62-63	Course made good **Note 5**	Unsigned 16 bits; LSB= $180/2^{15}$ deg, [0 to 360 deg]
Field 28	Bytes 64-65	Speed over ground **Note 5**	Unsigned 16 bits; LSB= $65.536/2^{15}$, [0 to 65.536 m/s]
Field 29	Bytes 66-67	Direction of the current **Note 11**	Unsigned 16 bits; LSB= $180/2^{15}$ deg, [0 to 360 deg]
Field 30	Bytes 68-69	Speed of the current **Note 10**	Unsigned 16 bits; LSB= $65.536/2^{15}$, [0 to 65.536 m/s]
Field 31	Bytes 70-75	Spare	Set to 0
Field 32	Byte 76	Checksum	**Note 6**
Field 33	Byte 77	Terminator	0xAA

Table 9 - Byte status 1

7|6|5|4|3|2|1|0	Parameter	Description	Attached INS status bits
# # # # # 0 0 0	Built-In-Test Equipment **Note 7**	Level 0 – No failure	N/A
# # # # # 0 0 1		Level 1 – Anomaly	When one of those INS User status bit is set to 0: DVL_RECEIVED_VALID GPS_RECEIVED_VALID GPS2_RECEIVED_VALID EMLOG_RECEIVED_VALID DEPTH_RECEIVED_VALID TIME_RECEIVED_VALID **Note 8**
# # # # # 0 1 0		Level 2 – Warning	When one of those INS User status bit is set to 1: ALTITUDE_SATURATION CPU_OVERLOAD TEMPERATURE_ERR INPUT_x_ERR OUTPUT_x_ERR
# # # # # 0 1 1		Level 3 – Failure (Attitude data not valid)	When one of those INS User status bit is set to 1: DEGRADED_MODE HRP_INVALID DYNAMIC_EXCEEDED
# # # # # 1 0 0		Level 4 – Failure (no data valid)	INS User status FAILURE_MODE set to 1
# # # 0 0 # # #	Mode	Navigation – Sea	INS User status NAVIGATION set to 1
# # # 0 1 # # #		Navigation – Quay	INS Algo 2 status ZUPT_MODE_VALID set to 1
# # # 1 0 # # #		Alignment	INS User status ALIGNMENT set to 1
# # # 1 1 # # #		Maintenance	INS System status 2 SIMULATION_MODE set to 1
# 0 0 # # # # #	Attitude and Heading Reference validity	Data valid	N/A
# 0 1 # # # # #		Data degraded	When one of those INS User status bit is set to 1: DEGRADED_MODE FINE_ALIGNMENT
# 1 0 # # # # #		Data not valid	INS User status HRP_INVALID set to 1
x # # # # # # #	/	Reserved	N/A

Table 10 - Byte status 2

7|6|5|4|3|2|1|0	Parameter	Description	Attached INS status bits
# # # # # # # 0	GNSS time validity	Data valid	INS User status TIME_RECEIVED_VALID set to 1
# # # # # # # 1		Data not valid	INS User status TIME_RECEIVED_VALID set to 0
# # # # # # 0 #	GNSS position Validity **Note 8**	Data valid	INS User status GPS_RECEIVED_VALID set to 1 or GPS2_RECEIVED_VALID set to 1
# # # # # # 1 #		Data not valid	When one of those INS User status bit is set to 0: GPS_RECEIVED_VALID GPS2_RECEIVED_VALID
# # # # # x # #	/	Reserved	N/A
# # # # 0 # # #	Log speed **Note 8**	Data valid	INS User status EMLOG_RECEIVED_VALID set to 1
# # # # 1 # # #		Data not valid	INS User status EMLOG_RECEIVED_VALID set to 0
# # # 0 # # # #	Navigation data	Data valid	**Note 5**
# # # 1 # # # #		Data not valid	
# # x # # # # #	/	Reserved	N/A
# 0 # # # # # #	Simulation	Operational	INS System status 2 SIMULATION_MODE set to 0
# 1 # # # # # #		Simulation mode	INS System status 2 SIMULATION_MODE set to 1
0 # # # # # # #	INS Identification	when last digit of IP address is even	No INS status is related to this bit.
1 # # # # # # #		when last digit of IP address is odd	

Table 11 - Bit status

Bit N°	User Status	Message description
7	N/A	Reserved as spare
6	9	FOG_ANOMALY
5	10	ACC _ANOMALY
4	11	TEMPERATURE_ERR
3	12	CPU_OVERLOAD
2	13	DYNAMIC_EXCEEDED
1	14	SPEED_SATURATION
0	15	ALTITUDE_SATURATION

Note 1:

The date corresponds to the number of the day in the year. If the date is not included in the messages received from the GNSS, bytes 7 and 8 of the field 6 are set to 1. The time corresponds to the GNSS time in seconds since the beginning of the GNSS day. When the INS is powered on, time is reset to 0 and until reception of a valid GNSS message including a valid time, the time emitted in messages will be the INS internal system time (number of seconds since the start).
When a GNSS time is received, the time is set at this value, and then it is updated by INS internal clock until the next time received in a GNSS message.
The date and UTC synchronization time can be received in ZDA, RMC or UTC messages, if sent by GNSS.

Note 2:
The attitude rate data corresponds to the time derivation of the attitude data, respect to the LRS telegram attitude data sign convention (ISO Convention or Inverse ISO Convention).

Note3:
The depth is positive under sea level.
When depth sensor altitude mode is selected, the depth corresponds to the INS altitude.
For a surface ship (for Stabilization, GNSS or Hydro altitude mode), the depth corresponds to the heave (the mean value is then 0).

Note 4:
The position accuracy corresponds to the maximum error estimation (3σ value = 99% probability). The position correlation corresponds to: σLat.Lon/(σLat. σLon) where σLat.Lon is the cross-covariance of latitude and Longitude.

Note 5:
The Navigation data bit (bit 4 of the status 2) is set to 1 when one of the following condition is met:

- The speed over ground (field 28) < 1 knot (then course made good (field 27) is set to 0).
- The Built-In-Test Equipment status (see status 1) is on level 3 or 4: i.e: When one of those INS User status bit is set to 1: DEGRADED_MODE, HRP_INVALID DYNAMIC_EXCEEDED, FAILURE_MODE.
- The INS is not in navigation mode at sea situation: i.e: INS User status NAVIGATION is set to 0.
- The current standard deviation > 95% of the current standard deviation set in the Web-Based Graphical User Interface.

Note 6:
One byte is used and it represents the checksum of the message. It is an 8-bit word. The checksum is a binary addition without carry modulo 256 of all data bytes, NUMDATA and IDENT included. Therefore, Header, Checksum and Terminator are excluded from this addition.

Note 7:
For the parameter Built-in-Test Equipment, the level 4 is checked first, then level 3, then level 2 and finally level 1.

Note 8:

Only input sensors that are configured through the Web-Based Graphical User Interface are taken into account. In Simulation mode, sensors flags are not managed and the operator should not take them into account.

Note 9:

Sign convention of roll and pitch:

	ISO Convention	Inverse ISO Convention	INS Convention
Roll > 0	for left-port side up	for left-port side down	for left-port side up
Pitch > 0	for front-side bow up	for front side-bow down	for front side-bow down

Note 10:

The speed of current is computed as follow :

$$\sqrt{(Vc_{north})^2 + (Vc_{east})^2}$$

Note 11:

The direction of current is issued from the angle

$$\alpha = arc\tan\left(\frac{Vc_{east}}{Vc_{north}}\right)$$

Direction of current	$Vc_{east} > 0$	$Vc_{east} < 0$
$Vc_{north} > 0$	abs(α)	2*PI – abs(α)
$Vc_{north} < 0$	PI – abs(α)	PI + abs(α)

If is Vc_{north} is 0, the direction of current is not calculated and set to 0.

LRS100 32 IC - LRS100 32 IIC

Standard: Binary protocol. IC=ISO Convention, IIC= Inverse ISO Convention.

Data sent: Status, date, time ref GNSS, heading, roll, pitch, heading rate, roll rate, pitch rate, latitude, longitude, depth, latitude accuracy, longitude accuracy, position correlation, GNSS latitude, GNSS longitude, North velocity, East velocity, down velocity, Log speed, course made good, speed over ground, direction of the current, speed of the current.

Data frame: 78 bytes – For data coded on several bytes, the bytes are sent MSB byte first. All signed integer are coded as two's complement. This protocol telegram is assumed to be sent at the rate of 10 Hz.

Message <F0><F1><F2>…..<F18>			
Field 0	Byte 0 Byte 1	Header	0x5A 0xA5
Field 1	Byte 2	NUMDATA	0x1A
Field 2	Byte 3	IDENT	0x01
Field 3	Byte 4	Status 1	**Table 1**
Field 4	Byte 5	Status 2	**Table 2**
Field 5	Bytes 6-7	Heading	Unsigned 16 bits; LSB= $180/2^{15}$ deg, [0 to 360 deg] Positive and increasing when bow turns to starboard.
Field 6	Bytes 8-9	Roll	Signed 16 bits; LSB= $90/2^{15}$ deg, ±90 deg
		For LRS100_32_IIC	Positive for left-port side down. Warning: Opposite sign of INS usual convention Inverse ISO Convention **Note 9**
		For LRS100_32_IC	Positive for left-port side up. ISO Convention **Note 9**
Field 7	Bytes 10-11	Pitch	Signed 16 bits; LSB= $90/2^{15}$ deg, ±90 deg
		For LRS100_32_IIC	Positive for front side-bow down. Inverse ISO Convention **Note 9**
		For LRS100_32_IC	Positive for front side-bow up. Warning: Opposite sign of INS usual convention ISO Convention **Note 9**
Field 8	Bytes 12-13	Heading rate **Note 2**	Signed 16 bits; LSB= $1/2^{15}$ rd/s, ±1 rd/s Positive when heading angle increases
Field 9	Bytes 14-15	Roll rate **Note 2**	Signed 16 bits; LSB= 1/215 rd/s, ±1 rd/s Positive when roll angle increases
Field 10	Bytes 16-17	Pitch rate **Note 2**	Signed 16 bits; LSB= $1/2^{15}$ rd/s, ±1 rd/s Positive when pitch angle increases
Field 11	Bytes 18-19	North Velocity	Signed 16 bits; LSB= $65.536/2^{15}$, ±65.536 m/s
Field 12	Bytes 20-21	East Velocity	Signed 16 bits; LSB= 65.536/215 , ±65.536 m/s
Field 13	Bytes 22-23	Down Velocity	Signed 16 bits; LSB= 65.536/215 , ±65.536 m/s
Field 14	Bytes 24-25	North acceleration	Signed 16 bits; LSB= 327.68/215 , ±327.68 m/s^2
Field 15	Bytes 26-27	East acceleration	Signed 16 bits; LSB= 327.68/215 , ±327.68 m/s^2

Message <F0><F1><F2>…..<F18>			
Field 16	Bytes 28-29	Down acceleration **Note 12**	Signed 16 bits; LSB= 327.68/215 , ±327.68 m/s^2
Field 17	Byte 30	Checksum	**Note 6**
Field 18	Byte 31	Terminator	0xAA

Table 12 - Byte status 1

7|6|5|4|3|2|1|0	Parameter	Description	Attached INS status bits
# # # # # 0 0 0	Built-In-Test	Level 0 – No failure	N/A
# # # # # 0 0 1	Equipment **Note 7**	Level 1 – Anomaly	When one of those INS User status bit is set to 0: DVL_RECEIVED_VALID GPS_RECEIVED_VALID GPS2_RECEIVED_VALID EMLOG_RECEIVED_VALID DEPTH_RECEIVED_VALID TIME_RECEIVED_VALID **Note 8**
# # # # # 0 1 0		Level 2 – Warning	When one of those INS User status bit is set to 1: ALTITUDE_SATURATION CPU_OVERLOAD TEMPERATURE_ERR INPUT_x_ERR OUTPUT_x_ERR
# # # # # 0 1 1		Level 3 – Failure (Attitude data not valid)	When one of those INS User status bit is set to 1: DEGRADED_MODE HRP_INVALID DYNAMIC_EXCEEDED
# # # # # 1 0 0		Level 4 – Failure (no data valid)	INS User status FAILURE_MODE set to 1
# # # 0 0 # # #	Mode	Navigation – Sea	INS User status NAVIGATION set to 1
# # # 0 1 # # #		Navigation – Quay	INS Algo 2 status ZUPT_MODE_VALID set to 1
# # # 1 0 # # #		Alignment	INS User status ALIGNMENT set to 1
# # # 1 1 # # #		Maintenance	INS System status 2 SIMULATION_MODE set to 1

7|6|5|4|3|2|1|0	Parameter	Description	Attached INS status bits
# 0 0 # # # # #	Attitude and Heading Reference validity	Data valid	N/A
# 0 1 # # # # #		Data degraded	When one of those INS User status bit is set to 1: DEGRADED_MODE FINE_ALIGNMENT
# 1 0 # # # # #		Data not valid	INS User status HRP_INVALID set to 1
x # # # # # # #	/	Reserved	N/A

Table 13 - Byte status 2

7|6|5|4|3|2|1|0	Parameter	Description	Attached INS status bits
# # # # # # # 0	GNSS time validity	Data valid	INS User status TIME_RECEIVED_VALID set to 1
# # # # # # # 1		Data not valid	INS User status TIME_RECEIVED_VALID set to 0
# # # # # # 0 #	GNSS position Validity **Note 8**	Data valid	INS User status GPS_RECEIVED_VALID set to 1 or GPS2_RECEIVED_VALID set to 1
# # # # # # 1 #		Data not valid	When one of those INS User status bit is set to 0: GPS_RECEIVED_VALID GPS2_RECEIVED_VALID
# # # # # x # #	/	Reserved	N/A
# # # # 0 # # #	Log speed **Note 8**	Data valid	INS User status EMLOG_RECEIVED_VALID set to 1
# # # # 1 # # #		Data not valid	INS User status EMLOG_RECEIVED_VALID set to 0
# # # 0 # # # #	Navigation data	Data valid	**Note 5**
# # # 1 # # # #		Data not valid	
# # x # # # # #	/	Reserved	N/A
# 0 # # # # # #	Simulation	Operational	INS System status 2 SIMULATION_MODE set to 0
# 1 # # # # # #		Simulation mode	INS System status 2 SIMULATION_MODE set to 1
0 # # # # # # #	INS Identification	when last digit of IP address is even	No INS status is related to this bit.
1 # # # # # # #		when last digit of IP address is odd	

Table 14 - Bit status

Bit N°	User Status	Message description
7	N/A	Reserved as spare
6	9	FOG_ANOMALY
5	10	ACC _ANOMALY
4	11	TEMPERATURE_ERR
3	12	CPU_OVERLOAD
2	13	DYNAMIC_EXCEEDED
1	14	SPEED_SATURATION
0	15	ALTITUDE_SATURATION

Note 1:

The date corresponds to the number of the day in the year. If the date is not included in the messages received from the GNSS, bytes 7 and 8 of the field 6 are set to 1. The time corresponds to the GNSS time in seconds since the beginning of the GNSS day. When the INS is powered on, time is reset to 0 and until reception of a valid GNSSmessa ge including a valid time, the time emitted in messages will be the INS internal system time (number of seconds since the start).

When a GNSS time is received, the time is set at this value, and then it is updated by INS internal clock until the next time received in a GNSS message.

The date and UTC synchronization time can be received in ZDA, RMC or UTC messages, if sent by GNSS.

Note 2:

The attitude rate data corresponds to the time derivation of the attitude data, respect to the LRS telegram attitude data sign convention (ISO Convention or Inverse ISO Convention).

Note3:

The depth is positive under sea level .

When depth sensor altitude mode is selected, the depth corresponds to the INS altitude.

For a surface ship (for Stabilization, GPS or Hydro altitude mode), the depth corresponds to the heave (the mean value is then 0).

Note 4:

The position accuracy corresponds to the maximum error estimation (3σ value = 99% probability). The position correlation corresponds to: σLat.Lon/(σLat. σLon) where σLat.Lon is the cross-covariance of latitude and Longitude.

Note 5:

The Navigation data bit (bit 4 of the status 2) is set to 1 when one of the following condition is met:

- The speed over ground (field 28) < 1 knot (then course made good (field 27) is set to 0).
- The Built-in-Test Equipement status (see status 1) is on level 3 or 4: i.e: When one of those INS User status bit is set to 1: DEGRADED_MODE, HRP_INVALID DYNAMIC_ EXCEEDED, FAILURE_MODE.

- The INS is not in navigation mode at sea situation: i.e: INS User status NAVIGATION is set to 0.
- The current standard deviation > 95% of the current standard deviation set in the Web-Based Graphical User Interface.

Note 6:
One byte is used and it represents the checksum of the message. It is an 8-bit word. The checksum is a binary addition without carry modulo 256 of all data bytes, NUMDATA and IDENTincluded. Therefore, Header, Checksum and Terminator are excluded from this addition.

Note 7:
For the parameter Built-in-Test Equipment, the level 4 is checked first, then level 3, then level 2 and finally level 1.

Note 8:
Only input sensors that are configured through the Web-Based Graphical User Interface are taken into account. In Simulation mode, sensors flags are not managed and the operator should not take them into account.

Note 9:
Sign convention of roll and pitch:

	ISO Convention	**Inverse ISO Convention**	**INS Convention**
Roll > 0	for left-port side up	for left-port side down	for left-port side up
Pitch > 0	for front-side bow up	for front side-bow down	for front side-bow down

Note 10:
The speed of current is computed as follow :

$$\sqrt{(Vc_{north})^2 + (Vc_{east})^2}$$

Note 11:
The direction of current is issued from the angle

$$\alpha = arc\tan\left(\frac{Vc_{east}}{Vc_{north}}\right)$$

Direction of current	$Vc_{east} > 0$	$Vc_{east} < 0$
$Vc_{north} > 0$	abs(α)	2*PI – abs(α)
$Vc_{north} < 0$	PI – abs(α)	PI + abs(α)

If is Vc_{north} is 0, the direction of current is not calculated and set to 0

Note 12:
The down acceleration is compensated from the g measurement.

LRS100 35 IC - LRS100 35 IIC

Standard: Binary protocol. IC=ISO Convention, IIC= Inverse ISO Convention.

Data sent: Status, date, time ref GNSS, heading, roll, pitch, heading rate, roll rate, pitch rate, latitude, longitude, depth, latitude accuracy, longitude accuracy, position correlation, GNSS latitude, GNSS longitude, North velocity, East velocity, down velocity, Log speed, course made good, speed over ground, direction of the current, speed of the current.

Data frame: 78 bytes – For data coded on several bytes, the bytes are sent MSB byte first. All signed integer are coded as two's complement. This protocol telegram is assumed to be sent at the rate of 10 Hz.

Message <F0><F1><F2>…..<F18>			
Field 0	Byte 0 Byte 1	Header	0x5A 0xA5
Field 1	Byte 2	NUMDATA	0x1D
Field 2	Byte 3	IDENT	0x03
Field 3	Byte 4	Status 1	**Table 1**
Field 4	Byte 5	Status 2	**Table 2**
Field 5	Bytes 6-8	Heading	Unsigned 24 bits; LSB= $180/2^{23}$ deg, [0 to 360 deg] Positive and increasing when bow turns to starboard.
Field 6	Bytes 9-11	Roll	Signed 24 bits; LSB= $90/2^{23}$ deg, ±90 deg
		For LRS100_35_IIC	Positive for left-port side down. Warning: Opposite sign of INS usual convention Inverse ISO Convention **Note 9**
		For LRS100_35_IC	Positive for left-port side up. ISO Convention **Note 9**
Field 7	Bytes 12-14	Pitch	Signed 24 bits; LSB= $90/2^{23}$ deg, ±90 deg
		For LRS100_35_IIC	Positive for front side-bow down. Inverse ISO Convention **Note 9**
		For LRS100_35_IC	Positive for front side-bow up. Warning: Opposite sign of INS usual convention ISO Convention **Note 9**
Field 8	Bytes 15-16	Heading rate **Note 1**	Signed 16 bits; LSB= $1/2^{15}$ rd/s, ±1 rd/s Positive when heading angle increases
Field 9	Bytes 17-18	Roll rate **Note 1**	Signed 16 bits; LSB= $1/2^{15}$ rd/s, ±1 rd/s Positive when roll angle increases
Field 10	Bytes 19-20	Pitch rate **Note 1**	Signed 16 bits; LSB= $1/2^{15}$ rd/s, ±1 rd/s Positive when pitch angle increases
Field 11	Bytes 21-22	North Velocity	Signed 16 bits; LSB= $65.536/2^{15}$, ±65.536 m/s
Field 12	Bytes 23-24	East Velocity	Signed 16 bits; LSB= $65.536/2^{15}$, ±65.536 m/s
Field 13	Bytes 25-26	Down Velocity	Signed 16 bits; LSB= $65.536/2^{15}$, ±65.536 m/s
Field 14	Bytes 27-28	North acceleration	Signed 16 bits; LSB= $327.68/2^{15}$, ±327.68 m/s²
Field 15	Bytes 29-30	East acceleration	Signed 16 bits; LSB= $327.68/2^{15}$, ±327.68 m/s²

Message <F0><F1><F2>…..<F18>			
Field 16	Bytes 31-32	Down acceleration **Note 12**	Signed 16 bits; LSB= $327.68/2^{15}$, ±327.68 m/s²
Field 17	Byte 33	Checksum	**Note 6**
Field 18	Byte 34	Terminator	0xAA

Table 15 - Byte status 1

7|6|5|4|3|2|1|0	Parameter	Description	Attached INS status bits
# # # # # 0 0 0	Built-In-Test	Level 0 – No failure	N/A
# # # # # 0 0 1	Equipment **Note 7**	Level 1 – Anomaly	When one of those INS User status bit is set to 0: DVL_RECEIVED_VALID GPS_RECEIVED_VALID GPS2_RECEIVED_VALID EMLOG_RECEIVED_VALID DEPTH_RECEIVED_VALID TIME_RECEIVED_VALID **Note 8**
# # # # # 0 1 0		Level 2 – Warning	When one of those INS User status bit is set to 1: ALTITUDE_SATURATION CPU_OVERLOAD TEMPERATURE_ERR INPUT_x_ERR OUTPUT_x_ERR
# # # # # 0 1 1		Level 3 – Failure (Attitude data not valid)	When one of those INS User status bit is set to 1: DEGRADED_MODE HRP_INVALID DYNAMIC_EXCEEDED
# # # # # 1 0 0		Level 4 – Failure (no data valid)	INS User status FAILURE_MODE set to 1
# # # 0 0 # # #	Mode	Navigation – Sea	INS User status NAVIGATION set to 1
# # # 0 1 # # #		Navigation – Quay	INS Algo 2 status ZUPT_MODE_VALID set to 1
# # # 1 0 # # #		Alignment	INS User status ALIGNMENT set to 1
# # # 1 1 # # #		Maintenance	INS System status 2 SIMULATION_MODE set to 1

7|6|5|4|3|2|1|0	Parameter	Description	Attached INS status bits
# 0 0 # # # # #	Attitude and Heading Reference validity	Data valid	N/A
# 0 1 # # # # #		Data degraded	When one of those INS User status bit is set to 1: DEGRADED_MODE FINE_ALIGNMENT
# 1 0 # # # # #		Data not valid	INS User status HRP_INVALID set to 1
x # # # # # # #	/	Reserved	N/A

Table 16 - Byte status 2

7|6|5|4|3|2|1|0	Parameter	Description	Attached INS status bits
# # # # # # # 0	GNSS time validity	Data valid	INS User status TIME_RECEIVED_VALID set to 1
# # # # # # # 1		Data not valid	INS User status TIME_RECEIVED_VALID set to 0
# # # # # # 0 #	GNSS position Validity **Note 8**	Data valid	INS User status GPS_RECEIVED_VALID set to 1 or GPS2_RECEIVED_VALID set to 1
# # # # # # 1 #		Data not valid	When one of those INS User status bit is set to 0: GPS_RECEIVED_VALID GPS2_RECEIVED_VALID
# # # # # x # #	/	Reserved	N/A
# # # # 0 # # #	Log speed **Note 8**	Data valid	INS User status EMLOG_RECEIVED_VALID set to 1
# # # # 1 # # #		Data not valid	INS User status EMLOG_RECEIVED_VALID set to 0
# # # 0 # # # #	Navigation data	Data valid	**Note 5**
# # # 1 # # # #		Data not valid	
# # x # # # # #	/	Reserved	N/A
# 0 # # # # # #	Simulation	Operational	INS System status 2 SIMULATION_MODE set to 0
# 1 # # # # # #		Simulation mode	INS System status 2 SIMULATION_MODE set to 1
0 # # # # # # #	INS Identification	when last digit of IP address is even	No INS status is related to this bit.
1 # # # # # # #		when last digit of IP address is odd	

Table 17 - Bit status

Bit N°	User Status	Message description
7	N/A	Reserved as spare
6	9	FOG_ANOMALY
5	10	ACC _ANOMALY
4	11	TEMPERATURE_ERR
3	12	CPU_OVERLOAD
2	13	DYNAMIC_EXCEEDED
1	14	SPEED_SATURATION
0	15	ALTITUDE_SATURATION

Note 1:

The date corresponds to the number of the day in the year. If the date is not included in the messages received from the GNSS, bytes 7 and 8 of the field 6 are set to 1. The time corresponds to the GNSS time in seconds since the beginning of the GNSS day. When the INS is powered on, time is reset to 0 and until reception of a valid GNSS message including a valid time, the time emitted in messages will be the INS internal system time (number of seconds since the start).

When a GNSS time is received, the time is set at this value, and then it is updated by INS internal clock until the next time received in a GNSS message.

The date and UTC synchronization time can be received in ZDA, RMC or UTC messages, if sent by GNSS.

Note 2:

The attitude rate data corresponds to the time derivation of the attitude data, respect to the LRS telegram attitude data sign convention (ISO Convention or Inverse ISO Convention).

Note3:

The depth is positive under sea level .

When depth sensor altitude mode is selected, the depth corresponds to the INS altitude.

For a surface ship (for Stabilization, GPS or Hydro altitude mode), the depth corresponds to the heave (the mean value is then 0).

Note 4:

The position accuracy corresponds to the maximum error estimation (3σ value = 99% probability). The position correlation corresponds to: σLat.Lon/(σLat. σLon) where σLat.Lon is the cross-covariance of latitude and Longitude.

Note 5:

The Navigation data bit (bit 4 of the status 2) is set to 1 when one of the following condition is met:

- The speed over ground (field 28) < 1 knot (then course made good (field 27) is set to 0).
- The Built-In-Test Equipement status (see status 1) is on level 3 or 4: i.e: When one of those INS User status bit is set to 1: DEGRADED_MODE, HRP_INVALID DYNAMIC_ EXCEEDED, FAILURE_MODE.

- The INS is not in navigation mode at sea situation: i.e: INS User status NAVIGATION is set to 0.
- The current standard deviation > 95% of the current standard deviation set in the Web-Based Graphical User Interface.

Note 6:

One byte is used and it represents the checksum of the message. It is an 8-bit word. The checksum is a binary addition without carry modulo 256 of all data bytes, NUMDATA and IDENTincluded. Therefore, Header, Checksum and Terminator are excluded from this addition.

Note 7:

For the parameter Built-In-Test Equipment, the level 4 is checked first, then level 3, then level 2 and finally level 1.

Note 8:

Only input sensors that are configured through the Web-Based Graphical User Interface are taken into account. In Simulation mode, sensors flags are not managed and the operator should not take them into account.

Note 9:

Sign convention of roll and pitch:

	ISO Convention	Inverse ISO Convention	INS Convention
Roll > 0	for left-port side up	for left-port side down	for left-port side up
Pitch > 0	for front-side bow up	for front side-bow down	for front side-bow down

Note 10:

The speed of current is computed as follow :

$$\sqrt{(Vc_{north})^2 + (Vc_{east})^2}$$

Note 11:

The direction of current is issued from the angle

$$\alpha = arc\tan\left(\frac{Vc_{east}}{Vc_{north}}\right)$$

Direction of current	$Vc_{east} > 0$	$Vc_{east} < 0$
$Vc_{north} > 0$	abs(α)	2*PI – abs(α)
$Vc_{north} < 0$	PI – abs(α)	PI + abs(α)

If is Vc_{north} is 0, the direction of current is not calculated and set to 0

Note 12:

The down acceleration is compensated from the g measurement.

MDL2

Standard: Output MDL ASCII standard.
Data sent: Heading, Roll, Pitch, Gyro status.
Data frame: HhhhhP±xxxxR±yyyyQ<CR><LF>
where:
hhhh is the heading in degrees multiplied by 10 range [0.0°, 359.9°]
xxxx is the pitch in degrees multiplied by 100, range [+89.99°, -89.99°]
yyyy is the roll in degrees multiplied by 100, range [+89.99°, -89.99°]
Q is the gyro status as follows:
M Gyro alignment mode (1)
E Gyro reports good data (2)
N reports invalid data (3)

(1) Q=M when flag USR_INS_ALIGNMENTis raised
(2) Q=E when M and N condition are false
(3) Q=N whenever any of the following flags are raised :
USR_INS_CPU_OVERLOAD | USR_INS_DEGRADED_MODE | USR_INS_FAILURE_MODE | USR_INS_HRP_UNVALID

MDL TRIM CUBE

Standard: Output MDL ASCII standard.
Data sent: Heading, pitch and roll values.
Data frame:

HxxxxP±yyyyR±zzzz<CR><LF>		
H	'H' character	
xxxx	heading value in degrees multiplied by 10 (no sign character)	Four digits range [0000,3599]
P	'P' character	
±yyyy	pitch value in degrees multiplied by 100 with its sign character '+' or ' –'	Four digits after the sign character range [-8999,+8999]
R	'R' character	
±zzzz	roll value in degrees multiplied by 100 with its sign character '+' or ' –'	Four digits after the sign character range [-8999,+8999]

MINIFOG_OTG_MSG8

Standard: ASCII.
Data sent: 'HaaaaP+bbbbR+ccccX+ddddY+eeeeZ+ffffQ<CR><LF>'
Data frame:

MINIFOG_OTG_MSG8 (OceanTools Gyro & Rotation Rates)		
Heading delimiter	'H'	Fixed character
Heading	'aaaa'	Value : In degrees, LSB = 0.1° Inside in [0°:360°[
Pitch delimiter	'P	Fixed character
Pitch Sign	'+'	Value :'+' or '-' positive bow up Warning: Opposite sign of usual convention.
Pitch	'bbbb'	Value : in degrees, LSB = 0.1° inside [-90°:90°]
Roll delimiter	'R'	Fixed character
Roll Sign	'+'	Value :'+' or '-' positive port up
Roll	'cccc'	Value : in degrees, LSB = 0.1° inside [-90°:90°]
Heading rate delimiter	'X'	Fixed character
Heading rate sign	'+'	Value : '+' or '-' Positive clockwise
Heading rate	'dddd'	Value : In degrees/second, LSB = 0.01 °/s range]-100°/s:100°/s[
Pitch rate delimiter	'Y'	Fixed Character
Pitch rate sign	'+'	Value : '+' or '-' positive bow up Warning: Opposite sign of INS usual convention.
Pitch rate	'eeee'	Value : in degrees/second, LSB = 0.01 °/s range]-100°/s:100°/s[
Roll rate delimiter	'Z'	Fixed Character
Roll rate sign	'+'	Value : '+' or '-' positive port up
Roll rate	'ffff'	Value : in degrees/second, LSB = 0.01 °/s range]-100°/s:100°/s[
Gyro Status	'Q'	Value : 'E' or 'D' Status of Gyro is defined with the help of user status, see Note1
End point	Carriage return	Value : <CR>
	Line feed	Value : <LF>

Note 1: Gyro status shall be:

- 'E' when word "HRP invalid" is "false" (available on IXBLUE "User status" message)

- Otherwise it shall be ‘D’.

Example:
A packet reading “H3214P+0132R+0012X+0012Y+0000Z+0085E<cr><lf>”
Wild be: Heading 321.4°, Pitch 13.2° Up, Roll 1.2° to Port.
Heading rotation 0.12°/sec increasing heading, Pitch rotation 0°/sec pitch, Roll rotation 0.85°/sec to Port.

NAV AND CTD

Standard: Binary protocol. BLUEFIN proprietary protocol.
Data sent: Transverse velocity, Longitudinal velocity, DVL Altitude, Vertical velocity, GNSS Latitude, GNSS Longitude, GNSS Altitude, USBL Latitude, USBL Longitude, USBL Altitude, LBL Latitude, LBL Longitude, LBL Altitude, LBL Beacon ID, LBL Range, Time.
Data frame: The frame contains 19 fields – 74 bytes.

Message <F0><F1><F2>…..<F19>			
Field 0	Byte 0	0x24	Synchronization byte
Field 1	Byte 1	0xAA	Synchronization byte
Field 2	Bytes 2 to 5	System Status	32 LSB bits of the INS System status 32 bits integer ; MSB are sent first
Field 3	Bytes 6 to 9	Algorithm Status	32 LSB bits of the INS Algorithm status 32 bits integer ; MSB are sent first
Field 4	Bytes 10 to 13	Algorithm Status	32 MSB bits of the INS Algorithm status 32 bits integer ; MSB are sent first
Field 5	Bytes 14 to 17	Heading	IEEE floating point format, radians (0 to 2Pi)
Field 6	Bytes 18 to 21	Roll	IEEE floating point format, radians (+/-Pi)
Field 7	Bytes 22 to 25	Pitch	IEEE floating point format, radians (+/-Pi/2)
Field 8	Bytes 26 to 29	North speed	IEEE floating point format, m/s
Field 9	Bytes 30 to 33	East speed	IEEE floating point format, m/s
Field 10	Bytes 34 to 37	Vertical speed	IEEE floating point format, m/s
Field 11	Bytes 38 to 41	Latitude	Signed 32 bits +/-231 = +/- 180° MSB are sent first
Field 12	Bytes 42 to 45	Longitude	Signed 32 bits +/-231 = +/- 180° MSB are sent first
Field 13	Bytes 46 to 49	Altitude '+' Up	IEEE floating point format, meters
Field 14	Bytes 50 to 53	Depth sensor	IEEE floating point format, meters
Field 15	Bytes 54 to 57	Conductivity*	IEEE floating point format, mS.cm
Field 16	Bytes 58 to 61	Temperature*	IEEE floating point format, °C
Field 17	Bytes 62 to 65	Pressure*	IEEE floating point format, decibar
Field 18	Bytes 66 to 69	Sound Velocity*	IEEE floating point format, m/s
Field 19	Bytes 70 to 73	Time	Bit 0 to bit 4 : 00000 Bit 5 to bit 14 : Milliseconds

Message <F0><F1><F2>…..<F19>			
			Bit 15 to bit 20 : Seconds Bit 21 to bit 26 : Minutes Bit 27 to bit 31 : Hours

* Conductivity, Temperature, Pressure and Sound Velocity are the values received with the $BFCTD data frame from the USBL LBL CTD input protocol.
Otherwise, the values are 0.

NAV BHO

Standard: ASCII NMEA 0183.

Data sent: This protocol outputs INS computed position values in an ASCII frame, some characters of this output frame are set to fixed values.

Data frame: is composed of:

- $PHZDA, refer to section 4.1.5
- $PHGGA, refer to section 4.2.2
- $PHVTG, refer to section 4.2.11
- $HEHDT, refer to section 4.1.7

$PASHR,hhmmss.sss,H.HH,T,aR.RR,bP.PP,cD.DD,r.rrr,p.ppp,h.hhh,x,y*hh<CR><LF>

hhmmss.sss	is the UTC of the data	
H.HH	is the heading in degrees	2 digits after decimal point
T	is a fixed character = ‘T’	
aR.RR	is the roll in degrees and a, its sign character ‘+’ when port up, ‘–‘ when port down	2 digits after decimal point
bP.PP	is the pitch in degrees and b, its sign character ‘–‘ when bow down, ‘+’ when bow up Warning: Opposite sign of INS usual convention	2 digits after decimal point
cD.DD	is the heave in meters and c, its sign character ‘–’ when INS goes up, ‘+’ when INS goes down Warning: Opposite sign of INS usual convention	2 digits after decimal point
r.rrr	is the roll standard deviation	3 digits after decimal point
p.ppp	is the pitch standard deviation	3 digits after decimal point
h.hhh	is the heading standard deviation	3 digits after decimal point
x	is the GNSS aiding status flag 1 when GNSS received, otherwise 0	
y	is the sensor error status flag 1 when FOG or ACC error, otherwise 0	
hh	is the checksum	

(*) Last GNSS values received. When no GNSS has been received since power-up, these fields are null.

(**) INS calculated data

(***) The quality indicator is managed as follows:

INS does not copy the quality indicator received on GGA input to GGA output.

The quality factor is set with respect to a correspondence table between INS calculated SD and Quality indicator in GGA telegram (refer to section 1.4).

(****) The ZDA sentence is always sent at 1 Hz whatever the chosen refresh rate.

(*****) 5 digits after the decimal point in Military mode.

WARNING

Some empty fields are allowed in --GGA and –VTG data frames. See samples hereafter $--GGA,161229.487,3723.2475,N,12158.3416,W,1,07,1.0,9.0,M,,,,0000*18

$--GGA,064036.289,4836.5375,N,00740.9373,E,1,04,3.2,200.2,M,,,,0000*0E

$--VTG,309.62,T,,M,0.13,N,0,2,K*6

NAV BHO LONG

Standard: ASCII NMEA 0183.

Data sent: This protocol outputs INS computed position values in an ASCII frame, some characters of this output frame are set to fixed values.

Data frame: is composed of:

$PHZDA, refer to section 4.1.5

$PHGGA, refer to section 4.2.2

$PHGST,hhmmss.ss,x.xxx,y.yyy,z.zzz,a.aaa,x.xxx,y.yyy,z.zzz*hh<CR><LF>

hhmmss.ss	is the UTC time of the GGA fix associated with this sentence	(**) 3 digits after decimal point
x.xxx	is the RMS value of the standard deviation of the range inputs to the navigation process	
y.yyy	is the standard deviation of semi-major axis of error ellipse (meters)	(**) 3 digits after decimal point
z.zzz	is the standard deviation of semi-minor axis of error ellipse (meters)	(**) 3 digits after decimal point
a.aaa	is the orientation of semi-major axis of error ellipse (degrees from true North)	(**) 3 digits after decimal point
x.xxx	is the standard deviation of latitude error, in meters	(**) 3 digits after decimal poin
y.yyy	is the standard deviation of longitude error, in meters	(**) 3 digits after decimal point
z.zzz	is the standard deviation of altitude error, in meters	(**) 3 digits after decimal point
hh	is the checksum	

$PHVTG, refer to section 4.2.11

$HEHDT, refer to section 4.1.7

$PASHR,hhmmss.sss,H.HH,T,aR.RR,bP.PP,cD.DD,r.rrr,p.ppp,h.hhh,x,y*hh<CR><LF>

hhmmss.sss	is the UTC of the data	
H.HH	is the heading in degrees	2 digits after decimal point
T	is a fixed character = 'T'	
aR.RR	is the roll in degrees and a, its sign character '+' when port up, '–' when port down	2 digits after decimal point
bP.PP	is the pitch in degrees and b, its sign character '–' when bow down, '+' when bow up Warning: Opposite sign of INS usual convention	2 digits after decimal point
cD.DD	is the heave in meters and c, its sign character '–' when INS goes up, '+' when INS goes down Warning: Opposite sign of INS usual convention	2 digits after decimal point
r.rrr	is the roll standard deviation	3 digits after decimal point
p.ppp	is the pitch standard deviation	3 digits after decimal point
h.hhh	is the heading standard deviation	3 digits after decimal point
x	is the GNSS aiding status flag 1 when GNSS received, otherwise 0	
y	is the sensor error status flag 1 when FOG or ACC error, otherwise 0	
hh	is the checksum	

(*) Last GNSS values received. When no GNSS has been received since power-up, these fields are null.

(**) INS calculated data

(***) The quality indicator is managed as follows:
INS does not copy the quality indicator received on GGA input to GGA output.
The quality factor is set with respect to a correspondence table between INS calculated SD and Quality indicator in GGA telegram (refer to section 1.4).

(****) The ZDA sentence is always sent at 1 Hz whatever the chosen refresh rate.

(*****) 5 digits after the decimal point in Military mode.

WARNING

Some empty fields are allowed in --GGA and –VTG data frames. See samples hereafter $--GGA,161229.487,3723.2475,N,12158.3416,W,1,07,1.0,9.0,M,,,,0000*18
$--GGA,064036.289,4836.5375,N,00740.9373,E,1,04,3.2,200.2,M,,,,0000*0E
$--VTG,309.62,T,,M,0.13,N,0,2,K*6

NAV BHO 2M

Standard: ASCII .
Data sent: This protocol outputs INS computed position values in an ASCII frame.
Data frame: is composed of:
$PHZDA, refer to section 4.1.5
$PHGGA, refer to section 4.2.2
$PHVTG, refer to section 4.2.11
$HEHDT, refer to section 4.1.7

$PASHR,hhmmss.sss,H.HH,T,aR.RR,bP.PP,cD.DD,r.rrr,p.ppp,h.hhh,x,y*hh<CR><LF>

hhmmss.sss	is the UTC of the data	
H.HH	is the heading in degrees	2 digits after decimal point
T	is a fixed character = 'T'	
aR.RR	is the roll in degrees and a, its sign character '+' when port up, '–' when port down	2 digits after decimal point
bP.PP	is the pitch in degrees and b, its sign character '–' when bow down, '+' when bow up Warning: Opposite sign of INS usual convention	2 digits after decimal point
cD.DD	is the heave in meters and c, its sign character '–' when INS goes up, '+' when INS goes down Warning: Opposite sign of INS usual convention	2 digits after decimal point
r.rrr	is the roll standard deviation	3 digits after decimal point
p.ppp	is the pitch standard deviation	3 digits after decimal point
h.hhh	is the heading standard deviation	3 digits after decimal point
x	is the GNSS aiding status flag 1 when GNSS received, otherwise 0	
y	is the sensor error status flag 1 when FOG or ACC error, otherwise 0	
hh	is the checksum	

(*) Last GNSS values received. When no GNSS has been received since power-up, these fields are null.

(**) INS calculated data

(***) The quality indicator is managed as follows:
INS does not copy the quality indicator received on GGA input to GGA output.
The quality factor is set with respect to a correspondence table between INS calculated SD and Quality indicator in GGA telegram (refer to section 1.4).

(****) The ZDA sentence is always sent at 1 Hz whatever the chosen refresh rate.

(*****) 5 digits after the decimal point in Military mode.

WARNING

Some empty fields are allowed in --GGA and –VTG data frames. See samples hereafter $--GGA,161229.487,3723.2475,N,12158.3416,W,1,07,1.0,9.0,M,,,,0000*18
$--GGA,064036.289,4836.5375,N,00740.9373,E,1,04,3.2,200.2,M,,,,0000*0E
$--VTG,309.62,T,,M,0.13,N,0,2,K*6

NAV BINARY

Standard: Binary protocol.
Data sent: Status, Heading, Roll, Pitch, Heading Rate, Roll Rate, Pitch Rate, Latitude, Longitude, Altitude, Heave, North speed, East speed, Vertical speed, Status, Time.
Data frame: 16 bytes.

Message <71><F1><F2>…..<F16><CRC>			
Field 0	Byte 0	0x71	Synchronization byte
Field 1	Byte 1 to 4	UTC data timestamp	Unsigned 32 bits integer in s
Field 2	Bytes 5	UTC data timestamp	Unsigned 8 bits integer in hundredths of seconds
Field 3	Bytes 6 to 9	Latitude	Signed 32 bits in radians (+/-2^{31} = +/- 180°) Sign "+" North of the equator
Field 4	Bytes 10 to 13	Longitude	Signed 32 bits in radians (+/-2^{31} = +/- 180°) Sign "+" East of
Field 5	Bytes 14 to 17	Altitude	Signed 32 bits in centimeters
Field 6	Bytes 18 to 19	Heave	Signed 16 bits in centimeters Sign "+" in down direction Warning: Opposite sign of product usual convention
Field 7	Bytes 20 to 21	North speed	Signed 16 bits in centimeters/sec
Field 8	Bytes 22 to 23	East speed	Signed 16 bits in centimeters/sec
Field 9	Bytes 24 to 25	Down speed	Signed 16 bits in centimeters/sec Warning: Opposite sign of product usual convention
Field 10	Bytes 26 to 27	Roll*	Signed 16 bits in radians (+/- 2^{15}= +/- 180°) Sign "+" when port side up
Field 11	Bytes 28 to 29	Pitch*	Signed 16 bits in radians (+/- 2^{15}= +/- 180°) Sign "+" when bow up. Warning: Opposite sign of product usual convention
Field 12	Bytes 30 to 31	Heading*	Unsigned 16 bits in radians (2^{16}= 360°)
Field 13	Bytes 32 to 33	XV1 rotation rate **	Signed 16 bits in Radians /sec (+/- 2^{15}= +/- 180°/sec)
Field 14	Bytes 34 to 35	XV2 rotation rate **	Signed 16 bits in Radians /sec (+/- 2^{15}= +/- 180°/sec) Warning: Opposite sign of product usual convention
Field 15	Bytes 36 to 37	XV3 rotation rate **	Signed 16 bits in Radians /sec (+/- 2^{15}= +/- 180°/sec) Warning: Opposite sign of product usual convention

* In non- military mode , the precision of heading, roll and pitch data are limited to 0.001° to comply with export regulation.
** In non-military mode, the precision of rotation rate data is limited to 3.6 deg/h to comply with export regulation.

*** CRC computation is given hereafter:

```
unsigned short blkcrc(unsigned char* bufptr, unsigned len)
{
unsigned char i;
unsigned short data;
unsigned short crc = 0xffff;
    if (len == 0)
        return ~crc;
    do
    {
        for (i = 0, data = (unsigned short)(0xff & *bufptr++); i < 8; i++, data >>= 1)
        {
        if ((crc & 0x0001) ^ (data & 0x0001))
        {
        crc = (crc >> 1) ^ 0x8408;
        }
        else
        {
        crc >>= 1;
        }
        }
        } while (--len);
        crc = ~crc;
        data = crc;
        crc = (crc << 8) | ((data >> 8) & 0xff);
        return crc;
    }
```

NAV BINARY 1

Standard:Binary protocol.

Data sent: Status, Heading, Roll, Pitch, Heading Rate, Roll Rate, Pitch Rate, Latitude, Longitude, Altitude, Heave, North speed, East speed, Vertical speed, Status, Time.

Data frame: 17 fields - 50 bytes. Big Endian (MSB sent first).

Message <F0><F1><F2>…..<F16><F17>			
Field 0	Byte 0	0x71	Synchronization byte
Field 1	Bytes 1 to 4	UTC data timestamp	32 bits integer in s
Field 2	Byte 5	UTC data timestamp	8 bits integer in hundredths of seconds
Field 3	Bytes 6 to 13	Latitude	Signed 64 bits in degrees (+/-2^{63} = +/- 180°) Sign "+" North of equator
Field 4	Bytes 14 to 21	Longitude	Signed 64 bits in degrees (+/-2^{63} = +/- 180°) Sign "+" East of Greenwich
Field 5	Bytes 22 to 25	Altitude	Signed 32 bits in centimeters Sign "+" Up direction
Field 6	Bytes 26 to 27	Heave	Signed 16 bits in centimeters Sign "+" in down direction
Field 7	Bytes 28 to 29	North speed	Signed 16 bits in centimeters/sec
Field 8	Bytes 30 to 31	East speed	Signed 16 bits in centimeters/sec
Field 9	Bytes 32 to 33	Down speed	Signed 16 bits in centimeters/sec
Field 10	Bytes 34 to 35	Roll*	Signed 16 bits in degrees (+/- 2^{15}= +/- 180°)) Sign "+" when port side up
Field 11	Bytes 36 to 37	Pitch*	Signed 16 bits in degrees (+/- 2^{15}= +/- 180°)) Sign "+" when bow up. Warning: Opposite sign of INS usual convention
Field 12	Bytes 38 to 39	Heading*	Unsigned 16 bits in degrees (2^{16}= 360°)
Field 13	Bytes 40 to 41	X_{V1} rotation rate **	Signed 16 bits in degrees /sec (+/- 2^{15}= +/- 180°/sec)
Field 14	Bytes 42 to 43	X_{V2} rotation rate **	Signed 16 bits in degrees /sec (+/- 2^{15}= +/- 180°/sec) Warning: Opposite sign of INS usual convention
Field 15	Bytes 44 to 45	X_{V3} rotation rate **	Signed 16 bits in degrees /sec (+/- 2^{15}= + /- 180°/sec) Warning: Opposite sign of INS usual convention
Field 16	Bytes 46 to 47	Status	Unsigned 16 bits : 0x0000 when data valid 0x00AA when data invalid Data are not valid if one of the following User Status bit is set to 1: USR_INS_FOG_ERROR USR_INS_ACC_ERROR USR_INS_TEMP_ERROR

Message <F0><F1><F2>.....<F16><F17>			
			USR_INS_DYNAMIC_EXCEEDED USR_INS_SPEED_SATURATION USR_INS_ALTI_SATURATION USR_INS_ALIGNMENT USR_INS_DEGRADED_MODE USR_INS_FAILURE_MODE
Field 17	Bytes 48 to 49	Checksum (CRC)***	Unsigned 16 bits computed on bytes 1 to 47

* In non-military mode , the precision of heading, roll and pitch data are limited to 0.001° to comply with export regulation.
** In non-military mode, the precision of rotation rate data is limited to 3.6 deg/h to comply with export regulation.
This protocol is derived from NAV_BINARY. Latitude and Longitude resolution have been extended from 32 bits to 64 bits.

*** CRC computation is given hereafter:

```
unsigned short blkcrc(unsigned char* bufptr, unsigned len)
{
unsigned char i;
unsigned short data;
unsigned short crc = 0xffff;
    if (len == 0)
        return ~crc;
    do
    {
        for (i = 0, data = (unsigned short)(0xff & *bufptr++); i < 8; i++, data >>= 1)
        {
        if ((crc & 0x0001) ^ (data & 0x0001))
        {
        crc = (crc >> 1) ^ 0x8408;
        }
        else
        {
        crc >>= 1;
        }
        }
        } while (–len);
        crc = ~crc;
        data = crc;
        crc = (crc << 8) | ((data >> 8) & 0xff);
        return crc;
    }
}
}
} while (–len);
crc = ~crc;
data = crc;
```

```
crc = (crc << 8) | ((data >> 8) & 0xff);
return crc;
}
```

NAV BINARY HR

Standard: Binary protocol.
Data sent:Status, Heading, Roll, Pitch, XV1, XV2 and XV3 rotation rates, Latitude, Longitude, Altitude, Heave, North speed, East speed, Vertical speed, Status, Time.
Data frame: This protocol is derived from NAV BINARY protocol with higher resolution on roll and pitch.

Message <71><F1><F2>.....<F16><CRC>			
Field 0	Byte 0	0x71	Synchronization byte
Field 1	Byte 1 to 4	UTC data timestamp	32 bits integer in s
Field 2	Bytes 5	UTC data timestamp	8 bits integer in hundredths of seconds
Field 3	Bytes 6 to 9	Latitude	Signed 32 bits in radians (+/-2^{31} = +/-180°) Sign "+" North of equator
Field 4	Bytes 10 to 13	Longitude	Signed 32 bits in radians (+/-2^{31} = +/-180°)) Sign "+" East of Greenwich
Field 5	Bytes 14 to 17	Altitude	Signed 32 bits in centimeters
Field 6	Bytes 18 to 19	Heave	Signed 16 bits in centimeters Sign "+" in down direction
Field 7	Bytes 20 to 21	North speed	Signed 16 bits in centimeters/sec
Field 8	Bytes 22 to 23	East speed	Signed 16 bits in centimeters/sec
Field 9	Bytes 24 to 25	Down speed	Signed 16 bits in centimeters/sec
Field 10	Bytes 26 to 27	Roll*	Signed 16 bits in radians (+/- 2^{15}= +/- 45°) Sign "+" when port side up
Field 11	Bytes 28 to 29	Pitch*	Signed 16 bits in radians (+/- 2^{15}= +/- 45°) Sign "+" when bow up. Warning: Opposite sign of INS usual convention
Field 12	Bytes 30 to 31	Heading*	Unsigned 16 bits in radians (2^{16}= 360°)
Field 13	Bytes 32 to 33	X_{V1} rotation rate **	Signed 16 bits in Radians /sec (+/- 2^{15}= +/- 180°/sec)
Field 14	Bytes 34 to 35	X_{V2} rotation rate **	Signed 16 bits in Radians /sec (+/- 2^{15}= +/-180°/sec) Warning: Opposite sign of INS usual convention
Field 15	Bytes 36 to 37	X_{V3} rotation rate **	Signed 16 bits in Radians /sec (+/- 2^{15}= +/-180°/sec) Warning: Opposite sign of INS usual convention
Field 16	Bytes 38 to 39	Status	Unsigned 16 bits 0x0000 when data valid 0x00AA when data invalid Data are not valid if one of the following User Status bit is set to 1 : USR_INS_FOG_ERROR USR_INS_ACC_ERROR USR_INS_TEMP_ERROR USR_INS_DYNAMIC_EXCEEDED USR_INS_SPEED_SATURATION USR_INS_ALTI_SATURATION

Message <71><F1><F2>…..<F16><CRC>			
			USR_INS_ALIGNMENT USR_INS_DEGRADED_MODE USR_INS_FAILURE_MODE
Field 17	Bytes 40 to 41	Checksum (CRC)***	Unsigned 16 bits CRC16 computed on bytes 1 to 39

* In non-military mode , the precision of heading, roll and pitch data are limited to 0.001° to comply with export regulation.
** In non-military mode, the precision of rotation rate data is limited to 3.6 deg/h to comply with export regulation.

*** CRC computation is given hereafter:

```
unsigned short blkcrc(unsigned char* bufptr, unsigned len)
{
unsigned char i;
unsigned short data;
unsigned short crc = 0xffff;
    if (len == 0)
        return ~crc;
    do
    {
        for (i = 0, data = (unsigned short)(0xff & *bufptr++); i < 8; i++, data >>= 1)
        {
        if ((crc & 0x0001) ^ (data & 0x0001))
        {
        crc = (crc >> 1) ^ 0x8408;
        }
        else
        {
        crc >>= 1;
        }
        }
        } while (–len);
        crc = ~crc;
        data = crc;
        crc = (crc << 8) | ((data >> 8) & 0xff);
        return crc;
    }
```

NAVIGATION

Standard: Output NMEA 0183 compatible.
Data sent: Heading, Attitude, Position, Status.
Data frame:

$HEHDT,x.xxxxx,T*hh<CR><LF>

x.xxx	is the true heading in degrees	Float, 3 digits after the decimal point. (*)
T	is a fixed character = ‘T’	
hh	is the checksum	

$PIXSE,ATITUD,x.xxx,y.yyy*hh<CR><LF>

x.xxx	is the roll in degrees	Float, 3 digits after the decimal point. (*)
y.yyy	is the pitch in degrees	Float, 3 digits after the decimal point. (*)
hh	is the checksum	

$PIXSE,POSITI,x.xxxxxxxx,y.yyyyyyyy,z.zzz*hh<CR><LF>

x.xxxxxxxx	is the latitude in degrees	Float, 8 digits after decimal point
y.yyyyyyyy	is the longitude in degrees*	Float, 8 digits after decimal point
z.zzz	is the altitude in meters	Float, 3 digits after decimal point
hh	is the checksum	

*Longitude negative is from Greenwich meridian to east and positive to west.

$PIXSE,STATUS,hhhhhhhh,llllllll,jjjjjjjj*hh<CR><LF>

hhhhhhhh	is the hexadecimal value of the 32 bits of the INS System status 1
llllllll	is the hexadecimal value of the 32 bits of the INS System status 2
jjjjjjjj	is the hexadecimal value of the 32 bits of the INS System status 3
hh	is the checksum

(*) 5 digits after the decimal point in Military mode.

NAVIGATION HDLC

Standard: Binary protocol. THALES proprietary binary protocol.

Data sent: Time, Latitude, Longitude, Altitude, Heave, North Velocity , East Velocity, Down Velocity, Heading, Roll, Pitch, Heading rate, Roll rate, Pitch rate, Data validity, Checksum.

Data frame: 22 fields - 47 bytes.

Message <F1><F1><F2>…..<F22>			
Field 1	Byte 0	Start marker	Fixed 0x02 value
Field 2	Byte 1 to 3	Data time tag	Unsigned integer. LSB= 2^{-10} ms= 0.9765625 µs
Field 3	Byte 4	reserved	Set to 0 by default
Field 4	Byte 5 to 8	Latitude	Signed integer. '+': North of equator. LSB=90°*2^{-30}.
Field 5	Byte 9 to 12	Longitude	Signed integer. '+': East of Greenwich. LSB=90°*2^{-30}.
Field 6	Byte 13 to 16	Altitude	Signed integer. '+' above sea level. LSB= 1 cm.
Field 7	Byte 17 to 18	Heave	Signed integer. '+' when down. LSB= 1 cm
Field 8	Byte 19 to 20	North Velocity	Signed integer. LSB= 1 cm/s
Field 9	Byte 21 to 22	East Velocity	Signed integer. LSB= 1 cm/s
Field 10	Byte 23 to 24	Down Velocity	Signed integer. LSB= 1 cm/s
Field 11	Byte 25 to 27	Heading	Unsigned integer. MSB= 180°
Field 12	Byte 28	Reserved	Set to 0 by default
Field 13	Byte 29 to 31	Roll	Signed integer. '+' Starboard down. LSB=90°*2^{-22}
Field 14	Byte 32	Reserved	Set to 0 by default
Field 15	Byte 33 to 35	Pitch	Signed integer.'+' Bow down. LSB=90°*2^{-22}
Field 16	Byte 36	Reserved	Set to 0 by default
Field 17	Byte 37 to 38	Heading rate	Signed integer.LSB=45°/s*2^{-14}.
Field 18	Byte 39 to 40	Roll rate	Signed integer.LSB=45°/s*2^{-14}.
Field 19	Byte 41 to 42	Pitch rate	Signed integer.LSB=45°/s*2^{-14}.
Field 20	Byte 43	INS source	Bit 0 to 1: = 0 (by default) Bit 2: = 0 : Data fields invalid; 1: Data fields valid
Field 21	Byte 44	Reserved	Set to 0 by default
Field 22	Byte 45 to 46	Checksum	CRC16-Modbus. Computed on bytes 1 to 44.

Note 1: MSB is sent first then LSB (big-endian convention). Least significant bit (LSB) sent first.

Note 2: Two's complement notation is used for signed integers.

Note 3: The resolution of rotation rate data is limited to 3.6 deg/h to comply with export regulation.

Note 4: Time is UTC time if INS is time synchronized with GNSS time. Otherwise it is INS internal clock time. Time is code over 0-24 h.

Note 5: Altitude value depends on the Altitude Computation Mode (refer to user manual).

Note 6: Validity Bit 2 of Field 20 description

Bit 2= OR (Bit 9 to 15, Bit 28, Bit 30, Bit 31)of INS User status.

NAVIGATION LONG

Standard: Binary protocol. Output navigation.

Data sent: Status, Heading, Roll, Pitch, North speed, East speed, Vertical speed in the terrestrial reference mark, Latitude, Longitude, Altitude, Standard deviation of speeds , HRP and position, Status, Time.

Data frame: 24 Fields – 90 Bytes; All multi-byte integers and floating point fields are sent MSB first. Signed integers are two's complement coded.

Message <F0><F1>…<F23>			
Field 0	Byte 0	Header byte 1	Fixed value = 0x24
Field 1	Byte 1	Header byte 2	Fixed value = 0xAA
Field 2	Bytes 2 to 5	INS User Status	32 bits of the INS User status
Field 3	Bytes 6 to 9	INS Algorithm Status 1	32 LSB bits of the INS Algorithm status
Field 4	Bytes 10 to 13	INS Algorithm Status 2	32 MSB bits of the INS Algorithm status
Field 5	Bytes 14 to 17	Heading	IEEE Float 32 bits; radians ; range = [0 ; 2π[Increasing when bow turns to east
Field 6	Bytes 18 to 21	Roll	IEEE Float 32 bits; radians range = [-π ; π[Positive when port side rises
Field 7	Bytes 22 to 25	Pitch	IEEE Float 32 bits; radians range = [-π/2 ; π/2[Positive when bow down
Field 8	Bytes 26 to 29	North speed	IEEE Float 32 bits; meters/second
Field 9	Bytes 30 to 33	East speed	IEEE Float 32 bits; meters/second
Field 10	Bytes 34 to 37	Vertical speed	IEEE Float 32 bits; meters/second Positive up
Field 11	Bytes 38 to 41	Latitude	Signed 32 bit integer; range = [-180°:+180°[LSB = 180° / 231 = 83.82.10^{-9} °
Field 12	Bytes 42 to 45	Longitude	Signed 32 bit integer; range = [-180°:+180°[LSB = 180° / 231 = 83.82.10^{-9} °
Field 13	Bytes 46 to 49	Altitude	IEEE Float 32 bits; meters Positive up
Field 14	Bytes 50 to 53	Validity time of the data	Bits 27 to 31: hours Bits 21 to 26: minutes Bits 15 to 20: seconds Bits 0 to 14 : microseconds (LSB = 50µs)
Field 15	Bytes 54 to 57	Heading error standard deviation	IEEE Float 32 bits; radians
Field 16	Bytes 58 to 61	Roll error standard deviation	IEEE Float 32 bits; radians
Field 17	Bytes 62 to 65	Pitch error standard deviation	IEEE Float 32 bits; radians
Field 18	Bytes 66 to 69	North speed error standard deviation	IEEE Float 32 bit ; meters/second
Field 19	Bytes 70 to 73	East speed error standard deviation	IEEE Float 32 bits; meters/second

Message <F0><F1>…<F23>			
Field 20	Bytes 74 to 77	Vertical speed error standard deviation	IEEE Float 32 bits; meters/second
Field 21	Bytes 78 to 81	Latitude speed error standard deviation	IEEE Float 32 bits; meters
Field 22	Bytes 82 to 85	Longitude error standard deviation	IEEE Float 32 bits; meters
Field 23	Bytes 86 to 89	Altitude error standard deviation	IEEE Float 32 bits; meters

NAVIGATION SHORT

Standard: Binary protocol. Output Navigation.

Data sent: Status, Heading , Roll, Pitch, North speed, East speed, Vertical speed in the terrestrial reference mark, Latitude, Longitude, Altitude, Time. **Data frame**: The frame contains 22 fields – 54 bytes.

In mode not military, the precision of heading, roll and pitch data are limited to 0.001° to comply with export regulation.

Message <24><AA><F2>…..<F13><CRC>			
Field 0	Byte 0	Header 1	Integer 8 bits Value: 0x24
Field 1	Byte 1	Header 2	Value: 0xAA
Field 2	Bytes 2 to 5	User Status	32 bits of the INS User status 32 bits integer ; MSB are sent first
Field 3	Bytes 6 to 13	Algorithm Status	64 bits of the INS Algorithm status 32 bits integer ; MSB are sent first
Field 4	Bytes 14 to 17	Heading	Floating IEEE in Radians
Field 5	Bytes 18 to 21	Roll	Floating IEEE in Radians (+ if port side rises)
Field 6	Bytes 22 to 25	Pitch	Floating IEEE in Radians (+ if the prow is inserted)
Field 7	Bytes 26 to 29	North speed in the terrestrial reference frame	Floating IEEE in Meters/second
Field 8	Bytes 30 to 33	East speed in the terrestrial reference frame	Floating IEEE in Meters/second
Field 9	Bytes 34 to 37	Vertical speed in the terrestrial reference frame	Floating IEEE in Meters/second
Field 10	Bytes 38 to 41	Latitude	Integer (+/- 2^{31}= +/- 180°)
Field 11	Bytes 42 to 45	Longitude	Integer (+/- 2^{31}= +/- 180°)
Field 12	Bytes 46 to 49	Altitude	Floating IEEE in Meters
Field 13	Bytes 50 to 53	Time	5 Bits integer : Hour 6 Bits integer : Min 6 Bits integer: Sec 10 Bits integer: msec 5 Bits: 00000

OCTANS STANDARD

Standard: Output NMEA 0183 compatible.

Data sent: Heading, Roll, Pitch, Position, Linear speed, Compensation values, Status, Date and time if ZDA option selected.

Data frame: Heading, Roll, Pitch, Position, Linear Speeds, User Status.

NMEA $HEHDT Frame refer to section 4.1.7
NMEA $PHTRO Frame refer to section 4.2.9
NMEA $PHLIN Frame refer to section 4.2.5
NMEA $PHSPD Frame refer to section 4.2.8
NMEA $PHCMP Frame refer to section 4.2.1
NMEA $PHINF Frame refer to section 4.2.4.

PHINS STANDARD

Standard: Output NMEA 0183 compatible.

Data sent: Heading, Attitude, Position, Speed, Standard deviations, Sensors input, Status. The maximal refresh rate is 20 Hz.

Data frame:

NMEA $HEHDT Frame refer to section 4.1.7.

NMEA $HETHS Frame refer to section 4.1.8.

$PIXSE,ATITUD,x.xxx,y.yyy*hh<CR><LF>

x.xxx	is the roll in degrees	3 digits after the decimal point. (*)
y.yyy	is the pitch in degrees	3 digits after the decimal point. (*)
hh	is the checksum	

(*) 5 digits after the decimal point in military mode

$PIXSE,POSITI,x.xxxxxxxx,y.yyyyyyyy,z.zzz*hh<CR><LF>

x.xxxxxxxx	is the latitude in degrees	8 digits after decimal point
y.yyyyyyyy	is the longitude in degrees*	8 digits after decimal point
z.zzz	is the altitude in meters	3 digits after decimal point
hh	is the checksum	

*Longitude negative is from Greenwich meridian to east and positive to west.

$PIXSE,SPEED_,x.xxx,y.yyy,z.zzz*hh<CR><LF>

x.xxx	Speed X_{East} in m/s (East speed)	3 digits after decimal point
y.yyy	Speed X_{North} in m/s (North speed)	3 digits after decimal point
z.zzz	Speed X_{UP} in m/s (Sign “+” towards up side)	3 digits after decimal point
hh	is the checksum	

$PIXSE,UTMWGS,c,nn,x.xxx,y.yyy,z.zzz*hh<CR><LF>

C	is the latitude UTM zone (character)	
nn	is the longitude UTM zone (integer)	
x.xxx	is the east UTM position in meter	3 digits after decimal point
y.yyy	is the north UTM position in meter	3 digits after decimal point
z.zzz	is the altitude in meters	3 digits after decimal point
hh	is the checksum	

$PIXSE,HEAVE_,x.xxx,y.yyy,z.zzz*hh<CR><LF>

x.xxx	is the surge in meters (signed)	3 digits after the decimal point
y.yyy	is the sway in meters (signed)	3 digits after the decimal point
z.zzz	is the heave in meters (signed)	3 digits after the decimal point
hh	is the checksum	

$PIXSE,STDHRP,x.xxx,y.yyy,z.zzz*hh<CR><LF>

x.xxx	is the heading std dev (degrees)	3 digits after the decimal point
y.yyy	is the roll std dev (degrees)	3 digits after the decimal point
z.zzz	is the pitch std dev (degrees)	3 digits after the decimal point
hh	is the checksum	

$PIXSE,STDPOS,x.xx,y.yy,z.zz*hh<CR><LF>

x.xx	is the latitude std dev (meters)	2 digits after the decimal point
y.yy	is the longitude std dev (meters)	2 digits after the decimal point
z.zz	is the altitude std dev (meters)	2 digits after the decimal point
hh	is the checksum	

$PIXSE,STDSPD,x.xxx,y.yyy,z.zzz*hh<CR><LF>

x.xxx	north speed std dev (m/s)	3 digits after the decimal point
y.yyy	east speed std dev (m/s)	3 digits after the decimal point
z.zzz	vertical speed std dev (m/s)	3 digits after the decimal point
hh	is the checksum	

$PIXSE,TIME__, hhmmss.ssssss*hh<CR><LF>

hhmmss.ssssss	is the validity time of the computed data transmitted in the UTC time reference frame if available otherwise in the system time reference frame. 6 digits after the decimal point.
hh	is the checksum

$PIXSE,LOGIN_,x.xxx,y.yyy,z.zzz,m.mmm,hhmmss.ssssss*hh<CR><LF>

Last data received from the log bottom track sensor		
x.xxx	is the XS1 longitudinal DVL speed in m/s	3 digits after the decimal point
y.yyy	is the XS2 transverse DVL speed in m/s	3 digits after the decimal point
z.zzz	is the XS3 vertical DVL speed in m/s	3 digits after the decimal point
m.mmm	is the log heading misalignment Kalman estimation in degrees	3 digits after the decimal point
hhmmss.ssssss	is the validity time of the log data received in the UTC time reference frame if available otherwise in thesystem time reference frame	
hh	is the checksum	

$PIXSE,LOGDVL,x.xx,y.yy,z.zz*hh<CR><LF>

Last raw data received from the log sensor		
x.xx	is the DVL set sound velocity in water in m/s	2 digits after the decimal point
y.yy	is the measured compensation sound velocity in m/s	2 digits after the decimal point
z.zz	is the DVL distance to bottom in meters	2 digits after the decimal point
hh	is the checksum	

$PIXSE,LOGWAT,x.xxx,y.yyy,z.zzz,n.nnn,e.eee,N.NNN,E.EEE,hhmmss.ssssss*hh<CR><LF>

Last data received from the log water track sensor		
x.xxx	is the XS1 longitudinal DVL speed in m/s	3 digits after the decimal point
y.yyy	is the XS2 transverse DVL speed in m/s	3 digits after the decimal point
z.zzz	is the XS3 vertical DVL speed in m/s	3 digits after the decimal point
n.nnn	is the north current speed in in m/s	3 digits after the decimal point
e.eee	is the east current speed in m/s	3 digits after the decimal point
N.NNN	is the north current speed std dev in m/s	3 digits after the decimal point
E.EEE	is the east current speed std dev in m/s	3 digits after the decimal point
hhmmss.ssssss	is the validity time of the log WT data received in the UTC time reference frame if available otherwise in the system time reference frame	6 digits after decimal point
hh	is the checksum	

$PIXSE,GPSIN_,x.xxxxxxx,y.yyyyyyy,z.zzz,hhmmss.ssssss,q*hh<CR><LF>

Last data received from the GNSS1 sensor		
x.xxxxxxx	is the latitude in degrees	8 digits after the decimal point
y.yyyyyyy	is the longitude in degrees	8 digits after the decimal point
z.zzz	is the altitude in meters	3 digits after the decimal point
hhmmss.ssssss	is the validity time of the GNSS data received in the UTC time reference frame if available otherwise in the system time reference frame	6 digits after the decimal point
q	is the GNSS quality indicator 0 and ^{3} 5 Fix not valid 1 GPS SPS Mode Fix not valid 2 Differential Mode, SPS Mode, Fix not valid 3 GPS PPS Mode, Fix not valid 4 GPS RTK Mode	
hh	is the checksum	

$PIXSE,GP2IN_,x.xxxxxxx,y.yyyyyyy,z.zzz,hhmmss.ssssss,q*hh<CR><LF>

Last data received from the GNSS2 sensor		
x.xxxxxxx	Is the latitude in degrees	8 digits after the decimal point
y.yyyyyyy	is the longitude in degrees	8 digits after the decimal point
z.zzz	is the altitude in meters	3 digits after the decimal point
hhmmss.ssssss	is the validity time of the GNSS2 data received in the UTC time reference frame if available otherwise in the system time reference frame	6 digits after the decimal point
q	is the GNSS quality indicator 0 and ^{3} 5 Fix not valid 1 GPS SPS Mode fix valid 2 Differential Mode, SPS Mode, fix valid 3 GPS PPS Mode, fix valid 4 GPS RTK Mode	
hh	is the checksum	

$PIXSE,GPMIN_,x.xxxxxxx,y.yyyyyyy,z.zzz,hhmmss.ssssss,q*hh<CR><LF>

Last data received from the manual GNSS sensor		
x.xxxxxxx	is the latitude in degrees	8 digits after the decimal point
y.yyyyyyy	is the longitude in degrees	8 digits after the decimal point
z.zzz	is the altitude in meters	3 digits after the decimal point
hhmmss.ssssss	is the validity time of the Manual GNSS data received in the UTC time reference frame if available otherwise in the system time reference frame	6 digits after the decimal point
q	is the GNSS quality indicator 0 and ^{3} 5 Fix not valid 1 GPS SPS Mode fix valid 2 Differential Mode, SPS Mode, fix valid 3 GPS PPS Mode, fix valid 4 GPS RTK Mode	
hh	is the checksum	

$PIXSE,DEPIN_,x.xxx,hhmmss.ssssss*hh<CR><LF>

Last data received from the depth sensor		
x.xxx	is the depth in meters	3 digits after the decimal point
hhmmss.ssssss	is the validity time of the depth sensor data received in the UTC time reference frame if available otherwise in the system time reference frame	6 digits after the decimal point
hh	is the checksum	

$PIXSE,USBIN_,x.xxxxxxxx,y.yyyyyyyy,z.zzz,d.dd,hhmmss.ssssss,n,ccccccc*hh<CR><LF>

Last data received from the USBL sensor		
x.xxxxxxxx	is the latitude in degrees	8 digits after the decimal point
y.yyyyyyyy	is the longitude in degrees	8 digits after the decimal point
z.zzz	is the altitude in meters	3 digits after the decimal point
d.dd	is the delay (age of data) in seconds	2 digits after the decimal point
hhmmss.ssssss	is the validity time of the USBL data received in the UTC time reference frame if available otherwise in the system time reference frame	
n	number of configured beacon for display	
ccccccc	TP code of the beacon	7 ASCII characters
hh	is the checksum	

$PIXSE,LBLIN_,x.xxxxxxx,y.yyyyyyyy,z.zzz,n,r.rrr,hhmmss.ssssss*hh<CR><LF>

Last data received from the LBL		
x.xxxxxxxx	is the latitude of the beacon in degrees	8 digits after the decimal point
y.yyyyyyyy	is the longitude of the beacon in degrees	8 digits after the decimal point
z.zzz	is the altitude of the beacon in meters	3 digits after the decimal point
n	is the beacon internal index (i.e : 0, 1, 2 or 3)	
r.rrr	is the range in meters	3 digits after the decimal point
hhmmss.ssssss	is the validity time of the LBL data received in the UTC time reference frame if available otherwise in the system time reference frame	6 digits after the decimal point
hh	is the checksum	

$PIXSE,UTCIN_,hhmmss.ssssss*hh<CR><LF>

Last UTC received		
hhmmss.ssssss	is the UTC time received	6 digits after the decimal point
hh	is the checksum	

$PIXSE,LMNIN_,x.xxx,n.nnn,e.eee,N.NNN,E.EEE,hhmmss.ssssss*hh<CR><LF>

Last data received from the EM Log sensor		
x.xxx	is the XS1 speed in m/s	3 digits after the decimal point
n.nnn	is the speed of the north current in m/s	3 digits after the decimal point
e.eee	is the speed of the east current in m/s	3 digits after the decimal point
N.NNN	is the north current speed std dev in m/s	3 digits after the decimal point
E.EEE	is the east current speed std dev in m/s	3 digits after the decimal point
hhmmss.ssssss	is the validity time of the EM log data received in the UTC time reference frame if available otherwise in the system time reference frame	6 digits after the decimal point
hh	is the NMEA checksum	

$PIXSE,DDRECK,x.xxxxxxxx,y.yyyyyyyy,z.zzz,m.mmm,f.fffffff,p.ppp,q.qqq,r.rrrrrr,s.sss*hh<CR><LF>

x.xxxxxxxx	is the dead reckoning latitude in degrees	8 digits after the decimal point
y.yyyyyyyy	is the dead reckoning longitude in degrees	8 digits after the decimal point
z.zzz	is the dead reckoning altitude in meters	3 digits after the decimal point
m.mmm	is the heading misalignment dead reckoning estimation in degrees	3 digits after the decimal point
f.ffffff	is the scale factor dead reckoning estimation (*)	6 digits after the decimal point
p.ppp	is the pitch dead reckoning estimation in degrees	3 digits after the decimal point
q.qqq	is the heading misalignment dead reckoning standard deviation estimation in degrees	3 digits after the decimal point
r.rrrrrrr	is the scale factor dead reckoning standard deviation estimation (*)	6 digits after the decimal point
s.sss	is the pitch dead reckoning standard deviation estimation in degrees	3 digits after the decimal point
hh	is the checksum	

(*) 0.00123 means 0.123% scale factor correction.

$PIXSE,CALCHK,x.xx,y.yy,z.zz*hh<CR><LF>

Calibration check frame. This frame is only transmitted when calibration verification is active	
x.xx	is the travelled distance from start of verification line in meters
y.yy	is the current vertical error in meters
z.zz	is the current horizontal error in meters
hh	is the checksum

$PIXSE,ALGSTS,hhhhhhhh,llllllll *hh<CR><LF>

INS Algo status (refer to section 5.3)	
hhhhhhhh	is the hexadecimal value of INS Algo status1 (LSB)
llllllll	is the hexadecimal value of INS Algo status 2 (MSB)
hh	is the checksum

$PIXSE,STATUS,hhhhhhhh,llllllll *hh<CR><LF>

INS System status (refer to section 5.4)	
hhhhhhhh	is the hexadecimal value of INS System status 1 (LSB)
llllllll	is the hexadecimal value of INS System status 2 (MSB)
hh	is the checksum

$PIXSE,SORSTS,hhhhhhhh,llllllll *hh<CR><LF> (*)

INS Sensor status (refer to section 5.2)	
hhhhhhhh	is the hexadecimal value of INS Sensor status 1 (LSB)
llllllll	is the hexadecimal value of INS Sensor status 2 (MSB)
hh	is the checksum

(*) This telegram is only sent on the repeater link data flow.

$PIXSE,ALGSTX,hhhhhhhh,llllllll,jjjjjjjj,kkkkkkkk*hh<CR><LF>

Extended INS Algorithm status (refer to section 5.3)	
hhhhhhhh	is the hexadecimal value of INS Algo status3
llllllll	is the hexadecimal value of INS Algo status4
jjjjjjjj	is the hexadecimal value of INS Algo status5
kkkkkkkk	is the hexadecimal value of INS Algo status6
hh	is the checksum

(*) For Subsea product, this field is filled with “0” value.

$PIXSE,SYSSTX,hhhhhhhh*hh<CR><LF>

Extended System status (refer to section 5.4)	
hhhhhhhh	is the hexadecimal value of System status 3
hh	is the checksum

$PIXSE,USRSTS,hhhhhhhh*hh<CR><LF>

Extended User status (refer to section 5)	
hhhhhhhh	is the hexadecimal value of INS User status
hh	is the checksum

$PIXSE,HT_STS,hhhhhhhh *hh<CR><LF>

INS high-level repeater status (refer to section 5)	
hhhhhhhh	is the hexadecimal value of INS high-level repeater status(*)
hh	is the checksum

(*) This frame is output in the PHINS STANDARD protocol but is mainly intended to be used by iXblue legacy serial MMI iXrepeater.

$PIXSE,FOGEST,x.xxxx,y,yyyy,z,zzzz*hh <CR><LF>

x.xxxx	is the error FOG X estimation in 10-3 °/h
y.yyyy	is the error FOG Y estimation in 10-3 °/h
z .zzzz	is the error FOG Z estimation in 10-3 °/h
hh	is the checksum

$PIXSE,ACCEST,x.xxxx,y,yyyy,z,zzzz*hh <CR><LF>

x.xxxx	is the error ACC X estimation in µg
y.yyyy	is the error ACC Y estimation in µg
z .zzzz	is the error ACC Z estimation in µg
hh	is the checksum

$PIXSE,LOGEST,x.xxxx*hh <CR><LF>

x.xxxx	is the error log estimation in °
hh	is the checksum

(**) Data frames LOGIN_, LOGVDL, GPSIN_, DEPIN_, USBIN_, LBLIN_ … are sent only when updated data is received from the external sensor (DDRECK if dead reckoning mode is turned on).

POSIDONIA 6000

Standard: ASCII NMEA 0183
Data sent: Latitude, Hemisphere, Longitude, Direction, GNSS mode, Norm speed, Roll, Pitch, Heading
Data frame:

INS does not copy the quality indicator received on GGA input to GGA output.
During initial alignment (at power-up or after a system restart) the quality factor is fixed to 6.

$CAPACK,LLll.lll,a,LLll.lll,b,m,SS.SS,sR.RR,sP.PP,sH.HH*hh<CR><LF>

LLll.lll	is the latitude in degrees (LL) and in minutes (ll.lll)	3 digits after decimal point
a	is 'N' for Northern hemisphere, 'S' for Southern hemisphere	
LLll.lll	is the longitude in deg (LL) and in minutes (ll.lll)	3 digits after decimal point
b	is 'E' for East, 'W' for West	
m	is the GNSS quality indicator*	
SS.SS	is the horizontal speed norm in knots	2 digits after decimal point
sR.RR	is the roll in degrees and s is the sign character ' + ' for port up and ' - ' for port down. -180° to +180°.	2 digits after decimal point
sP.PP	is the pitch in degrees and s is the sign character ' + ' for bow down and ' - ' for bow up. -90° to +90°.	2 digits after decimal point
sH.HH	is the heading in degrees and s is the sign character even if always ' + ' for INS. +0 to +360°.	2 digits after decimal point
hh	is the checksum	

POS MV GRP111

Standard: Binary protocol. POS MV Group 111 Aplanix proprietary protocol.
Data sent: Smart Heave, Real time heave.
Data frame: The frame contains a header, 15 fields with 84 bytes in binary format. Reference document: POSMV 320 V3 Ethernet SCSI ICD 30 January 2003.
LSB are sent first and MSB in last position (Little Endian)

Message <F0><F1><F2>…..<F15>				
Field 0	Byte 0 to 3	$GRP	char	Group start
Field 1	Bytes 4 to 5	111	ushort	Group ID
Field 2	Bytes 6 to 7	76	ushort	Bytes count
Field 3	Bytes 8 to 15	Time 1	double	Seconds (NOT USED)
Field 4	Bytes 16 to 23	Time 2	double	Seconds (NOT USED)
Field 5	Bytes 24 to 31	Distance tag	double	meters (NOT USED)
Field 6	Bytes 32	Time types	byte	Value: bits 0-3 (*) 0: Time 1: INS time 1: Time 1: GNSS time (NOT USED) 2: Time 1: UTC time Value: bits 4-7 (*) 0: Time 2: INS time (NOT USED) 1: Time 2: GNSS time (NOT USED) 2: Time 2: UTC time (NOT USED) 3: Time 2: User time (NOT USED)
Field 7	Byte 33	Distance type		Value: bits 0-3 0: N/A 1: INS distance (NOT USED)
Field 4	Bytes 34 to 37	Smart Heave	float	Delayed heave in meters, positive down
Field 5	Bytes 38 to 41	Smart Heave RMS	float	Delayed heave standard deviation in meters (NOT USED)
Field 6	Bytes 42 to 45	Status	ulong	Bit 0=1 : Smart heave Valid Bit1=1 : Real-Time heave Valid Bit 2 to 31 : Reserved
Field 7	Bytes 46 to 49	Real Time Heave	float	in meters, positive down
Field 8	Bytes 50 to 53	Real Time Heave RMS	float	standard deviation in meters (NOT USED)
Field 9	Bytes 54 to 61	Heave Time 1	double	Delayed heave time (**) seconds
Field 10	Bytes 62 to 69	Heave Time 2	double	(NOT USED)
Field 11	Bytes 70 to 73	Rejected IMU data count	ulong	(NOT USED)
Field 12	Bytes 74 to 77	Out of range IMU data count	ulong	(NOT USED)
Field 13	Bytes 78 to 79	Pad	byte	Set to 0
Field 14	Bytes 80 to 81	Checksum	ushort	16 bit sum of all data in the

Message <F0><F1><F2>…..<F15>				
				data group: byte 34 to 79
Field 15	Bytes 82 to 83	$#	char	Group end

(*) Only INS time or UTC time will be flagged. UTC time is flagged when INS time synchronization with GNSS is valid (ZDA or ZDA+PPS valid). By default INS should be time synchronized with GNSS time. It is highly recommended to use PPS pulse for accurate timing.

(**) If INS is time synchronized with GNSS (ZDA+PPS) Time is UTC seconds in the week (0-604800= 7 x 86400sec per day). Otherwise it is time since power on of the INS.
Both Smart Heave (100 s delayed heave) and real time heave are time-matched to “Heave Time 1”.

All data not output by INS, labeled “NOT USED”, will be set to 0 default value.

POSTPROCESSING

Please note that this protocol is not detailed in this documentation.

To get more information about this protocol, contact iXblue.

PRDID

Standard: Output NMEA 0183 compatible.
Data sent: Pitch, Roll, Heading.
Data frame:

$PRDID,aPPP.PP,bRRR.RR,HHH.HH*hh<CR><LF>

aPPP.PP	is the pitch in degrees and a, its sign character '–' when bow down, '+' when bow up Warning: Opposite sign of product usual convention	2 digits after decimal point
bRRR.RR	is the roll angle in degrees and a, its sign character '–' for port down, '+' for port up	2 digits after decimal point
HHH.HH	is the heading in degrees.	
hh	is the checksum	

PRDID TSS

Standard: Output NMEA 0183 compatible.
Data sent: Pitch, Roll, Heading.
Data frame:

$PRDID,aPPP.PP,bRRR.RR,HHH.HH*hh<CR><LF>

aPPP.PP	is the pitch in degrees and a, its sign character '–' when bow down, '+' when bow up Warning: Opposite sign of INS usual convention	(*) 2 digits after the decimal point
bRRR.RR	is the roll angle in degrees and a, its sign character '–' for port down, '+' for port up	(*) 2 digits after the decimal point
HHH.HH	is the heading in degrees	
hh	is the checksum	

(*) The attitude angles are computed with respect to TSS convention.
Roll and Pitch are referenced to the local vertical acceleration.
The formula calculation with respect to INS standard convention (Euler Angle or Tate Bryant) is given hereafter:
$Roll_{TSS} = Sin^{-1}$ (Sin ($Roll_{TB}$) x Cos ($Pitch_{TB}$)) and $Pitch_{TSS} = Pitch_{TB}$

PRECISE ZDA

Standard: ASCII NMEA 0183

Data sent: Latitude, Hemisphere, Longitude, Direction, GNSS mode, Norm speed, Roll, Pitch, Heading

Data frame:

Following frame is sent at 1Hz:

$GPZDA,hhmmss.ss,dd,mm,yyyy,hh,mm*hh<CR><LF>		
hhmmss.ssssss	UTC time	**
dd	UTC day	**
mm	UTC month	**
yyyy	UTC year	**
hh	local zone hours	*
mm	local zone minutes	*
hh	checksum	**

* Copy of last GNSS values received. When no GNSS has been received since power-up, these fields are null.

** INS calculated data.

The time in the message has 6 digits after the comma. It corresponds to the transmission of the PPS LIKE output pulse, if this pulse is enabled.

PTNL GGK

Standard: Output NMEA 0183 compatible.

Data sent: The $PTNLG, GGK is provided to simulate a Trimble GPS output. Some characters of this output frame are set to fixed values and some empty fields are allowed.

See samples hereafter when there is no ZDA input :

$PTNL,GGK,000527.01,,4852.2000122,N,00200.0000013,E,6,03,613.8,EHT0.000,M*7C

Time management in INS: If INS has never received GNSS, date starts on 1st January 2006. Otherwise date is maintained on GNSS time as soon as GNSS time is received.

If GNSS is lost, INS will maintain time with its internal clock.

If INS is synchronized with GNSS time, the time tags are UTC time, otherwise they are INS time (time since power-up of the system).

$PTNL,GGK,hhmmss.ss,ddmmyy,LLII.IIIIIII,a, LLII.IIIIIII,a,x,y.z.z,EHTx.xxx,M*hh<CR><LF>		
GGK	Message ID	Fixed value
hhmmss.ss	UTC time of position fix.	**
ddmmyy	UTC date of position fix **(empty field allowed)**.	**
LLII.IIIIIII	Latitude in degrees (LL) and in minutes (II.IIIIIII)	**
a	'N' for Northern hemisphere, 'S' for Southern hemisphere	**
LLII.IIIIIII	Longitude in deg (LLL) and in minutes (II.IIIIIII)	**
a	'E' for East, 'W' for West	**
x	GNSS quality indicator***	**
y	Number of satellites in fix	*
z.z	DOP of fix (1 digit after coma)	**
EHTx.xxx	Height above ellipsoid (3 digits after comma)	**
M	Ellipsoid height measured in meters.	*
hh	Checksum	**

* Copy of last GNSS values received. When no GNSS has been received since power-up, these fields are null.

** The INS calculated data.

*** The quality indicator is managed as follows:

- The INS does not copy the quality indicator received on GGA input to GGA output.
- The quality factor is set with respect to a correspondence table between INS calculated SD and Quality indicator in GGA telegram (refer to section 1.4).

Sample telegram:

$PTNL,GGK,180432.00,101300,4027.0279123,N,08704.8570697,W,4,07,1.7,EHT178.340,M*69

RDI PD11

Standard: ASCII NMEA 0183

Data sent: There are 3 NMEA0183 sentences containing sensor and navigational data. All data are here INS computed output data. All data fields are variable width. Empty data fields will indicate missing or invalid data.

Note:

- Data is output in telegram $PRDIH:
- if GNSS and/or DVL bottom track data is valid.
- Data is output in telegram $PRDII:
- If GNSS or DVL water track data is valid.
- If no GNSS, DVL water track and DVL bottom track are valid, no data is output in $PRDIH and $PRDII telegrams.
- In all cases data in the $PRDIG, $PRDIH and $PRDII telegram are INS data.

Data frame:

$PRDIG,H,x.xxx,P,x.xxx,R,x.xxx,D,x.xxx*hh<CR><LF>			
H	Heading ID	Fixed character	
x.xxx	Heading	0.000 to 360.000 deg	3 digits after decimal point
P	Pitch ID	Fixed character	
x.xxx	Pitch	-90.000 to 90.000 deg	3 digits after decimal point
R	Roll ID	Fixed character	
x.xxx	Roll	-180.000 to 180.000 deg	3 digits after decimal point
D	Depth ID	Fixed character	
x.xxx	Depth below surface	In meters	3 digits after decimal point
hh	Is the checksum		

$PRDIH,R,x.x,S,x.xxx,C,x.xxx*hh<CR><LF>			
R	Range to bottom ID	Fixed character	
x.x	Range to bottom	In meters	1 digit after decimal point
S	Speed over ground ID	Fixed character	
x.xxx	Speed over ground	In meters/second	3 digit after decimal point
C	Course over ground ID	Fixed character	
x.xxx	Course of speed over ground	0.000 to 360.000 deg	3 digit after decimal point
hh	Is the checksum		

$PRDII,S,x.xxx,C,x.xxx*hh<CR><LF>			
S	Speed relative to water ID	Fixed character	
x.xxx	Speed relative to water	In meters/second	3 digit after decimal point
C	Course of speed relative to water ID	Fixed character	
x.xxx	Course of speed relative to water	0.000 to 360.000 deg	3 digit after decimal point
hh	Is the checksum		

RDI PING

Standard: ASCII

Data sent: RDI Start Pinging command

Data frame:

When this protocol is selected on the INS output port that is connected to the DVL input, the INS will send a CS command each time the INS is powered on. This can avoid issue of DVL stopping to ping when INS is powered on. At INS power on, the DVL can sometimes interprets initial flow sent on the output of INS as a break command.

CS<CR><LF>	
CS	Start Ping Command

RDI SYNC

Standard: ASCII

Data frame:

This output protocol is used to trigger acoustic emission at the rate chosen by this output protocol.

DVL should be set with CF10110 instead of CF11110 (default value).

Sending the F1 character triggers the acoustic output at the chosen output frequency of this protocol. In this case the firmware will time tag the date a t_emission + (accoustic turnaround propagation time/2). Acoustic turnaround propagation time= Bottom_Range/(Sound speed * cos (30°)).

This has the advantage of having a more accurate time_stamping since we are not affected by variable processing time that is not accurately known.

F1<CR><LF>		
F1	Space + enter	Fixed character

RIEGL

Standard: ARINC 705 (aviation standard).
Data sent: Latitude, longitude, altitude, roll, pitch, yaw.
Data frame: Data is separated by space characters.

Example: 50759.013 48.33841614 15.93149532 471.005 -0.8156 7.1238 71.1383

t.ttt x.xxxxxxxx y.yyyyyyyy z.zzz r.rrrr p.pppp h.hhhh<CR><LF>

t.ttt	UTC time stamp in seconds (0-24 h) (*)	Long double, 3 digits after the decimal point
x.xxxxxxxx	Latitude in degrees	Float, 8 digits after the decimal point
y.yyyyyyyy	Longitude in degrees	Float, 8 digits after the decimal point
z.zzz	Altitude in meters	Float, 3 digits after the decimal point
r.rrrr	Roll in degrees, positive port up, [-180° to +180°]	Float, 4 digits after the decimal point
p.pppp	Pitch in degrees, positive bow up, [-90° to +90°]	Float, 4 digits after the decimal point
h.hhh	Heading in degrees, [0° to 360°]	Float, 4 digits after the decimal point

(*) INS time or UTC time if INS is synchronized with GNSS time.

S40 NAV10

Standard: Binary protocol.

Data sent: Status, Date, GNSS Time, Heading, Roll, Pitch, Heading rate, Roll rate, Pitch rate, Latitude, Longitude, Depth, Latitude accuracy, Longitude accuracy, position correlation, GNSS latitude, GNSS longitude, North Velocity, East Velocity, Down Velocity, Log speed, Course made good, Speed over ground, Direction of the current, Speed of the current.

Data frame: The frame contains 78 bytes. For data coded on several bytes, the bytes are sent MSB byte first. All signed integer are coded as two's complement.

This protocol telegram is assumed to be sent at the rate of 10 Hz.

Message <F0><F1><F2>…..<F33>			
Field 0	Byte 0 Byte 1	Header	0x5A 0xA5
Field 1	Byte 2	NUMDATA	0x48
Field 2	Byte 3	IDENT	0x02
Field 3	Byte 4	Status 1	**Table 1**
Field 4	Byte 5	Status 2	**Table 2**
Field 5	Byte 6	BITE Status	**Table 3**
Field 6	Bytes 7-8	Date **Note 1**	Unsigned 16 bits; LSB= 1 day, [1 to 366 days]
Field 7	Bytes 9-11	Time Ref GNSS	Unsigned 24 bits; LSB= 0.01 s, [0 to 86400 s]
Field 8	Bytes 12-13	Spare	Set to 0
Field 9	Bytes 14-15	Heading	Unsigned 16 bits; LSB= $180/2^{15}$ deg, [0 to 360 deg] Positive and increasing when bow turns to starboard.
Field 10	Bytes 16-17	Roll	Signed 16 bits; LSB= $90/2^{15}$ deg, ±90 deg Positive for left-port side down. Warning: Opposite sign of INS usual convention Inverse ISO Convention **Note 9**
Field 11	Bytes 18-19	Pitch	Signed 16 bits; LSB= $90/2^{15}$ deg, ±90 deg Positive for front side-bow down. Inverse ISO Convention **Note 9**
Field 12	Bytes 20-21	Heading rate **Note 2**	Signed 16 bits; LSB= $45/2^{15}$ deg/s, ±45 deg/s Positive when heading angle increasing
Field 13	Bytes 22-23	Roll rate **Note 2**	Signed 16 bits; LSB= $45/2^{15}$ deg/s, ±45 deg/s Positive when roll angle increasing
Field 14	Bytes 24-25	Pitch rate **Note 2**	Signed 16 bits; LSB= $45/2^{15}$ deg/s, ±45 deg/s Positive when pitch angle increasing
Field 15	Bytes 26-29	Latitude	Signed 32 bits; LSB= $90/2^{31}$ deg, ±90 deg Positive in northern hemisphere.
Field 16	Bytes 30-33	Longitude	Signed 32 bits; LSB= $180/2^{31}$ deg,

Message <F0><F1><F2>…..<F33>			
			±180 deg Positive East of Greenwich meridian, up to 180 deg.
Field 17	Bytes 34-35	Depth **Note 3**	Signed 16 bits; LSB= $655.36/2^{15}$, ±655.36 m
Field 18	Bytes 36-39	Latitude accuracy **Note 4**	Unsigned 32 bits; LSB= $10800/2^{31}$ Nm, [0 to 10800 Nm]
Field 19	Bytes 40-43	Longitude accuracy **Note 4**	Unsigned 32 bits; LSB= $10800/2^{31}$ Nm, [0 to 10800 Nm]
Field 20	Bytes 44-45	Position correlation **Note 4**	Signed 16 bits; LSB= $1/2^{15}$, ±1
Field 21	Bytes 46-49	GNSS Latitude	Signed 32 bits; LSB= $90/2^{31}$ deg, ±90 deg Positive in northern hemisphere.
Field 22	Bytes 50-53	GNSS Longitude	Signed 32 bits; LSB= $180/2^{31}$ deg, ±180 deg Positive East of Greenwich meridian, up to 180 deg.
Field 23	Bytes 54-55	North Velocity	Signed 16 bits; LSB= $65.536/2^{15}$, ±65.536 m/s
Field 24	Bytes 56-57	East Velocity	Signed 16 bits; LSB= $65.536/2^{15}$, ±65.536 m/s
Field 25	Bytes 58-59	Down Velocity	Signed 16 bits; LSB= $65.536/2^{15}$, ±65.536 m/s
Field 26	Bytes 60-61	Log Speed	Signed 16 bits; LSB= $65.536/2^{15}$, ±65.536 m/s

Navigation Data			
Field 27	Bytes 62-63	Course made good **Note 5**	Unsigned 16 bits; LSB= 180/215 deg, [0 to 360 deg]
Field 28	Bytes 64-65	Speed over ground **Note 5**	Unsigned 16 bits; LSB= 65.536/215, [0 to 65.536 m/s]
Field 29	Bytes 66-67	Direction of the current **Note 11**	Unsigned 16 bits; LSB= 180/215 deg, [0 to 360 deg]
Field 30	Bytes 68-69	Speed of the current **Note 10**	Unsigned 16 bits; LSB= 65.536/215, [0 to 65.536 m/s]
Field 31	Bytes 70-75	Spare	Set to 0
Field 32	Byte 76	Checksum	**Note 6**
Field 33	Byte 77	Terminator	0xAA

Table 18 - Byte STATUS 1

7|6|5|4|3|2|1|0	Parameter	Description	Attached INS status bits
# # # # # 0 0 0	Built- In- Test Equipment **Note 7**	Level 0 – No failure	N/A
# # # # # 0 0 1		Level 1 – Anomaly	When one of those INS User status bit is set to 0: DVL_RECEIVED_VALID GPS_RECEIVED_VALID GPS2_RECEIVED_VALID EMLOG_RECEIVED_VALID DEPTH_RECEIVED_VALID TIME_RECEIVED_VALID **Note 8**
# # # # # 0 1 0		Level 2 – Warning	When one of those INS User status bit is set to 1: ALTITUDE_SATURATION CPU_OVERLOAD TEMPERATURE_ERR INPUT_x_ERR OUTPUT_x_ERR
# # # # # 0 1 1		Level 3 – Failure (Attitude data not valid)	When one of those INS User status bit is set to 1: DEGRADED_MODE HRP_INVALID DYNAMIC_EXCEEDED
# # # # # 1 0 0		Level 4 – Failure (no data valid)	INS User status FAILURE_MODE set to 1
# # # 0 0 # # #	Mode	Navigation – Sea	INS User status NAVIGATION set to 1
# # # 0 1 # # #		Navigation – Quay	INS Algo 2 status ZUPT_MODE_VALID set to 1
# # # 1 0 # # #		Alignment	INS User status ALIGNMENT set to 1
# # # 1 1 # # #		Maintenance	INS System status 2 SIMULATION_MODE set to 1
# 0 0 # # # # #	Attitude and Heading Reference validity	Data valid	N/A
# 0 1 # # # # #		Data degraded	When one of those INS User status bit is set to 1: DEGRADED_MODE FINE_ALIGNMENT
# 1 0 # # # # #		Data not valid	INS User status HRP_INVALID set to 1
x # # # # # # #	/	Reserved	N/A

Table 19 - Byte STATUS 2

7|6|5|4|3|2|1|0	Parameter	Description	Attached INS status bits
# # # # # # # 0	GNSS time validity	Data valid	INS User status TIME_RECEIVED_VALID set to 1
# # # # # # # 1		Data not valid	INS User status TIME_RECEIVED_VALID set to 0
# # # # # # 0 #	GNSS position Validity **Note 8**	Data valid	INS User status GPS_RECEIVED_VALID set to 1 or GPS2_RECEIVED_VALID set to 1
# # # # # # 1 #		Data not valid	When one of those INS User status bit is set to 0: GPS_RECEIVED_VALID GPS2_RECEIVED_ VALID
# # # # # x # #	/	Reserved	N/A
# # # # 0 # # #	Log speed **Note 8**	Data valid	INS User status EMLOG_RECEIVED_VALID set to 1
# # # # 1 # # #		Data not valid	INS User status EMLOG_RECEIVED_VALID set to 0
# # # 0 # # # #	Navigation data	Data valid	**Note 5**
# # # 1 # # # #		Data not valid	
# # x # # # # #	/	Reserved	N/A
# 0 # # # # # #	Simulation	Operational	INS System status 2 SIMULATION_MODE set to 0
# 1 # # # # # #		Simulation mode	INS System status 2 SIMULATION_MODE set to 1
0 # # # # # # #	INS Identification	when last digit of IP address is even	No INS status is related to this bit.
1 # # # # # # #		when last digit of IP address is odd	

Table 20 - BITE STATUS

iXblue has defined the following table for the BITE status.

Bit N°	7	6	5	4	3	2	1	0
User status bits	N/A	9	10	11	12	13	14	15
Message description	Reserved as spare	FOG_ANOMALY	ACC_ANOMALY	TEMPERATURE_ERR	CPU_OVERLOAD	DYNAMIC_EXCEEDED	SPEED_SATURATION	ALTITUDE_SATURATION

Note 1:

The date corresponds to the number of the day in the year. If the date is not included in the messages received from the GNSS, bytes 7 and 8 of the field 6 are set to 1. The time corresponds to the GNSS time in seconds since the beginning of the GNSS day. When the INS is powered on, time is reset to 0 and until reception of a valid GNSS message including a valid time, the time emitted in messages will be the INS internal system time (number of seconds since the start).

When a GNSS time is received, the time is set at this value, and then it is updated by INS internal clock until the next time received in a GNSS message.

The date and UTC synchronization time can be received in ZDA, RMC or UTC messages, if sent by GNSS.

Note 2:

The attitude rate data corresponds to the time derivation of the attitude data, respect to the S40 telegram attitude data sign convention (ISO Convention or Inverse ISO Convention).

Note 3:

The depth is positive under sea level . When depth sensor altitude mode is selected, the depth corresponds to the INS altitude. For a surface ship (for Stabilization, GPS or Hydro altitude mode), the depth corresponds to the heave (the mean value is then 0).

Note 4:

The position accuracy corresponds to the maximum error estimation (3σ value = 99% probability). The position correlation corresponds to: σLat.Lon/(σLat. σLon) where σLat.Lon is the cross-covariance of latitude and Longitude.

Note 5:

The Navigation data bit (bit 4 of the status 2) is set to 1 when one of the following condition is met:

- The speed over ground (field 28) < 1 knot (then course made good (field 27) is set to 0).
- The Built-In-Test Equipement status (see status 1) is on level 3 or 4: i.e: when one of those INS User status bit is set to 1: DEGRADED_MODE, HRP_INVALID DYNAMIC_ EXCEEDED, FAILURE_MODE.
- The INS is not in navigation mode at sea situation: i.e: INS User status NAVIGATION is set to 0
- The current standard deviation > 95% of the current standard deviation set in the Web-Based Graphical User Interface

Note 6:

One byte is used and it represents the checksum of the message. It is an 8-bit word. Checksum is a binary addition without carry modulo 256 of all data bytes, NUMDATA and IDENT. Therefore, Header, Checksum and Terminator are excluded from this addition.

Note 7:

For the parameter Built in Test Equipment, the level 4 is checked first, than level 3, then level 2 and finally level 1.

Note 8:
Only input sensors that are configured through the Web-Based Graphical User Interface are taken into account. In Simulation mode, sensors flags are not managed and the operator should not take them into account.

Note 9:
Sign convention of roll and pitch:

	ISO Convention	Inverse ISO Convention	INS Convention
Roll > 0	for left-port side up	for left-port side down	for left-port side up
Pitch > 0	for front-side bow up	for front side-bow down	for front side-bow down

Note 10:
The speed of current is computed as follow:

$$\sqrt{(Vc_{north})^2 + (Vc_{east})^2}$$

Note 11:
The direction of current is issued from the angle:

$$\alpha = arc\tan\left(\frac{Vc_{east}}{Vc_{north}}\right)$$

Direction of current	$Vc_{east} > 0$	$Vc_{east} < 0$
$Vc_{north} > 0$	abs(α)	2*PI – abs(α)
$Vc_{north} < 0$	PI – abs(α)	PI + abs(α)

If Vc_{north} is 0, the direction of current is not calculated and set to 0.

Note 12:
The down acceleration is compensated from the g measurement.

S40 NAV100

Standard: Binary protocol.

Data sent: Status, Heading, Roll, Pitch, Heading rate, Roll rate, Pitch rate, North Velocity, East Velocity, Down Velocity, North acceleration, East acceleration, Down acceleration.

Data frame: The frame contains 32 bytes. For data coded on several bytes, the bytes are sent MSB byte first. All signed integer are coded as two's complement. This protocol telegram is assumed to be sent at the rate of 100 Hz.

Message <F0><F1><F2>…..<F18>			
Field 0	Byte 0 Byte 1	Header	0x5A 0xA5
Field 1	Byte 2	NUMDATA	0x1A
Field 2	Byte 3	IDENT	0x01
Field 3	Byte 4	Status 1	**Table 1**
Field 4	Byte 5	Status 2	**Table 2**
Field 5	Bytes 6-7	Heading	Unsigned 16 bits; LSB= $180/2^{15}$ deg, [0 to 360 deg] Positive and increasing when bow turns starboard.
Field 6	Bytes 8-9	Roll	Signed 16 bits; LSB= $90/2^{15}$ deg, ±90 deg Positive for left-port side down. **Warning**: Opposite sign of INS usual convention Inverse ISO Convention **Note 9**
Field 7	Bytes 10-11	Pitch	Signed 16 bits; LSB= $90/2^{15}$ deg, ±90 deg Positive for front side-bow down. Inverse ISO Convention **Note 9**
Field 8	Bytes 12-13	Heading rate **Note 2**	Signed 16 bits; LSB= $45/2^{15}$ deg/s, ±45 deg/s Positive when heading angle increasing
Field 9	Bytes 14-15	Roll rate **Note 2**	Signed 16 bits; LSB= $45/2^{15}$ deg/s, ±45 deg/s Positive when roll angle increasing
Field 10	Bytes 16-17	Pitch rate **Note 2**	Signed 16 bits; LSB= $45/2^{15}$ deg/s, ±45 deg/s Positive when pitch angle increasing
Field 11	Bytes 18-19	North Velocity	Signed 16 bits; LSB= $65.536/2^{15}$, ±65.536 m/s
Field 12	Bytes 20-21	East Velocity	Signed 16 bits; LSB= $65.536/2^{15}$, ±65.536 m/s
Field 13	Bytes 22-23	Down Velocity	Signed 16 bits; LSB= $65.536/2^{15}$, ±65.536 m/s
Field 14	Bytes 24-25	North acceleration	Signed 16 bits; LSB= $327.68/2^{15}$, ±327.68 m/s²
Field 15	Bytes 26-27	East acceleration	Signed 16 bits; LSB= $327.68/2^{15}$, ±327.68 m/s²
Field 16	Bytes 28-29	Down acceleration Note 12	Signed 16 bits; LSB= $327.68/2^{15}$, ±327.68 m/s²
Field 17	Byte 30	Checksum	**Note 6**
Field 18	Byte 31	Terminator	0xAA

Table 21 - Byte STATUS 1

7|6|5|4|3|2|1|0	Parameter	Description	Attached INS status bits
# # # # # 0 0 0	Built- In- Test Equipment **Note 7**	Level 0 – No failure	N/A
# # # # # 0 0 1		Level 1 – Anomaly	When one of those INS User status bit is set to 0: DVL_RECEIVED_VALID GPS_RECEIVED_VALID GPS2_RECEIVED_VALID EMLOG_RECEIVED_VALID DEPTH_RECEIVED_VALID TIME_RECEIVED_VALID **Note 8**
# # # # # 0 1 0		Level 2 – Warning	When one of those INS User status bit is set to 1: ALTITUDE_SATURATION CPU_OVERLOAD TEMPERATURE_ERR INPUT_x_ERR OUTPUT_x_ERR
# # # # # 0 1 1		Level 3 – Failure (Attitude data not valid)	When one of those INS User status bit is set to 1: DEGRADED_MODE HRP_INVALID DYNAMIC_EXCEEDED
# # # # # 1 0 0		Level 4 – Failure (no data valid)	INS User status FAILURE_MODE set to 1
# # # 0 0 # # #	Mode	Navigation – Sea	INS User status NAVIGATION set to 1
# # # 0 1 # # #		Navigation – Quay	INS Algo 2 status ZUPT_MODE_VALID set to 1
# # # 1 0 # # #		Alignment	INS User status ALIGNMENT set to 1
# # # 1 1 # # #		Maintenance	INS System status 2 SIMULATION_MODE set to 1
# 0 0 # # # # #	Attitude and Heading Reference validity	Data valid	N/A
# 0 1 # # # # #		Data degraded	When one of those INS User status bit is set to 1: DEGRADED_MODE FINE_ALIGNMENT
# 1 0 # # # # #		Data not valid	INS User status HRP_INVALID set to 1
x # # # # # # #	/	Reserved	N/A

Table 22 - Byte STATUS 2

7|6|5|4|3|2|1|0	Parameter	Description	Attached INS status bits
# # # # # # # 0	GPS time validity	Data valid	INS User status TIME_RECEIVED_VALID set to 1
# # # # # # # 1		Data not valid	INS User status TIME_RECEIVED_VALID set to 0
# # # # # # 0 #	GPS position Validity **Note 8**	Data valid	INS User status GPS_RECEIVED_VALID set to 1 or GPS2_RECEIVED_VALID set to 1
# # # # # # 1 #		Data not valid	When one of those INS User status bit is set to 0: GPS_RECEIVED_VALID GPS2_RECEIVED_VALID
# # # # # x # #	/	Reserved	N/A
# # # # 0 # # #	Log speed **Note 8**	Data valid	INS User status EMLOG_RECEIVED_VALID set to 1
# # # # 1 # # #		Data not valid	INS User status EMLOG_RECEIVED_VALID set to 0
# # # 0 # # # #	Navigation data	Data valid	**Note 5**
# # # 1 # # # #		Data not valid	
# # x # # # # #	/	Reserved	N/A
# 0 # # # # # #	Simulation	Operational	INS System status 2 SIMULATION_MODE set to 0
# 1 # # # # # #		Simulation mode	INS System status 2 SIMULATION_MODE set to 1
0 # # # # # # #	INS Identification	when last digit of IP address is even	No INS status is related to this bit.
1 # # # # # # #		when last digit of IP address is odd	

Table 23 - BITE STATUS

iXblue has defined the following table for the BITE status.

Bit N°	7	6	5	4	3	2	1	0
User status bits	N/A	9	10	11	12	13	14	15
Message description	Reserved as spare	FOG_ANOMALY	ACC_ANOMALY	TEMPERATURE_ERR	CPU_OVERLOAD	DYNAMIC_EXCEEDED	SPEED_SATURATION	ALTITUDE_SATURATION

Note 1:

The date corresponds to the number of the day in the year. If the date is not included in the messages received from the GPS, bytes 7 and 8 of the field 6 are set to 1. The time corresponds to the GPS time in seconds since the beginning of the GPS day. When the INS is powered on, time is reset to 0 and until reception of a valid GPS message including a valid time, the time emitted in messages will be the INS internal system time (number of seconds since the start).

When a GPS time is received, the time is set at this value, and then it is updated by INS internal clock until the next time received in a GPS message.

The date and UTC synchronization time can be received in ZDA, RMC or UTC messages, if sent by GPS.

Note 2:

The attitude rate data corresponds to the time derivation of the attitude data, respect to the S40 telegram attitude data sign convention (ISO Convention or Inverse ISO Convention).

Note 3:

The depth is positive under sea level . When depth sensor altitude mode is selected, the depth corresponds to the INS altitude. For a surface ship (for Stabilization, GPS or Hydro altitude mode), the depth corresponds to the heave (the mean value is then 0).

Note 4:

The position accuracy corresponds to the maximum error estimation (3σ value = 99% probability). The position correlation corresponds to: σLat.Lon/(σLat. σLon) where σLat.Lon is the cross-covariance of latitude and Longitude.

Note 5:

The Navigation data bit (bit 4 of the status 2) is set to 1 when one of the following condition is met:

- The speed over ground (field 28) < 1 knot (then course made good (field 27) is set to 0).
- The Built-In-Test Equipement status (see status 1) is on level 3 or 4: i.e: when one of those INS User status bit is set to 1: DEGRADED_MODE, HRP_INVALID DYNAMIC_ EXCEEDED, FAILURE_MODE.
- The INS is not in navigation mode at sea situation: i.e: INS User status NAVIGATION is set to 0
- The current standard deviation > 95% of the current standard deviation set in the Web-Based Graphical User Interface

Note 6:

One byte is used and it represents the checksum of the message. It is an 8-bit word. The checksum is a binary addition without carry modulo 256 of all data bytes, NUMDATA and IDENT. Therefore, Header, Checksum and Terminator are excluded from this addition.

Note 7:

For the parameter Built in Test Equipment, the level 4 is checked first, then level 3, then level 2 and finally level 1.

Note 8:

Only input sensors that are configured through the MMI are taken into account. In Simulation mode, sensors flags are not managed and the operator should not take them into account.

Note 9:

Sign convention of roll and pitch:

	ISO Convention	Inverse ISO Convention	INS Convention
Roll > 0	for left-port side up	for left-port side down	for left-port side up
Pitch > 0	for front-side bow up	for front side-bow down	for front side-bow down

Note 10:

The speed of current is computed as follow:

$$\sqrt{(Vc_{north})^2 + (Vc_{east})^2}$$

Note 11:

The direction of current is issued from the angle:

$$\alpha = arc\tan\left(\frac{Vc_{east}}{Vc_{north}}\right)$$

Direction of current	$Vc_{east} > 0$	$Vc_{east} < 0$
$Vc_{north} > 0$	abs(α)	2*PI – abs(α)
$Vc_{north} < 0$	PI – abs(α)	PI + abs(α)

If Vc_{north} is 0, the direction of current is not calculated and set to 0.

Note 12:

The down acceleration is compensated from the g measurement.

SEANAV ID1

Standard: Binary standard protocol.
Data received: Status, Latitude, Longitude, Altitude, Date, Roll, Pitch, Heading, Vehicle frame velocities, Vehicle frame accelerations, Heading rate, Roll rate, Pitch rate, Time, Heave, Surge, Sway.
Data frame: 56 bytes - For data coded on several bytes, the bytes are sent LSB byte first. All signed integer are coded as two's complement.

Message <F0><F1>…<F11>			
Field 0	Byte 0	Header byte	0xAA
Field 1	Byte 1	ID	0x01
Field 2	Byte 2	Message length	0x34
Field 3	Bytes 3-4	Transmission counter	Unsigned 16 bits integer. Cycling from 0 to 65535.
Field 4	Byte 5	Alignment status	**Note 2**
Field 5	Byte 6	Navigation aid status	**Note 3**
Field 6	Bytes 7-10	Latitude	Signed 32 bits integer. +/-90 deg. LSB = $180/2^{31}$ deg Positive in north hemisphere and increasing to north pole
Field 7	Bytes 11-14	Longitude	Signed 32 bits integer. +/-180 deg. LSB = $180/2^{31}$ deg Positive and increasing towards east from Greenwich, negative west from Greenwich
Field 8	Bytes 15-18	INS Depth (-INS Altitude)	Signed 32 bits integer. LSB = 1 cm, positive down
Field 9	Bytes 19-20	Date	Unsigned 16 bits integer. b15 b14 b13 b12 b11 b10 b9 : year since 1970 b8 b7 b6 b5 : month [1,12] b4 b3 b2 b1 b0 : day [1,31]
Field 10	Bytes 21-22	Roll	Signed 16 bits integer. +/-180 deg. LSB = $180/2^{15}$ deg Positive when port side up
Field 11	Bytes 23-24	Pitch	Signed 16 bits integer. +/-90deg. LSB =$180/2^{15}$ deg Positive when bow up **Warning**: Opposite sign of INS usual convention.
Field 12	Bytes 25-26	Heading	Unsigned 16 bits integer. [0,360°[. LSB =$180/2^{15}$ deg 0° at North and increasing from North to East
Field 13	Bytes 27-28	XV1 Velocity	Signed 16 bits integer. +/-32 m/s. LSB = $32/2^{15}$ m/s Positive when moving forward
Field 14	Bytes 29-30	XV2 Velocity	Signed 16 bits integer. +/-32 m/s. LSB = $32/2^{15}$ m/s Positive when moving toward port side
Field 15	Bytes 31-32	XV3 Velocity	Signed 16 bits integer. +/-32 m/s. LSB = $32/2^{15}$ m/s Positive when moving upwards
Field 16	Bytes 33-34	XV1 longitudinal acceleration	Signed 16 bits integer. +/-32 m/s2. LSB = $32/2^{15}$ m/s^2 Positive when accelerating forward

Message <F0><F1>…<F11>			
Field 17	Bytes 35-36	XV2 transverse acceleration	Signed 16 bits integer. +/-32 m/s2. LSB = $32/2^{15}$ m/s^2 Positive when accelerating to port
Field 18	Bytes 37-38	XV3 vertical acceleration	Signed 16 bits integer. +/-32 m/s2. LSB = $32/2^{15}$ m/s^2 Positive when accelerating upwards, compensated from gravity
Field 19	Bytes 39-40	Roll rate **Note 1**	Signed 16 bits integer. +/-4 rad/s. LSB = $4/2^{15}$ rad/s Sign of the SEANAV_ID1 roll derivative
Field 20	Bytes 41-42	Pitch rate **Note 1**	Signed 16 bits integer. +/-4 rad/s. LSB = $4/2^{15}$ rad/s Sign of the SEANAV_ID1 pitch derivative Warning: Opposite sign of INS usual convention
Field 21	Bytes 43-44	Heading rate **Note 1**	Signed 16 bits integer. +/-4 rad/s. LSB = $4/2^{15}$ rad/s Sign of the SEANAV_ID1 heading derivative
Field 22	Bytes 45-48	INS Time	b31 : **Note 6** b30 b29….b4 b3 b2 b1 b0 : Time in day since 00:00:00 [0, 86400 s[LSB = $1/2^{14}$ sec
Field 23	Bytes 49-50	Surge	Signed 16 bits integer. +/-64 m. LSB = $64/2^{15}$ m Positive forward
Field 24	Bytes 51-52	Sway	Signed 16 bits integer. +/-64 m. LSB = $64/2^{15}$ m Positive port
Field 25	Bytes 53-54	Heave	Signed 16 bits integer. +/-64 m. LSB = $64/2^{15}$ m Positive upwards
Field 26	Byte 55	Checksum	XOR of all the bytes from 1 to 54 Initial value = 0x00

Note 1: These fields are attitude rates and are given in the INS reference frame (X1, X2, X3). They are not XV1, XV2, XV3 rotations rates.

Note 2: Alignment status byte.

Function	Value	Links with INS status words
IDLE **Note 4**	= 0x00	INS System status 2 WAIT_FOR_POSITION
Coarse stationary alignment **Note 5**	= 0x01	INS User status & INS System status 2 ALIGNMENT AND NOT WAIT_FOR_POSITION AND NOT Alignment mode (Note 3)
Fine stationary alignment – Not complete	= 0x02	INS User status FINE_ALIGNMENT AND NOT Alignment mode (Note 3)
Fine stationary alignment – Complete	= 0x03	INS User status NOT (ALIGNMENT OR FINE_ALIGNEMENT) AND NOT Alignment mode (Note 3)

Function	Value	Links with INS status words
Coarse GNSS alignment Note 5	= 0x04	INS User status & INS System status 2 ALIGNMENT AND NOT WAIT_FOR_POSITION AND Alignment mode (Note 3)
Fine GNSS alignment – Not complete	= 0x05	INS User status FINE_ALIGNMENT AND Alignment mode (Note 3)
Fine GNSS alignment – Complete	= 0x06	INS User status NOT (ALIGNMENT OR FINE_ALIGNEMENT) AND Alignment mode (Note 3)
Doppler/EM Log alignment – Not complete	= 0x07 Not used	NA
Spare	= 0x08 Not used	NA
Aided navigation	= 0x09 Not used	NA
NO GO (System fault) Note 5	= 0x0A	INS User status CPU_OVERLOAD OR TEMPERATURE_ERR OR INPUT_x_ERR OR OUTPUT_x_ERR OR FAILURE_MODE

Note 3: Navigation aid status byte

Function	Bit #	Value	Links with INS status words
Spare	0	= 0	NA
Alignment mode	1	= 0 without position = 1 with position	INS System status 2 GPS_DETECTED OR GPS2_DETECTED seen at least once during coarse alignment with a rejection mode set to Always True or to Automatic Reacquisition. When the SEANAV_ ID1 alignment mode flag is set to one, it will remain set to one for GPS dropouts.
ZUPT mode	2	= 0 ZUPT processing off = 1 ZUPT processing on	INS Algorithm status 2 ZUPT_MODE_VALID OR AUTOSTATICBENCH_ ZUPT_ MODE_ VALID
Spare	3	= 0	NA
Spare	4	= 0	NA

Function	Bit #	Value	Links with INS status words
GNSS data	5,6	= 0,1 GPS data accepted	INS Algorithm status 1 & 2 GPS_VALID OR GPS2_VALID
		= 1,0 GPS data rejected	INS Algorithm status 1 GPS_REJECTED AND GPS2_REJECTED
Spare	7	= 0	NA

Note 4: When the system is in IDLE or NO GO state, all the bytes from 7 to 54 are set to 0x00. The NO GO state has priority over all the other states.

Note 5: During Coarse stationary alignment or Coarse GNSS alignment state, all the bytes from 49 to 54 are set to 0x00. Other fields are available, but with degraded accuracy. Full performance is only reached after fine alignment is completed.

Note 6: Logic of the MSB b_{31} of the INS Time field.

MSB b31	= 1 GNSS time not yet received In this case the time in this field is internal time = 0 GNSS time available In this case the time in this field is UTC time	INS System status 2 SYS2_UTC_DETECTED seen at least once. When b31 is set to 0, it will remain set to 0 even for GPS dropouts.

SEAPATH

Standard: Binary standard. **Binary format 11 protocol.**

Data sent: Time, Latitude, Longitude, Altitude, Heave, North Velocity, East Velocity, Down Velocity, Roll, Pitch, Heading, Roll Rate, Pitch Rate, Yaw Rate, Status.

Data frame: 18 fields - 42 bytes. All multi-byte data is sent MSB first.

Message <F0><F1><F2>…..<F17>			
Field 0	Byte 0	'q'	Synchronization byte
Field 1	Bytes 1 to 4	System time (Synchro with GNSS time)	32 bits integer (seconds)
Field 2	Byte 5	System time (Synchro with GNSS time) Fraction of seconds	Unsigned 8 bits integer 0.01 seconds (0 to 99)
Field 3	Bytes 6 to 9	Latitude '+': North of equator	Signed 32 bits integer +/-2^{31} = +/- 180°
Field 4	Bytes 10 to 13	Longitude '+': East of Greenwich	Signed 32 bits integer +/-2^{31} = +/- 180°
Field 5	Bytes 14 to 17	Altitude	Signed 32 bits integer (centimeters)
Field 6	Bytes 18 to 19	Heave "+" when down	Signed 16 bits integer (centimeters) Warning: Opposite sign of product usual convention
Field 7	Bytes 20 to 21	North Velocity	Signed 16 bits integer (centimeters/second)
Field 8	Bytes 22 to 23	East Velocity	Signed 16 bits integer (centimeters/second)
Field 9	Bytes 24 to 25	Down Velocity	Signed 16 bits integer (centimeters/second)
Field 10	Bytes 26 to 27	Roll Sign "+" when port side up	Signed 16 bits integer +/-2^{15} = +/- 180°
Field 11	Bytes 28 to 29	Pitch Sign "+" when bow up	Signed 16 bits integer +/-2^{15} = +/- 180° Warning: Opposite sign of product usual convention
Field 12	Bytes 30 to 31	Heading	Unsigned 16 bits integer 2^{15} = 180°
Field 13	Bytes 32 to 33	Roll rate (see **Note 1**) Positive when roll increases	Signed 16 bits integer +/-2^{15} = +/-180°/s
Field 14	Bytes 34 to 35	Pitch rate (see **Note 1**) Positive when pitch increases	Signed 16 bits integer +/-2^{15} = +/-180°/s Warning: Opposite sign of product usual convention
Field 15	Bytes 36 to 37	Heading rate (see **Note 1**) Positive when heading increases	Signed 16 bits integer +/-2^{15} = +/-180°/s
Field 16	Bytes 38 to 39	Status (see **Note 2**)	0x00AA : Invalid data 0x0000 : Data valid
Field 17	Bytes 40 to 41	Checksum (CRC) (see **Note 3**)	Computed on bytes 1 to 39

Note 1: on dual use products, rotation rate resolution is limited to 3.6 deg/h to comply with export regulation.

Note 2: data is flagged as invalid during coarse alignment (ALIGNMENT bit in INS algorithm status 1).

Note 3: CRC computation is given hereafter:

```
unsigned short blkcrc(unsigned char* bufptr, unsigned len)
{
unsigned char i;
unsigned short data;
unsigned short crc = 0xffff;
	if (len == 0)
		return ~crc;
	do
	{
		for (i = 0, data = (unsigned short)(0xff & *bufptr++); i < 8; i++, data >>= 1)
		{
			if ((crc & 0x0001) ^ (data & 0x0001))
			{
				crc = (crc >> 1) ^ 0x8408;
			}
			else
			{
				crc >>= 1;
			}
		}
	} while (--len);
	crc = ~crc;
	data = crc;
	crc = (crc << 8) | ((data >> 8) & 0xff);
	return crc;
}
```

SEATEX DHEAVE

Standard: Binary protocol. Binary PFreeHeave® Kongsberg Seatex proprietary protocol.
Data sent: Smart Heave™ and Time validity of data.
Data frame: The frame contains a header, 6 fields with 13 bytes in binary format. The signed integers are represented as two-complement numbers. For multi-byte elements, the MSB (Most Significant Byte) is transmitted first. The PFreeHeave® output is delayed by a few minutes due to the processing.

Message <F0><F1><F2>…..<F6>				
Field 0	Byte 0	0xAA	Unsigned	Header
Field 1	Byte 1	0x52	Unsigned	Header
Field 2	Bytes 2 to 5	Time (*)	Integer	In seconds
Field 3	Bytes 6 to 7	0 to 9999	Unsigned	Time, fraction of second: 0.0001 s
Field 4	Bytes 8 to 9	Smart Heave™	Integer	Delayed heave, in centimeters, positive down
Field 5	Byte 10	Status word	Bit-fields	0 : Heave is valid 1 : Heave is invalid
Field 6	Bytes 11 to 12	Checksum (**)	Unsigned	See **Note 1**

(*) The time fields contain time of validity of the data. The integer seconds part of time is counted from 1970-01-01 UTC time, ignoring leap seconds.

(**) Checksum is calculated as a 16-bit Block Cyclic Redundancy Check of all bites between, but not including the Header and Checksum fields. The CRC algorithm is describes in Note 1.

Note 1: Cyclic redundancy check algorithm
The 16-bit Block Cyclic Redundancy Check algorithm used to calculate the checksum in some formats is described in C and Fortran source code below.
C code:

```
#define POLY 0x8408
unsigned short blkcrc(
unsigned char *bufptr, /* message buffer */
unsigned long len /* number of bytes */
)
{
unsigned char i;
unsigned short data;
unsigned short crc = 0xffff;
if (len == 0L) {
return ~crc;
}
do {
for (i=0, data = (unsigned short) (0xff & *bufptr++);
i < 8;
```

```
i++, data >>= 1) {
if ((crc & 0x0001) ^ (data & 0x0001)) {
crc = (crc >> 1) ^ POLY;
} else {
crc >>= 1;
}
}
} while (--len);
crc = ~crc;
data = crc;
crc = (crc << 8) | ((data >> 8) & 0xff);
return crc;
}
```

SENSOR RD

Standard: Binary protocol. BLUEFIN proprietary protocol. It is linked to the input USBL-LBL-CTD protocol.

Data sentIMU outputs (on IMU products only), Transverse velocity, Longitudinal velocity, DVL Altitude, Vertical velocity, GNSS Latitude, GNSS Longitude, GNSS Altitude, USBL Latitude, USBL Longitude, USBL Altitude, LBL Latitude, LBL Longitude, LBL Altitude, LBL Beacon ID, LBL Range, UTC Time.

Data frame: 24 fields – 88 bytes.

Binary frame description

Message <F0><F1><F2>…..<F23>			
Field 0	Byte 0	0x24	Synchronization byte
Field 1	Byte 1	0xF0	Synchronization byte
Field 2	Bytes 2 to 5	XV1 delta rotation **(note 1)**	32 bits IEEE floating point format, rad (positive clockwise around bow direction)
Field 3	Bytes 6 to 9	XV2 delta rotation **(note 1)**	32 bits IEEE floating point format, rad (positive clockwise towards port side)
Field 4	Bytes 10 to 13	XV3 delta rotation (note 1)	32 bits IEEE floating point format, rad (positive clockwise towards up vector)
Field 5	Bytes 14 to 17	XV1 delta velocity **(note 1)**	32 bits IEEE floating point format, m/s (positive towards the bow)
Field 6	Bytes 18 to 21	XV2 delta velocity **(note 1)**	32 bits IEEE floating point format, m/s (positive towards port side)
Field 7	Bytes 22 to 25	XV3 delta velocity **(note 1)**	32 bits IEEE floating point format, m/s (positive towards up side)
Field 8	Bytes 26 to 29	Speed XV2	32 bits IEEE floating point format, m/s (positive towards port side)
Field 9	Bytes 30 to 33	Speed XV1	32 bits IEEE floating point format, m/s (positive towards the bow)
Field 10	Bytes 34 to 37	DVL Altitude	32 bits IEEE floating point format, meters
Field 11	Bytes 38 to 41	Speed XV3	32 bits IEEE floating point format, m/s (positive towards up side)
Field 12	Bytes 42 to 45	GNSS Latitude	Signed 32 bits integer $+/-2^{31}$ = +/-180° ; MSB are sent first
Field 13	Bytes 46 to 49	GNSS Longitude	Signed 32 bits integer $+/-2^{31}$ = +/-180° ; MSB are sent first
Field 14	Bytes 50 to 53	GNSS Altitude	32 bits IEEE floating point format, meters
Field 15	Bytes 54 to 57	USBL Latitude	Signed 32 bits integer $+/-2^{31}$ = +/-180° ; MSB are sent first
Field 16	Bytes 58 to 61	USBL Longitude	Signed 32 bits integer $+/-2^{31}$ = +/-180° ; MSB are sent first
Field 17	Bytes 62 to 65	USBL Altitude	32 bits IEEE floating point format, meters
Field 18	Bytes 66 to 69	LBL Latitude	Signed 32 bits integer

Message <F0><F1><F2>…..<F23>			
			+/-2^{31} = +/-180° ; MSB are sent first
Field 19	Bytes 70 to 73	LBL Longitude	Signed 32 bits integer +/-2^{31} = +/-180° ; MSB are sent first
Field 20	Bytes 74 to 77	LBL Altitude	32 bits IEEE floating point format, meters
Field 21	Bytes 78 to 79	LBL Beacon ID **(note 2)**	2 ASCII bytes
Field 22	Bytes 80 to 83	LBL Range	32 bits IEEE floating point format, meters
Field 23	Bytes 84 to 87	UTC Time	32 bit integer structured as follows: Bit 0 to bit 4 : 00000 Bit 5 to bit 14 : Milliseconds Bit 15 to bit 20 : Seconds Bit 21 to bit 26 : Minutes Bit 27 to bit 31 : Hours

Note 1: These fields contain thermally and mechanically compensated angular and velocity increments, corrected from coning and sculling (IMU outputs) and are only available on IMU product versions. They are forced to 0 on standard INS.

Note 2: The Beacon ID contains the last two digits of the beacon ID value received with the $BFLBL data frame from the USBL-LBL-CTD input protocol, from 0 to 99. If no USBL is received, the value is set to 0.

SIMRAD EM

Standard: Binary protocol. Output Simrad proprietary protocol.

Data sent: Status, Roll, Pitch, Heave and Heading.

Data frame: The frame contains 6 fields - 10 bytes. Except the heading, each sent data is two complemented coded. LSB are sent first.

Message <F0><F1><F2>…..<F5>			
Field 0	Byte 0	Sensor Status	0x90 if ok 0x9A if alignment
Field 1	Byte 1	Synchronization byte	0x90
Field 2	Bytes 2 to 3	Roll	+/-180° ; LSB = 0.01° Sign “+” when port up
Field 3	Bytes 4 to 5	Pitch	+/-180° ; LSB = 0.01° Sign “+” when bow up **Warning**: Opposite sign of INS usual convention.
Field 4	Bytes 6 to 7	Heave (*)	+/-10 m ; LSB = 0.01 m Sign +” when INS goes up
Field 5	Bytes 8 to 9	Heading	0° to 360° ; LSB = 0.01°

(*) The heave corresponds to the lever arm set on the output port.

SIMRAD EM HEAVE2

Standard: Binary protocol. Output Simrad proprietary protocol.

Data sent: Status, Roll, Pitch, Heave 2 and Heading.

Data frame: 6 fields - 10 bytes. Except the heading, each sent data is two complemented coded. LSB are sent first.

Message <F0><F1><F2>…..<F5>			
Field 0	Byte 0	Sensor Status	0x90 if ok 0x9A if alignment
Field 1	Byte 1	Synchronization byte	0x90
Field 2	Bytes 2 to 3	Roll (*)	+/-180° ; LSB = 0.01° Sign "+" when port up
Field 3	Bytes 4 to 5	Pitch (*)	+/-180° ; LSB = 0.01° Sign "+" when bow up **Warning**: Opposite sign of INS usual convention
Field 4	Bytes 6 to 7	Heave 2 (**)	+/-10 m ; LSB = 0.01 m Sign +" when INS goes up
Field 5	Bytes 8 to 9	Heading	0° to 360° ; LSB = 0.01°

(*) The attitude angles are computed with respect to TSS convention.
Roll and Pitch are referenced to the local vertical acceleration.
The formula calculation with respect to INS standard convention (Euler Angle or Tate Bryant) is given hereafter:

$$Roll_{TSS} = Sin^{-1} (Sin (Roll_{TB}) \times Cos (Pitch_{TB})) \text{ and } Pitch_{TSS} = Pitch_{TB}$$

(**) This outputs the "specific heave" and corresponds to the lever arm set on the output port.

SIMRAD EM TSS

Standard: Binary protocol. Output Simrad proprietary protocol.
Data sent: Status, Roll, Pitch, Heave and Heading.
Data frame: 6 fields - 10 bytes. LSB are sent first.

Message <F0><F1><F2>.....<F5>			
Field 0	Byte 0	Sensor Status	0x90 if ok 0x9A if alignment
Field 1	Byte 1	Synchronization byte	0x90
Field 2	Bytes 2 to 3	Roll*	+/-180° ; LSB = 0.01° Sign "+" when port up
Field 3	Bytes 4 to 5	Pitch (*)	+/-180° ; LSB = 0.01° Sign "+" when bow up **Warning**: Opposite sign of INS usual convention
Field 4	Bytes 6 to 7	Heave (**)	+/-10 m ; LSB = 0.01 m Sign +" when INS goes up
Field 5	Bytes 8 to 9	Heading	0° to 360° ; LSB = 0.01°

(*) The attitude angles are computed with respect to TSS convention.
Roll and Pitch are referenced to the local vertical acceleration.
The formula calculation with respect to INS standard convention (Euler Angle or Tate Bryant) is given hereafter:

$$Roll_{TSS} = Sin^{-1}\,(Sin\,(Roll_{TB}) \times Cos\,(Pitch_{TB})) \text{ and } Pitch_{TSS} = Pitch_{TB}$$

(**) The heave corresponds to the lever arm set on the output port.

SPERRY ATT

Standard: Binary protocol.

Data sent: Status, Heading, Attitude, Rotation rates.

Data frame: 11 fields - 18 bytes - MSB first.

Message <F0><F1><F2>.....<F10>			
Field 0	Byte 0	0xAA	Header TP1
Field 1	Byte 1	0x55	Header TP2
Field 2	Byte 2 to 3	Status**	See status specification table 1
Field 3	Bytes 4 to 5	Heading	0° to 360° ; LSB = 180°/ 2^{15}
Field 4	Bytes 6 to 7	Roll	± 90°; LSB = 90°/ 2^{15} Sign "+" for port down **Warning**: Opposite sign of INS usual convention
Field 5	Bytes 8 to 9	Pitch	± 90°; LSB = 90°/ 2^{15} Sign "+" when bow down
Field 6	Bytes 10 to 11	Heading rotation rate*	± 45°/s; LSB = 45°/ 2^{15}
Field 7	Bytes 12 to 13	Roll rotation rate*	± 45°/s; LSB = 45°/ 2^{15} **Warning**: Opposite sign of INS usual convention
Field 8	Bytes 14 to 15	Pitch rotation rate*	± 45°/s; LSB = 45°/ 2^{15}
Field 9	Byte 16	CSUM checksum	Negative sum of all the bytes from 0 to 15
Field 10	Byte 17	CSUMN checksum	CSUM checksum 1's complement

* The precision of rotation rate data is limited to 3.6 deg/h to comply with export regulation.

** Status specification table 1

Function	**Bits**	**Value**	**Links with INS status words**
Time Tag	0 to 9	Fractions of seconds: Unsigned 10 bits integer ~0.00125 seconds (0 to 799)	None
Not used	10	set to 0	None
Alignment	11	0 if alignment , otherwise 01	NOT(Bit 1) of the 32 LSB bits of the INS Algorithm status 1. Warning: Opposite of INS convention.
System error	12 to 15	0000 : no error XXX1: system warning XX1X: system alarm X1XX: system malfunction 1XXX: system fail or not ready	XXX1: OR of bits 11, 15, 19, 23, 27 of the INS Algorithm status 1 and bits 3,7, 27 of INS Algorithm status 2 and bits 7,11,15 of the INS Algorithm status 3. XX1X : OR of bits 1 to 5 and 17 to 21 of the INS System status 1 , bit 27 of the INS System status 2 bits 0,1,2,4,5,6, 8 of the INS Sensor status 2 X1XX: OR of bits 28, 29 of the INS Algorithm status 1 1XXX: bit 26 of INS User status

SPERRY ATT STAN

Standard: Binary protocol.

Data sent: Status, Heading, Attitude, Rotation rates.

Data frame: 11 fields - 18 bytes - MSB first.

Message <F0><F1><F2>…..<F10>			
Field 0	Byte 0	0xAA	Header TP1
Field 1	Byte 1	0x55	Header TP2
Field 2	Byte 2 to 3	Status**	See status specification table 1
Field 3	Bytes 4 to 5	Heading	0° to 360° ; LSB = 180°/ 2^{15}
Field 4	Bytes 6 to 7	Roll	± 90° ; LSB = 90°/ 2^{15} Sign "+" for port up (STANAG 4222 convention)
Field 5	Bytes 8 to 9	Pitch	± 90° ; LSB = 90°/ 2^{15} Sign "+" when bow up (STANAG 4222 convention) **Warning**: Opposite sign of INS usual convention
Field 6	Bytes 10 to 11	Heading rotation rate*	± 45°/s ; LSB = 45°/ 2^{15}
Field 7	Bytes 12 to 13	Roll rotation rate*	± 45°/s ; LSB = 45°/ 2^{15} **Warning**: Opposite sign of INS usual convention
Field 8	Bytes 14 to 15	Pitch rotation rate*	± 45°/s ; LSB = 45°/ 2^{15}
Field 9	Byte 16	CSUM checksum	Negative sum of all the bytes from 0 to 15
Field 10	Byte 17	CSUMN checksum	CSUM checksum 1's complement

* The precision of rotation rate data is limited to 3.6 deg/h to comply with export regulation.

** Status specification table 1

Function	**Bits**	**Value**	**Links with INS status words**
Time Tag	0 to 9	Fractions of seconds: Unsigned 10 bits integer ~0.00125 seconds (0 to 799)	None
Not used	10	set to 0	None
Alignment	11	0 if alignment , otherwise 01	NOT(Bit 1) of the 32 LSB bits of the INS Algorithm status 1. Warning: Opposite of INS convention.
System error	12 to 15	0000 : no error XXX1 : system warning XX1X : system alarm X1XX : system malfunction 1XXX : system fail or not ready	XXX1 : OR of bits 11, 15, 19, 23, 27 of the INS Algorithm status 1, and bits 3,7, 27 of INS Algorithm status 2, and bits 7,11,15 of the INS Algorithm status 3. XX1X : OR of bits 1 to 5 and 17 to 21 of the INS System status 1 , bit 27 of the INS System status 2 bits 0,1,2,4,5,6, 8 of the INS Sensor status 2 X1XX : OR of bits 28, 29 of the INS Algorithm status 1 1XXX : bit 26 of INS User status

STOLT OFFSHORE

Standard: ASCII. Output NMEA 0183 compatible with Seatex MRU system.
Data sent: Heading, Roll, Pitch, Roll speed, Pitch speed, Heading speed.
Data frame:

$PSXN,S,ddd,X1,X2,X3,X4,X5,X6,*hh<CR><LF> Note 3		
S	Status S = 10 for data valid S = 11 for data invalid	Note 1
ddd	User identification = 014	Three fixed ASCII characters
X1	Pitch in radian Warning: Opposite sign of Rovins usual convention	Note 2
X2	Roll in radian	Note 2
X3	Heading in radian	Note 2
X4	Pitch speed in radian/second, positive when PSXN pitch value increases Warning: Opposite sign of Rovins usual convention	Note 2
X5	Roll speed in radian/second, positive when roll increases	Note 2
X6	Heading speed in radian/second, positive when heading decreases	Note 2
hh	Checksum	Note 2

Note 1:
S = 11 when one of those INS User status bit is set to 1
FOG_ANOMALY
ACC_ANOMALY
TEMPERATURE_ERR
CPU_OVERLOAD
DYNAMIC_EXCEEDED
SPEED_SATURATION
ALTITUDE_SATURATION
ALIGNMENT
DEGRADED_MODE
FAILURE_MODE

Note 2: x1, x2, x3, x4, x5, x6 are written as floats in scientific format (for example –2.5648e01)
Note 3: STOLT OFFSHORE Telegram sample :
$PSXN,11,014,-4.000e-03,-1.350e-02,1.254e-01,0.000e+00,0.000e+00,0.000e+00,*0B<CR><LF>

STOLT OFFSHORE2

Standard: ASCII. Output NMEA 0183 compatible with Seatex MRU system.
Data sent: Heading, Roll, Pitch, Roll speed, Pitch speed, Heading speed.
Data frame:

$PSXN,S,dd,X1,X2,X3,X4,X5,X6*hh<CR><LF		
S	Status S = 10 when no error occurs S = 11 when one or more error occurs	Note 1
dd	User identification = 14	Two fixed ASCII characters
X1	Pitch in radian Warning: Opposite sign of Rovins usual convention	Note 2
X2	Roll in radian	Note 2
X3	Heading in radian	Note 2
X4	Pitch speed in radian/second, positive when PSXN pitch value increases Warning: Opposite sign of Rovins usual convention	Note 2
X5	Roll speed in radian/second, positive when roll increases	Note 2
X6	Heading speed in radian/second, positive when heading decreases	Note 2
hh	Checksum	

Note 1:
S = 11 when one of those INS User status bit is set to 1
FOG_ANOMALY
ACC_ANOMALY
TEMPERATURE_ERR
CPU_OVERLOAD
DYNAMIC_EXCEEDED
SPEED_SATURATION
ALTITUDE_SATURATION
ALIGNMENT
DEGRADED_MODE
FAILURE_MODE
Note 2: x1, x2, x3, x4, x5, x6 are written as floats in scientific format (for example –2.5648e01)
Note 3: STOLT OFFSHORE 2 Telegram sample :
$PSXN,11,14,-4.000e-03,-1.350e-02,1.254e-01,0.000e+00,0.000e+00,0.000e+00*17<CR><LF>

TECHSAS

Standard: Output NMEA 0183 compatible.

Data sent: Heading, Roll, Pitch, Heave, standard deviations for Heading, Roll and Pitch, Status flags

Data frame:

$PASHR,hhmmss.sss,H.HH,T,aR.RR,bP.PP,cD.DD,r.rrr,p.ppp,h.hhh,x,y*hh<CR><LF>		
hhmmss.sss	is the UTC of the data	
H.HH	is the heading in degrees, no sign character	2 digits after decimal point
T	is a fixed character = ‘T’	
aR.RR	is the roll in degrees and a, its sign character ‘+’ when port up, ‘–‘ when port down	2 digits after decimal point
bP.PP	is the pitch in degrees and b, its sign character ‘–‘ when bow down, ‘+’ when bow up Warning: Opposite sign of INS usual convention	2 digits after decimal point
cD.DD	is the heave in meters and c, its sign character ‘–’ when INS goes up ‘+’ when INS goes down Warning: Opposite sign of INS usual convention	
r.rrr	is the roll standard deviation	3 digits after decimal point
p.ppp	is the pitch standard deviation	3 digits after decimal point
h.hhh	is the heading standard deviation	3 digits after decimal point
x	is the GNSS aiding status flag 1 when GNSS received and valid, otherwise 0	x = 1 when the following INS User status bit is set to 1 GPS_RECEIVED_VALID otherwise x = 0
y	is the sensor error status flag 1 when ACC or FOG error, otherwise 0	y = 1 when one of those INS User status bit is set to 1 FOG_ ANOMALY ACC_ANOMALY otherwise y = 0
hh	is the checksum	

TECHSAS TSS

Standard: Output NMEA 0183 compatible.

Data sent: Heading, Roll, Pitch, Heave, standard deviations for Heading, Roll and Pitch, Status flags

Data frame:

$PASHR,hhmmss.sss,H.HH,T,aR.RR,bP.PP,cD.DD,r.rrr,p.ppp,h.hhh,x,y*hh<CR><LF>

hhmmss.sss	is the UTC of the data	
H.HH	is the heading in degrees	Float, 2 digits after decimal point
T	is a fixed character = 'T'	
aR.RR	is the roll in degrees and a, its sign character (*) '+' when port up, '–' when port down	Float, 2 digits after decimal point
bP.PP	is the pitch in degrees and b, its sign character (*) '–' when bow down, '+' when bow up Warning: Opposite sign of INS usual convention	Float, 2 digits after decimal point
cD.DD	is the heave in meters and c, its sign character '–' when INS goes up, '+' when INS goes down Warning: Opposite sign of INS usual convention	Float, 2 digits after decimal point
r.rrr	is the roll standard deviation	Float, 3 digits after decimal point
p.ppp	is the pitch standard deviation	Float, 3 digits after decimal point
h.hhh	is the heading standard deviation	Float, 3 digits after decimal point
x	is the GNSS aiding status flag 1 when GNSS received and valid, otherwise 0	
y	is the sensor error status flag 1 when ACC or FOG error, otherwise 0	
hh	is the checksum	

* The attitude angles are computed with respect to TSS convention. Roll and Pitch are referenced to the local vertical acceleration. The formula calculation with respect to INS standard convention (Euler Angle or Tate Bryant) is given hereafter:

$$Roll_{TSS} = Sin\text{-}1\ (Sin\ (Roll_{TB})\ x\ Cos\ (Pitch_{TB}))\ \text{and}\ Pitch_{TSS} = Pitch_{TB}$$

TMS CCV IMBAT

Standard: Binary protocol. Output Thomson Marconi Sonar proprietary protocol.
Data sent: Roll, Pitch, Heave, Heading, Linear accelerations, Rotations rates, Status.
Data frame: The frame contains 13 fields - 24 bytes. LSB are sent first.

Message <F0><F1>…<F12>			
Field 0	Bytes 0 to 1	Header	Fixed value = 0x0090
Field 1	Bytes 2 to 3	Roll	+/-180° ; LSB = 180° / 2^{15} Sign "+" when port up
Field 2	Bytes 4 to 5	Pitch	+/-180° ; LSB = 180° / 2^{15} Sign "+" when bow up **Warning**: Opposite sign of INS usual convention.
Field 3	Bytes 6 to 7	Heave	+/-327 m ; LSB = 327m / 2^{15} Sign "+" when INS goes down **Warning**: Opposite sign of INS usual convention.
Field 4	Bytes 8 to 9	Heading	0° to 360° ; LSB = 360° / 2^{16}
Field 5	Bytes 10 to 11	XV1 acceleration*	+/-0.5 g; LSB = 0.5g / 2^{15}
Field 6	Bytes 12 to 13	XV2 acceleration*	+/-0.5 g; LSB = 0.5g / 2^{15}
Field 7	Bytes 14 to 15	XV3 acceleration*	+/-0.5 g; LSB = 0.5g / 2^{15}
Field 8	Bytes 16 to 17	XV1 rotation rate*	+/-20°/s; LSB = 20°/s / 2^{15}
Field 9	Bytes 18 to 19	-XV2 rotation rate*	+/-20°/s; LSB = 20°/s / 2^{15} **Warning**: Opposite sign of INS usual convention.
Field 10	Bytes 20 to 21	-XV3 rotation rate*	+/-20°/s; LSB = 20°/s / 2^{15} **Warning**: Opposite sign of INS usual convention.
Field 11	Byte 22	Status	0xFF if Ok 0xAA if Alignment 0x00 if Error
Field 12	Byte 23	End of sentence	Fixed value = 0x91

*To comply with export regulation, the precision of rotation rate data is limited to 3.6 deg/h and the precision of acceleration data is limited to 1 mg. Accelerations are compensated from g.

TOKIMEC_PTVF

Standard: Output NMEA 0183 compatible.

Data frame:

The number of characters in the string (including carriage return line feed) is always 75.

$PTVF,abbbbP,cddddR,eee.eT,fggg.gPR,hiii.iRR,jkkk.kAR,lmm.mN,yyyMD,000sAL*nn<CR><LF>		
abbbb	is the Pitch in degrees bb(deg)bb(min) sign character a : [-] bow up / [space] bow down	
cdddd	is the Roll in degrees dd(deg)dd(min) sign character c : [-] port up / [space] port down	Warning: Opposite sign of INS usual convention
eee.e	is the Heading in degrees no sign character	[0 to 359.9] 0° at North and increasing when bow turns starboard.
fggg.g	is the Pitch Rate in degrees/sec sign character f : [-] bow up / [space] bow down Note 1	Sign of the INS pitch derivative
hiii.i	is the Roll Rate in degrees/sec sign character h : [-] port up / [space] port down Note 1	Sign of the INS roll derivative Warning: Opposite sign of INS usual convention
jkkk.k	is the Heading Rate in degrees/sec sign character j : [-] CCW / [space] CW Note 1	Sign of the INS heading derivative
lmm.m	is the vessel speed in Knots sign character l : [-] is astern / [space] ahead	
yyy	NOT USED.	3 digits fixed to 000
s	Hexadecimal value of the Status flag (one digit) Note 2	
hh	checksum of all in string but $ and * characters	NMEA Checksum

Note 1: These data are attitude rates and are given in the INS reference frame (X_1, X_2, X_3).

Note 2: Status flag

Meaning	Value	Links with INS status words
IDLE	= 0	INS System status 2 WAIT_FOR_POSITION
Coarse stationary alignment	= 1	INS User status & INS System status 2 ALIGNMENT **AND NOT** WAIT_FOR_POSITION AND NOT Alignment mode (Note 3)
Fine stationary alignment Not completed	= 2	INS User status FINE_ALIGNMENT **AND NOT** Alignment mode (Note 3)
Fine stationary alignment Completed	= 3	INS User status NOT (ALIGNMENT OR FINE_ALIGNEMENT) **AND NOT** Alignment mode (Note 3)
Coarse GNSS alignment	= 4	INS User status & INS System status 2 ALIGNMENT **AND NOT** WAIT_FOR_POSITION **AND** Alignment mode (Note 3)
Fine GNSS alignment Not completed	= 5	INS User status FINE_ALIGNMENT **AND** Alignment mode (Note 3)
Fine GNSS alignment Completed	= 6	INS User status NOT (ALIGNMENT OR FINE_ALIGNEMENT) **AND** Alignment mode (Note 3)
Aided navigation	= 9 Not used	NA
System failure (This state has priority over all the other states)	= A	INS User status CPU_OVERLOAD **OR** TEMPERATURE_ERR **OR** INPUT_x_ERR **OR** OUTPUT_x_ERR **OR** FAILURE_MODE

Note 3: Logic of the Alignment mode flag

Alignment mode	= 0 without position = 1 with position	INS System status 2 GPS_DETECTED **OR** GPS2_DETECTED seen at least once during coarse alignment with a rejection mode set to Always True or to Automatic Reacquisition. When this alignment mode flag is set to one, it will remain set to one for GNSS dropouts.

TSS1 DMS

Standard: Output TSS proprietary protocol

Data sent: Roll, Pitch, Heave, Linear accelerations, Status Accelerations are compensated from earth gravity. All values are saturated to maximum value.

Data frame: 27 bytes in ASCII format.

:XXAAAA<SP>MHHHHQMRRRR<SP>MPPPP<CR><LF>		
:	Header character	0x3A
XX	Horizontal acceleration (in plane XV1, XV2)	Acc: 0 to 9.81 m.s-2 Unit: 3.83 cm.s-2 (ASCII representation of a 8 bits unsigned integer hexadecimal value)
AAAA	XV3 acceleration	Acc: ± 20.48 m.s-2 Unit: 0.0625 cm.s-2 Sign "+" when system goes up (ASCII representation of a 16 bits signed 2 complement integer hexadecimal value)
<SP>	Space character	0x20
MHHHH	Heave	Heave: ± 99 m Unit: 1 cm M is the space character when system goes up M is the minus character when system goes down
Q	Status character	'h' for alignment mode 'H' for nominal mode
MRRRR	Roll	Roll ± 90° Unit : 0.01° M is the space character when port up M is the minus character when port down
<SP>	Space character	0x20
MPPPP	Pitch	Pitch ± 90° Unit : 0.01° Sign "+" when bow up M is the space character when bow up M is the minus character when bow down Warning: Opposite sign of usual convention
<CR><LF>	End of frame	0x0D 0x0A

TSS335B

Standard: Output TSS proprietary protocol
Data sent: Roll, Pitch, Heave, Linear accelerations, Status Accelerations are compensated from earth gravity. All values are saturated to maximum value.
Data frame: 27 bytes in ASCII format.

:XXAAAA<SP>MHHHHQMRRRR<SP>MPPPP<CR><LF>		
:	Header character	0x3A
XX	Horizontal acceleration (in plane XV1, XV2)	Acc: 0 to 9.81 $m.s^{-2}$ Unit: 3.83 $cm.s^{-2}$ (ASCII representation of a 8 bits unsigned integer hexadecimal value)
AAAA	XV3 acceleration	Acc: ± 20.48 $m.s^{-2}$ Unit: 0.0625 $cm.s^{-2}$ Sign "+" when system goes up (ASCII representation of a 16 bits signed 2 complement integer hexadecimal value)
<SP>	Space character	0x20
MHHHH	Heave	Heave: ± 99 m Unit: 1 cm M is the space character when system goes up M is the minus character when system goes down
Q	Status character	'?' for alignment mode space character for nominal mode
MRRRR	Roll	Roll ± 90° Unit : 0.01° M is the space character when port up M is the minus character when port down
<SP>	Space character	0x20
MPPPP	Pitch	Pitch ± 90° Unit : 0.01° Sign "+" when bow up M is the space character when bow up M is the minus character when bow down Warning: Opposite sign of usual convention
<CR><LF>	End of frame	0x0D 0x0A

VTG GGA

Standard: Output NMEA 0183 compatible.

Data sent: This protocol outputs INS computed position values in an ASCII frame. Some characters of this output frame are set to fixed values.

Data frame:

- NMEA $--PHGGA Frame, see section 4.2.2
- NMEA $--PHVTG Frame, see section 4.2.11.

VTG GGU

Standard: Output NMEA 0183 compatible.

Data sent: This protocol outputs INS UTM WGS84 computed position and speed values in an ASCII frame.

Data frame:

- NMEA $--PHVTG Frame, see section 4.2.11.

$PHGGU,xxxxxxxx.x,a,yyyyyyyy.y,b,hhmmss.ss,*hh<CR><LF>		
xxxxxxxx.x	is the UTM WGS84 easting coordinate	1 digit after decimal point
a	is a fixed character 'E'	
yyyyyyyy.y	is the UTM WGS84 northing coordinate	1 digit after decimal point
b	is a fixed character 'N'	
hhmmss.ss	is the UTC of the position	
hh	is the checksum	

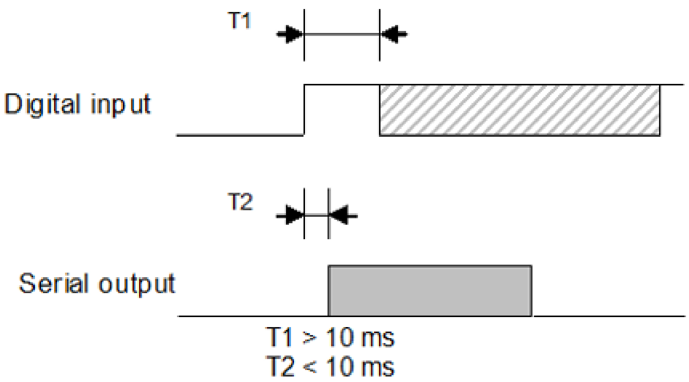
7 Pulses Interfaces Specification

7.1 Input Pulses Specification

7.1.1 Input Pulses Functional Specification

Four input pulses are available in the system, but only part of them may be present on external connectors. Following protocols can be configured:

- **PPS rising/falling + Time**: In this configuration, the system waits for an input PPS pulse and associated UTC time should be provided in following time frame. UTC time validity corresponds to rising/falling edge of PPS pulse.
- **Time + PPS rising/falling**:In this configuration, the system waits for an input PPS pulse and associated UTC time should be provided in following time frame. UTC time validity corresponds to rising/falling edge of PPS pulse.
- **Event marker rising/falling**: In this configuration, the input pulse will be used to time stamp external event. Associated counts will be logged in EVENT MARKER protocol (see protocol description for details).
- **Event marker rising/falling PP**: In this configuration, the input pulse will be used to time stamp external event. Associated counts will belogged in POST PROCESSING protocol (see protocol description for details).
- **Synchro out X**: In this configuration, input pulse is used to trigger protocol output on associated port X during next available output slot (a slot is available each 5 ms).



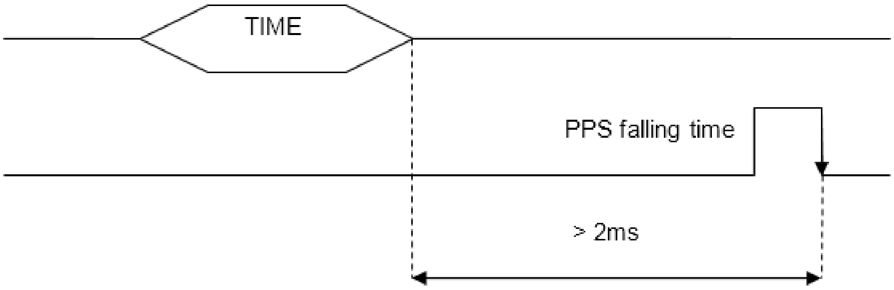
The table below details pulses and protocols available:

Table 24 - Pulses and Protocols available

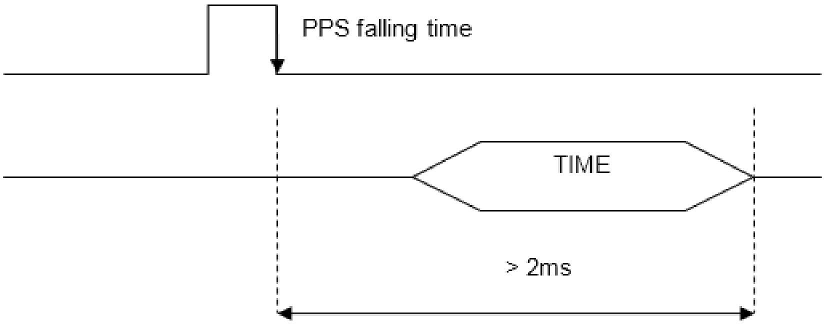
Pulse A	Pulse B	Pulse C
PPS rising/falling + Time Time + PPS rising/falling Synchro out X Event marker rising/falling Event marker rising/falling PP	Idem Pulse A	Idem Pulse A

7.1.2 Time/PPS Input Constraints

When configured in Time + PPS, the Time frame content must be fully received at least 2 ms before the PPS signal to be correctly taken into account:



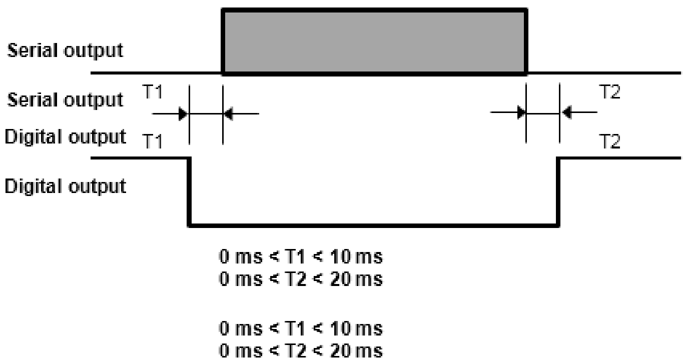
When configured in PPS + Time, the PPS must be sent at least 2ms before the Time frame is entirely received to be correctly taken into account:



7.2 Output Pulses Functional Specification

Output pulses are available in the system ; only part of them may be present on external connectors. For each output pulse, the following protocols can be configured:

- **Serial out X RTC:** In this configuration, the output pulse will be active during selected port output (the pulse will reflect the envelope of selected serial output). Pulse width thus depends on serial port baudrate and frame length.



- **PPS Like**: in this configuration, the pulse will be triggered once per second, at the moment when the ZDA frame is sent by the system in the GPS LIKE protocol. This pulse can be used to synchronize the system with another equipment in case no PPS is available from an external UTC reference.

- **Distance traveled** rising/falling: In this configuration, the pulse will be output each time the traveled distance increases by specified step set in pulse out scale factor. For example, if step is 1, the pulse will be output each meter. Pulse width is 5ms in this mode.
- **Timer rising/falling**: in this configuration, the pulse will be output at specified period in seconds. The time resolution for the output period is derived from the 200Hz base is 5ms, so the output period must be a multiple of 5ms.
- **System failure**: In this configuration associated pulse will be set to Failure State when at least one of the following user status bit is set:
  - Bit 11: TEMPERATURE ERROR
  - Bit 13: DYNAMIC EXCEEDED
  - Bit 14: SPEED SATURATION
  - Bit 15: ALTITUDE SATURATION
  - Bit 26: HRP INVALID
  - Bit 27: ALIGNMENT
  - Bit 31: FAILURE MODE

  In all other cases, the pulse is set to Idle State.

  When this pulse is set, system outputs should not be trusted and must be considered invalid for external equipments

The table below details pulses and protocols available

Table 25 - Pulses and Protocols available

Pulse A	Pulse B	Failure pulse
Serial out X RTC PPS Like Distance Traveled Timer	Serial out X RTC PPS Like Distance Traveled Timer	System Failure

8 Control Command Description

This section describes the Rovins configuration and monitoring commands which can be used during operation. These commands are sent directly through the repeater port to check settings and modify Rovins or external sensors configuration, to reset the unit, and save the configuration to PROM.
However, commands described below are exactly equivalent to the ones which are sent by the Web-Based Graphical User Interface. They may be useful when launching and using the Web-Based Graphical User Interface during operation is not possible.

When Rovins is configured during initial installation, the Web-Based Graphical User Interface is not mandatory to operate the INS.
Customer can control/monitor Rovins operation using control commands through the repeater port over serial or on Ethernet port 8110.

8.1 General Conventions

8.1.1 Command Syntax

All frames are compatible with NMEA 0183 standard and are formatted as follows:
$PIXSE,CONFIG,NAME[,xxx,…,..y]*hh<CR><LF> for generic configuration commands
$PIXSE,TEXT__,NAME [,xxx,…,..y]*hh<CR><LF> for generic configuration text retrieve commands
$PHCNF,NAME[,xxx,…,..y]*hh<CR><LF> for Rovinsspecific configuration commands
$PHTXT,NAME [,xxx,…,..y]*hh<CR><LF> for Rovins specific configuration text retrieve commands

$ is a header and "NAME" depends on the command.
Brackets **[ ]** indicates optional parameters, depending on the command.

Most commands can be used either to send configuration parameters "xxx,…,..y" to the system, or to ask for the current value for the parameter.
In such case, the ",xxx,…,..y" should be replaced by ",,". "hh" is the checksum of the sentence, and allows for a control during the transmission.
It is calculated by exclusive-OR'ing (XOR) the 8 bits (no start bits or stop bits) of each character in the sentence, excluding "$" and "*".

The hexadecimal values of the most significant and least significant 4 bits of the result are converted to two ASCII (0-9, A-F) for transmission.
The most significant character is transmitted first.
The checksum field is required in all transmitted sentences.
All frames should be ended by the two characters <CR><LF> (0D 0A hexadecimal).
A web based NMEA calculator is provided next page to help computing checksums.

8.1.2 How to Send and Receive Commands

The Rovins will listen for COMMAND protocol on repeater flow, which is available on repeater connector, digital connector and Ethernet.

Repeater serial configuration is fixed and set to 57 600 bauds, odd parity bit, 2 stop bits.
Repeater Ethernet configuration is fixed to TCP IP port 8110 in server mode.

If a command is not correctly formatted, Rovins will ignore it and no answer will be sent. If a parameter is out of allowed range, it will be ignored by Rovins and set to default value (0 or none). Thus, to confirm that a command was correctly handled by Rovins, the control application should systematically send the command to change a parameter, and then send the read back command to check stored value. After all changes are made, the application can send a SAVE command to store parameters in non-volatile memory inside Rovins.

8.1.3 NMEA Checksum Web Page

To get a simple NMEA computation tool, copy following code into a new file (nmea.html) and save it, and open this file with your internet navigator:

```
<html><head><title>NMEA MTK checksum calculator</title>
<script><!--
function updateChecksum(cmd)
{
var checksum = 0;
for(var i = 0; i < cmd.length; i++) checksum = checksum ^
cmd.charCodeAt(i);
var hexsum = Number(checksum).toString(16).toUpperCase();
if (hexsum.length < 2) hexsum = ("00" + hexsum).slice(-2);
settext(document.getElementById("output"), "$" + cmd + "*" +
hexsum);
}
function settext(span, text)
{
if (!span.hasChildNodes()) {
span.appendChild(span.ownerDocument.createTextNode(text));
return;
} else span.firstChild.nodeValue = text;
}
--></script></head><body>
<h1>MTK NMEA checksum calculator</h1>
<p>This is a simple calculator to compute the checksum field of
NMEA frames.</p>
<p> The checksum is simple, just an XOR of all the bytes between
the <tt>$</tt> and the <tt>*</tt> (not including the delimiters
themselves), and written in hexadecimal.</p>
<p>For this to work you'll need to be using a browser that supports
JavaScript and DHTML
(most modern browsers do).</p>
<div style="margin:1em; padding: 2em; background: #ddddff;">
<form onsubmit="document.getElementById('commandfld').select();
return false;">
<table>
```

```
<tr><th align=right>Command:</th><td><tt>$<input id="commandfld"
size=80 type="text" onchange="updateChecksum(this.value);"
value="PIXSE,CONFIG,WAKEUP">*</tt></td></tr>
<tr><th align=right>With checksum:</th><td><span id="output"
style="font-family: monospace;"></span></td></tr>
</table></form></div>
<script>updateChecksum(document.getElementById
("commandfld").value);</script>
<hr></body></html>
```

8.2 General System Configuration

8.2.1 Communication Mode

To start the communication with Rovins in User Mode:

Message	**$PIXSE,CONFIG,WAKEUP*40<CR><LF>**
Title	Starting Communication with the Rovins in User Mode

Commands can be sent directly without having to call WAKEUP before.

8.2.2 Save to EEPROM

Save all parameters into EEPROM:

Message	**$PIXSE,CONFIG,SAVE__*5C<CR><LF>**
Title	Save Configuration to EEPROM

Please note the 2 underscores ("_") characters at the end of this frame.

8.2.3 Software System Reboot

To reboot the system by software, following command can be used:

Message	**$PIXSE,CONFIG,RESET_*57<CR><LF>**
Title	Restart Rovins algorithm

Rovins will restart with the configuration saved into PROM. It is recommended to perform a save to PROM command (see section 8.2.2) before resetting if you changed the settings and want to keep them.

8.2.4 Reset to factory DSP parameters

To reset DSP parameters to factory defaults, following command can be used:

Message	**$PIXSE,CONFIG,RSTDSP*4F<CR><LF>**
Title	Reset DSP parameters to factory defaults

This command resets only DSP parameters (lever arms, system orientation, algorithm modes, initial position, etc.). Default parameters will only apply at next reboot. To completely reset system settings, use RSTMPC command as well.

8.2.5 Reset to factory MPC parameters

To reset MPC parameters to factory defaults, following command can be used:

Message	**$PIXSE,CONFIG,RSTMPC*56<CR><LF>**
Title	Reset MPC parameters to factory defaults

This command resets only MPC parameters (input and output protocols, sensor interfaces, etc.). Default parameters will apply only after next reboot. To completely reset system settings, use RSTDSP command as well.

8.2.6 Web-Based Graphical User Interface Passwords Reset

To reset Web-Based Graphical User Interface passwords, following command can be used:

Message	**$PIXSE,CONFIG,PWDRST*4B<CR><LF>**
Title	Reset Web-Based Graphical User Interface Passwords

8.2.7 System Errors Log Reset

To reset System Error Log, following command can be used:

Message	**$PIXSE,CONFIG,ERRRST*4D<CR><LF>**
Title	Reset System Errors log

8.2.8 Starting Position

To enter the starting position used in static alignment process:

Message	**$PIXSE,CONFIG,STPOS,x.x,y.y,z.z*hh<CR><LF>**		
Title	Manual Position		
Data Field	**Semantics**	**Unit**	**Type**
x.x	Latitude, positive north	Degree	float
y.y	Longitude, positive east	Degree	float
z.z	Altitude, positive up	m	float

To retrieve current starting position:

$PIXSE,CONFIG,STPOS,,*16<CR><LF>

8.2.9 Starting UTM Position

To enter the starting UTM position used in static alignment process:

Message	**$PIXSE,CONFIG,STUTM,c,i,x.x,y.y,z.z*hh<CR><LF>**		
Title	Manual UTM Position		
Data Field	**Semantics**	**Unit**	**Type**
c	Northing area letter	N/A	char
i	Easting area index	N/A	int
x.x	Easting offset in area	m	float
y.y	Northing offset in area	m	float
z.z	Altitude	m	float

To retrieve current UTM starting position:

$PIXSE,CONFIG,STUTM,,*16<CR><LF>

8.2.10 Initial Position

To enter initial position used in static alignment process:

Message	**$PIXSE,CONFIG,MANPOS,x.x,y.y,z.z*hh<CR><LF>**		
Title	Manual Position		
Data Field	**Semantics**	**Unit**	**Type**
x.x	Latitude, positive north	Degree	float
y.y	Longitude, positive east	Degree	float
z.z	Altitude, positive up	m	float

To retrieve current initial position: **$PIXSE,CONFIG,MANPOS,,*53<CR><LF>**

8.2.11 Manual Attitude

To enter manual attitude for used in denied GNSS:

Message	$PIXSE,CONFIG,MANATT,a.a,b.b*hh<CR><LF>		
Title	Manual Position		
Data Field	Semantics	Unit	Type
a.a	heading	degree	float
b.b	heading standard deviation	degree	float

To retrieve manual attitude:

$PIXSE,CONFIG,MANATT,,*53<CR><LF>

8.2.12 Manual UTM Position

To enter initial UTM position used in static alignment process:

Message	**$PIXSE,CONFIG,UTMWGS,c,i,x.x,y.y,z.z*hh<CR><LF>**		
Title	Manual UTM Position		
Data Field	**Semantics**	**Unit**	**Type**
c	Northing area letter	**N/A**	char
i	Easting area index	**N/A**	int
x.x	Easting offset in area	**m**	float
y.y	Northing offset in area	**m**	float
z.z	Altitude	**m**	float

To retrieve current UTM initial position:

$PIXSE,CONFIG,UTMWGS,,*52<CR><LF>

8.2.13 Heading, roll and pitch fine misalignments

To configure user attitude biases (fine misalignment with respect to Subsea vehicle frame XV1, XV2, XV3):

Message	**$PIXSE,CONFIG,BIAS__,x.x,y.y,z.z *hh<CR><LF>**		
Title	Bias Configuration		
Data Field	**Semantics**	**Unit**	**Type**
x.x	Heading misalignment	Degree	float
y.y	Roll misalignment	Degree	float
z.z	Pitch misalignment	Degree	float

To retrieve the biases:

$PIXSE,CONFIG,BIAS__,,*44<CR><LF

8.2.14 Axis Orientation

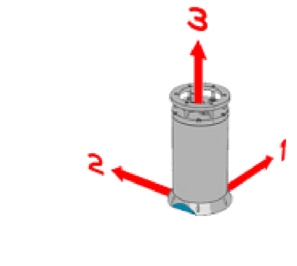
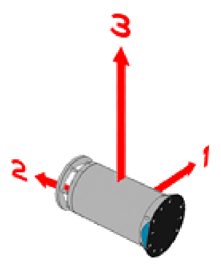
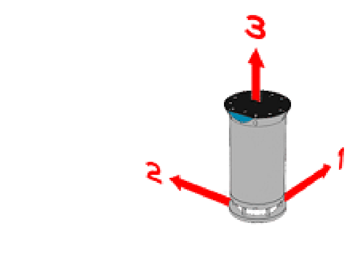
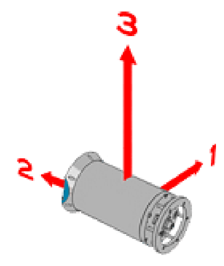
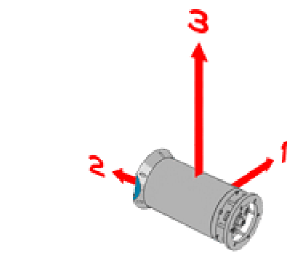
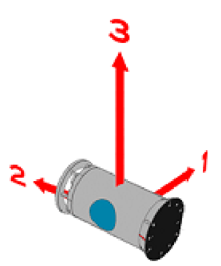
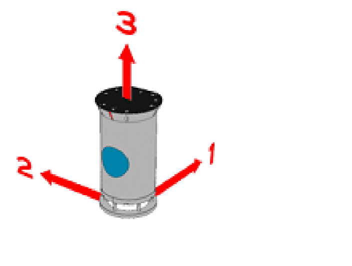
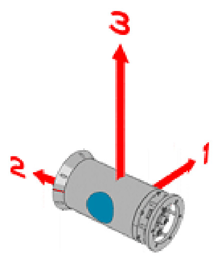
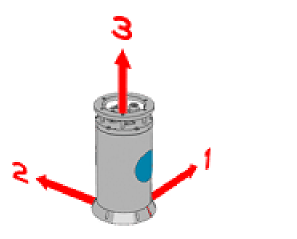
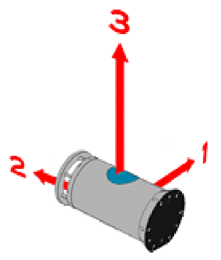
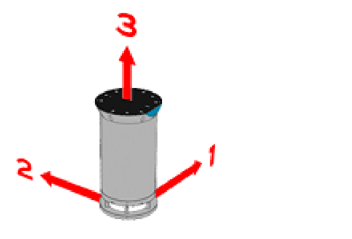
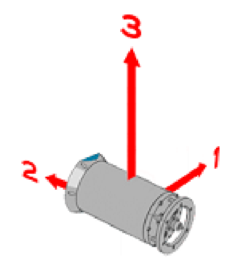
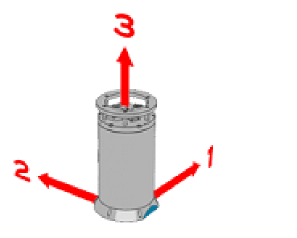
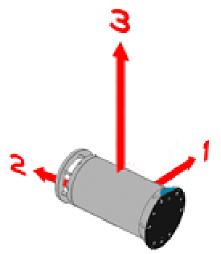
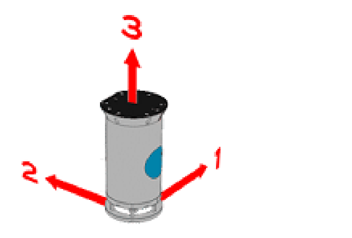
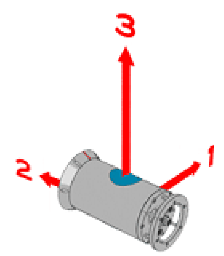
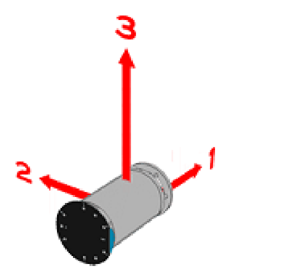
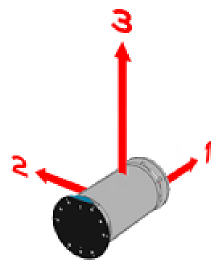
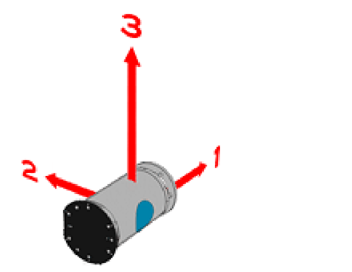
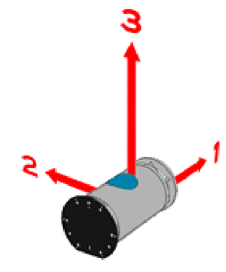
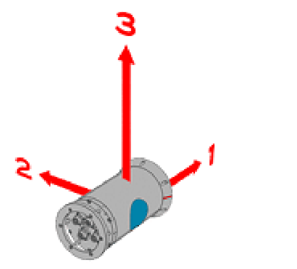
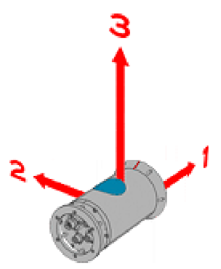
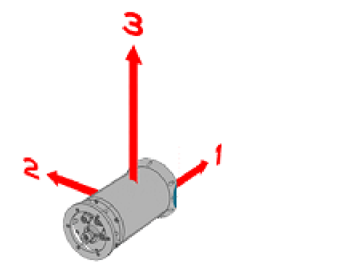
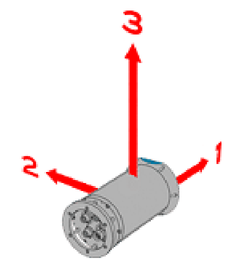
To enter an axis orientation (rough misalignment):

Message	**$PIXSE,CONFIG,AXISOR,i*hh<CR><LF>**		
Title	Axis Orientation		
Data Field	**Semantics**	**Syntax**	**Type**
i	Index of the rough misalignment	**See table next page for index / orientation correspondence and checksum**	int

To retrieve the axis orientation, also called the rough misalignment:

$PIXSE,CONFIG,AXISOR,,*43<CR><LF

Table 26 - Matching between index i and INS axis orientation with associated checksum (hh)

index	orientation	index	orientation	index	orientation	index	orientation
0 **(5F)**		1 (5E)		2 (5D)		**3** **(5C)**	
4 **(5B)**		5 (5A)		6 (59)		**7** **(58)**	
8 **(57)**		9 (56)		10 (6E)		**11** **(6F)**	
12 **(6C)**		13 (6D)		14 (6A)		**15** **(6B)**	
16 **(68)**		17 (69)		18 (66)		**19** **(67)**	
20 **(6D)**		21 (6C)		22 (6F)		**23** **(6E)**	

8.2.15 Main Lever Arms

To configure the main monitoring point lever arms from the Rovins:

Message	$PIXSE,CONFIG,LEVARM,x.x,y.y,z.z*hh<CR><LF>		
Title	Lever Arm Configuration		
Data Field	**Semantics**	**Unit**	**Type**
x.x	X_{V1} lever arm	**m**	Float
y.y	X_{V2} lever arm	**m**	Float
z.z	X_{V3} lever arm	**m**	Float

To retrieve main monitoring point lever arms from the Rovins:

$PIXSE,CONFIG,LEVARM,,*5C<CR><LF>

8.2.16 Secondary Lever Arms

To configure secondary monitoring point lever arms from the Rovins:

Message	$PIXSE,CONFIG,SECLVX,x.x,y.y,z.z*hh<CR><LF>		
Title	Lever Arm Configuration		
Data Field	**Semantics**	**Unit**	**Type**
X	Lever arm (‘A’, ‘B’ or ‘C’)	**N/A**	Char
x.x	X_{V1} lever arm	**m**	Float
y.y	X_{V2} lever arm	**m**	Float
z.z	X_{V3} lever arm	**m**	Float

To retrieve secondary monitoring point lever arms from the Rovins:

$PIXSE,CONFIG,SECLVA,,*53<CR><LF>
$PIXSE,CONFIG,SECLVB,,*50<CR><LF>
$PIXSE,CONFIG,SECLVC,,*51<CR><LF>

8.2.17 Center of Gravity Position

To configure center of gravity position relative to Rovins:

Message	$PIXSE,CONFIG,COG___,x.x,y.y,z.z*hh<CR><LF>		
Title	Lever Arms Configuration		
Data Field	Semantics	Unit	**Type**
x.x	X_{V1} cog	m	Float
y.y	X_{V2} cog	m	Float
z.z	X_{V3} cog	m	Float

Please note the 3 underscore(''_'') characters after COG header.

To retrieve center of gravity from the INS:

$PIXSE,CONFIG,COG___,,*49<CR><LF>

8.2.18 ZERO VELOCITY UPDATE

Enabling a ZUPT mode is equivalent to sending a speed sensor forced to 0m/s in the INS. Following modes are available:

- Static 10m/s : this mode sends a 0m/s speed input with 10m/s standard deviation
- Static 0.1m/s: this mode sends a 0m/s speed input with 0.1m/s standard deviation
- Autostatic 0.01m/s: this mode detects system movements and when no rotation larger than 10°/h is detected, it enables a null speed entry with 0.01m/s standard deviation.
- Autostatic bench: this mode detects movements and when no rotation larger than 10°/h is detected, it enables a null speed entry with 0.01m/s standard deviation and a null rotation of 10°/h standard deviation.
- Fixed Position: in this mode, the INS is considered static, and this information is used by the Kalman filter to estimate its parameters.

To configure the Zero Velocity Update Mode:

Message	**$PIXSE,CONFIG,ZUP___,i*hh<CR><LF>**		
Title	Zero Velocity Update Mode		
Data Field	**Semantics**	**Syntax**	**Type**
i	Index of ZUPT mode	**See table below**	int

Please note the 3 underscore (''_'') characters after ZUP header.

i	**ZUPT mode**
0	None
1	Static 10m/s
2	Static 0.1m/s
3	Auto static 0.01m/s
4	Autostatic bench 0.01m/s
6	Fixed position
7	Autostatic Position

To retrieve the Zero Velocity Update Mode:

$PIXSE,CONFIG,ZUP___,,*5D<CR><LF>

8.2.19 Turn on/off DVL Calibration Mode

To control the DVL calibration process:

Message	$PIXSE,CONFIG,DDRECK,i*hh<CR><LF>		
Title	Dead Reckoning Configuration		
Data Field	**Semantics**	**Syntax**	**Type**
i	Index of Dead Reckoning Mode	0: Stop calibration and ignore estimated calibration values 1: Start calibration mode and reset estimations 2: Stop calibration and save estimated calibration values for odometer calibration	int

Comments:

- When calibration (Dead Reckoning) is started (index set to 1), the $PIXSE,DDRECK frame is output from the PHINS Standard protocol, to provide for estimated misalignment (heading and pitch) and scale factor during calibration + associated standard deviation. For more details, refer to PHINS STANDARD protocol description.
- When calibration is stopped (index set to 0), the misalignments and scale factor estimations are not accounted for, and the misalignment parameters are kept unchanged.
- When calibration is stopped (index set to 2), current DVL misalignments and scale factor parameters are automatically corrected with new calibration estimation. A "Save to PROM" command is required afterward to permanently save these values.

To retrieve current calibration mode:

$PIXSE,CONFIG,DDRECK,,*42<CR><LF>

Message	$PIXSE,CONFIG,DDRECK,i*hh<CR><LF>	
Title	Dead Reckoning Configuration	
Data Field	**Syntax**	**Type**
i	0: Calibration OFF	int
	1: Calibration NONE	
	2: Calibration COARSE	
	3: Calibration FINE	
	4: Calibration COMPLETE	

The DVL calibration process is not stopped automatically by the algorithm, but the Index value is set to 4 when all parameters estimated internally has been finished to converge. The frame associated to this estimation is also described on section 8.3.1.2 (LOGCAL command).

8.2.20 Turn ON/OFF DVL Calibration Check Mode

To control the DVL calibration check process:

Message	$PIXSE,CONFIG,CALCHK,i*hh<CR><LF>		
Title	Calibration Check		
Data Field	**Semantics**	**Syntax**	**Type**
i	Calibration Check State	0: STOP 1: START	int

To retrieve current calibration check mode:

$PIXSE,CONFIG,CALCHK,,*53<CR><LF>

8.2.21 Configuration Starting Mode

The frame used to define the starting mode is:

Message	$PIXSE,CONFIG,START_,i*hh<CR><LF>		
Title	Starting Mode		
Data Field	**Semantics**	**Syntax**	**Type**
i	Index for Starting Mode	0 : Immediate run after power on 1 : Wait for Position 2 : Restore Position 3 : Restore Attitude 4 : Emulation mode	int

To retrieve the starting mode:

$PIXSE,CONFIG,START_,,*42<CR><LF>

8.2.22 Altitude Calculation Mode

The frame used to define the mode to compute the altitude is:

Message	$PIXSE,CONFIG,ALTMDE,i*hh<CR><LF>		
Title	Altitude Calculation Mode		
Data Field	**Semantics**	**Syntax**	**Type**
i	Index	**See table below**	int

To retrieve the altitude computation mode:

$PIXSE,CONFIG,ALTMDE,,*48<CR><LF>

Table 27 - Altitude Mode for Rovins

i	Altitude mode
0	Stabilization
1	GNSS
2	Depth
3	Hydro

8.2.23 Heave Parameters

The frame used to define the sea state is:

Message	$PIXSE,HVECNF,i*hh<CR><LF>		
Title	Heave Parameters		
Data Field	**Semantics**	**Syntax**	**Type**
I	Index	0: Slight Sea (< 1.2 m) 1: Moderate Sea (< 2.5 m) 2: Rough Sea (> 2.5 m) 3: Harbors and Channels	int

To retrieve current sea state mode:

$PIXSE,CONFIG,HVECNF,,*4D<CR><LF>

8.2.24 Static Convergence Selection

The frame used to enable/disable static convergence algorithm:

Message	$PIXSE,CONFIG,CVSTAT,i*hh<CR><LF>		
Title	Static convergence selection		
Data Field	**Semantics**	**Syntax**	**Type**
I	Start	0: Disable static convergence algorithm 1: Enable static convergence algorithm	int

To retrieve current convergence mode:

$PIXSE,CONFIG,CVSTAT,,*5A<CR><LF>

8.2.25 Go to Navigation

The frame used to switch to navigation mode is:

Message	$PIXSE,CONFIG,GONAV_*53<CR><LF>
Title	Switch to navigation mode

8.2.26 UTM Zone Mode

The frame used to define the UTM Zone mode is:

Message	**$PIXSE,CONFIG,UTMEXT_,i*hh<CR><LF>**		
Title	UTM Zone		
Data Field	**Semantics**	**Syntax**	**Type**
i	UTM Zone Mode	0 : Standard Mode 1 : Extended Mode	int

To retrieve the UTM Zone mode:

$PIXSE,CONFIG,UTMEXT,,*58<CR><LF>

8.2.27 Geoïdal Separation

The frame used to configure the Geoïdal Separation:

Message	$PIXSE,CONFIG,GEOSEP,s.s*hh<CR><LF>		
Title	Geoïdal Separation		
Data Field	Semantics	**Unit**	Type
s.s	Separation	**m**	float

To retrieve the Geoïdal Separation:

$PIXSE,CONFIG,GEOSEP,,*56<CR><LF>

8.2.28 Geodetic Convention Setup

The frame used to configure the geodetic convention:

Message	**$PIXSE,CONFIG,MODGEO,i*hh<CR><LF>**		
Title	Geodetic convention		
Data Field	**Semantics**	**Unit**	**Type**
i	Coordinate mode 0/Others: Lat/Long 1:UTM 2:Polar 3:UPS 4:OSGB 5:MGRS 6:GEOREF 7:ECEF	**-**	int

To retrieve the geodetic convention:

$PIXSE,CONFIG,MODGEO,,*56<CR><LF>

8.2.29 Advanced Filtering Setup

8.2.29.1 Setup Activation

The frame used to enable/disable the “Advanced Filtering mode” is:

Message	$PIXSE,CONFIG,ADVFIL,i*hh<CR><LF>		
Title	ADVanced FILtering mode		
Data Field	**Semantics**	**Unit**	**Type**
i	Advanced filtering mode activation: 0: OFF 1: ON	-	int

To retrieve the advanced filtering setup parameters is:

$PIXSE,CONFIG,ADVFIL,,*4D<CR><LF>

8.2.29.2 Minimum Rejection

The frame used to set the minimum rejection ratio for input position is:

Message	$PIXSE,CONFIG,ADVREJ,i,j,k.k*hh<CR><LF>		
Title	ADVanced filtering minimum REJection		
Data Field	**Semantics**	**Unit**	**Type**
i	Sensors switch input 0: GNSS 1: USBL	-	int
j	Minimum mode rejection 0: OFF 1: ON	-	int
k.k	SD value to set (-1 to ignore)	m	float

This command is ignored if the Advanced Filtering mode is OFF.

To retrieve the advanced filtering minimum rejection ratio parameters is:

$PIXSE,CONFIG,ADVREJ,,*53<CR><LF>

Message	$PIXSE,**C**ONFIG,ADVREJ,i,j,k.k,l.l*hh<CR><LF>		
Title	ADVanced filtering minimum REJection		
Data Field	**Semantics**	**Unit**	**Type**
i	Minimum mode rejection for GNSS 0: OFF 1: ON	-	int
j	Minimum mode rejection for USBL 0: OFF 1: ON	-	int
k.k	GNSS Minimum SD value	m	float
l.l	USBL Minimum SD value	m	float

8.2.29.3 Input Standard Deviation

The frame used to set the input standard deviation associated to input sensors is:

Message	**$PIXSE,CONFIG,ADVISD,i,j*hh<CR><LF>**		
Title	ADVanced filtering Input Standard Deviation		
Data Field	**Semantics**	**Unit**	**Type**
i	Sensors switch input SD 0: DVL Bottom Track 1: DVL Water Track 2: EMLOG 3: DEPTH	-	int
j	SD value to set in (-1 to ignore): - Meters for DEPTH - Meters per Second for DVL (BT/WT) or EMLOG	m m/s	float

This command is ignored if the Advanced Filtering mode is OFF.

To retrieve the advanced filtering input standard deviation parameters is:

$PIXSE,CONFIG,ADVISD,,*50<CR><LF>

Message	**$PIXSE,CONFIG,ADVISD,i.i,j.j,k.k,l.l*hh<CR><LF>**		
Title	ADVanced filtering Input Standard Deviation		
Data Field	**Semantics**	**Unit**	**Type**
i.i	DVL Bottom Track SD value	m/s	float
j.j	DVL Water Track SD value	m/s	float
k.k	EMLOG SD value	m/s	float
l.l	DEPTH SD value	m	float

8.3 External Sensor Configuration

8.3.1 DVL Configuration

8.3.1.1 Lever Arm

To configure the lever arm from Rovins to the DVL:

Message	**$PIXSE,CONFIG,LOGLV_,x.x,y.y,z.z*hh<CR><LF>**		
Title	DVL Lever Arm		
Data Field	**Semantics**	**Unit**	**Type**
x.x	X_{V1} lever arm	m	float
y.y	X_{V2} lever arm	m	float
z.z	X_{V3} lever arm	m	float

To retrieve the lever arm from Rovins to the DVL:

$PIXSE,CONFIG,LOGLV_,,*5C<CR><LF>

8.3.1.2 DVL calibration (misalignments and scale factor)

To manually configure the DVL calibration:

Message	**$PIXSE,CONFIG,LOGCAL,x.x,y.y,z.z,s.s*hh<CR><LF>**		
Title	DVL Calibration		
Data Field	**Semantics**	**Unit**	**Type**
x.x	misalignment / unit around X_{V1} (roll)	Deg	float
y.y	misalignment / unit around X_{V2} (pitch)	Deg	float
z.z	misalignment /unit around X_{V3} (heading)	Deg	float
s.s	Scale factor correction percentage		float

To retrieve the DVL calibration:

$PIXSE,CONFIG,LOGCAL,,*57<CR><LF>

Message	**$PIXSE,CONFIG,LOGCAL,o.o,p.p,q.q,r.r,x.x,y.y,z.z,s.s*hh<CR><LF>**		
Title	DVL Calibration		
Data Field	**Semantics**	**Unit**	Type
o.o	misalignment / unit around XV1 (roll) estimation	Deg	float
p.p	misalignment / unit around XV2 (pitch) estimation	Deg	float
q.q	misalignment /unit around XV3 (heading) estimation	Deg	float
r.r	Scale factor correction estimation (*)		float
x.x	SD misalignment / unit around XV1 (roll) estimation (**)	Deg	float
y.y	SD misalignment / unit around XV2 (pitch) estimation (**)	Deg	float
z.z	SD misalignment /unit around XV3 (heading) estimation (**)	Deg	float
s.s	SD Scale factor correction estimation (*)		float

(*) for example 0.00123 give 0.123%

(**) only available when calibration process is running.

Note: These estimations are also available through the PHINS STANDARD output frame (refer to section 8.2.20).

8.3.1.3 DVL Interface

To configure the DVL interface (Rovins interface to receive data from log):

Message	$PIXSE,CONFIG,LOGINT,i*hh<CR><LF>		
Title	DVL Interface		
Data Field	Semantics	Syntax	Type
i	Interface	0: None 1: Port A 2: Port B 3: Port C 4: Port D 5: Port E 6 :Port F 7 :Port G	int

The interface should be configured in accordance with the instrument.

To retrieve the Rovins interface for DVL data reception:

$PIXSE,CONFIG,LOGINT,,*4A<CR><LF>

8.3.1.4 Sound velocity compensation

Rovins can compensate the DVL measurement with the velocity of sound received from an external sensor.

To configure the interface to get the real time velocity of sound:

Message	$PIXSE,CONFIG,LOGSND,i*hh<CR><LF>		
Title	Sound Velocity compensation		
Data Field	Semantics	Syntax	Type
i	Interface	0: None 1: Port A 2: Port B 3: Port C 4: Port D 5: Port E 6 :Port F 7 :Port G	int

The interface should be configured in accordance with the instrument.

To retrieve the Rovins interface to which the sound velocity sensor is connected:

$PIXSE,CONFIG,LOGSND,,*40<CR><LF>

8.3.1.5 Rejection Filter configuration for Bottom Track

To configure the Rejection Filter mode for DVL data in Bottom Track mode:

Message	$PIXSE,CONFIG,LOGKFM,i*hh<CR><LF>		
Title	Rejection Filter Configuration for Bottom Track		
Data Field	Semantics	Syntax	Type
i	Mode Index	0: Not Active Always True 1: Not Active Always False 2: Active Automatic reacquisition 3: Active Manual reacquisition	int

To retrieve the Rejection Filter mode for DVL data in Bottom Track mode:

$PIXSE,CONFIG,LOGKFM,,*59<CR><LF>

8.3.1.6 Rejection Filter configuration for Water Track

To configure the Rejection Filter mode for DVL data in Water Track mode:

Message	$PIXSE,CONFIG,LOGWTM,i*hh<CR><LF>		
Title	Rejection Filter Configuration for Water Track		
Data Field	Semantics	Syntax	Type
i	Mode Index	0: Not Active Always True 1: Not Active Always False 2: Active Automatic reacquisition 3: Active Manual reacquisition	int

To retrieve the Rejection Filter mode for DVL data in Water Track mode:

$PIXSE,CONFIG,LOGWTM,,*57<CR><LF>

8.3.1.7 Coupling Mode

To configure the DVL Coupling Mode:

Message	$PIXSE,CONFIG,LOGCPL,i*hh<CR><LF>		
Title	DVL Coupling mode		
Data Field	Semantics	Syntax	Type
i	Coupling Mode	0: DVL not coupled to the system 1: DVL coupled to the system	int

To retrieve the DVL coupling mode:

$PIXSE,CONFIG,LOGCPL,,*46<CR><LF>

8.3.1.8 DVL Type

To configure the DVL type:

Message	$PIXSE,CONFIG,LOGTYP,i*hh<CR><LF>		
Title	DVL Type		
Data Field	**Semantics**	**Syntax**	**Type**
i	DVL type	0: Piston DVL 1: Phased Array DVL	int

To retrieve the DVL type:

$PIXSE,CONFIG,LOGTYP,,*44<CR><LF>

8.3.2 DVL2 CONFIGURATION

8.3.2.1 Lever Arm

To configure the lever arm from Rovins to the DVL2:

Message	$PIXSE,CONFIG,LOG2LV_,x.x,y.y,z.z*hh<CR><LF>		
Title	DVL2 Lever Arm		
Data Field	**Semantics**	**Unit**	**Type**
x.x	X_{V1} lever arm	m	float
y.y	X_{V2} lever arm	m	float
z.z	X_{V3} lever arm	m	float

To retrieve the lever arm from Rovins to the DVL2:

$PIXSE,CONFIG,LOG2LV_,,*21<CR><LF>

8.3.2.2 DVL2 calibration (misalignments and scale factor)

To manually configure the DVL2 calibration:

Message	$PIXSE,CONFIG,LOG2CAL,x.x,y.y,z.z,s.s*hh<CR><LF>		
Title	DVL2 Calibration		
Data Field	**Semantics**	**Unit**	**Type**
x.x	misalignment / unit around X_{V1} (roll)	Deg	float
y.y	misalignment / unit around X_{V2} (pitch)	Deg	float
z.z	misalignment /unit around X_{V3} (heading)	Deg	float
s.s	Scale factor correction percentage		float

8.3.2.3 DVL2 Interface

To configure the DVL2 interface (Rovins interface to receive data from log):

Message	$PIXSE,CONFIG,LOG2INT,i*hh<CR><LF>		
Title	DVL2 Interface		
Data Field	Semantics	Syntax	Type
i	Interface	0: None 1: Port A 2: Port B 3: Port C 4: Port D 5: Port E 6 :Port F 7 :Port G	int

The interface should be configured in accordance with the instrument.
To retrieve the Rovins interface for DVL2 data reception:

$PIXSE,CONFIG,LOG2INT,,*4A<CR><LF>

8.3.2.4 Rejection Filter configuration for Bottom Track

To configure the Rejection Filter mode for DVL data in Bottom Track mode:

Message	$PIXSE,CONFIG,LOGKFM,i*hh<CR><LF>		
Title	Rejection Filter Configuration for Bottom Track		
Data Field	Semantics	Syntax	Type
i	Mode Index	0: Not Active Always True 1: Not Active Always False 2: Active Automatic reacquisition 3: Active Manual reacquisition	int

To retrieve the Rejection Filter mode for DVL data in Bottom Track mode:

$PIXSE,CONFIG,LOGKFM,,*59<CR><LF>

8.3.2.5 Rejection Filter configuration for Water Track

To configure the Rejection Filter mode for DVL data in Water Track mode:

Message	**$PIXSE,CONFIG,LOGWTM,i*hh<CR><LF>**		
Title	Rejection Filter Configuration for Water Track		
Data Field	**Semantics**	**Syntax**	**Type**
i	Mode Index	0: Not Active Always True 1: Not Active Always False 2: Active Automatic reacquisition 3: Active Manual reacquisition	int

To retrieve the Rejection Filter mode for DVL data in Water Track mode:

$PIXSE,CONFIG,LOGWTM,,*57<CR><LF>

8.3.2.6 Coupling Mode

To configure the DVL2 Coupling Mode:

Message	**$PIXSE,CONFIG,LOG2CPL,i*hh<CR><LF>**		
Title	DVL2 Coupling mode		
Data Field	**Semantics**	**Syntax**	**Type**
i	Coupling Mode	0: DVL not coupled to the system 1: DVL coupled to the system	int

To retrieve the DVL coupling mode:

$PIXSE,CONFIG,LOG2CPL,,*3B<CR><LF>

8.3.2.7 DVL Type

To configure the DVL type:

Message	**$PIXSE,CONFIG,LOGTYP,i*hh<CR><LF>**		
Title	DVL Type		
Data Field	**Semantics**	**Syntax**	**Type**
i	DVL type	0: Piston DVL 1: Phased Array DVL	int

To retrieve the DVL type:

$PIXSE,CONFIG,LOGTYP,,*44<CR><LF>

8.3.3 DVL COMMAND

To send command to DVL:

Message	**$PIXSE,CONFIG,DVLCMD,x,y*hh<CR><LF>**		
Title	DVL command		
Data Field	**Semantics**	**Syntax**	**Type**
x	ID DVL	1: DVL 2: DVL2	int
y	ID command Or Sub Command	ID command: 0: DVL Stop pinging 1: DVL Start Pinging 2: DVL Single Ping (see note 1) Sub Command: string: sub ASCII command (256 chars max)	int or string

Frame examples:

- DVL2 Stop pinging:
  $PIXSE,CONFIG,DVLCMD,2,0*4B<CR><LF>
- Sending a sub ASCII command to DVL1:
  $PIXSE,CONFIG,DVLCMD,1,CS*68<CR><LF>

Note 1: Refer to Teledyne RDI user guide to configure your DVL into single ping or periodic ping mode.

Follow this procedure to use this function:

- Set flow control of RDI DVL to CFx0xxx (Single Ping), CFx1xxx being for periodic ping (default is CF11110)
- To engage the single ping mode, send a first command “1: DVL Start Pinging”
- Send single ping command (“2: DVL Single Ping”) periodically
- To disengaged the single ping mode, send “0: DVL Stop pinging” command

8.3.4 EM Log Configuration

8.3.4.1 EM Log Lever Arm

To configure the lever arm from the Rovins to the EM Log sensor:

Message	**$PIXSE,CONFIG,LMNLV_,x.x,y.y,z.z*hh<CR><LF>**		
Title	EM Log Lever Arm		
Data Field	**Semantics**	**Unit**	**Type**
x.x	X_{V1} lever arm	m	float
y.y	X_{V2} lever arm	m	float
z.z	X_{V3} lever arm	m	float

To retrieve the lever arm from the Rovins to the EM Log sensor:

$PIXSE,CONFIG,LMNLV_,,*57<CR><LF>

8.3.4.2 EM Log Interface

To configure the EM Log interface (Rovins interface to receive data from EM Log sensor):

Message	**$PIXSE,$PIXSE,CONFIG,LMNINT,i*hh<CR><LF>**		
Title	EM Log Interface		
Data Field	**Semantics**	**Syntax**	**Type**
i	Interface	0: None 1: Port A 2: Port B 3: Port C 4: Port D 5: Port E	int

The interface should be configured in accordance with the instrument.

To retrieve the Rovins interface to which EM Log should be connected:

$PIXSE,CONFIG,LMNINT,,*41<CR><LF>

8.3.4.3 EM Log Rejection Filter

To configure the EM Log Rejection Filter mode:

Message	**$PIXSE,CONFIG,LMNKFM,i*hh<CR><LF>**		
Title	Rejection Filter Configuration for EM Log		
Data Field	**Semantics**	**Syntax**	**Type**
i	Mode Index	0: Not Active Always True 1: Not Active Always False 2: Active Automatic reacquisition 3: Active Manual reacquisition	int

To retrieve the EM Log Rejection Filter mode :

$PIXSE,CONFIG,LMNKFM,,*52<CR><LF>

8.3.5 GNSS Configuration

8.3.5.1 GNSS Lever Arm

To configure the lever arm from the Rovins to the GNSS:

Message	**$PIXSE,CONFIG,GPSLV_,x.x,y.y,z.z*hh<CR><LF>**		
Title	GNSS Lever Arm		
Data Field	**Semantics**	**Unit**	**Type**
x.x	X_{V1} lever arm	**m**	float
y.y	X_{V2} lever arm	**m**	float
z.z	X_{V3} lever arm	**m**	float

To retrieve the lever arm from the Rovins to the GNSS:

$PIXSE,CONFIG,GPSLV_,,*5C<CR><LF>

8.3.5.2 GNSS Interface

To configure the GNSS interface (Rovins interface to receive data from GNSS):

Message	**$PIXSE,CONFIG,GPSINT,i*hh<CR><LF>**		
Title	GNSS Interface		
Data Field	**Semantics**	**Syntax**	**Type**
i	Interface	**0: None** **1: Port A** **2: Port B** **3: Port C** **4: Port D** **5: Port E** **200: Repeater**	int

The interface should be configured in accordance with the instrument.
To retrieve the Rovins interface to which GNSS should be connected:

$PIXSE,CONFIG,GPSINT,,*4A<CR><LF>

8.3.5.3 GNSS Rejection Filter

To configure the GNSS Rejection Filter mode:

Message	$PIXSE,CONFIG,GPSKFM,i*hh<CR><LF>		
Title	Rejection Filter Configuration for GNSS		
Data Field	**Semantics**	**Syntax**	**Type**
i	Mode Index	0: Not Active Always True 1: Not Active Always False 2: Active Automatic reacquisition 3: Active Manual reacquisition	int

To retrieve the GNSS Rejection Filter mode :

$PIXSE,CONFIG,GPSKFM,,*59<CR><LF>

8.3.6 GNSS2 Configuration

8.3.6.1 GNSS2 Lever Arm

To configure the lever arm from the Rovins to the GNSS2:

Message	$PIXSE,CONFIG,GPS2LV_,x.x,y.y,z.z*hh<CR><LF>		
Title	GNSS2 Lever Arm		
Data Field	**Semantics**	**Unit**	**Type**
x.x	X_{V1} lever arm	m	float
y.y	X_{V2} lever arm	m	float
z.z	X_{V3} lever arm	m	float

To retrieve the lever arm from the Rovins to the GNSS2:

$PIXSE,CONFIG,GPS2LV_,,*3D<CR><LF>

8.3.6.2 GNSS2 Interface

To configure the GNSS2 interface (Rovins interface to receive data fromGNSS2):

Message	**$PIXSE,CONFIG,GPS2INT,i*hh<CR><LF>**		
Title	GNSS2 Interface		
Data Field	**Semantics**	**Syntax**	**Type**
i	Interface	0: None 1: Port A 2: Port B 3: Port C 4: Port D 5: Port E	int

The interface should be configured in accordance with the instrument.
To retrieve the Rovins interface to which GNSS2 should be connected:
$PIXSE,CONFIG,GPS2INT,,*2B<CR><LF>

8.3.6.3 GNSS2 Rejection Filter

To configure the GNSS2 Rejection Filter mode:

Message	**$PIXSE,CONFIG,GPS2KFM,i*hh<CR><LF>**		
Title	Rejection Filter Configuration for GNSS2		
Data Field	**Semantics**	**Syntax**	**Type**
i	Mode Index	0: Not Active Always True 1: Not Active Always False 2: Active Automatic reacquisition 3: Active Manual reacquisition	int

To retrieve the GNSS2 Rejection Filter mode :
$PIXSE,CONFIG,GPS2KFM,,*38<CR><LF>

8.3.7 MANUAL GNSS CONFIGURATION

8.3.7.1 Manual GNSS Lever Arm

To configure the lever arm from the Rovins to the manual GNSS:

Message	**$PIXSE,CONFIG,GPMLV_,x.x,y.y,z.z*hh<CR><LF>**		
Title	Manual GNSS Lever Arm		
Data Field	**Semantics**	**Unit**	**Type**
x.x	X_{V1} lever arm	m	float
y.y	X_{V2} lever arm	m	float
z.z	X_{V3} lever arm	m	float

To retrieve the lever arm from the Rovins to the manual GNSS:

$PIXSE,CONFIG,GPMLV_,,*42<CR><LF>

8.3.7.2 Manual GNSS Rejection Filter

To configure the manual GNSS Rejection Filter mode:

Message	**$PIXSE,CONFIG,GPMKFM,i*hh<CR><LF>**		
Title	Rejection Filter Configuration for manual GNSS		
Data Field	**Semantics**	**Syntax**	**Type**
I	Mode Index	0: Not Active Always True 1: Not Active Always False 2: Active Automatic reacquisition 3: Active Manual reacquisition	int

To retrieve the manual GNSS Rejection Filter mode:

$PIXSE,CONFIG,GPMKFM,,*47<CR><LF>

8.3.7.3 Manual GNSS Position Fix

To send a manual position fix:

Message	**$PIXSE,CONFIG,MANGPS,a.a,b.b,c.c,d.d,e.e,f.f*hh<CR><LF>**		
Title	Manual GNSS position fix		
Data Field	**Semantics**	**Unit**	**Type**
a.a	Latitude	degree	float
b.b	Longitude	degree	float
c.c	Altitude	m	float
d.d	Latitude standard deviation	m	float
e.e	Longitude standard deviation	m	float
f.f	Altitude standard deviation	m	float

To retrieve last manual position fix:

$PIXSE,CONFIG,MANGPS,,*5B<CR><LF>

8.3.8 Depth Sensor Configuration

8.3.8.1 Depth Lever Arm

To configure the lever arm from the Rovins to the depth sensor:

Message	$PIXSE,CONFIG,DEPLV_,x.x,y.y,z.z*hh<CR><LF>		
Title	Depth Lever Arm		
Data Field	Semantics	Unit	Type
x.x	X_{V1} lever arm	m	float
y.y	X_{V2} lever arm	m	float
z.z	X_{V3} lever arm	m	float

To retrieve the lever arm from the Rovins to the depth sensor:

$PIXSE,CONFIG,DEPLV_,,*49<CR><LF>

8.3.8.2 Depth Sensor Offset

To configure the offset that will be subtracted from depth sensor input value:

Message	$PIXSE,CONFIG,DEPOFS,x.x*hh<CR><LF>		
Title	Depth offset		
Data Field	Semantics	Unit	Type
x.x	Offset	m	float

To retrieve current depth offset:

$PIXSE,CONFIG,DEPOFS,,*56<CR><LF>

8.3.8.3 Zero Depth Sensor

To use current raw depth input as depth offset (calibrate the depth sensor):

Message	$PIXSE,CONFIG,DEPZER,,*41<CR><LF>
Title	Zero depth

8.3.8.4 Depth Sensor Interface

To configure the depth sensor interface (Rovins interface to receive data from depth sensor):

Message	**$PIXSE,CONFIG,DEPINT,i*hh<CR><LF>**		
Title	Depth Sensor Interface		
Data Field	**Semantics**	**Syntax**	**Type**
i	Interface	0: None 1: Port A 2: Port B 3: Port C 4: Port D 5: Port E	int

The interface should be configured in accordance with the instrument.
To retrieve the depth sensor interface (Rovins interface to receive data from depth sensor):

$PIXSE,CONFIG,DEPINT,,*5F<CR><LF>

8.3.8.5 Rejection Filter Mode for Depth Sensor

To configure the Rejection Filter mode for the depth sensor:

Message	**$PIXSE,CONFIG,GPS2KFM,i*hh<CR><LF>**		
Title	Rejection Filter Configuration for Depth Sensor		
Data Field	**Semantics**	**Syntax**	**Type**
i	Mode Index	0: Not Active Always True 1: Not Active Always False 2: Active Automatic reacquisition 3: Active Manual reacquisition	int

To retrieve the Rejection Filter mode for Depth Sensor::

$PIXSE,CONFIG,DEPKFM,,*4C<CR><LF>

8.3.9 USBL CONFIGURATION

8.3.9.1 USBL Lever Arm

To configure the lever arm from the Rovins to the USBL beacon:

Message	$PIXSE,CONFIG,USBLV_,x.x,y.y,z.z,i*hh<CR><LF>		
Title	USBL Lever Arm		
Data Field	Semantics	Unit	Type
x.x	X_{V1} lever arm	m	float
y.y	X_{V2} lever arm	m	float
z.z	X_{V3} lever arm	m	float
i	USBL beacon index (0, 1 or 2)	n/a	int

To retrieve the lever arm from the Rovins to the specified USBL beacon (i is the beacon index:

$PIXSE,CONFIG,USBLV_,,i*hh<CR><LF>

8.3.9.2 USBL Interface

To configure the USBL interface (Rovins interface to receive data from specified USBL beacon):

Message	$PIXSE,CONFIG,USBINT,i,ttt,j*hh<CR><LF>		
Title	USBL Interface		
Data Field	Semantics	Syntax	Type
i	Interface	0: None 1: Port A 2: Port B 3: Port C 4: Port D 5: Port E	int
ttt	Tp code	8 char max	string
j	index	USBL beacon index (0,1 or 2)	int

To retrieve the specified USBL beacon interface (j is the beacon index):

$PIXSE,CONFIG,USBINT,,j*hh<CR><LF>

8.3.9.3 USBL Rejection Filter Mode

To configure the Rejection Filter mode for the specified USBL beacon:

Message	**$PIXSE,CONFIG,USBKFM,i,j*hh<CR><LF>**		
Title	Rejection Filter Configuration for USBL		
Data Field	**Semantics**	**Syntax**	**Type**
i	Mode Index	0: Not Active Always True 1: Not Active Always False 2: Active Automatic reacquisition 3: Active Manual reacquisition	int

To retrieve the Rejection Filter mode for specified USBL beacon index j::

$PIXSE,CONFIG,USBKFM,,j*hh<CR><LF>

8.3.9.4 USBL Beacon Watch Selection

In PHINS STANDARD protocol, only one USBL beacon will be reported at a time.
To configure the USBL beacon to watch in PHINS STANDARD protocol:

Message	**$PIXSE,CONFIG,USBVIE,i*hh<CR><LF>**		
Title	USBL beacon watch selection		
Data Field	**Semantics**	**Syntax**	**Type**
I	Beacon index	USBL beacon index (0, 1 or 2)	int

To retrieve currently selected beacon:

$PIXSE,CONFIG,USBVIE,,*43<CR><LF>

8.3.9.5 Maximum Number of USBL Beacon

To retrieve maximum number of beacon, the firmware can manage:

$PIXSE,CONFIG,USBNBB,,*57<CR><LF>

This will return:

$PIXSE,CONFIG,USBNBB,3*48<CR><LF>

8.3.10 LBL CONFIGURATION

8.3.10.1 LBL Lever Arm

To configure the lever arm from the Rovins to the LBL computing point:

Message	$PIXSE,CONFIG,LBLLV_,x.x,y.y,z.z*hh<CR><LF>		
Title	LBL Lever Arm		
Data Field	Semantics	Unit	Type
x.x	X_{V1} lever arm	m	float
y.y	X_{V2} lever arm	m	float
z.z	X_{V3} lever arm	m	float

To retrieve the lever arm from the Rovins to the LBL computing point:

$PIXSE,CONFIG,LBLLV_,,*5A<CR><LF>

8.3.10.2 LBL Interface

To configure the LBL interface (Rovins interface to receive data from LBL):

Message	$PIXSE,CONFIG,LBLINT,i*hh<CR><LF>		
Title	LBL Interface		
Data Field	Semantics	Syntax	Type
i	Interface	0: None 1: Port A 2: Port B 3: Port C 4: Port D 5: Port E 6: Port F 7: Port G	int

To retrieve the LBL interface Rovins interface to receive data from LBL):

$PIXSE,CONFIG,LBLINT,,*4C<CR><LF>

8.3.10.3 LBL Rejection Filter Mode

To configure the LBL Rejection Filter mode:

Message	**$PIXSE,CONFIG,LBLKFM,i*hh<CR><LF>**		
Title	Rejection Filter Configuration for LBL		
Data Field	**Semantics**	**Syntax**	**Type**
i	Mode Index	0 : Not Active Always True 1 : Not Active Always False 2 : Active Automatic reacquisition 3 : Active Manual reacquisition	int

To retrieve the LBL Rejection Filter mode:

$PIXSE,CONFIG,LBLKFM,,*5F<CR><LF>

8.3.11 UTC (TIME SYNCHRONIZATION) INTERFACE

To configure the UTC interface (Rovins interface to receive data from UTC):

Message	**$PIXSE,CONFIG,UTCINT,i*hh<CR><LF>**		
Title	UTC Interface		
Data Field	**Semantics**	**Syntax**	**Type**
i	Interface	0: None 1: Port A 2: Port B 3: Port C 4: Port D 5: Port E	int

To retrieve the UTC interface (Rovins interface to receive data from UTC):

$PIXSE,CONFIG,UTCINT,,*4C<CR><LF>

8.3.12 UTC2 (TIME SYNCHRONIZATION) INTERFACE

To configure the UTC interface (Rovins interface to receive data from UTC):

Message	**$PIXSE,CONFIG,UTC2INT,i*hh<CR><LF>**		
Title	UTC Interface		
Data Field	**Semantics**	**Syntax**	**Type**
i	Interface	0: None 1: Port A 2: Port B 3: Port C 4: Port D 5: Port E	int

To retrieve the UTC interface (Rovins interface to receive data from UTC2):

$PIXSE,CONFIG,UTC2INT,,*4C<CR><LF>

8.4 Interfaces Configuration

8.4.1 Serial and Ethernet Commands

8.4.1.1 Serial I/O general parameters (parity and stop bit)

Used to configure the parity and stop bit for serial port X:

Message	**$PIXSE,CONFIG,RSCM_X,i,j[,k][,l]*hh<CR><LF>**		
Title	Serial interface configuration		
Data Field	**Semantics**	**Syntax**	**Type**
X	Port	A to G to configure input ports A to G A to E to configure output port A to E	int
I	parity	0: None 1: Even 2: Odd	int
J	number of stop bits	0: 0.5 stop bit 1: 1 stop bit 2: 1.5 stop bit 3: 2 stop bits	int
K	RFU	0	int
L	RFU	0	int

To retrieve parity and stop bit for serial port X (checksum hh depends on port X, see below) :

$PIXSE,CONFIG,RSCM_X,,*hh<CR><LF>

8.4.1.2 Serial/Ethernet input port configuration

To configure the input port X:

Message	**$PIXSE,CONFIG,RSIN_X,i,j[,k][,l]*hh<CR><LF>**		
Title	Input port configuration		
Data Field	**Semantics**	**Syntax**	**Type**
X	Port	A to G	char
i	Baud rate	See table "Baud rate index" below	int
j	Protocol	See table "List and index of input protocols for input ports" below	int
k	RFU	0	int
l	RFU	0	int

The RSIN command must be used to select input protocol in Ethernet mode too.

Changing input port baud rate will affect corresponding output port baud rate as baud rate generator is common to both input and output serial transceiver.

To retrieve configuration of serial input port X (checksum hh depends on port X, see below):

$PIXSE,CONFIG,RSIN_X,,*hh<CR><LF>

Table 28 - Baud rate index

i	Baud rate
0	600 bps
1	1200 bps
2	2400 bps
3	4800 bps
4	9600 bps
5	19200 bps
6	38400 bps
7	57600 bps
8	115200 bps
9	230400 bps
10	460800 bps
11	921600 bps

Table 29 - List and index of input protocols for input ports

Id	Input protocols
0	NONE
1	RDI PD6
2	RDI PD4
3	GPS
4	MICRO IN
5	SVP 70
6	EMLOG VBW
7	PAROSCIENTIFIC
8	APOS PSIM SSB
9	HALLIBURTON
10	USBL LBL CTD
11	Reserved
12	EXT SENSOR BIN
13	POSIDONIA
14	USBL INPUT
15	SEAKING 700
16	EMLOG VHW
17	LOG VBW
18	Reserved
19	Reserved
20	MINI SVS
21	CTD SBE
22	GAPS
23	RDI PD0
24	SVX2
25	Reserved
26	Reserved
27	IXSEA AUV
28	APOS PSIM LBP

Id	Input protocols
29	RAMSES PP
30	USBLBOX PP
31	RDI PD3
32	RDI PD3 RT
33	PRESSURE SENSOR
34	Reserved
35	STD_NMEA
36	STDBIN
37	SBE49
38	NRK DF21
39	TERPS8000

8.4.1.3 Serial/Ethernet output port configuration

To configure the Serial/Ethernet output on port X:

Message	$PIXSE,CONFIG,RSOUTX,i,j,k,l,m[,n][,o]*hh<CR><LF>		
Title	Output port configuration		
Data Field	**Semantics**	**Syntax**	**Type**
X	Port	A to E	char
i	baud rate	See "Baud rate index" table below	int
j	protocol	See "List and index of output protocols for output ports" table below	int
k	rate	Output rate in ms (100 = 100 ms = 10 Hz) Minimum value is 5 ms (200 Hz)	int
l	RS level	0: RS232 1: RS422	int
m	Lever arm	0: Main lever arm 1: Secondary lever arm 1, standard heave filter 2: Secondary lever arm 2, standard heave filter 3: Secondary lever arm 3, standard heave filter	int
o	Altitude reference	0: WGS84 Geoidal (default) 1: WGS84 Ellipsoidal	int

To retrieve configuration of serial output port X (checksum hh depends on port X):

$PIXSE,CONFIG,RSOUTX,,*hh<CR><LF>

Table 30 - List and index of output protocols for output ports

MARINS BKA-R
NONE
PHINS STANDARD
POST PROCESSING
NAV_SHORT
NAV_LONG
GPSLIKE
NAV_BINARY

MARINS BKA-R
VTGGGA
HEHDT
LONG_BIN_NAV
NAVIGATION
Reserved
Reserved
SPERRY_ATT
Reserved
BROADCAST_A
BROADCAST_B
BROADCAST_C
BROADCAST_D
BROADCAST_E
HEHDT_HEROT
VTG GGU
LONG_BIN_NAV_HR
IXSEA_ICCB_1
NAV_BINARY_HR
NAVIGATION_HDLC
Reserved
SEROUT_A_RTC
SEROUT_B_RTC
SEROUT_C_RTC
SEROUT_D_RTC
SEROUT_E_RTC
Reserved
Reserved
PTNL_GGK
NAV_BINARY_1
INHDT
S40_NAV10

MARINS BKA-R
S40_NAV100
ANSCHUTZ_STD20
Reserved
Reserved
LRS10_78_IIC
LRS10_78_IC
LRS100_32_IIC
LRS100_32_IC
LRS100_35_IIC
LRS100_35_IC
LONG_BIN_NAV_SM
Reserved
Reserved
Reserved
Reserved
Reserved
Reserved
HETHS_HEROT
SPERRY_ATT_STANAG
STDBIN_V2
STDBIN_V3
Reserved

Id	Output Protocols
0	NONE
1	PHINS STANDARD
2	POST PROCESSING
3	HALLIBURTON SAS
4	DORADO
5	CONTROL NO G
6	NAVIGATION SHORT
7	NAVIGATION LONG

Id	Output Protocols
8	GPS LIKE
9	IMU ASCII
10	IMU BINARY
11	BINARY NAV
12	VTG GGA
13	HEHDT
14	HEHDT FIXED
15	OCTANS STANDARD
16	GYROCOMPASS
17	GYROCOMPASS II
18	Reserved
19	SEAPATH
20	LONG BINARY NAV
21	BUC
22	EMT SDV GCS
23	HDMS
24	SIMRAD EM
25	SIMRAD EM TSS
26	SIMRAD EM TSSH2
27	HYDROGRAPHY
28	TECHSAS
29	TECHSAS TSS
30	PRDID
31	PRDID TSS
32	NAVIGATION
33	TSS1 DMS
34	TSS335B
35	TMS CCV IMBAT
36	POSIDONIA
37	NAV BHO
38	NAV BHO LONG

Id	Output Protocols
39	INDYN
40	DOLOG HRP
41	SENSOR RD
42	NAV AND CTD
43	EXT SENSOR BIN
44	Reserved
45	Reserved
46	SPERRY ATT
47	RIEGL
48	AUVG 3000
49	IXSEA TAH
50	Reserved
51	BROADCAST A
52	BROADCAST B
53	BROADCAST C
54	BROADCAST D
55	BROADCAST E
56	HEHDT HEROT
57	RDI SYNC
58	VTG GGU
59	LONG BINARY NAV HR
60	IXSEA ICCB1
61	RDI PD11
62	NAV BINARY HR
63	Reserved
64	STOLT OFFSHORE 2
65	SHORT GPS LIKE
66	POS MV GRP 111
67	SEATEX DHEAVE
68	HEAVE POST PRO
69	CONTROL

Id	Output Protocols
70	GAPS BIN
71	NAVIGATION HDLC
72	Reserved
73	SER OUT A RTC
74	SER OUT B RTC
75	SER OUT C RTC
76	SER OUT D RTC
77	SER OUT E RTC
78	Reserved
79	Reserved
80	PTNL GGK
81	NAV BINARY 1
82	KVH EXTENDED
83	Reserved
84	INHDT
85	S40 NAV 10
86	S40 NAV 100
87	ANSCHUTZ STD 20
88	Reserved
89	Reserved
90	LRS 10 78 IIC
91	LRS 10 78 IC
92	LRS 100 32 IIC
93	LRS 100 32 IC
94	LRS 100 35 IIC
95	LRS 100 35 IC
96	Reserved
97	MDL
98	Reserved
99	Reserved
100	PRECISE ZDA

Id	Output Protocols
101	LONG BIN NAV SM
102	Reserved
103	SEANAV ID1
104	TOKIMEC PTVF
105	RDI_PING
106	IMU RAW DATA
107	AIPOV
108	LONG BIN NAV HR 2
109	HETHS HEROT
110	RTCM_SIX
111	KINETIC_SCIENTIFIC
112	STDBIN_V2
113	SPERRY_ATT_STANAG
114	STOLT_OFFSHORE
115	STDBIN_V3
116	STANDARD2
117	INSITU
118	LRS_10_64_IIC
119	LRS_10_64_IC
120	LRS_100_33_IIC
121	LRS_100_33_IC
122	BHO_GRAVI
123	MINIFOG
124	NAV BHO2M
125	MDL2
126	KM BINARY
129	EVENT MARKER

8.4.1.4 Output port protocol datablock configuration

To configure the serial/Ethernet output on port X:

Message	$PIXSE,CONFIG,DBLOCX,aaa…a,i[,bbb…b,j,…,…]*hh<CR><LF>		
Title	Output port configuration		
Data Field	**Semantics**	**Syntax**	**Type**
X	Port	A to E	char
aaa…a	Datablock name	See list below	string
i	Datablock state	0 to do not output datablock 1 to output	int
bbb…b	Datablock name	See list below	string
j	Datablock state	0 to do not output datablock 1 to output	**int**
…	Datablock name	See list below	string
…	Datablock state	0 to do not output datablock 1 to output	int
…			

To retrieve configuration of protocol to output port X (checksum hh depends on port X):

$PIXSE,CONFIG,DBLOCX,,*hh<CR><LF>

List of datablock name for IXBLUE STD BIN output protocols:

- Attitude and Heading
- Attitude and Heading Std. Dev.
- Real Time Heave/Surge/Sway
- Smart Heave
- Heading/Roll/Pitch Rate
- Rot. Rate in Vessel Frame Compensated from Earth Rot.
- Accel. in Vessel Frame Compensated from gravity
- Position
- Position Std. Dev.
- Speed in Geographic Frame
- Speed Std. Dev in Geographic Frame
- Current in Geographic Frame
- Current Std. Dev. in Geographic Frame
- System Date
- INS Sensor Status
- INS Algorithm Status
- INS System Status
- INS User Status
- Heave Surge and Sway Speed
- Speed in Vessel Frame
- Accel. in Geographic Frame not Compensated from Gravity
- Course and Speed Over Ground

- Temperatures (ACC/FOG/ANA)
- Attitude Quaternion Data
- Attitude Quaternion Std. Dev.
- Raw Accel. in Vessel Frame not Compensated from Gravity
- Accel. Std Dev. in Vessel Frame
- Rot. Rate std. Dev. in Vessel Frame
- Rotation accelerations in vessel frame
- Rotation acceleration Std. Dev.
- Raw rot. Rate in Vessel Frame not Comp. from Earth Rot.
- UTC
- GNSS1
- GNSS2
- Manual GNSS
- EMLOG1
- USBL1
- USBL2
- USBL3
- LBL1
- LBL2
- LBL3
- LBL4
- Depth
- Sound Velocity
- DVL Ground Speed
- DVL Water Speed
- DVL2 Ground Speed
- DVL2 Water Speed

8.4.1.5 Output Device Selection

To select the device that will be used for data output on selected port:

Message	**$PHCNF,EDIROX,i*hh<CR><LF>**		
Title	Output device selection		
Data Field	**Semantics**	**Syntax**	**Type**
X	Port	A to E	char
I	Device	0: No device selected 1: Serial output on selected port 2: Ethernet output on selected port 3: Ethernet and serial output on selected port	int

To retrieve network configuration (where X is the port letter):

$PHCNF,EDIROX,,*hh<CR><LF>

8.4.1.6 Input Device Selection

To select the device that will be used for data input on selected port:

Message	$PHCNF,EDIRIX,i*hh<CR><LF>		
Title	Input device selection		
Data Field	**Semantics**	**Syntax**	**Type**
X	Port	A to E	char
I	Device	0: No device selected 1: Serial input on selected port 2: Ethernet input on selected port	int

To retrieve network configuration (where X is the port letter):

$PHCNF,EDIRIX,,*hh<CR><LF>

8.4.1.7 Port Forwarding Command

Use this command to send frames from repeater port to another port. This can be useful to configure external sensors through the Rovins:

Message	$PIXSE,CONFIG,TXx,D*hh<CR><LF>		
Title	Port forwarding		
Data Field	**Semantics**	**Syntax**	**Type**
x	Port	A: Port A, B: Port B, … E: Port E	char
D	Forwarded data	String to forward to selected port. The system will add <CR><LF> to the string and forward it to port X. End of string is detected by '*' character before checksum.	string

To send a break on selected output port, use "BREAK,t" as D string. This will generate a break of t milliseconds to selected port.

Example: to send frame 'TEST' to port A:

$PIXSE,CONFIG, TXA,TEST*0A<CR><LF>

8.4.2 ETHERNET CONFIGURATION

8.4.2.1 Network Setup Command

Use this command to configure Ethernet network settings. These settings will be effective after next reboot only.

Message	$PHCNF,ETHIP,D,I,N[,G][,H]*hh<CR><LF>		
Title	Ethernet configuration		
Data Field	**Semantics**	**Syntax**	**Type**
D	DHCP	0: disable DHCP at boot time 1: enable DHCP	Int
I	IP address	System IP address (i.e.: 192.168.36.100)	IP address
N	Netmask	Network mask (i.e.: 255.255.255.0)	IP address
G	Gateway	Gateway (i.e.: 192.168.36.254)	IP address
H	DNS	DNS (i.e.: 192.168.36.1)	IP address

To retrieve network configuration:

$PHCNF,ETHIP,,*3F<CR><LF>

8.4.2.2 IP Input Configuration

To configure the IP input settings on selected port:

Message	$PHCNF,ELCFIX,m,i,p*hh<CR><LF>		
Title	IP input settings		
Data Field	**Semantics**	**Syntax**	**Type**
X	Port	A to G	Char
m	Mode	IP input mode: 0: TCP server 1: TCP client 2: UDP point to point 3: UDP broadcast 4: UDP multicast	int
i	Destination IP	IP address to connect to. This field is ignored in TCP server and UDP broadcast modes.	IP address
p	IP Port	Port to connect/ listen to	int

To retrieve IP input settings on port X, use following command:

$PHCNF,ELCFIX,,*hh<CR><LF>

8.4.2.3 IP Output Configuration

To configure the IP output settings on selected port:

Message	**$PHCNF,ELCFOX,m,i,p*hh<CR><LF>**		
Title	IP output settings		
Data Field	**Semantics**	**Syntax**	**Type**
X	Port	A to G	Char
M	Mode	IP output mode: 0: TCP server 1: TCP client 2: UDP point to point 3: UDP broadcast 4: UDP multicast	Int
I	Destination IP	IP address to connect to. This field is ignored in TCP server and UDP broadcast modes.	IP address
P	IP Port	Port to connect/listen to	Int

To retrieve IP output settings on port X, use following command:

$PHCNF,ELCFOX,,*hh<CR><LF>

8.4.3 PULSES INTERFACE

8.4.3.1 Pulses Input Configuration

To configure pulse input X:

The protocol parameter is only used on factory protocols for now (heading, roll, pitch trigger). When not used, this parameter can be set to 0.0.

To retrieve configuration of input pulse port X (checksum hh depends on port X):

$PIXSE,CONFIG,IOIN_X,,*hh<CR><LF>

Table 31 - list and index of input protocols for pulse ports

Index	Input Pulse Type
0	NONE
1	PPS RISING + ZDA EDGE
2	PPS FALLING + ZDA EDGE
3	ZDA + PPS RISING EDGE

Index	Input Pulse Type
4	ZDA + PPS FALLING EDGE
5	SERIAL OUT SYNC A
6	SERIAL OUT SYNC B
7	SERIAL OUT SYNC C
8	SERIAL OUT SYNC D
9	SERIAL OUT SYNC E
10	EVENT MARKER RISING
11	EVENT MARKER FALLING
12	EVENT MARKER RISING PP
13	EVENT MARKER FALLING PP

8.4.3.2 Pulses Input Association

To associate the pulse input X to an UTC interface with:

Message	**$PIXSE,CONFIG,IOINAX,y*hh<CR><LF>**		
Title	Pulse Input		
Data Field	**Semantics**	**Syntax**	**Type**
X	Pulse index	A : Pulse A B : Pulse B C : Pulse C D : Pulse D	char
y	port	0: None 1: UTC1 2: UTC2	int

The command will not be taken into account if the selected pulse has been already associated to another mechanism (UTC or Event Marker). There is no protection mechanism which avoids to configure a nonexistent pulse.
Pulses associated to Rovins: Pulse A, Pulse B, Pulse C

To retrieve the associated UTC interface on pulse selected:

$PIXSE,CONFIG,IOINAX,,*hh<CR><LF>

8.4.3.3 Pulses Output Configuration

To configure pulse output X:

Message	$PIXSE,CONFIG,IOOUTX,x.x,i*hh<CR><LF>		
Title	Pulse Output		
Data Field	**Semantics**	**Syntax**	**Type**
X	Port	A: Port A B: Port B	char
x.x	parameter	Parameter for the pulse protocol	float
i	Protocol	See table below	int

The protocol parameter is only used on factory protocols for now (heading, roll, pitch trigger). When not used, this parameter can be set to 0.0.

To retrieve configuration of pulse output port X (checksum hh depends on port X) :

$PIXSE,CONFIG,IOOUTX,,*hh<CR><LF>

Table 32 - List and index of output protocols for pulse ports

Index	Output Pulse Type
0	NONE
1	SER OUT A RTC
2	SER OUT B RTC
3	SER OUT C RTC
4	SER OUT D RTC
5	SER OUT E RTC
6	TRAVELED DISTANCE RISING
7	TRAVELED DISTANCE FALLING
8	PPS LIKE
9	TIMER RISING
10	TIMER FALLING

8.5 Dynamic string retrieve commands

8.5.1 Generic text retrieve command

To retrieve a specific text for a command:

Message	**$PIXSE,TEXT__,list,i,j,c*hh<CR><LF>**		
Title	Text list retrieve		
Data Field	**Semantics**	**Syntax**	**Type**
list	List name	Name of command associated with the list	string
i	Section index	Index of list to retrieve for this command	int
j	String index	Index of string in the list	int
c	Language	Only English 'E' is supported	char

Example:
To retrieve first serial output protocol name, you should send:

$PIXSE,TEXT__,RSOUTX,1,0,E*35<CR><LF>

Rovins will then answer:

$PIXSE,TEXT__,RSOUTX,1,0,NONE*7A<CR><LF>

When no string is available, Rovins returns "________________" (16 x '_' character).
Thus, to retrieve all available output protocol names, you should send $PIXSE,CONFTEXT,RSOUT command and increment string index until the firmware answers no string available. Following table details all string retrieve functions and their parameters:

Table 33 - List name table

List name	**Section index**	**Retrieved list description**
RSOUTX	0	Port output baud rate strings
	1	Port output protocol names
	2	Port output lever arm names
	3	Port output level names
	4	Port output heave filter names
RSCM_X	0	Port input/output parity strings
	1	Port input/output stop bits strings
RSIN_X	0	Port input/output baud rate strings
	1	Port input protocol names

List name	Section index	Retrieved list description
IOIN_X	0	Input pulse protocol names
IOOUTX	0	Output pulse protocol names
STATUS	0	System status word bit names and attributes
ALGSTS	0	Algorithm status word bit names and attributes
HT_STS	0	High-level status word bit names and attributes
LOGINT	0	DVL bottom track sensor interface names
LOGSND	0	Sound velocity sensor interface names
LOGKFM	0	DVL bottom track rejection filter mode names
LOGWTM	0	DVL water track sensor interface names
GPSINT	0	GNSS sensor interface names
GPSKFM	0	GNSS sensor rejection filter mode names
GP2INT	0	GNSS2 sensor interface names
GP2KFM	0	GNSS2 sensor rejection filter mode names
DEPINT	0	Depth sensor interface names
DEPKFM	0	Depth sensor rejection filter mode names
USBINT	0	USBL sensor interface names
USBKFM	0	USBL sensor rejection filter mode names
LBLINT	0	LBL sensor interface names
LBLKFM	0	LBL sensor rejection filter mode names
UTCINT	0	UTC synchronization interface names
START_	0	Starting mode selection strings
ZUP__	0	ZUPT mode selection strings
ALTMDE	0	Altitude stabilization mode selection strings
SERNUM	0	INS Serial number (i and j parameters not used)
VERFRM	0	INS firmware version (i and j parameters not used)
VERLDR	0	INS loader version (i and j parameters not used)
EQP__	0	INS type name (i and j parameters not used)
GENPRO	0	System configuration when using the protocol generator

8.5.2 SPECIFIC TEXT RETRIEVE COMMAND

To retrieve a specific text for a command:

Message	$PHTXT,list,c,i,j*hh<CR><LF>		
Title	Text list retrieve		
Data Field	**Semantics**	**Syntax**	**Type**
list	List name	Name of command associated with the list	string
c	Language	Only English 'E' is supported	char
i	Section index	Index of list to retrieve for this command	int
j	String index	Index of string in the list	int

Example:
To retrieve first input interface selection name, you should send:

$PHTXT,EDIRIX,0,0,E*0E<CR><LF>

Rovins will then answer:

$PHTXT,EDIRIX,0,0,Serial_only___*7F<CR><LF>

When no string is available, Rovins returns "________________" (16 x '_' character).
Thus, to retrieve all available input interface names, you should send $PHTXT,EDIRX command and increment string index until the firmware answers no string available. Following table details all string retrieve functions and their parameters:

Table 34 - List name table

List name	Section index	Retrieved list description
EDIRIX	0	Port input interface selection name
EDIROX	0	Port output interface selection name
ELCFIX	0	IP input mode selection name
ELCFOX	0	IP output mode selection name

iXblue CONTACT - SUPPORT

For non-URGENT support:

- by email: support@ixblue.com
- using the form on the iXblue web site www.ixblue.com

For 24/7 URGENT SUPPORT:

- North America / NORAM
  +1 617 861 4589
- Europe Middle-East Africa Latin-America / EMEA-LATAM
  +33 1 30 08 98 98
- Asia Pacific / APAC
  +65 6747 7027

A Appendix A: Quaternion Definitions

Quaternion definition and operations

General expression of a quaternion

if q is a real number and $\underline{V} = \begin{bmatrix} V_x \\ V_y \\ V_z \end{bmatrix}$ a vector of the three-dimensional Euclidian space,

then the quaternion $Q = q + \underline{V}$ can be defined as:

$$Q = q + \underline{V} = \begin{bmatrix} q \\ V_x \\ V_y \\ V_z \end{bmatrix}$$

Where q and $\underline{V}$ are respectively the scalar part and the vector part of the quaternion Q.

Conjugate of a quaternion:

If $Q = \begin{bmatrix} q0 \\ q1 \\ q2 \\ q3 \end{bmatrix}$ is a quaternion, the conjugate $\overline{Q}\ of\ Q$ is given by:

$$\overline{Q} = \begin{bmatrix} q_0 \\ -q_1 \\ -q_2 \\ -q_3 \end{bmatrix}$$

Product of two quaternions

If $P = \begin{bmatrix} p0 \\ p1 \\ p2 \\ p3 \end{bmatrix}$ and $Q = \begin{bmatrix} q0 \\ q1 \\ q2 \\ q3 \end{bmatrix}$ are two quaternions,

the product of P by Q, written $P.Q$ is given by:

$$P.Q = \begin{bmatrix} p0 \\ p1 \\ p2 \\ p3 \end{bmatrix} \cdot \begin{bmatrix} q0 \\ q1 \\ q2 \\ q3 \end{bmatrix} = \begin{bmatrix} p0q0 - p1q1 - p2q2 - p3q3 \\ p0q1 + p1q0 + p2q3 - p3q2 \\ p0q2 - p1q3 + p2q0 + p3q1 \\ p0q3 + p1q2 - p2q1 + p3q0 \end{bmatrix}$$

Conjugation of a vector V by a quaternion Q

$$Q.\underline{V}.\overline{Q} = \begin{bmatrix} q0 \\ q1 \\ q2 \\ q3 \end{bmatrix} . \begin{bmatrix} 0 \\ Vx \\ Vy \\ Vz \end{bmatrix} . \begin{bmatrix} q0 \\ -q1 \\ -q2 \\ -q3 \end{bmatrix} = \begin{bmatrix} 0 \\ (q0^2 + q1^2 - q2^2 - q3^2)Vx + 2(q1q2 - q0q3)Vy + 2(q1q3 + q0q2)Vz \\ 2(q1q2 + q0q3)Vx + (q0^2 - q1^2 + q2^2 - q3^2)Vy + 2(q2q3 - q0q1)Vz \\ 2(q1q3 - q0q2)Vx + 2(q2q3 + q0q1)Vy + (q0^2 - q1^2 - q2^2 + q3^2)Vz \end{bmatrix}$$

Attitude quaternion definition

Definition

Quaternion Q_{Geob} represents the rotation required to get from geographical frame to body frame:

For a vector V, V_b being its expression in body frame and V_{geo} its expression in geographical frame, the transformation from geographical frame to body frame is done through V_{geo} conjugation by quaternion Q_{Geob} :

$$V_b = Q_{Geob}.V_{geo}.\overline{Q_{Geob}}$$

Here we note «.» the usual product in quaternion space and $\overline{Q_{geob}}$ the conjugate of quaternion Q_{Geob}.

Relationship between attitude quaternion and Euler angles

- H is heading angle, positive from North to East (indirect)
- P is pitch angle, positive when bow gets down
- R is roll angle, positive when port side gets up

[A'] and [A''] are intermediate frames where:

- [A'] is the image of geographical frame by heading rotation (-H, XUP)
- [A''] is the image of [A'] by pitch rotation (P, YA')
- [v] is the image of [A''] by roll rotation (R, XA'')

To get from geographical frame to vessel frame, we successively apply:

- A heading rotation of -H degrees around X_{UP}, represented by quaternion:

$$Q_{A'Geo} = \left(\cos\left(\frac{H}{2}\right) - \sin\left(\frac{H}{2}\right) X_{UP}\right)$$

- A pitch rotation P, around axis $Y_{A'}$ represented by the quaternion:

$$Q_{A''A'} = \left(\cos\left(\frac{P}{2}\right) + \sin\left(\frac{P}{2}\right) Y_{A'}\right)$$

- Roll rotation R around axis $X_{A''}$ represented by quaternion:

$$Q_{vA''} = \left(\cos\left(\frac{R}{2}\right) + \sin\left(\frac{R}{2}\right) X_{A''}\right)$$

The global transition quaternion from geographical frame to vessel frame is thus the product:

$$Q_{Geov} = \left(\cos\left(\frac{R}{2}\right) - \sin\left(\frac{R}{2}\right) X_{A''}\right) . \left(\cos\left(\frac{P}{2}\right) - \sin\left(\frac{P}{2}\right) Y_{A'}\right) . \left(\cos\left(\frac{H}{2}\right) + \sin\left(\frac{H}{2}\right) X_{UP}\right)$$

Or:

$$Q_{Geov} = \begin{bmatrix} cos\left(\frac{R}{2}\right) \\ -sin\left(\frac{R}{2}\right) \\ 0 \\ 0 \end{bmatrix} . \begin{bmatrix} cos\left(\frac{P}{2}\right) \\ 0 \\ -sin\left(\frac{P}{2}\right) \\ 0 \end{bmatrix} . \begin{bmatrix} cos\left(\frac{H}{2}\right) \\ 0 \\ 0 \\ sin\left(\frac{H}{2}\right) \end{bmatrix}$$

That can be developed as:

$$Q_{Geov} = \begin{bmatrix} cos\left(\frac{R}{2}\right).cos\left(\frac{P}{2}\right)cos\left(\frac{H}{2}\right) - sin\left(\frac{R}{2}\right).sin\left(\frac{P}{2}\right).sin\left(\frac{H}{2}\right) \\ -sin\left(\frac{R}{2}\right).cos\left(\frac{P}{2}\right).cos\left(\frac{H}{2}\right) - cos\left(\frac{R}{2}\right).sin\left(\frac{P}{2}\right)sin\left(\frac{H}{2}\right) \\ sin\left(\frac{R}{2}\right).cos\left(\frac{P}{2}\right).sin\left(\frac{H}{2}\right) - cos\left(\frac{R}{2}\right).sin\left(\frac{P}{2}\right).cos\left(\frac{H}{2}\right) \\ cos\left(\frac{R}{2}\right).cos\left(\frac{P}{2}\right).sin\left(\frac{H}{2}\right) + sin\left(\frac{R}{2}\right).sin\left(\frac{P}{2}\right).cos\left(\frac{H}{2}\right) \end{bmatrix}$$

Standard deviation of attitude quaternion

Definition

Attitude quaternion Q_{Geov} can be written as follows: $Q_{Geov}=Q_{bv}Q_{nb}Q_{Geon}$

Introducing the error quaternions for Q_{vb} and Q_{Geon}, the above can be rewritten as follows:

$$Q_{Geov} = Q_{bv}\hat{Q}_{nb}\left(1-\frac{\eta}{2}\right)\hat{Q}_{Geon}\left(1-\frac{\gamma}{2}\right) \quad (0.1)$$

Introducing the error quaternion for Q_{Geov} , we can write:

$$Q_{Geov} = \hat{Q}_{Geov}\left(1-\frac{\xi}{2}\right)$$

$$Q_{Geov} = Q_{bv}\hat{Q}_{Geob}\left(1-\frac{\xi}{2}\right) \quad (0.2)$$

with ξ defined as the (small) rotation vector to get from actual [Geo] frame to erroneous [Geo] frame.

Equating Eq. (1.3) and Eq.(1.6) yields:

$$\hat{Q}_{Geob}\left(1-\frac{\xi}{2}\right) = \hat{Q}_{nb}\left(1-\frac{\eta}{2}\right)\hat{Q}_{Geon}\left(1-\frac{\gamma}{2}\right)$$

$$\left(1-\frac{\xi}{2}\right) = \hat{Q}_{nGeo}\left(1-\frac{\eta}{2}\right)\hat{Q}_{Geon}\left(1-\frac{\gamma}{2}\right)$$

Neglecting the second order terms, we can write:

$$\left(1-\frac{\underline{\xi}}{2}\right)=1-\frac{\underline{\gamma}}{2}-\hat{Q}_{nGeo}\frac{\underline{\eta}}{2}\hat{Q}_{Geon}$$

$$\underline{\xi}=\underline{\gamma}+\hat{Q}_{nGeo}\underline{\eta}\hat{Q}_{Geon}$$

Finally, we can write the covariance matrix on vector $\underline{\xi}$, i.e., the so-called "attitude quaternion covariance", as follows:

$$\boxed{\sigma^2\left(\underline{\xi}\right)=\sigma^2\left(\underline{\gamma}\right)+\hat{C}_n^{Geo}\sigma^2\left(\underline{\eta}\right)\left(\hat{C}_n^{Geo}\right)^T}$$

The standard deviations on vector $\underline{\xi}$ components are noted:

$$\sigma\left(\underline{\xi}_1\right)$$

$$\sigma\left(\underline{\xi}_2\right)$$

$$\sigma\left(\underline{\xi}_3\right)$$

Nota: $\sigma^2\left(\underline{\gamma}\right)$, $\hat{C}_n^{Geo}$ and $\sigma^2\left(\underline{\eta}\right)$ are computed by the navigation algorithm.

Expression of Euler angles covariance from error vector covariance

The covariance of vector $\underline{\eta}$ is related to the covariance of the roll, pitch and yaw variances as follows:

$$\sigma^2\begin{pmatrix}\delta R\\ \delta T\\ -\delta A\end{pmatrix}=F\sigma^2\begin{pmatrix}v_x\\ v_y\\ v_z\end{pmatrix}F^t$$

where

$$F=\begin{pmatrix}-\dfrac{\cos\hat{A}}{\cos\hat{T}} & \dfrac{\sin\hat{A}}{\cos\hat{T}} & 0\\ -\sin\hat{A} & -\cos\hat{A} & 0\\ -\tan\hat{T}\cos\hat{A} & \tan\hat{T}\sin\hat{A} & -1\end{pmatrix}$$