Software Design Specification

Lodestar Messages

1 Document Revision History

Rev.	Comment	Name:	Date:
1	Document compiled and released for third party use	AZC	22/07/2014

2 Table of Contents

3 Introduction

This document defines the messages (also known as telegrams or sentences) used to pass data between the Lodestar and other systems such as aiding sensors, DP systems or user interfaces.

Lodestar is designed to be extremely versatile, and is used in a wide range of applications, each with its own requirements. Due to this, a significant number of messages have been defined, many of which are specially designed to be compatible with a certain system or manufacturer. For example, there are over 25 different messages that contain heading data. It is intended that the user selects only the message or messages that they require for interfacing to their equipment. This might be one message with pitch, roll and heading, or a heading message and a pitch and roll message. The number of different messages and their output rates will be restricted by the bandwidth or baud rate available on the port.

4 Description

This document describes every message that Lodestar uses. The following information is detailed for each message:

- Name, including aliases used in commands
- Type e.g. industry standard / proprietary
- Encoding – ASCII or Binary
- Whether it is an input or an output
- A description of the message and what it is used for
- The format of the message
- An example message
- Details of each field in the message, including units
- Multiplex Message ID

The multiplex Message ID is used as part of the multiplex protocol (see section 8). When enabled, the messages are wrapped in a protocol so that binary and ASCII data can be mixed into one stream for debug and logging purposes.

5 General Information

5.1 Inputs, Outputs and Logs

Messages can either be sent to the lodestar or received from the lodestar, however there are two types of messages sent out by Lodestar and there are three different sources of data within the lodestar.

Outputs are messages generated by the lodestar regularly, at a user specified frequency. These messages contain data generated by one of the two algorithms that are running independently in the lodestar: The AHRS algorithm and the AINS algorithm. When configuring the Lodestar, the user can specify the frequency and source of output messages (see section 5.3 below).

For example, the NMEA *HDT* message contains heading data. Both the AHRS and the AINS algorithm calculate heading, using different methods, so the HDT message can be generated by either source. For some messages, only one source will be valid. If an invalid source is selected the output message will be populated with null or empty fields. See sections 12 to 16 for details of valid sources for each message.

In a multiplexed stream (see section 8), the source of a message can be determined from its Message ID.

Outputs can be sent to any port or to the SD card.

Logs are generated in the Lodestar by an event, so in contrast to *outputs*, the frequency can not be specified by the user. For example, every input message can be echoed back as a log – the Lodestar sends out a copy of the input message as soon as it is received.
Another example of a log message is the ALARM message. This is generated if a fault occurs or a parameter is outside acceptable limits. Obviously this message is not generated at a fixed frequency.
As with outputs, logs can be sent to any port or to the SD card.

Inputs are messages sent in to the lodestar from an external device such as a transceiver or GPS unit. These can be at any frequency. All inputs can be logged, i.e. repeated to an output port or the SD card.

5.2 Multiplexed Format
In order to transmit many different types of messages across a single link, Lodestar uses a multiplex message format. This mode may be enabled or disabled by the user. See section 8 for details.

5.3 Output Rates
The output rate for most messages is user selectable. The rate required will generally be as high as practical, particularly for the attitude. For most applications 25 Hz should be more than sufficient but otherwise, the maximum data rate is limited by the number of bytes in the messages and baud rate of the channel. The data is valid on transmission of the first character of each message. The Lodestar will indicate if too many messages at too high a repetition rate are requested will not accept configuration that will overload a port. The DSP load is also monitored on a low priority task.

5.4 Lever Arms & Remote Points
Each Instrument or sensor on a vehicle or vessel has a position and misalignment in the vehicle coordinate frame. The position is an X, Y, Z distance from the central reference point (CRP). This is also known as a *lever arm*. The Lodestar in general will also have a lever arm and misalignment to the vehicle frame, but of course can be set as zero to use the Lodestar as the reference point of the vehicle. The lever arms are distances from the CRP in the forward, starboard and down directions of the vehicle or vessel.

A number of additional remote reference points in the vessel coordinate frame can be configured in the Lodestar. These are configured using forward, starboard and down coordinates and angular offsets relative to the vessel CRP.

Lodestar can be configured to output messages that take these remote offsets into account by selecting a remote output point for each message. Lodestar can also accept input data from sensors located at a remote point, and can itself be located somewhere other than the vehicle CRP. Note that by default messages are calculated for the location of the CRP, not the location of the Lodestar.

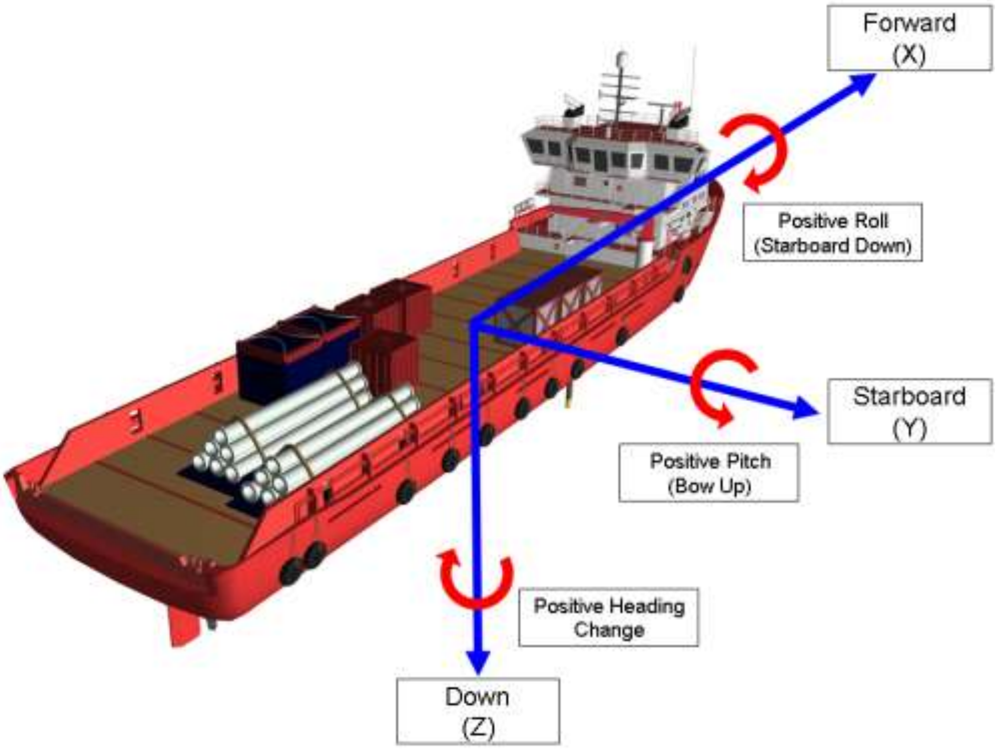
5.5 Coordinate System
The Lodestar x-axis points to the bow (forward) and the roll and pitch angles follow from that. Heave, surge and sway is in the local horizontal frame rotated by the heading. Unless otherwise specified (e.g for mounting angles), pitch (p) and roll (r) angles are what are known as Datawellian or Plumb Bob angles. They are referenced to the local horizontal (the Geoid) defined by the gravity vector.
If the tilt (τ) is the angle from the horizontal plane, then
$\sin^2(\tau) = \sin^2(p) + \sin^2(r)$

Figure 1 shows the definition of the coordinate frame used by Lodestar.

As the Lodestar can potentially be mounted upside down and back to front for ease of mechanical location, the mounting angles are given as Euler angles, which avoids singularities. These angles are quite different to the Datawellian or Plumb Bob angles generally used in marine surveying as they are with respect to the vehicle frame rather than the earth frame. The order of the angles is very important, as changing the order of rotations can result in completely different orientations.
The three angles, in order, from vehicle frame to output frame, are α, β, σ.

- α – Positive (anti-clockwise) rotation about the forward axis.
- β – Positive (anti-clockwise) rotation about the starboard axis.
- σ – Positive (anti-clockwise) rotation about the down axis.

Figure 1 - Lodestar Coordinate Frame



5.6 Invalid State Data

Unless otherwise specified, all messages will output zero (0) for data fields if in an invalid state, e.g.

- No attitude data available for output
- Output requested from an invalid source (e.g. INS output requested from a message that does not support INS output, or due to license restrictions)

5.7 SD Card Header Format

Data can be logged to the built-in SD card. Each SD card file has a header at the start, followed by the data packets. The header message is always packaged in the multiplex protocol, including a 48 bit SYS timestamp (see section 8.2). The multiplex message ID for the header is 244.

The header format is
`ddd,dddddd-ddd,d,yyyymmddhhmmss,d`

Field	Description
ddd	Build Number (ASCII, MS Digit first)
dddddd-ddd	IMU Serial Number (ASCII)
d	LOG Sequence Number (ASCII, MS Digit first)
yyyymmddhhmmss	UTC Date (ASCII)
d	Time System source (ASCII, Note 1)

Table 1 - SD Card Header Format

Note 1 – The source field can take the following values:

0 = No source
1 = RTC
2 = ZDA
3 = GGA
4 = ZDA_1PPS
5 = 1PPS

The header is then wrapped with the byte stuffing protocol described in section 8.1.

Example: `201,123456-789,4,20091028175049,2`

The example above contains the header without the byte stuffing protocol applied.

6 ASCII Messages

The messages are either in ASCII or binary format. The ASCII messages are all the messages or sentences that can be processed as text. This means that the messages are printable ASCII characters terminated with a carriage return and a linefeed character (`0x0D` & `0x0A` in hex). In this document the carriage return and a linefeed characters will be written as `<CR><LF>`.

6.1 Format

Many ASCII messages either use the TSS proprietary format or the NMEA 0183 & IEC-61162 standards.

TSS messages start with a colon ':' character and end with `<CR><LF>`. The messages are fixed length.

NMEA 0183 and IEC-61162 messages are more stringently defined, beginning with a dollar '$' or exclamation mark '!' character[1] and identifier string. The fields are comma separated and variable length and can be empty. The message ends with a star '*' character followed by a checksum. Although the length of a message is variable, the number of fields is not[2]. Commas are not omitted when a field is empty.

[1]Messages that start with '!' conform to special purpose encapsulation and non-delimited field composition rules. Please refer to the NMEA 0183 or IEC-61162 standard for more information.

6.2 NMEA talker codes

Standard NMEA sentences begin with a $ character, followed by two characters indicating the source of the message. Amongst others, these can be:

Magnetic Compass	HC
Gyro, North Seeking	HE
Gyro, Non-North Seeking	HN
Integrated Instrumentation	II
Integrated Navigation	IN

So according to the NMEA specification, it shall be HE for the Lodestar AHRS, and IN for the Lodestar INS.

6.3 NMEA Checksum

For NMEA messages there is a checksum field at the end of the message, before the <CR><LF> characters. The checksum field is the 8-bit exclusive OR (no start or stop bits) of all characters in the sentence, including ',' delimiters, between but not including the '$' and the '*' delimiters. The hexadecimal value of the most significant and least significant 4 bits of the result is converted to two ASCII characters (0-9,A-F) for transmission. The most significant character is transmitted first.

6.4 Valid Time

The messages do not generally specify a valid time. The time of validity of the data as expected by an external system will be on receipt of the first byte (e.g. '$' or ':'). If two different types of messages are output, the validity of the data in the message should be at the time of the first byte of the message. Some messages have UTC in a field, as a time of validity of the data.

6.5 Units

Units are SI units unless otherwise specified. All heading and attitude values are in decimal degrees unless otherwise specified. Sign convention depends on the message.

6.6 Packing Characters

For NMEA messages there are no spaces before or after values and there are no leading zeros, unless otherwise stated. The format "xx.xx" represents a variable number of significant figures and a variable number of decimals.
Numerical string values can have a sign character in front (+/-).

TSS fields are fixed length and may include a sign character, which is space ‘ ’ for positive, dash ‘-’ for negative. Some fields are separated by spaces and others are not. Please refer to the message descriptions in section0.

6.7 Decimal Places

Unless otherwise specified, the number of decimal places output for heading pitch and roll will be three decimal places. This is to avoid quantisation of the output, and is consistent with the binary output.

7 Binary Messages

7.1 Binary Data Formats

So that binary data can mixed with ASCII data on one serial port connection it is usually wrapped in the multiplex protocol (see section 8). Unless stated otherwise,

- Data is transmitted LSB (Least significant byte) first.
- Signed integers use 2’s complement representation
- Floating point numbers (float and double) use IEEE 754 representation

8 Multiplex Protocol

Lodestar supports communication of ASCII and binary messages via multiplexed serial and Ethernet links. Each message is given a unique ID and provided as the payload data within a *packet*, i.e. one packet contains one message. Separation of packets is achieved via unique packet start and end byte sequences and the “byte stuffing” technique.

8.1 Byte Stuffing

The scheme commonly used within Sonardyne for separation of packets is known as *byte stuffing*. The start and end of a packet is marked with unique ASCII control bytes sequences, namely DLE (0x10) & STX (0x02) at the start, and DLE & ETX (0x03) at the end. For every DLE byte found *within* the packet data an extra DLE byte is inserted (stuffed). This allows a parser of the resulting data stream to distinguish between bytes within a packet having the value DLE and the start/end of a packet.

Name	Byte	Description
`DLE`	`0x10`	Data Link Escape
`STX`	`0x02`	Start of Text
`ETX`	`0x03`	End of Text

Table 2 Start and end of packet characters

Example byte stuffed data (e.g. transmitted via serial link):
`0x65, DLE(0x10), DLE(0x10), ETX(0x03)`

Here the DLE / ETX pair does NOT indicate the end of a packet but rather the (original) byte sequence:
`0x65, 0x10, 0x03`

8.2 Packet Format

Each message is placed inside a ‘packet’ which consists of the fields shown below:

<table>
<tr><th>DLE STX</th><th colspan="4">ID</th><th>Timestamp (LSB first)</th><th>Payload (message data)</th><th>Checksum</th><th>DLE ETX</th></tr>
<tr><td rowspan="2"></td><td>TS</td><td>RES</td><td>SID [0-15]</td><td>MID [0-1023]</td><td rowspan="2"></td><td rowspan="2"></td><td rowspan="2"></td><td rowspan="2"></td></tr>
<tr><td>1 bit</td><td>1 bit</td><td>4 bits</td><td>10 bits (LSB)</td></tr>
<tr><td>2 bytes</td><td colspan="4">2 bytes</td><td>6 bytes</td><td>0 -2047bytes</td><td>1 byte</td><td>2 bytes</td></tr>
</table>

Table 3 Lodestar packet format for multiplexed messaging

The format is described below:

Name	Length	Description
DLE STX	2 bytes	Start of packet
TS	1 bit	If set indicates that the packet payload is pre-pended by 6 byte timestamp. If

		clear there is no packet timestamp
RES	1 bit	Reserved - any value to be expected
SID	4 bits	"Sender ID", "Source ID" or "System ID". Unless otherwise specified should be ignored by user or set to 0.
MID	10 bits	Message ID is used to identify the message type. Note that this is sent as big-endian i.e. the 2 MSBs are transmitted before the remaining 8 LSBs.
Timestamp	6 bytes	Packet timestamp using the Lodestar system time, i.e. time since power up or system reset, continues and is uncorrected, based on the internal crystal oscillator. LSB=1µS. Definition is message dependent and cannot be assumed to be time of validity of the message data. Only for Lodestar output/logging. SHOULD NOT BE PRESENT ON INPUT TO LODESTAR.
Data	0-2047 bytes	The byte-stuffed message itself
Checksum	1 byte	Exclusive-OR of the pre-byte-stuffed ID + data.
DLE ETX	2 bytes	End of packet

Table 4 Multiplexed format description

8.2.1 ID Field

The first 2 bytes after the DLE/STX identifies the message type (MID), the sender/source/system of the message (SID), a reserved bit (RES) and a bit indicating the existence of a packet time tag following the ID. The reserved bit can hold any value and should be set to 0 when sending to the Lodestar. The Source ID (SID) identifies the Sender/Sensor/System ID. Used only for advanced applications e.g. by Lodestar to output a navigation message for both vessel central reference point (CRP) and a remote point.

SID

The current use of the 4 bit SID depends on whether it is used in a message input to the Lodestar our output from the Lodestar.

For an output message, where applicable, the remote point index populates the 3 LSBs. Currently the MSB is unused – this may change soon and if so, this document will be updated.

For an input message the SID is generally 0000 but where necessary is used in the following way:
The 2 MSBs of the SID determine the source sensor type.
The 2 LSBs of the SID determine which number of the type is used.
For a GGA input which could be sourced from GPS or USBL, the 2 MSBs of the SID are: GPS = 00 and USBL = 01 while the 2 LSBs of the SID are GPS(main) = 00, GPS2 = 01; USBL(main) = 00, USBL2 = 01.
Currently, the GGA is the only input type which could contain a non-0000 SID.

Message ID

Every message type has a corresponding message ID. The MID for each message can be found in section 10. Note that MID = 255 can be used for any message that needs to be fed into the Lodestar e.g. for reasons of logging to its SD card.

The MID field is 6 bytes and is transmitted in big-endian form, i.e. the 2 most significant bits are transmitted before the 4 least significant bits. The code sample below demonstrates how to extract the message ID from a multiplex message

8.3 Code Example

A C code extract is shown below that removes the multiplex protocol, indicates when a complete message has been received and extracts the message ID. Note that this does not include checksum calculation or error handling. The code assumes that the received data is read into variable '`Databyte`' one byte at a time.

```
switch (Databyte){
    case 0x10:  //DLE
        if (DLEReceived == false){
            //first DLE - wait for the next DLE
            DLEReceived = true;
        }else{
            //DLE DLE found - copy one DLE to message buffer
```

```
            DLEReceived = false;
            message[datalength++] = Databyte;
        }
        break;

    case 0x02: //STX
        if (DLEReceived == true){
            // DLE STX - Start of the message found
            DLEReceived = false;

            DLE_STX_Found = true;
        }else{
            // Databyte is data
            message[datalength++] = Databyte;
        }
        break;

    case 0x03: //ETX
        if (DLEReceived == true){
            //DLE ETX found
            DLEReceived = false;
            if (DLE_STX_Found == true){
                  //DLE-STX/DLE-ETX pair found - complete message
                  MessageComplete = true;
            }
            DLE_STX_Found = false;
        }else{
            //Databyte is data
            message[datalength++] = Databyte;
        }
        break;

    default:    //Data
        if (DLEReceived == true){
            DLEReceived = false;
        }
        message[datalength++] = Databyte;
        break;
}

//Extract ID and checksum from complete message
if (MessageComplete == true)
{
    //get Source Id xx(2bits Source ID) (2bits Sensor ID)xx
    sourceId = (message[0] & 0x00) >> 4;

    //get Sensor Id
    sensorId = (message[0] & 0x3C) >> 2;

    //get Message ID
    mid = ((message[0] & 0x03) << 8) | (message[1]));

    //get chksum
    Chksum = message[datalength - 1];
}
```

8.4 Message Examples
Below are three examples of multiplexed messages showing each of the bytes transmitted with an explanation of the bits inside the ID field:

Message	DLE	STX	ID				DLE	ETX
Byte	1	2	3	4	5...10	11...(n-2)	n-1	n
1: Log CMD	0x10	0x02	0x82 10000010	0x00 00000000	(Timestamp)	(Data & checksum)	0x10	0x03
2: Log ZDA	0x10	0x02	0x80 10000000	0x3D 00111101	(Timestamp)	(Data & checksum)	0x10	0x03
3: Output NAV (RP 3)	0x10	0x02	0x0C 00001100	0xD5 11010101	(Data & checksum)		0x10	0x03
3: Log BIST	0x10	0x02	0x00 00000000	0xD9 11011001	(Data & checksum)		0x10	0x03

Table 5 Example multiplexed messages

Message 1 is a logged command, which has a timestamp pre-pended and has message ID 512.
Message 2 is a logged ZDA message, which has a timestamp pre-pended and has message ID 61.
Message 3 is a NAV output message with remote point 3 selected. As the message format already includes a timestamp, no additional timestamp is added. The message ID is 213.
Message 4 is a BIST message, which has message ID 217 and no additional timestamp.

8.5 ASCII Messages
ASCII messages transmitted or logged by Lodestar using the multiplexed protocol are generally and unless otherwise specified (message definition) pre pended by a 48 bit Lodestar system time tag as detailed previously. The meaning of the time tag (time of arrival / time of validity / other) is message type specific and may not be clearly defined. Lodestar users and system integrators should avoid depending on this time-tag and should instead make use of messages with appropriate (internal/payload) time tags. The packet time tags are intended for Lodestar diagnostics/support and system development and availability is subject to change.

9 Commands
Commands can be sent to the Lodestar on a correctly configured port. The Lodestar transmits a response to each command on the port that it was received. This response may consist of a number of messages:
1. An echo of the entered command. This may be disabled by the user.
2. Text containing the result if the command was a query. This may consist of multiple lines (and therefore multiple messages).
3. ‘`ok`’ if the command was executed successfully, or ‘`not ok`’ if the command failed.
4. Human readable error text indicating why a command failed may be generated in some cases.

9.1 Multiplexed Commands
If the multiplex protocol is enabled on a port then commands are sent to Lodestar using Message ID zero (0). All of the above responses are returned by the Lodestar via the port on which the command was received, also using MID 0.

9.2 Command Logging
Commands may also be logged (duplicated) on other outputs if required, however please note the following:
- If multiplexed then MID 512 is used for logging the command and the response instead of MID 0
- Only the command and the ‘ok’ / ’not ok’ response is logged. No other text associated with the response is logged.

The reason for using a different MID in the log is to avoid confusion when logging and commanding is carried out using the same port.

10 Message Summary
The table below lists all available messages, its associated multiplex message ID(s) and whether the message is an input, an output or a log message. Where the message name is followed by “,2”, the system can be configured to receive the same message type for a secondary sensor by appending a ‘2’ to

the end of the message name in the command. See command and control documentation for more information.

Category	Name	Also Known As	Input/ Output/ Log	AHRS MID	INS MID	notes
NMEA						
	ACK		I/L	91	91	
	ALARM		L	90	90	
	PRDDPT, 2		I/O/L		147	
	GGA	{GPS}, {SUSBL}	I/O/L	64	105	
	GST	{GPS}	I/L	76		
	TEMP		O	212		
	DBG, TXT		L	92	92	
	VTG	{GPS}	I/L	66		
	ZDA	{GPS}	I/L	61		
Proprietary Sonardyne ASCII						
	PRDSONDEPM, 2		I/L		145	
	PSONLBLBCN, 2	{PSONLBLBCN}	I/L		160	
	PSONLOBS	{PSONLBLBCN}	I/L		163	
	PSONLVR	{PSONLBLBCN}	I/L		161	
	PSONSS	{SVS}	I/L		146	
	COMMAND, CMD		I/L	0/512	0/512	IN COMMAND, LOG CMD
	SON2		O	120	120	
	TRG		L	110		
	SD HEADER			244		
Proprietary Other ASCII						
	SVS		I/L		143	
	PSIMSSB	{SUSBL}	I/L		152	
	PRDDIGIQM, 2		I/L		144	
	PRDDIGIQPSI, 2		I/L		158	
	PRDDIGIQKPA, 2		I/L		159	
Proprietary Sonardyne Binary						
	NAV		O		213	
	NAVQUAL		L		214	
	OBST[ZMD / VUSBL / GPSPOS / GPSVEL / SUSBL / XPOS / PDEPTH / SVS / DVL / LBL / ZUPT]		L		170-180	
	TMS		L	208		
	SETTINGS		L	216	216	
	BIST		L	217	217	
Proprietary Other Binary						
	PD4/PD5	{DVL}	I/L		140	
	PD0	{DVL}	I/L		141	

Table 6 - Message Summary

11 Message Details

The remainder of this document provides details of each message type that can be used by the Lodestar. The messages are grouped into the following categories:

ASCII Messages:

1. NMEA Messages – These are ASCII inputs or outputs that, unless otherwise stated, conform to the NMEA 0183 Standard for Interfacing Marine Electronic Devices.
2. Proprietary Sonardyne Messages – These messages are designed to provide commonly used data with the appropriate accuracy in the most efficient way possible (Note however, that Binary messages are inherently more efficient than ASCII). There are also a number of messages that are used to communicate between specific Sonardyne equipment and systems, or for logging purposes.
3. Other proprietary messages – These messages use formats that are specific to other manufacturers or devices. This allows easy integration into systems designed to use only those devices.

Binary Messages:

1. 'Industry Standard' messages – These messages are a proprietary format that has become widely used by many manufacturers in the industry
2. Proprietary Sonardyne messages – These messages can provide a significant amount of information in a much smaller data package than the Sonardyne ASCII messages. Other than this they have the same purpose.
3. Other proprietary messages – These binary messages use formats that are specific to other manufacturers or devices. This allows easy integration into systems designed to use only those devices.

12 ASCII – NMEA messages

Lodestar supports a number of standard messages defined by the National Maritime Electronics Association (NMEA). The details of these messages can be found in the NMEA 0183 V4.00 specification or in IEC 61162-1 edition 4. The table below lists the NMEA messages used by Lodestar. Note that some of these messages can be referred to by the 'GPS' message name. See section 6 for general information on NMEA messages.

Name	Also known as	Input / Output / Log	Multiplex Message ID	Talker Code
ACK		Input	91	HE
ALR	ALARM	Log	90	HE
DPT	PRDDPT	Input	147	Any
GGA		Input	64	Any
GST		Input	76	Any
TXT	DBG, TEMP, BIST	Log / Output	92 (Log) 212 (Temp) 217 (BIST)	HE
VTG		Input	66	Any
ZDA		Input	61	Any

Table 7 Supported NMEA 0182 / IEC61162-1 messages

12.1 Notes

ALR

This sentence is used to report an alarm condition on a device and its current state of acknowledgement. For example, the Lodestar will indicate if the external power supply has failed and the unit is running under the backup battery. When the external power is restored, an ALR report will also be sent.

TXT

- (Log) TXT – This contains information following an alarm condition. This sentence is used to provide more detailed information on the cause of an alarm condition reported by a device
- (Log) DBG – If the debug log is enabled then messages will be generated containing debug information using the NMEA 'TXT' message format. These messages have text identifier number 90 and multiplex message ID 92.
- (Log) BIST – If the BIST log is enabled on a non-multiplexed port then TXT messages will be generated containing a hex-ASCII representation of the status information. See section 15.1 for more information.
- (Output) TEMP - This message provides the temperature of the x, y and z accelerometer sensors and the x, y and z cases using the NMEA 'TXT' message format. These messages have a text identifier number 66 and multiplex message ID 212. The format of the text message is shown below

```
$HETXT,1,1,66,dd.d,dd.d,dd.d,dd.d,dd.d,dd.d*hh<cr><lf>
```

Field	Description
$HETXT	Header
1	Total number of messages
1	Message number
66	Text identifier
dd.d	X sensor temperature, degrees Celsius
dd.d	Y sensor temperature, degrees Celsius
dd.d	Z sensor temperature, degrees Celsius
dd.d	X case temperature, degrees Celsius
dd.d	Y case temperature, degrees Celsius
dd.d	Z case temperature, degrees Celsius
*hh	checksum
<CR><LF>	return plus linefeed

Table 8 TEMP message format

13 ASCII – Proprietary Sonardyne Messages

These messages have been designed by Sonardyne to provide a simple way of transmitting basic data from Lodestar. Note that unless stated otherwise, checksums are calculated in the same way as for NMEA 0183 messages, i.e. an XOR of all characters between the $ and *, represented in hexadecimal ASCII format.

13.1 Command

Name	Command
Also Known As	CMD
Type	ASCII – Sonardyne
Input / Output / Log	Input / Log
Length	Variable
Delimiter	None
Log Frequency	Generated whenever a command is received on a command port
Multiplex Message ID	0 (Input / Response) 512 (Log)

13.1.1 Description

Please see section 9 for an explanation of the command protocol, and SDS Command & Control.doc for full details of commands and syntax.

On a non-multiplexed link, the commands are in plain ASCII, terminated by a carriage return and line feed. On a multiplexed link, the message ID is either 0 or 512, as shown in the table below.

Message name	Multiplex Message ID	Description
COMMAND	0	ASCII strings used to configure and control the Lodestar. Responses on the commanding port also have message ID 0
CMD	512	Logged commands. Can be output from any port.

13.2 PSONDEPM

Name	PRDSONDEPM
Also Known As	PSONDEP
Type	ASCII – Sonardyne
Input / Output / Log	Input
Length	Variable
Delimiter	Comma
Multiplex Message ID	145

13.2.1 Description

The purpose of this proprietary string is to support depth input into Sonardyne software and instruments from a non-specific source. This string is already used in several subsea positioning applications with both Ranger and Fusion software.

13.2.2 Format

```
$PSONDEP,x.xx,y.y,c*hh<cr><lf>
```

13.2.3 Fields

Field	Description	Units
`$PSONDEP`	Start character and format identifier	
`x.xx`	Depth	Defined by units field
`y.y`	Observation error (note 1)	
`c`	Units, M = metres (note 2)	
`*hh`	Terminator and checksum	

<cr><lf>	Return plus linefeed	

13.2.4 Example
`$PSONDEP,2001.63,,M*1A <cr><lf>`

13.2.5 Notes

Note 1 – Observation error has no standard/common convention; it is dependent on transmitting system (not recommended for use).

Note 2 – Supported units are M (metres)

13.3 PSONBCN

Name	PSONBCN
Also Known As	PSONLBLBCN
Type	ASCII – Sonardyne
Input / Output / Log	Input
Length	Variable
Delimiter	Comma
Multiplex Message ID	160

13.3.1 Description
This proprietary string contains the real world position of a calibrated (fixed) beacon. It is sent to Lodestar along with an associated $PSONLVR and $PSONUOBS message. These messages provide acoustic aiding to the INS algorithm.

13.3.2 Format
`$PSONBCN,tttttttt.tttttt,ddd,d.ddd,d.ddd,d.ddd,d.ddd,ddddd,dd.d,d.d*hh<cr><lf>`

13.3.3 Fields

Field	Description	Units
$PSONBCN	Start character and format identifier	
tttttttt.tttttt	UTC / INS timestamp (note 1)	Seconds
ddd	Beacon address	
d.ddd	Latitude	Degrees
d.ddd	Longitude	Degrees
d.ddd	Depth	Metres
d.ddd	TAT (Turn Around Time)	ms
ddddd	Carrier frequency	Hz
dd.d	Horizontal error radius	Metres
d.d	Depth error	Metres
*hh	Terminator and checksum	
<cr><lf>	Return plus linefeed	

13.3.4 Example
```
$PSONBCN,922.672222,2306,28.2236437,-88.5303721,1693.373,200.000,25500,0.0,
0.0*59<cr><lf>
```

13.3.5 Notes

Note 1 – Time stamp should be identical to a corresponding acoustic observation (PSONLOBS message) but this is not compulsory. May be omitted (empty) or set to zero.

- The timestamp field can contain up to 6 decimal places for microsecond resolution
- The Latitude and Longitude fields can contain up to 7 decimal places for a resolution of approximately 1cm at the Equator.
- The $PSONLVR and $PSONBCN messages will normally be expected before the $PSONUOBS message. If either the $PSONLVR or $PSONBCN message is missing then the preceding

message will be used. If the most recent PSONLVR or PSONBCN is older than 180s then the observation is not used.

13.4 PSONLOBS

Name	PSONLOBS
Type	ASCII – Sonardyne
Input / Output / Log	Input
Length	Variable
Delimiter	Comma
Multiplex Message ID	163

13.4.1 Description

This proprietary string is used for inputting LBL observations to Lodestar. It is sent to Lodestar along with an associated $PSONLVR and $PSONBCN message. These messages provide acoustic aiding to the INS algorithm.

13.4.2 Format

```
$PSONLOBS,ttttttt.ttttt,ddd,ddd.ddd,dddd.ddd,dddd.ddd,dd.d,dd.d,d,c*hh<cr><lf>
```

13.4.3 Fields

Field	Description	Units
`$PSONLOBS`	Start character and format identifier	
`ttttttt.ttttt`	UTC / INS system timestamp (Note 1)	Seconds
`ddd`	Beacon address	
`ddd.ddd`	Travel time (Note 2)	µSecond
`dddd.ddd`	Sound speed at beacon, as determined by acoustic system	m/s
`dddd.ddd`	Sound speed for range, as determined/used by acoustic system	m/s
`dd.d`	Signal to noise	dB
`dd.d`	Signal level (5G: dB relative to 1µPascal at 1 metre) (6G: dB relative to full scale voltage [DBV-12])	dB
`d`	Cross correlation (6G only, not populated for 5G) (Note 3)	
`c`	Status flag (A = accept, V = invalid) (Note 4)	
`*hh`	Terminator and checksum	
`<cr><lf>`	Return plus linefeed	

13.4.4 Example

```
$PSONLOBS,-39201.186643,1706,444750.000,1485.000,1485.000,71.0,-2.0,89.0,A*57
<cr><lf>
```

13.4.5 Notes

Note 1 – Time stamp is time of transceiver reception of the acoustic response signal (beginning). This can be expressed in the INS system time base, or in UTC. UTC timestamp is number of seconds since midnight with a negative sign to indicate UTC rather than Lodestar system time.
e.g. 10:53:21.186643UTC should be sent as '-39201.186643'.

Note 2 – Two way travel time including TAT

Note 3 – Depending on the implementation the cross correlation may or may not be present

Note 4 – The status flag can take the following values:
A: Acoustic system suggests observation is OK
V: Acoustic system suggests observation is bad and should not be used.

- The $PSONLVR and $PSONBCN messages will normally be expected before the $PSONLOBS message. If either the $PSONLVR or $PSONBCN message is missing then the preceding message will be used. If the most recent PSONLVR or PSONBCN is older than 180s then the observation is not used.

13.5 PSONLVR

Name	PSONLVR
Type	ASCII – Sonardyne
Input / Output / Log	Input
Length	Variable
Delimiter	Comma
Multiplex Message ID	161

13.5.1 Description

This proprietary message provides information about lever arms from the acoustic system to Lodestar. It is sent to Lodestar along with an associated $PSONLVR message. These messages provide acoustic aiding to the INS algorithm.

13.5.2 Format

```
$PSONLVR,tttttttt.tttttt,d.dd,d.dd,d.dd,d.dd,d.dd,d.dd,d.dd,d.dd,d.dd,d.dd,d.dd,d.dd,d.dd,d.dd*hh<cr><lf>
```

13.5.3 Fields

Field	Description	Units
`$PSONLVR`	Start character and format identifier	
`tttttttt.tttttt`	UTC / INS Timestamp (Note 1)	Seconds
`d.dd`	Transceiver pitch correction (Note 2)	Degrees
`d.dd`	Transceiver roll correction (Note 2)	Degrees
`d.dd`	Transceiver heading correction (Note 2)	Degrees
`d.dd`	Transceiver distance starboard of CRP	Metres
`d.dd`	Transceiver distance forward of CRP	Metres
`d.dd`	Transceiver distance below CRP	Metres
`d.dd`	Depth of CRP (Note)	Metres
`d.dd`	GPS antenna distance starboard of CRP	Metres
`d.dd`	GPS antenna distance forward of CRP	Metres
`d.dd`	GPS antenna distance below CRP	Metres
`d.dd`	IMU distance starboard of CRP	Metres
`d.dd`	IMU distance forward of CRP	Metres
`d.dd`	IMU distance below CRP	Metres
`d.dd`	IMU mounting angle alpha (Note 3)	Degrees
`d.dd`	IMU mounting angle beta (Note 3)	Degrees
`d.dd`	IMU mounting angle gamma (Note 3)	Degrees
`*hh`	Terminator and checksum	
`<cr><lf>`	Return plus linefeed	

13.5.4 Example

```
$PSONLVR,1798.772679,,,,-16.740,15.770,14.754,0.0,-2.390,1.700,-116.600,-16.740,15.770,14.546,0.129,-0.308,3.725*4B<cr><lf>
```

13.5.5 Notes

Note 1 – Time stamp should be identical to a corresponding acoustic observation (PSONLOBS message) but this is not compulsory. May be omitted (empty) or set to zero.

Note 2 – Do not populate transceiver orientation fields for LBL operation.

Note 3 – Angles alpha, beta and gamma express the Euler rotation angle sequence from INS vessel frame into IMU frame.

- Fields can be null (empty) if unknown. If data is present (including zeros), the values configured in Lodestar will be overwritten.

- See "SDS Kalman.doc" for exact definition of direction cosine matrix from mounting angles.
- The $PSONLVR and $PSONBCN messages will normally be expected before the $PSONUOBS message. If either the $PSONLVR or $PSONBCN message is missing then the preceding message will be used. If the most recent PSONLVR or PSONBCN is older than 180s then the observation is not used.
- If the message is generated by an LBL system then it is referred to as PSONLBLBCN and uses message ID 161

13.6 PSONSS

Name	PSONSS
Also Known As	SVS
Type	ASCII – Sonardyne
Input / Output / Log	Input
Length	Variable
Delimiter	Comma
Multiplex Message ID	146

13.6.1 Description

The purpose of this proprietary string is to support sound speed input into Sonardyne software and instruments from a non-specific source.

13.6.2 Format

```
$PSONSS,x.x,y.y,c*hh<cr><lf>
```

13.6.3 Fields

Field	Description	Units
`$PSONSS`	Start character and format identifier	
`x.x`	Depth (Note 1)	Units
`y.y`	Sound Speed	Units per second
`c`	Units (Note 2)	
`*hh`	Terminator and checksum	
`<cr><lf>`	Return plus linefeed	

13.6.4 Example

```
$PSONSS,1991.00,1502.00,M*65<cr><lf>
```

13.6.5 Notes

Note 1 – The depth information is ignored and can be a null field.

Note 2 – Supported units are M (metres) or F (US survey feet).

- In order to accept this message, the Lodestar should be configured to receive the message type 'SVS' on the desired port. The SVS type must then be configured to PSONSS. See SDS Lodestar Command & Control.doc for more information. See also VALEPORT message.

13.7 PSONTMS

Name	PSONTMS
Type	ASCII – Sonardyne
Input / Output / Log	Output
Source	AHRS
Length	Variable
Delimiter	Comma
Multiplex Message ID	129

13.7.1 Description

The purpose of this proprietary string is to export the synchronous System Time with the Lodestar's estimate of UTC time. Both time values are represented as floating point decimal string values, to microsecond resolution.

13.7.2 Format

`$PSONTMS,tttttttt.tttttt,ssssssss.ssssss,d,S*hh<cr><lf>`

13.7.3 Fields

Field	Description	Units
`$PSONTMS`	Start character and format identifier	
`tttttttt.tttttt`	INS System Timestamp, seconds from start up. µs resolution	Seconds
`ssssssss.ssssss`	Coordinated Universal Time (UTC), timestamp. The number of seconds elapsed since midnight 1970-01-01 not counting UTC leap seconds. µs resolution.	Seconds
`d`	Source of most recent UTC sync (Note 1)	
`S`	Status flag (A = accept, V = invalid) (Note 2)	
`*hh`	Terminator and checksum	
`<cr><lf>`	Return plus linefeed	

13.7.4 Example

`$PSONTMS,983.010838,1384511829.802214,4,A*08<cr><lf>`

13.7.5 Notes

Note 1 – The source of the most recent RTC to UTC update can be one of the following:

0 = No Source
1 = Lodestar RTC
2 = Standalone ZDA
3 = Standalone GGA
4 = ZDA & 1PPS

Note 2 – The status flag will be 'A' if the selected reference for UTC has been available and accepted within the past k1 (60) seconds OR the time system expected accuracy of UTC is < k2 (5ms). For example, if using and losing ZDA/1PPS then status flag will continue to be 'A' for ~1000 seconds. Otherwise the status flag will be 'V'.

- Note that the selected reference for UTC is the currently selected source, which is not necessarily the source which was most recently used to sync to UTC.
- Lodestar follows the Posix convention for expression of UTC time as a single number: The number of seconds elapsed since midnight 1970-01-01 not counting UTC leap seconds.
- The system mode values are the same as the binary Time System Data message (MID = 211)

13.8 TRG

Name	TRG
Also Known As	PSONTRG
Type	ASCII – Sonardyne
Input / Output / Log	Log
Source	AHRS
Length	Variable
Delimiter	Comma
Frequency	Sent on every trigger pulse detected or generated
Multiplex Message ID	110

13.8.1 Description

The purpose of this proprietary string is to record the 1 PPS input or any other trigger input or output such as DVLs and responders, and to log the trigger parameters.

13.8.2 Format

`$PSONTRG,tttttttttttt,hhmmss.ssssss,d,c,c,tttttttt,tttttttt*hh <CR><LF>`

13.8.3 Fields

Field	Description	Units
`$PSONTRG`	Start character and format identifier	
`tttttttttttt`	Trigger time in Lodestar System Time, seconds from start up, resolution to µs. 12 hex digits.	Seconds
`hhmmss.ssssss`	Trigger time converted to Lodestar Time System, seconds from start up, resolution to µs.	Seconds
`d`	Trigger port number (1 to 4)	
`c`	Trigger direction, (A = input, B = output)	
`c`	Sign of the trigger edge, “+” or “-“	
`tttttttt`	Width of the trigger pulse, resolution to µs. 8 hex digits.	Seconds
`tttttttt`	Period of the output pulse, resolution to µs. 8 hex digits.	Seconds
`*hh`	Terminator and checksum	
`<cr><lf>`	Return plus linefeed	

13.8.4 Example

`$PSONTRG,00003FE06FAE,094020.500365,4,B,+,0000C350,000F4240*63<cr><lf>`

13.8.5 Notes

- Lodestar follows the Posix convention for expression of UTC time as a single number: The number of seconds elapsed since midnight 1970-01-01 not counting UTC leap seconds.
- The time parameter has been included for compatibility with other Sonardyne systems. It duplicates the record time tag in the multiplexed message stream as the message may be output directly on a serial port.
- The width and period parameters are null fields if the trigger is an input.

13.9 SON2

Name	SON2
Type	ASCII – Sonardyne
Input / Output / Log	Output
Source	AHRS
Length	Fixed, 38
Delimiter	None
Multiplex Message ID	120

13.9.1 Description

This Sonardyne proprietary string outputs heave, surge, sway, pitch, roll and heading in a fixed field format.

13.9.2 Format

```
:hhmmssssMRRRRRRMPPPPPPMHHHHHHMVVVS<cr><lf>
```

13.9.3 Fields

Field	Description	Units
:	Start character	
hhmmsssss	UTC time, hours, minutes, seconds and milliseconds	
M	Space if positive, minus if negative	
RRRRRR	Roll	0.001 Degrees
M	Space if positive, minus if negative	
PPPPPP	Pitch	0.001 Degrees
M	Space if positive, minus if negative	
HHHHHH	Heading	0.001 Degrees
M	Space	
VVV	Estimated variance	
S	Status flag, U, u, A, a, V, v, G or g	
<cr><lf>	Terminator, return plus linefeed	

13.9.4 Example

```
:152424103-001141 002279 010189 002U
```

13.9.5 Notes

- Positive roll is starboard down, port up.
- Positive pitch is bow up, stern down.
- The angle measurements are in thousandths (i.e. x 1000 degrees)
- The motion measurements contained in the data string will be in real time, valid for the instant when the System begins to transmit the string.
- Motion measurements include ASCII-coded decimal values.
- Roll and pitch measurements are in degrees in the range –90° to +90°.
- The Status flag can take one of the following values:
    - A or a: Both VTG and GGA aiding are present
    - V or v: VTG only
    - G or g: GGA only
    - U or u: Neither VTG nor GGA are available.

  Upper case indicates that the gyrocompass has settled. Lower case indicates that the gyrocompass is settling.
- If outputting u or U status, as soon as a VTG and/or GGA is received the status changes appropriately. However if VTG and/or GGA is not seen, it takes 5 seconds for the new (lesser) status to be updated on the message.

14 ASCII – Other Proprietary Messages

These messages are proprietary messages defined by third parties.

14.1 DigiQuartz Pressure Sensor Data Formats (PRDDIGIQ...)

Name	PRDDIGIQ[KPA/M/PSI]
Type	ASCII – Proprietary
Input / Output / Log	Input
Length	Variable
Delimiter	None
Multiplex Message ID	159/144/158

14.1.1 Description

These messages contain data from a Paroscientific Digiquartz depth sensor. The units cannot be determined from the message data so must be specified by using the relevant message name, i.e. if the sensor is configured to output depth in kPa then the Lodestar should be configured to receive the PRDDIGIQKPA message. The supported units and their respective message names are listed below:

Message Name	Multiplex Message ID	Units	Notes
PRDDIGIQKPA	159	kPa	psi × 6.894757
PRDDIGIQM	144	Metres of H_2O	psi × 0.7030696
PRDDIGIQPSI	158	psi	psi × 1.0

Table 9 supported DigiQuartz messages

14.1.2 Format

`*ddssx.xx<cr><lf>`

14.1.3 Fields

Field	Description	Units
`*`	Start character	
`dd`	Destination address, always ‘00’	
`ss`	Source address, ‘01’ unless depth sensor is configured with a different address.	
`x.xx`	Pressure	Specified by message name
`<cr><lf>`	Terminator, return plus linefeed	

14.1.4 Example

`*000114.573<cr><lf>`

14.1.5 Notes

- Sensor is the Series 8000 intelligent sensor. Specific model used: 8CB4000-I-505.
- The sensor should be configured to continuous measurement mode (P4). This can be set by sending the command `*0100P4<cr><lf>`
- See also PRDDIGIQO, which is a binary pressure message format.

14.2 PSIMSSB

Name	PSIMSSB
Type	ASCII – Proprietary
Input / Output / Log	Input
Length	Variable
Delimiter	Comma
Multiplex Message ID	152

14.2.1 Description

This proprietary Kongsberg / Simrad message format contains the position of a SSBL transponder. It is preceded by an associated USBL measurement.

14.2.2 Format
`$PSIMSSB,hhmmss.ss,ccc,a,ccc,a,a,a,x.x,x.x,x.x,x.x,a,x.x,x.x*hh<cr><lf>`

14.2.3 Fields

Field	Description	Units
$PSIMSSB	Start character and format identifier	
hhmmss.ss	Empty or UTC time of reception	
ccc	Transponder code, e.g. B01, B33, B47 (Note 1)	
a	Status, A = OK, V = Invalid (Note 2)	
ccc	Empty or 3 character error code (Note 3)	
a	Co-ordinate system, C = Cartesian, P = Polar, U = UTM, R = Radians (Note 4)	
a	Orientation, H = vessel heading up, N = North, E = East (Note 4)	
a	Software filter, M = Measured, F = Filtered, P = Predicted	
x.x	X co-ordinate (Note 4, 5)	Defined by co-ordinate system field
x.x	Y co-ordinate (Note 4, 5)	Defined by co-ordinate system field
x.x	Depth (Note 5)	Metres
x.x	Expected accuracy of the position (Note 6)	Defined by co-ordinate system field
a	Additional information, N,C,I,D or T (Note 7)	
x.x	First additional value (may be empty) (Note 7)	Defined by additional information field
x.x	Second additional value (may be empty) (Note 7)	Defined by additional information field
*hh	Terminator and checksum	
<cr><lf>	Return plus linefeed	

14.2.4 Example
`$PSIMSSB,091430.22,B18,A,,U,E,M,217682.28,626751.82,131.88,0.81,N,,*7E`

14.2.5 Notes

Note 1 – The measurements in the message relate to the transponder with the three character ASCII code contained in the ‘transponder code’ field. The characters are the same as the ones used on the HPR display and in the HPR operator manual.

Note 2 – The status field is ‘A’ when the measured position is OK and ‘V’ when the position is not OK or is missing. The ‘error code’ field provides further quality or error information.

Note 3 – The possible error codes are shown in the table below:

Code	Explanation	Implication	Status Field
NRy	No reply is received.	No position is calculated	V
AmX	Ambiguity error in the X direction.	No position is calculated.	V
AmY	Ambiguity error in the Y direction.	No position is calculated.	V
Rej	The position is measured OK, but rejected by the SW filter in either the transceiver or in the HSC 400.		A
Mi2	The second pulse of the transponder reply is missing.	Both ‘additional info’ fields are empty. Status Field is V if sensor type is depth.	A/V
Mi3	The third pulse of the transponder reply is missing.	Second ‘additional info’ field is empty.	A
Pre	No position is measured. The position is predicted by the Kalman filter in the operator station.		A

VRU	The VRU connected to the system has reported error.	Position not calculated with correct roll & pitch	V
GYR	The gyro connected to the system has reported error.	Position not calculated with correct heading.	V
ATT	Attitude sensor (VRU or Gyro) connected to the system has reported error.	Position not calculated with correct heading, roll and pitch.	V
ExD	External depth used in calculation of position.	Position OK.	A
ExM	External depth wanted but not received.	Position calculated without locked depth.	V
???	The system has reported an unknown error.		V

Table 10 PSIMSSB error codes

Note 4 – The contents of the X, Y and Depth co-ordinate fields are controlled by settings in the USBL system. The 'Co-ordinate system' and 'Orientation' fields should be used to decode the X and Y fields according to the table below:

USBL System settings	PSIMSSB fields		PSIMSSB coordinates of transponder position	
CO-ORD	**Coordinate system**	**Orientation**	**X co-ordinate**	**Y co-ordinate**
Polar	P	H	Horizontal range	Bearing in degrees
Cartesian X/Y	C	H	Starboard	Forwards
Cartesian N/E	C	N	North	East
Cartesian E/N	C	E	East	North
UTM N/E	U	N	Northings	Eastings
UTM E/N	U	E	Eastings	Northings
Radians	R	N	Latitude	Longitude

Table 11 PSIMSSB co-ordinate systems

The Polar and Cartesian X/Y coordinates are the position of the transponder relative to the vessel. They are in metres, except the Bearing which is an angle between 0 to 360 degrees.
The Eastings and the Northings are the UTM coordinates of the transponder. The Latitude and the Longitude 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 better than 0.01m.

The geographic co-ordinates are in the USBL system presentation datum.

Note 5 – When no position is calculated, the position fields (X, Y and Depth) are all empty.

Note 6 – The expected accuracy of the position is based on the covariance data calculated for each position. It is equal to the statistical sum of the Major and Minor semi-axes of the error ellipse displayed around the position.

Note 7 – The Additional information field indicates what additional sensor data has been retrieved from the transponder. The 'First additional value' and 'Second additional value' fields are filled with this data. The table below lists the types of data that can be included and the corresponding character that is used in the 'Additional information' field. If the transponder is in beacon mode then no additional data will be retrieved and the additional value fields will be empty.

Additional informat	First addition	Second additional	Description

ion	al value	value	
C	Bearing		Bearing in degrees
I	X inclination	Y inclination	Inclination values in degrees, also used for diffinclination.
D	Depth		Depth sensor reading in metres
T	Time		Time from transponder to transducer in seconds
N			Empty fields, used when transponder in beacon mode.

Table 12 PSIMSSB additional information fields

When using this message in INS (Lodestar & Janus):

- All USBL filtering, including 'depth aiding' should be disabled
- The PSIMSSB message will only be recognised by the Lodestar parser if:
    - The number of fields is 15
    - The checksum is correct
    - The Orientation is 'H', 'N' or 'E'
    - All of the X, Y, Depth and Expected accuracy fields are present
- The PSIMSSB message will only be used for INS aiding if:
    - The system time is good
    - The status field is 'A'
    - The Co-ordinate system is 'R'
    - The Software filter is 'M'
- The time of validity (TOV) for the PSIMSSB message is computed as follows:
    - If the UTC field is populated then it is converted to TOV
    - If the UTC field is not populated then TOV is taken to be the time of arrival (TOA)
    - If there is a 'Time' value in the additional information then this is subtracted from the (above) TOV.

14.3 VALEPORT

Name	VALEPORT
Also Known As	SVS
Type	ASCII – Proprietary
Input / Output / Log	Input
Length	Variable
Delimiter	None
Multiplex Message ID	143

14.3.1 Description

This string outputs sound velocity in metres per second from the Valeport Mini SVS.

14.3.2 Format

`<space>x.x<cr><lf>`

14.3.3 Fields

Field	Description	Units
<space>	Start character ' '	
x.x	Sound velocity	m/s
<cr><lf>	Terminator, return plus linefeed	

14.3.4 Example

` 1562.331<cr><lf>`

14.3.5 Notes

- In order to accept this message, the Lodestar should be configured to receive the message type 'SVS' on the desired port. The SVS type must then be configured to VALEPORT. See SDS Lodestar Command & Control.doc for more information. See also PSONSS message.

15 Binary – Proprietary Sonardyne Messages

Binary data formats make more efficient use of the communications bandwidth than ASCII messages. This makes them the preferred choice for high frequency outputs or detailed information.

In order to use binary messages, the port must be configured for multiplexed communications. See section 8 for details of the multiplex protocol.

15.1 BIST

Name	BIST
Type	Binary – Sonardyne
Input / Output / Log	Log
Log Frequency	0.5 Hz (every 2 seconds)
Length	Fixed, 48 bytes
Multiplex Message ID	217

15.1.1 Description

The Built-In Self Test (BIST) message provides a summary of errors detected within a BIST cycle (2 seconds) and additional LS state information. Unless explicitly stated, an error indication is NOT persisted across cycles if the underlying cause has cleared or error event has not re-occurred.

BIST is generic, supporting all LS AHRS/INS applications. Each application will use only the relevant subset of the information provided as defined in the application specific ICD. Where possible, BIST information is accumulated into high level “go/no go” bits, thus simplifying the interface/use and reducing BIST change induced maintenance.

15.1.2 Format

The BIST message is logically subdivided into 5 bit-blocks as shown below:

Byte#	Field name	Unit(LSB)	Data type	Note
1-6 / 6	Time Tag	10^{-6} sec	Uint48	System time
7-14 / 8	Firmware Version		Uint64	FW Version: Major, Minor, Interim, Build. 16 bits each, Build number and LSB first.
15-22 / 8	IMU		Bits (x64)	
23-30 / 8	Comms		Bits (x64)	
31-38 / 8	CCA		Bits (x64)	Circuit Card Assembly
39-40 / 2	AHRS		Bits (x16)	
41-48 / 8	AINS		Bits (x64)	

Table 13 BIST message blocks

Each of the blocks are described in the tables below.

IMU bit#	Field name	Note / bit set
0	bNoGo	Serious hardware issue (no go) (Note 1)
1	bISANotOk	ISA / IMU data not ok
2	bFWNotStarted	Firmware not started
3	bGyroPwrNotOk	Gyro power not ok
4	bXGyroProblem	X gyro problem – tripped or sensor status not ‘Normal’
5	bYGyroProblem	Y gyro problem – tripped or sensor status not ‘Normal’
6	bZGyroProblem	Z gyro problem – tripped or sensor status not ‘Normal’
7	bExtPwrNotOk	External power supply not ok
8	bBatteryNotOk	Battery not ok
9	bRTCNotOk	Real Time Clock not ok
10	bXAccelSensorTempNotOk	X accelerometer sensor temperature not ok (> 80 ºC)
11	bYAccelSensorTempNotOk	Y accelerometer sensor temperature not ok (> 80 ºC)
12	bZAccelSensorTempNotOk	Z accelerometer sensor temperature not ok (> 80 ºC)
13	bXAccelCaseTempNotOk	X accelerometer case temperature not ok (> 80 ºC)
14	bYAccelCaseTempNotOk	Y accelerometer case temperature not ok (> 80 ºC)
15	bZAccelCaseTempNotOk	Z accelerometer case temperature not ok (> 80 ºC)
16	bAccelRangeNotOk	Accelerometer out of range
17	bAHRSResultNotOk	AHRS result not ok (Note 2)
18-31		Spare
32	bShutdownReq	Shutdown requested
33	bCurrentFlashNotUsed	Current flash not used
34	bPICNotAuth	PIC not authenticated
35-63		Spare

Table 14 BIST – IMU block

Note 1 – “IMU no go” is a high level bit indicating serious HW error causing all AHRS/INS applications to fail or be unreliable. Bit is currently cumulative (or) result of bits 1-6,10-12, 16-17, 32-34 and (bit7 & bit8), i.e. “no go” is set if both 7 and 8 are set.

Note 2 – The “AHRS result not ok” bit is set if any of bits 1, 4-6, 10-12, 16 are set. When these bits become low, bit 17 remains latched high until a GC RST followed by an INS RST is done. None of these bits must be high between the GC RST and the INS RST otherwise bit 17 remains high.

Comms bit#	Field name	Note / bit set
0	bUART0Rx	UART0 rx
1	bUART1Rx	UART1 rx
2	bUART2Rx	UART2 rx
3	bUART3Rx	UART3 rx
4	bUART4Rx	UART4 rx
5	bUART0Tx	UART0 tx
6	bUART1Tx	UART1 tx
7	bUART2Tx	UART2 tx
8	bUART3Tx	UART3 tx
9	bUART4Tx	UART4 tx
10	bTRIG1Rx	TRIG1 rx
11	bTRIG2Rx	TRIG2 rx
12	bTRIG3Rx	TRIG3 rx
13	bTRIG4Rx	TRIG4 rx
14	bUART0GeneralError	UART0 Rx General error (timetag overflow or frame or parity error)
15	bUART1GeneralError	UART1 Rx General error (timetag overflow or frame or parity error)
16	bUART2GeneralError	UART2 Rx General error (timetag overflow or frame or parity error)
17	bUART3GeneralError	UART3 Rx General error (timetag overflow or frame or parity error)
18	bUART4GeneralError	UART4 Rx General error (timetag overflow or frame or parity error)
19	bUART0TxOverflowError	UART0 Tx Overflow error
20	bUART1TxOverflowError	UART1 Tx Overflow error
21	bUART2TxOverflowError	UART2 Tx Overflow error
22	bUART3TxOverflowError	UART3 Tx Overflow error
23	bUART4TxOverflowError	UART4 Tx Overflow error
24	bTRIG1_RxOverrun	TRIG1 Rx Overrun
25	bTRIG2_RxOverrun	TRIG2 Rx Overrun
26	bTRIG3_RxOverrun	TRIG3 Rx Overrun
27	bTRIG4_RxOverrun	TRIG4 Rx Overrun
28	bSD	SD Rx
29	bETHERNET0_Rx	Ethernet Port0 rx
30	bETHERNET0_Tx	Ethernet Port0 tx
31	bETHERNET1_Rx	Ethernet Port1 rx
32	bETHERNET1_Tx	Ethernet Port1 tx
33	bETHERNET2_Rx	Ethernet Port2 rx
34	bETHERNET2_Tx	Ethernet Port2 tx
35	bETHERNET3_Rx	Ethernet Port3 rx
36	bETHERNET3_Tx	Ethernet Port3 tx
37	bETHERNET4_Rx	Ethernet Port4 rx
38	bETHERNET4_Tx	Ethernet Port4 tx
39	bETHERNET0_Error	Ethernet Port0 Error
40	bETHERNET1_Error	Ethernet Port1 Error
41	bETHERNET2_Error	Ethernet Port2 Error
42	bETHERNET3_Error	Ethernet Port3 Error
43	bETHERNET4_Error	Ethernet Port4 Error
44	bPTP_Rx	PTP UDP port 319 & 320 Rx
45	bPTP_Tx	PTP UDP prot 319 & 320 Tx
46	bPTP_Error	PTP Error
47	bETHERNET_Reset	Ethernet Reset
48	bMuxChecksumError	Multiplex Protocol Checksum error
49-63		Unused

Table 15 BIST – Comms block

- The Ethernet index entries are in increasing port number order so to determine which Ethernet port is for which BIST entry the SETTINGS & CMDS msgs should be referred to.

CCA bit#	Field name	Note / bit set
0	bCurrentFlashUsed	Current Flash used
1	bOldFlashUsed	Old Flash used
2	bCurrentFactoryUsed	Current Factory used
3	bOldFactoryUsed	Old Factory used
4	bPICUserFactory	PIC Factory user
5	bPICUserEngineer	PIC Engineer user
6	bPICUserCustomer	PIC Customer user
7	bPICIMUBitSet	PIC App – IMU bit set
8	bPICAppAll	PIC App – All
9	bPICAppAHRSOnly	PIC App – AHRS only
10	bPICAppOptUSBL	PIC App – optimised USBL
11	bPICAppDPINS	PIC App – DPINS
12	bPICAppSubsea	PIC App – Subsea
13	bPICAppGPSINS	PIC App – GPSINS
14	bPICAppMetrology	PIC App – Metrology
15	bUARTStartupNotOk	Problem setting up UARTs at startup
16	bSDNotInit	SD card not initialised
17	bSDDeleteNOTOK	SD card file or directory deletion not ok
18	bTcvr1PwrNotOk	TCVR1 power problem – either should be on when off or off when on
19	bTcvr2PwrNotOk	TCVR2 power problem – either should be on when off or off when on
20	bTcvr1PwrTripped	TCVR1 power tripped
21	bTcvr2PwrTripped	TCVR2 power tripped
22	bBatteryProblem	Battery problem. Either discharge not allowed as temp>60, or charge not allowed as temp > 50.
23	bSDRW	Problem reading from or writing to the SD card
24	bPICAppSubseaLite	PIC App – Subsea Lite
25	bPICHWIssue	PIC HW problem
26-31		Spare
32	bFWIssue	Firmware issue
33-63		Spare

Table 16 BIST – CCA block

AHRS bit#	Field name	Note / bit set
0		Reserved
1	bGCNotSettled	Unsettled
2	bGCNotPosAided	Not position aided for > 10s
3	bGCNotVelAided	Not velocity aided for > 10s
4-15		

Table 17 BIST – AHRS block

AINS bit#	Field name	Note / bit set
0		Spare
1	bNotInit	AINS uninitialized
2	bNoOrient	No init orientation (AHRS not settled)
3	bNoVel	No init velocity (less than x seconds old) for initialisation from enabled sensors (Note 1)
4	bNoPos	No init position available (less than x seconds old) for initialisation from enabled sensors (Note 1)
5	bNoDepth	No init depth available (less than x seconds old) for initialisation from enabled sensors (Note 1)
6-30		Spare
31	bZMDBLarge	ZMD Bias large (Note 2)

32	bPosLimit	Horizontal position 1DRMS high – INS auto re-initialize (Note 3)
33	bHdgInteg	Heading unreasonable (Note 4)
34	bAttInteg	Attitude unreasonable (Note 5)
35	bGyroBLarge	Gyro (bias) error large (Note 6)
36	bAccBLarge	Accel (bias) error large (Note 7)
37-63		Spare

Table 18 BIST – AINS block

Note 1 – x = Kalman delay + 1

Note 2 – ZMD Bias large when bias > (ZMD cmd MRms2 value) for > 120s.

Note 3 – Horizontal position 1DRMS high when 1DRMS > 1000m (limit can be set by user) for more than 5s.

Note 4 – Heading unreasonable when abs(INS hdg – AHRS hdg) > 2 degrees for more than 5s.

Note 5 – Attitude unreasonable when abs (INS roll – AHRS roll) or abs (INS pitch – AHRS pitch) > 0.2 degrees for more than 5s.

Note 6 – Gyro bias large when bias > 3 sigma cmd value for > 120s.

Note 7 – Accelerometer bias large when bias > 3 sigma cmd value for > 120s.

Note 8 – Bits 32-63 = 0 when AINS is not initialised (bit 1 = 1).

15.2 NAV

Name	Nav
Also Known As	Navigation Data
Type	Binary – Sonardyne
Input / Output / Log	Output
Source	INS
Length	Fixed, 46 bytes
Multiplex Message ID	213

15.2.1 Description

The navigation (Nav) data message is the generic navigation output from Lodestar INS and usually used in conjunction with the navigation quality message (NavQual).

Nav and NavQual are intended for advanced users including internal (Sonardyne) and external system integrators.

Nav values are valid for the vehicle CRP / frame, except acceleration which is valid for the IMU zero point but expressed in vehicle frame. For best accuracy it is recommended to use CRP=IMU zero point.

The INS algorithm is the only source for the NAV message.

All data is transmitted LSB first

15.2.2 Format

Byte#	Field name	Units	Min, max *approximate	Data Type	Notes

1-6 / 6	timeTag	10^{-6} seconds	0, 2.8×10^{8}*	Uint48	System time
7-10 / 4	lat	$2^{-31}\times90$ deg	-90, +90	Int32	Latitude, Note 1
11-14 / 4	lon	$2^{-31}\times180$ deg	-180, +180	Int32	Longitude, Note 2
15-18 / 4	depth	10^{-3} metres	-2×10^{6}, $+2\times10^{6}$*	Int32	Depth below sea level, Note 3
19-20 / 2	altitude	10^{-2} metres	0, 655*	Uint16	Height above seabed, Note 4
21-22 / 2	roll	$2^{-15}\times180$ deg	-180, +180	Int16	Note 5
23-24 / 2	pitch	$2^{-15}\times180$ deg	-90, +90	Int16	Note 6
25-26 / 2	heading	$2^{-15}\times180$ deg	0, 360	Uint16	Note 7
27-28 / 2	vx	10^{-3} m/s	-32.8, +32.8*	Int16	X-velocity
29-30 / 2	vy	10^{-3} m/s	-32.8, +32.8*	Int16	Y-velocity
31-32 / 2	vz	10^{-3} m/s	-32.8, +32.8*	Int16	Z-velocity
33-34 / 2	wx	10^{-2} deg/s	-328, +328*	Int16	Angular rate about x axis
35-36 / 2	wy	10^{-2} deg/s	-328, +328*	Int16	Angular rate about y axis
37-38 / 2	wz	10^{-2} deg/s	-328, +328*	Int16	Angular rate about z axis
39-40 / 2	ax	10^{-3} m/s^{2}	-32.8, +32.8* (-3.3, +3.3 G)	Int16	X-acceleration
41-42 / 2	ay	10^{-3} m/s^{2}	-32.8, +32.8* (-3.3, +3.3 G)	Int16	X-acceleration
43-44 / 2	az	10^{-3} m/s^{2}	-32.8, +32.8* (-3.3, +3.3 G)	Int16	X-acceleration
45-46 / 2	mode	N/A		Bit16	Note 8

Note 1 – Latitude, north is positive. 0.5cm resolution.

Note 2 – Longitude, east is positive. 1cm resolution at equator.

Note 3 – Depth, down is positive.

Note 3 – Height above seabed as measured by the DVL.

Note 4 – Roll is the angle between the y-axis and horizontal. Roll is positive when y is pointed below the horizontal (starboard down).

Note 5 – Pitch is the angle between the x-axis and horizontal. Pitch is positive when x is pointed above the horizontal (bow up).

Note 6 – Heading is the angle between North and projection of the x-axis onto the horizontal (measured about the down direction).

Note 7 – The mode register bits are described below:
0: Data valid
1: INS initialised
2: INS application not enabled
3: Altitude old – no new value from DVL since last altitude output in this message
4-14: Reserved
15: System failure

- The Nav message will be generated even if the INS is not initialised, however all fields (except timeTag) will be zero.

15.3 NAVQUAL

Name	NavQual
Also Known As	Navigation quality estimate
Type	Binary – Sonardyne
Input / Output / Log	Log
Log Frequency	~1Hz
Length	Fixed, 50 bytes
Multiplex Message ID	214

15.3.1 Description

The navigation quality (NavQual) message reports the expected accuracy of the data given in the NAV message.

All data is transmitted LSB first

15.3.2 Format

Byte#	Field name	Units	Data Type	Notes
1-6 / 6	timeTag	10^{-6} seconds	Uint48	System time
7-10 / 4	posMajor	Metres	Float32	Horizontal position 1σ error ellipse (Note 1): - semi-major axis
11-14/ 4	posMinor	Metres	Float32	- semi-minor axis
15-18 / 4	dirPMajor	Degrees	Float32	- direction of semi-major axis
19-22 / 4	stdDepth	Metres	Float32	1σ depth error
23-26 / 4	stdLevN	Degrees	Float32	1σ level error about North (Note 2)
27-30 / 4	stdLevE	Degrees	Float32	1σ level error about East (Note 2)
31-34 / 4	stdHeading	Degrees	Float32	1σ heading error
35-38 / 4	velMajor	m/s	Float32	Horizontal velocity 1σ error ellipse (Note 3): - semi-major axis
39-42 / 4	velMinor	m/s	Float32	- semi-minor axis
43-46 / 4	dirVMajor	Degrees	Float32	- direction of semi-major axis
47-50 / 4	velDown	m/s	Float32	1σ down velocity error

Note 1 –

- Horizontal position 1DRMS = sqrt(posMajor^2 + posMinor^2)
- CEP(50%) ~= 0.589 × (posMajor + posMinor)
- Error ellipse (1σ) is 39.4% probability (i.e. 39.4% likelihood that true value is within ellipse)
- 95% percent probability error ellipse is 2.447 × 1σ error ellipse

Note 2 – Roll & pitch 1σ ~= max(stdLevN,stdLevE) for roll, pitch << 45deg.

Note 3 – Velocity RMS = sqrt(velMajor^2+ velMinor^2)

15.4 Observation status messages (OBST…)

Name	OBST[…]
Also Known As	Observation status
Type	Binary – Sonardyne
Input / Output / Log	Log
Log Frequency	Same as aiding frequency
Length	Fixed
Multiplex Message ID	170-180

15.4.1 Description

For each observation source (GPS, DVL, …), there is an associated observation status message type, which gives information about how the message was treated by the INS. These messages are listed in the table below:

Message name	Message ID	Description	Associated observation message(s)
obStDVL	178	DVL	PD0, PD4/5
obStGpsPos	172	GPS position	GGA, GST (future)
obStLBL	179	LBL range	PSONLOBS
obStPDepth	176	Pressure depth	PRDDPT, PSONDEPM, PRDDIGIQxx
obStSUSBL	174	Vehicle USBL position (TPDR)	PSIMSSB, GGA
obStSVS	177	Sound Velocity Sensor	VALEPORT, PSONSS

obStZMD	170	Zero Mean Depth	NA
obStZUPT	180	Zero Velocity Update	NA

Table 19 List of observation status messages

Every message contains three (mostly) generic fields as detailed in Table 19 which may be followed by further observation source specific fields.

Byte#	Field name	Units	Data Type	Notes
1-6 / 6	timeTag	10^{-6} Seconds	Uint48	System time, Note 1
7-8 / 2	reject		Bits x16	Rejection status bits, see Table 21
9-12 / 4	mahad		Float32	Mahalanobis distance, Note 2

Table 20 Generic observation status fields

Note 1 – Identical to the time tag of the associated (raw) observation data message. Unless otherwise specified, time tag is the time of arrival of the data at LS, i.e. time of the first message or packet byte.

Note 2 – Indicates how well the observation matched the INS/Kalman expectation.

The 4 MSB of the reject field are shown below. These bits are present in all observation status messages. The remaining 12 LSB are type specific as defined in the following sub-sections.

Bit	Name	Description
0 (LSB) - 11	-	Type specific
12	misc	Miscellaneous: This bit is used to denote a problem not catered for by the other bits. It is set if the Kalman hasn't used this sensor data and none of the other rejection bits (0..11,13..15) are set. This may occur when the aiding data is not used by the Kalman filter as it has not yet initialised, but only if no other rejection bits are set. Another reason it could be set is if a general firmware error caused the observation to not be used.
13	ttag	Time tag issue: Any of the following: • UTC time tag cannot be processed due to insufficient time sync • Observation latency too large • Observation time tag reasonability test failed
14	sig	Sigma test: Observation failed Kalman statistical testing. 0 otherwise. If bit 15 is set then SIG is 0 (test not performed).
15 (MSB)	dis	Disabled: Aiding source is disabled for AINS use (but still generating data). If this bit is set then all other bits may be set to 0 (not performed) or they may be set according to their otherwise specified functionality.

Table 21 Generic rejection bits

If the rejection field is all 0 then the observation was accepted.

15.4.2 OBSTDVL

Name	OBSTDVL
Also Known As	DVL observation status
Type	Binary – Sonardyne
Input / Output / Log	Log
Log Frequency	Same as DVL aiding frequency
Length	Fixed, 35 bytes
Multiplex Message ID	178

Rejection Bit	Name	Description
0	-	=0
1	-	=0

2	svs	SVS bad (no good SVS observation in previous 60s)
3	cfg	Sensor configuration unsupported
4	evel	'Error velocity' too high (observation noisy / outlier)
5	bsts	Bottom status bad
6	zbr	Zero beam range detected (no range measured on 1 or more beams)
7	zvel	Zero velocity measured (can be an outlier due to sensor error)
8	tout	Timeout – too long since last DVL observation
9	velcu	Velocity change unreasonable (since last observation)
10	-	=0
11	-	=0

Table 22 OBSTDVL specific rejection bits

Byte#	Field name	Units	Data Type	Notes
1-12 / 12	-	-	-	Generic, see Table 20
13 / 1	obmsgtype	-	Uint8	DVL message type: 0 = RDI PD4 1 = RDI PD5 2 = RDI PD0 8 = LinkQuest PD4
14-17 / 4	sv	m/s	Float32	Sound speed used by Kalman filter
18-23 / 6	timeTag	10^{-6} Seconds	Uint48	System time of validity
24-27 / 4	ResidualVx	m/s	Float32	Residual, Vx
28-31 / 4	ResidualVy	m/s	Float32	Residual, Vy
32-35 / 4	ResidualVz	m/s	Float32	Residual, Vz

Table 23 OBSTDVL specific fields

15.4.3 OBSTGPSPOS

Name	OBSTGPSPOS
Also Known As	GNSS position observation status
Type	Binary – Sonardyne
Input / Output / Log	Log
Log Frequency	Same as GNSS aiding frequency
Length	Fixed, 24 bytes
Multiplex Message ID	172

Rejection Bit	Name	Description
0	-	=0
1	-	=0
2	-	=0
3	-	=0
4	3DQual	Quality value unavailable (or unusable only when INS GPS USEVERTICAL set to 0)
5	qi	GGA quality indicator unacceptable. Acceptable values are configurable - see "GPS QUALITY" in command & control documentation.
6	-	=0
7	-	=0
8	-	=0
9	-	=0
10	-	=0
11	-	=0

Table 24 OBSTGPSPOS specific rejection bits

Byte#	Field name	Units	Data Type	Notes

1-12 / 12	-	-	-	Generic, see Table 20
13-16 / 4	ResidualLat	rad	Float32	Residual, Latitude
17-20 / 4	ResidualLon	rad	Float32	Residual, Longitude
21-24 / 4	ResidualDepth	m	Float32	Residual, Depth. Zero if 2D

Table 25 OBSTGPSPOS specific fields

15.4.4 OBSTLBL

Name	OBSTLBL
Also Known As	LBL observation status
Type	Binary – Sonardyne
Input / Output / Log	Log
Log Frequency	Once per LBL observation
Length	Fixed, 22 bytes
Multiplex Message ID	179

Rejection Bit	Name	Description
0	-	=0
1	-	=0
2	maxpred	Abs(observed range – predicted range) >= LS C&C limit
3	rrate	Range rate Observed range rate >= LS C&C limit
4	range	Range Below min range or larger than max range specified by LS C&C limit
5	tprevObs	Minimum previous observations Not enough previous observations in previous time specified by LS C&C limit
6	rsl	Reduced signal level Signal level - filtered signal level <= -LS C&C limit
7	rsnr	Reduced signal to noise ratio SNR - filtered SNR <= -LS C&C limit
8	sl	Signal Level Observed signal level <= LS C&C limit
9	snr	Signal to noise ratio Observed SNR <= LS C&C limit
10	lbi	Lvr or Bcn Info missing • Transceiver lever arm or beacon configuration is undefined or, • Age of PSONLBLLVR or PSONLBLBCN messages exceeded 180 seconds.
11	-	=0

Table 26 OBSTLBL specific rejection bits

Byte#	Field name	Units	Data Type	Notes
1-12 / 12	-	-	-	Generic, see Table 20
13-14 / 2	bcnid	-	Uint16	Beacon ID (address)
15-18 / 4	sv	m/s	Float32	Sound speed (range)
19-22 / 4	ResidualRange	s	Float32	Residual, range

Table 27 OBSTLBL specific fields

15.4.5 OBSTPDEPTH

Name	OBSTPDEPTH
Also Known As	Depth observation status
Type	Binary – Sonardyne
Input / Output / Log	Log
Log Frequency	Once per depth observation
Length	Fixed, 17 bytes
Multiplex Message ID	176

Rejection Bit	Name	Description
0	-	=0
1	-	=0
2	-	=0
3	-	=0
4	-	=0
5	Bathy	Bathy (Winson) system devices flags not ok
6	-	=0
7	-	=0
8	-	=0
9	-	=0
10	-	=0
11	-	=0

Table 28 OBSTPDEPTH specific rejection bits

Byte#	Field name	Units	Data Type	Notes
1-12 / 12	-	-	-	Generic, see Table 20
13 / 1	obmsgtype	-	Uint8	0 – Keller 1 – PSONDEP 2 – DigiQuartz (m) 3 – DigiQuartz (psi) 4 – DigiQuartz (kPa) 5 – $__DPT 6 – PRDDIGIQO 7 – Winson 8 – Valeport SVX2 (DBAR)
14-17 / 4	ResidualDepth	m	Float32	Residual, Depth

Table 29 OBSTPDEPTH specific fields

15.4.6 OBSTSUSBL

Name	OBSTSUSBL
Also Known As	Vehicle (transponder) USBL position observation status
Type	Binary – Sonardyne
Input / Output / Log	Log
Log Frequency	Once per observation
Length	Fixed, 27 bytes
Multiplex Message ID	174

Rejection Bit	Name	Description
0	-	=0
1	-	=0
2	-	=0
3	ra	Reject Acoustic • Acoustic observation status flag = ‘V’ (invalid). • PSIMSSB.Coordinate_system setting not OK (should be ‘R’ – radians). • PSIMSSB.SW_filter setting not OK (should be ‘M’ – measured). • GGA quality indicator unacceptable. See C&C “GPS QUALITY”.
4	-	=0
5	-	=0
6	-	=0
7	-	=0
8	-	=0
9	lbi	Lvr / Bcn info missing • Transponder/responder lever arm is undefined
10	-	=0

11	-	=0

Table 30 OBSTSUSBL specific rejection bits

Byte#	Field name	Units	Data Type	Notes
1-12 / 12	-	-	-	Generic, see Table 20
13-14 / 2	bcnid	-	Uint16	Beacon ID (address)
15 / 1	obmsgtype	-	Uint8	Observation type: 0 – GGA 1 – $PSIMSSB
16-19 / 4	ResidualLat	rad	Float32	Residual, latitude
20-23 / 4	ResidualLon	rad	Float32	Residual, longitude
24-27 / 4	ResidualDepth	m	Float32	Residual, depth. Zero if 2D

Table 31 OBSTSUSBL specific fields

15.4.7 OBSTSVS

Name	OBSTSVS
Also Known As	Sound velocity observation status
Type	Binary – Sonardyne
Input / Output / Log	Log
Log Frequency	Once per observation
Length	Fixed, 13 bytes
Multiplex Message ID	177

Note: Sound velocity (at vehicle) is not by itself used as a Kalman observation but indirectly to compensate DVL and LBL observations. The MISC rejection bit is not set.

Rejection Bit	Name	Description
0	-	=0
1	-	=0
2	-	=0
3	-	=0
4	-	=0
5	-	=0
6	unr	Measured sound speed unreasonable
7	-	=0
8	-	=0
9	-	=0
10	-	=0
11	-	=0

Table 32 OBSTSVS specific rejection bits

Byte#	Field name	Units	Data Type	Notes
1-12 / 12	-	-	-	Generic, see Table 20
13 / 1	obmsgtype	-	Uint8	Observation type: 0 – VALEPORT 1 – $PSONSS 2 – Manual 3 – Auto

Table 33 OBSTSVS specific fields

15.4.8 OBSTZMD

Name	OBSTZMD
Also Known As	Zero mean depth observation status
Type	Binary – Sonardyne
Input / Output / Log	Log
Log Frequency	Once per Kalman cycle

Length	Fixed, 16 bytes
Multiplex Message ID	170

This is generated once per Kalman cycle if and only if ZMD is enabled. Since ZMD is a “virtual” sensor it has no associated observation message and no message specific rejection bits. TimeTag is Kalman time of observation.

Byte#	Field name	Units	Data Type	Notes
1-12 / 12	-	-	-	Generic, see Table 20
13-16 / 4	ResidualDepth	m	Float32	Residual, Depth

Table 34 OBSTZMDspecific fields

15.4.9 OBSTZUPT

Name	OBSTZUPT
Also Known As	Zero velocity update observation status
Type	Binary – Sonardyne
Input / Output / Log	Log
Log Frequency	Once per Kalman cycle
Length	Fixed, 20 bytes
Multiplex Message ID	180

This is generated once per Kalman cycle if and only if ZUPT is enabled. Since ZUPT is a “virtual” sensor it has no associated observation message and no message specific rejection bits. TimeTag is Kalman time of observation.

Byte#	Field name	Units	Data Type	Notes
1-12 / 12	-	-	-	Generic, see Table 20
13-16 / 4	ResidualLat	rad	Float32	Residual, Latitude
17-20 / 4	ResidualLon	rad	Float32	Residual, Longitude
21-24 / 4	ResidualDepth	m	Float32	Residual, Depth. Zero if 2D

Table 35 OBSTZUPT specific fields

15.5 SETTINGS

Name	SETTINGS
Type	Binary – Sonardyne
Input / Output / Log	Log
Log Frequency	~10^{-3}Hz (15 min period)
Length	Variable
Multiplex Message ID	216

15.5.1 Description

Previously this message was to output the internal state of the Settings structure. However, as this is larger than the ASCII result of the SYS CMDS LIST, less maintainable and less immediately understandable to a human this message now outputs periodically a result of a SYS CMDS LIST.
The main issue with logging this information is its size. Logging huge data sizes in a single multiplex message is not feasible for lower baud rates and could cause loss of IMU data.

The data is broken down into N messages and the first two bytes of the Data section (Payload) in the Multiplex protocol consists of information of its sequence. The maximum size of data in each message is restricted to 512 bytes, without DLE byte stuffing.

15.5.2 Format

Byte#	Field name	Units	Data Type	Notes
1	N	NA	Uint8	Number of packets
2	n	NA	Uint8	Current packet number
3+	Data	NA	ASCII	SYS CMDS LIST output

15.6 TMS

Name	TMS
Also Known As	Time system data
Type	Binary – Sonardyne
Input / Output / Log	Log
Log Frequency	~1Hz
Length	Fixed, 32 bytes
Multiplex Message ID	208

15.6.1 Description

This message contains the current state of the time system, including configuration and relationship between system time and UTC.

15.6.2 Format

Byte#	Field name	Units	Data Type	Notes
1-6 / 6	sysTime	10^{-6} Seconds	Uint48	System time (and message time tag)
7-14 / 8	utcTime	10^{-6} Seconds	Uint64	UTC Time – seconds since midnight 1970.01.01
15-20 / 6	timeSinceUpdate	10^{-6} Seconds	Uint48	Time since last accepted UTC time update, e.g. from ZDA/PPS
21-24 / 4	stdDev	Seconds	Float32	Expected standard deviation of UTC time field
25 / 1	Source	-	Uint8	Source of most recent UTC sync (Note 1)
26 / 1	ppsRising	-	Uint8	PPS pulse type: 0 = PPS valid on rising edge 1 = PPS valid on falling edge
27 / 1	zdaCount	-	Uint8	LS byte of ZDA message count
28 / 1	ppsCount	-	Uint8	LS byte of PPS message count
29 / 1	zdaRejCount	-	Uint8	LS byte of ZDA message rejection count
30 / 1	ppsRejCount	-	Uint8	LS byte of PPS signal rejection count
31 / 1	ppsZdaProcCount	-	Uint8	LS byte of accepted PPS/ZDA pairs
32 / 1	filtResetCount	-	Uint8	LS byte of UTC filter reset count

15.6.3 Notes

Note 1 – Source of RTC (real time clock) to UTC update:

- 0 = No source
- 1 = Lodestar RTC
- 2 = Standalone ZDA
- 3 = Standalone GGA
- 4 = ZDA & 1PPS

Note 2 – The last 7 fields (ppsRising to filtResetCount) are for advanced users only and may be subject to change

Example: Converting IMU time tag from [sys] to [utc]:

imu.timeTag[sys] = 1234567890 µsec

Get the preceding timeSys message:

timeSys.sysTime = 1234101010 µsec
timeSys.utcTime = 1254273030984001 µsec
timeSys.stdDev = 0.0000124 sec   low std.dev. => UTC can be trusted

imu.timeTag [utc] = imu.timeTag[sys] + (timeSys.utcTime – timeSys.sysTime)
= 1234567890 + (1254273030984001 – 1234101010) µsec

= 1254273031450881 µsec = 20090930T011031 (ISO 8601) = 2009-09-30 01:10:31

16 Binary – Other Proprietary Messages

These binary messages are based on formats defined by other manufacturers

16.1 PD0

Name	PD0
Type	Binary – Proprietary
Input / Output / Log	Input
Length	Variable
Multiplex Message ID	141

16.1.1 Description

This proprietary message is generated by the Teledyne RDI DVL (e.g. Workhorse Navigator) when it has been configured using the PD0 command. See “WorkHorse Commands and Output Data Format” for details. Note that the encoding must be binary. Hex-ASCII is not supported for PD0.

The PD0 output data format assumes the instrument is stationary and the bottom is moving. The PD4/5 assumes the opposite.

16.1.2 Format

Bytes	**Field name**	**Notes**
6 + [2 × No. of data types]	HEADER	Always output
59	FIXED LEADER DATA	
65	VARIABLE LEADER DATA	
2 + 8 per depth cell	VELOCITY	WD-command WP-command
2 + 4 per depth cell	CORRELATION MAGNITUDE	
2 + 4 per depth cell	ECHO INTENSITY	
2 + 4 per depth cell	PERCENT GOOD	
85	BOTTOM TRACK DATA	BP-command
2	RESERVED	Always output
2	CHECKSUM	

16.1.3 Header Format

Byte#	**Field name**	**Notes**
1	Header ID	Stores the header identification byte (7Fh)
2	Data source ID	Stores the data source identification byte (7Fh for the Work-Horse ADCP)
3-4	Number of bytes in ensemble	This field contains the number of bytes from the start of the current ensemble up to, but not including, the 2-byte checksum
5	Spare	Undefined
6	Number of data types	This field contains the number of data types selected for collection. By default, fixed/variable leader, velocity, correlation magnitude, echo intensity, and percent good are selected for collection. This field will therefore have a value of six (4 data types + 2 for the Fixed/Variable Leader data).
7-8	Offset for data type #1	These field contains the internal memory address offset where the Navigator will store information for data type #n Adding “1” to this offset number gives the absolute Binary Byte number in the ensemble where Data Type #1 begins (the first byte of the ensemble is Binary Byte #1).
9-10	Offset for data type #2	
11-12	Offset for data type #3	
…	…	Sequence continues for up to n data types
(2n+5)-(2n+6) / 2	Offset for data type #n	

16.1.4 Fixed Leader Data Format

Byte#	**Field name**	**Notes**
1-2 / 2	FIXED LEADER ID	Stores the Fixed Leader identification word (00 00h).

3 / 1	CPU F/W VER.	Contains the version number of the CPU firmware.
4 / 1	CPU F/W REV.	Contains the revision number of the CPU firmware.
5-6 / 2	SYSTEM CONFIGURATION	This field defines the Navigator hardware configuration. Convert this field (2 bytes, LSB first) to binary and interpret as follows: LSB (Byte 5): BITS 76543210 -----000 75-kHz SYSTEM -----001 150-kHz SYSTEM -----010 300-kHz SYSTEM -----011 600-kHz SYSTEM -----100 1200-kHz SYSTEM -----101 2400-kHz SYSTEM ----0--- CONCAVE BEAM PAT. ----1--- CONVEX BEAM PAT. --00---- SENSOR CONFIG #1 --01---- SENSOR CONFIG #2 --10---- SENSOR CONFIG #3 -0------ XDCR HD NOT ATT. -1------ XDCR HD ATTACHED 0------- DOWN FACING BEAM 1------- UP-FACING BEAM MSB (Byte 6): BITS 76543210 ------00 15E BEAM ANGLE ------01 20E BEAM ANGLE ------10 30E BEAM ANGLE ------11 OTHER BEAM ANGLE 0100---- 4-BEAM JANUS CONFIG 0101---- 5-BM JANUS CFIG DEMOD) 1111---- 5-BM JANUS CFIG.(2 DEMOD) Example: Hex 5249 (i.e., hex 49 followed by hex 52) identifies a 150-kHz system, convex beam pattern, down-facing, 30E beam angle, 5 beams (3 demods).
7 / 1	REAL/SIM FLAG	This field is set by default as real data (0).
8 / 1	LAG LENGTH	Lag Length. The lag is the time period between sound pulses. This is varied, and therefore of interest in, at a minimum, for the WM5, WM8 and WM11 and BM7 commands.
9 / 1	NUMBER OF BEAMS	Contains the number of beams used to calculate velocity data (not physical beams). The Navigator needs only three beams to calculate water-current velocities. The fourth beam provides an error velocity that determines data validity. If only three beams are available, the Navigator does not make this validity check.
10 / 1	NUMBER OF CELLS {WN}	Contains the number of depth cells over which the Navigator collects data (WN-command). Scaling: LSD = 1 depth cell; Range = 1 to 128 depth cells
11-12 / 2	PINGS PER ENSEMBLE {WP}	Contains the number of pings averaged together during a data ensemble (WP-command). If WP = 0, the Navigator does not collect the WD water-profile data. Note: The Navigator automatically extends the ensemble interval (TE) if the product of WP and time per ping (TP) is greater than TE (i.e., if WP x TP > TE). Scaling: LSD = 1 ping; Range = 0 to 16,384 pings
13-14 / 2	DEPTH CELL LENGTH {WS}	Contains the length of one depth cell (WS-command). Scaling: LSD = 1 centimeter; Range = 1 to 6400 cm (210 feet)
15-16 / 2	BLANK AFTER TRANSMIT {WF}	Contains the blanking distance used by the Navigator to allow the transmit circuits time to recover before the receive cycle begins (WF-command). Scaling: LSD = 1 centimeter; Range = 0 to 9999 cm (328 feet)
17 / 1	PROFILING MODE {WM}	Contains the Signal Processing Mode. This field will always be set to 1.
18 / 1	LOW CORR THRESH	Contains the minimum threshold of correlation that water profile data

	{WC}	can have to be considered good data (WCcommand). Scaling: LSD = 1 count; Range = 0 to 255 counts
19 / 1	NO. CODE REPS	Contains the number of code repetitions in the transmit pulse. Scaling: LSD = 1 count; Range = 0 to 255 counts
20 / 1	%GD MINIMUM {WG}	Contains the minimum percentage of water-profiling pings in an ensemble that must be considered good to output velocity data. Scaling: LSD = 1 percent; Range = 1 to 100 percent
21-22 / 2	ERROR VELOCITY MAXIMUM {WE}	This field, initially set by the WE-command, contains the actual threshold value used to flag water-current data as good or bad. If the error velocity value exceeds this threshold, the Navigator flags all four beams of the affected bin as bad. Scaling: LSD = 1 mm/s; Range = 0 to 5000 mm/s
23 / 1	TPP MINUTES	These fields, set by the TP-command, contain the amount of time between ping groups in the ensemble. NOTE: The Navigator automatically extends the ensemble interval (set by TE) if (WP x TP > TE).
24 / 1	TPP SECONDS	
25 / 1	TPP HUNDREDTHS {TP}	
26 / 1	COORDINATE TRANSFORM {EX}	Contains the coordinate transformation processing parameters (EX-command). These firmware switches indicate how the Navigator collected data. xxx00xxx = NO TRANSFORMATION (BEAM COORDINATES) xxx01xxx = INSTRUMENT COORDINATES xxx10xxx = SHIP COORDINATES xxx11xxx = EARTH COORDINATES xxxxx1xx = TILTS (PITCH AND ROLL) USED IN SHIP OR EARTH TRANSFORMATION xxxxxx1x = 3-BEAM SOLUTION USED IF ONE BEAM IS BELOW THE CORRELATION THRESHOLD SET BY THE WC-COMMAND xxxxxxx1 = BIN MAPPING USED
27-28 / 2	HEADING ALIGNMENT {EA} LSB	Contains a correction factor for physical heading misalignment (EA-command). Scaling: LSD = 0.01 degree; Range = -179.99 to 180.00 degrees
29-30 / 2	HEADING BIAS {EB}	Contains a correction factor for electrical/magnetic heading bias (EB-command). Scaling: LSD = 0.01 degree; Range = -179.99 to 180.00 degrees
31 / 1	SENSOR SOURCE {EZ}	Contains the selected source of environmental sensor data (EZ-command). These firmware switches indicate the following. FIELD DESCRIPTION x1xxxxxx = CALCULATES EC (SPEED OF SOUND) FROM ED, ES, AND ET xx1xxxxx = USES ED FROM DEPTH SENSOR xxx1xxxx = USES EH FROM TRANSDUCER HEADING SENSOR xxxx1xxx = USES EP FROM TRANSDUCER PITCH SENSOR xxxxx1xx = USES ER FROM TRANSDUCER ROLL SENSOR xxxxxx1x = USES ES (SALINITY) FROM CONDUCTIVITY SENSOR xxxxxxx1 = USES ET FROM TRANSDUCER TEMPERATURE SENSOR NOTE: If the field = 0, or if the sensor is not available, the Navigator uses the manual command setting. If the field = 1, the Navigator uses the reading from the internal sensor or an external synchro sensor (only applicable to heading, roll, and pitch). Although you can enter a “2” in the EZ-command string, the Navigator only displays a 0 (manual) or 1 (int/ext sensor).
32 / 1	SENSORS AVAILABLE	This field reflects which sensors are available. The bit pattern is the same as listed for the EZ-command (above).
33-34 / 2	BIN 1 DISTANCE	This field contains the distance to the middle of the first depth cell (bin). This distance is a function of depth cell length (WS), the profiling mode (WM), the blank after transmit distance (WF), and speed of sound. Scaling: LSD = 1 centimeter; Range = 0 to 65535 cm (2150 feet)

35-36 / 2	XMIT PULSE LENGTH BASED ON {WT}	This field, set by the WT-command, contains the length of the transmit pulse. When the Navigator receives a <BREAK> signal, it sets the transmit pulse length as close as possible to the depth cell length (WS-command). This means the Navigator uses a WT command of zero. However, the WT field contains the actual length of the transmit pulse used. Scaling: LSD = 1 centimeter; Range = 0 to 65535 cm (2150 feet)
37-38 / 2	(starting cell) WP REF LAYER AVERAGE {WL} (ending cell)	Contains the starting depth cell (LSB, byte 37) and the ending depth cell (MSB, byte 38) used for water reference layer averaging (WL-command). Scaling: LSD = 1 depth cell; Range = 1 to 128 depth cells
39 / 1	FALSE TARGET THRESH {WA}	Contains the threshold value used to reject data received from a false target, usually fish (WA-command). Scaling: LSD = 1 count; Range = 0 to 255 counts (255 disables)
40 / 1	SPARE	Contains the CX-command setting. Range = 0 to 5
41-42 / 2	TRANSMIT LAG DISTANCE	This field, determined mainly by the setting of the WMcommand, contains the distance between pulse repetitions. Scaling: LSD = 1 centimeter; Range = 0 to 65535 centimeters
43-50 / 8	CPU BOARD SERIAL NUMBER	Contains the serial number of the CPU board.
51-52 / 2	SYSTEM BANDWIDTH {WB}	Contains the WB-command setting. Range = 0 to 1
53 / 1	SPARE	Spare
54 / 1	BASE FREQUENCY INDEX	Base frequency index
55-59 / 4	SPARE	Spare

16.1.5 Variable Leader Data Format

Byte#	Field name	Notes
1-2 / 2	VARIABLE LEADER ID 80h	Stores the Variable Leader identification word (80 00h).

3-4 / 2	ENSEMBLE NUMBER	This field contains the sequential number of the ensemble to which the data in the output buffer apply. Scaling: LSD = 1 ensemble; Range = 1 to 65,535 ensembles NOTE: The first ensemble collected is #1. At “rollover,” we have the following sequence:
		1 = ENSEMBLE NUMBER 1
		
		65535 = ENSEMBLE NUMBER 65,535 | ENSEMBLE
		0 = ENSEMBLE NUMBER 65,536 | #MSB FIELD
		1 = ENSEMBLE NUMBER 65,537 | (BYTE 12 INCR.)

5 / 1	RTC YEAR {TS}	These fields contain the time from the Navigator’s real-time clock (RTC) that the current data ensemble began. The TS-command (Set Real-Time Clock) initially sets the clock. The Navigator does account for leap years.
6 / 1	RTC MONTH {TS}	
7 / 1	RTC DAY {TS}	
8 / 1	RTC HOUR {TS}	
9 / 1	RTC MINUTE {TS}	
10 / 1	RTC SECOND {TS}	
11 / 1	RTC HUNDREDTHS {TS}	
12 / 1	ENSEMBLE # MSB	This field increments each time the Ensemble Number field (bytes 3,4) “rolls over.” This allows ensembles up to 16,777,215. See Ensemble Number field above.

13-14 / 2	BIT RESULT	This field contains the results of the Navigator’s Built-in Test function. A zero code indicates a successful BIT result.
		(BYTE 14 RESERVED FOR FUTURE USE)
		`BYTE 13  BYTE 14`
		`1xxxxxxx xxxxxxxx = RESERVED`
		`x1xxxxxx xxxxxxxx = RESERVED`
		`xx1xxxxx xxxxxxxx = RESERVED`
		`xxx1xxxx xxxxxxxx = DEMOD 1 ERROR`

		xxxx1xxx xxxxxxxx = DEMOD 0 ERROR xxxxx1xx xxxxxxxx = RESERVED xxxxxx1x xxxxxxxx = TIMING CARD ERROR xxxxxxx1 xxxxxxxx = RESERVED
15-16 / 2	SPEED OF SOUND {EC}	Contains either manual or calculated speed of sound information (EC-command). Scaling: LSD = 1 meter per second; Range = 1400 to 1600 m/s
17-18 / 2	DEPTH OF TRANSDUCER {ED}	Contains the depth of the transducer below the water surface (ED-command). This value may be a manual setting or a reading from a depth sensor. Scaling: LSD = 1 decimeter; Range = 1 to 9999 decimeters
19-20 / 2	HEADING {EH}	Contains the Navigator heading angle (EH-command). This value may be a manual setting or a reading from a heading sensor. Scaling: LSD = 0.01 degree; Range = 000.00 to 359.99 degrees
21-22 / 2	PITCH (TILT 1) {EP}	Contains the Navigator pitch angle (EP-command). This value may be a manual setting or a reading from a tilt sensor. Positive values mean that Beam #3 is spatially higher than Beam #4. Scaling: LSD = 0.01 degree; Range = -20.00 to +20.00 degrees
23-24 / 2	ROLL (TILT 2) {ER}	Contains the Navigator roll angle (ER-command). This value may be a manual setting or a reading from a tilt sensor. For up-facing Navigators, positive values mean that Beam #2 is spatially higher than Beam #1. For down-facing Navigators, positive values mean that Beam #1 is spatially higher than Beam #2. Scaling: LSD = 0.01 degree; Range = -20.00 to +20.00 degrees
25-26 / 2	SALINITY {ES}	Contains the salinity value of the water at the transducer head (ES-command). This value may be a manual setting or a reading from a conductivity sensor. Scaling: LSD = 1 part per thousand; Range = 0 to 40 ppt
27-28 / 2	TEMPERATURE {ET}	Contains the temperature of the water at the transducer head (ET-command). This value may be a manual setting or a reading from a temperature sensor. Scaling: LSD = 0.01 degree; Range = -5.00 to +40.00 degrees
29 / 1	MPT MINUTES	This field contains the Minimum Pre-Ping Wait Time between ping groups in the ensemble.
30 / 1	MPT SECONDS	
31 / 1	MPT HUNDREDTHS	
32 / 1	HDG STD DEV	These fields contain the standard deviation (accuracy) of the heading and tilt angles from the gyrocompass/ pendulums. Scaling (Heading): LSD = 1°; Range = 0 to 180° Scaling (Tilts): LSD = 0.1°; Range = 0.0 to 20.0°
33 / 1	PITCH STD DEV	
34 / 1	ROLL STD DEV	
35 / 1	ADC CHANNEL 0	These fields contain the outputs of the Analog-to-Digital Converter (ADC) located on the DSP board. The ADC sequentially samples one of the eight channels per ping group (the number of ping groups per ensemble is the maximum of the WP). These fields are zeroed at the beginning of the deployment and updated each ensemble at the rate of one channel per ping group. For example, if the ping group size is 5, then: END OF ENSEMBLE No. CHANNELS UPDATED Start All channels = 0 1 0, 1, 2, 3, 4 2 5, 6, 7, 0, 1 3 2, 3, 4, 5, 6 4 7, 0, 8, 2, 3 ... ... Here is the description for each channel: CHANNEL DESCRIPTION 0 XMIT CURRENT 1 XMIT VOLTAGE 2 AMBIENT TEMP 3 PRESSURE (+) 4 PRESSURE (-) 5 ATTITUDE TEMP
36 / 1	ADC CHANNEL 1	
37 / 1	ADC CHANNEL 2	
38 / 1	ADC CHANNEL 3	
39 / 1	ADC CHANNEL 4	
40 / 1	ADC CHANNEL 5	
41 / 1	ADC CHANNEL 6	
42 / 1	ADC CHANNEL 7	

		6 ATTITUDE 7 CONTAMINATION SENSOR Note that the ADC values may be “noisy” from sample to sample, but are useful for detecting long-term trends.
43-46 / 4	ERROR STATUS WORD (ESW) {CY?}	Contains the long word containing the bit flags for the CY? Command. The ESW is cleared (set to zero) between each ensemble. Note that each number above represents one bit set – they may occur in combinations. For example, if the long word value is 0000C000 (hexadecimal), then it indicates that both a cold wake-up (0004000) and an unknown wake-up (00008000) occurred. Byte 43 BITS 7 6 5 4 3 2 1 0 x x x x x x x 1 Bus Error exception x x x x x x 1 x Address Error exception x x x x x 1 x x Illegal Instruction exception x x x x 1 x x x Zero Divide exception x x x 1 x x x x Emulator exception x x 1 x x x x x Unassigned exception x 1 x x x x x x Watchdog restart occurred 1 x x x x x x x Battery Saver power Byte 44 x x x x x x x 1 Pinging x x x x x x 1 x Not Used x x x x x 1 x x Not Used x x x x 1 x x x Not Used x x x 1 x x x x Not Used x x 1 x x x x x Not Used x 1 x x x x x x Cold Wakeup occurred 1 x x x x x x x Unknown Wakeup occurred Byte 45 x x x x x x x 1 Clock Read error occurred x x x x x x 1 x Unexpected alarm x x x x x 1 x x Clock jump forward x x x x 1 x x x Clock jump backward x x x 1 x x x x Not Used x x 1 x x x x x Not Used x 1 x x x x x x Not Used 1 x x x x x x x Not Used Byte 46 x x x x x x x 1 Not Used x x x x x x 1 x Not Used x x x x x 1 x x Not Used x x x x 1 x x x Power Fail (Unrecorded) x x x 1 x x x x Spurious level 4 intr (DSP) x x 1 x x x x x Spurious level 5 intr (UART) x 1 x x x x x x Spurious level 6 intr (CLOCK) 1 x x x x x x x Level 7 interrupt occurred
47-48 / 2	SPARE	Reserved for TRDI use.
49-52 / 4	PRESSURE	Contains the pressure of the water at the transducer head relative to one atmosphere (sea level). Output is in decapascals. Scaling: LSD=1 deca-pascal; Range=0 to 4,294,967,295 deca-pascals
53-56 / 4	PRESSURE SENSOR VARIANCE	Contains the variance (deviation about the mean) of the pressure sensor data. Output is in deca-pascals. Scaling: LSD=1 deca-pascal; Range=0 to 4,294,967,295 deca-pascals
57 / 1	SPARE	Spare
58 / 1	RTC CENTURY	These fields contain the time from the Navigator’s Y2K compliant real-time clock (RTC) that the current data ensemble began. The TT-command (Set Real-Time Clock) initially sets the clock. The Navigator does account for leap years.
59 / 1	RTC YEAR	
60 / 1	RTC MONTH	
61 / 1	RTC DAY	
62 / 1	RTC HOUR	
63 / 1	RTC MINUTE	
64 / 1	RTC SECOND	
65 / 1	RTC HUNDREDTH	

16.1.6 Velocity Data Format

Byte#	Field name	Notes
1-2 / 2	Velocity ID	Stores the velocity data identification word (00 01h).
3-4 / 2	DEPTH CELL #1, VELOCITY 1	Stores velocity data for depth cell #1, velocity 1.
5-6 / 2	DEPTH CELL #1, VELOCITY 2	Stores velocity data for depth cell #1, velocity 2.
7-8 / 2	DEPTH CELL #1, VELOCITY 3	Stores velocity data for depth cell #1, velocity 3.
9-10 / 2	DEPTH CELL #1, VELOCITY 4	Stores velocity data for depth cell #1, velocity 4.
11-12 / 2	DEPTH CELL #2, VELOCITY 1	These fields store the velocity data for depth cells 2 through 128 (depending on the setting of the WN-command). These fields follow the same format as listed above for depth cell 1.
13-14 / 2	DEPTH CELL #2, VELOCITY 2	
15-16 / 2	DEPTH CELL #2, VELOCITY 3	
17-18 / 2	DEPTH CELL #2, VELOCITY 4	
…	↓ (SEQUENCE CONTINUES FOR UP TO 128 CELLS) ↓	
1019-1020 / 2	DEPTH CELL #128, VELOCITY 1	
1021-1022 / 2	DEPTH CELL #128, VELOCITY 2	
1023-1024 / 2	DEPTH CELL #128, VELOCITY 3	
1025-1026 / 2	DEPTH CELL #128, VELOCITY 4	

16.1.7 Correlation Magnitude, Echo Intensity and Percent-Good Data Format

Byte#	Field name	Notes
1-2 / 2	ID CODE	Stores the correlation magnitude/echo intensity/percent-good data identification word (00 02h/00 03h/00 04h).
3 / 1	DEPTH CELL #1, FIELD #1	Stores correlation magnitude/echo intensity/percent-good data for depth cell #1, beam #1.
4 / 1	DEPTH CELL #1, FIELD #2	Stores correlation magnitude/echo intensity/percent-good data for depth cell #1, beam #2.
5 / 1	DEPTH CELL #1, FIELD #3	Stores correlation magnitude/echo intensity/percent-good data for depth cell #1, beam #3.
6 / 1	DEPTH CELL #1, FIELD #4	Stores correlation magnitude/echo intensity/percent-good data for depth cell #1, beam #4.
7 / 1	DEPTH CELL #2, FIELD #1	These fields store correlation magnitude/echo intensity/percent-good data for depth cells 2 through 128 (depending on the WN-command) for all four beams. These fields follow the same format as listed above for depth cell 1.
8 / 1	DEPTH CELL #2, FIELD #2	
9 / 1	DEPTH CELL #2, FIELD #3	
10 / 1	DEPTH CELL #2, FIELD #4	
	↓ (SEQUENCE CONTINUES FOR UP TO 128 BINS) ↓	
511 / 1	DEPTH CELL #128, FIELD #1	
512 / 1	DEPTH CELL #128, FIELD #2	
513 / 1	DEPTH CELL #128, FIELD #3	
514 / 1	DEPTH CELL #128, FIELD #4	

16.1.8 Header Format

Byte#	Field name	Notes
1	Header ID	Stores the header identification byte (7Fh)
2	Data source ID	Stores the data source identification byte (7Fh for the Work-Horse ADCP)
3-4	Number of bytes in ensemble	This field contains the number of bytes from the start of the current ensemble up to, but not including, the 2-byte checksum

5	Spare	Undefined
6	Number of data types	This field contains the number of data types selected for collection. By default, fixed/variable leader, velocity, correlation magnitude, echo intensity, and percent good are selected for collection. This field will therefore have a value of six (4 data types + 2 for the Fixed/Variable Leader data).
7-8	Offset for data type #1	These field contains the internal memory address offset where the Navigator will store information for data type #n Adding "1" to this offset number gives the absolute Binary Byte number in the ensemble where Data Type #1 begins (the first byte of the ensemble is Binary Byte #1).
9-10	Offset for data type #2	
11-12	Offset for data type #3	
…	…	Sequence continues for up to n data types
(2n+5)-(2n+6) / 2	Offset for data type #n	

16.1.9 Bottom Track Data Format

Byte#	Field name	Notes
1-2 / 2	BOTTOM-TRACK ID	Stores the bottom-track data identification word (06 00h).
3-4 / 2	BT PINGS PER ENSEMBLE {BP}	Stores the number of bottom-track pings to average together in each ensemble (BP-command). If BP = 0, the ADCP does not collect bottom-track data. The ADCP automatically extends the ensemble interval (TE) if BP x TP > TE. Scaling: LSD = 1 ping; Range = 0 to 999 pings
5-6 / 2	BT DELAY BEFORE RE-ACQUIRE {BD}	Stores the number of ADCP ensembles to wait after losing the bottom before trying to reacquire it (BD-command). Scaling: LSD = 1 ensemble; Range = 0 to 999 ensembles
7 / 1	BT CORR MAG MIN {BC}	Stores the minimum correlation magnitude value (BCcommand). Scaling: LSD = 1 count; Range = 0 to 255 counts
8 / 1	BT EVAL AMP MIN {BA}	Stores the minimum evaluation amplitude value (BAcommand). Scaling: LSD = 1 count; Range = 1 to 255 counts
9 / 1	BT PERCENT GOOD MIN {BG}	Stores the minimum percentage of bottom-track pings in an ensemble that must be good to output velocity data (BGcommand).
10 / 1	BT MODE {BM}	Stores the bottom-tracking mode (BM-command).
11-12 / 2	BT ERR VEL MAX {BE}	Stores the error velocity maximum value (BE-command). Scaling: LSD = 1 mm/s; Range = 0 to 5000 mm/s (0 = did not screen data)
13-16 / 4	RESERVED	Reserved
17-18 / 2	BEAM#1 BT RANGE	Contains the two lower bytes of the vertical range from the ADCP to the sea bottom (or surface) as determined by each beam. This vertical range does not consider the effects of pitch and roll. When bottom detections are bad, BT Range = 0. See bytes 78 through 81 for MSB description and scaling. Scaling: LSD = 1 cm; Range = 0 to 65535 cm
19-20 / 2	BEAM#2 BT RANGE	
21-22 / 2	BEAM#3 BT RANGE	
23-24 / 2	BEAM#4 BT RANGE	
25-26 / 2	BEAM#1 BT VEL	The meaning of the velocity depends on the EX (coordinate system) command setting. The four velocities are as follows: a) Beam Coordinates: Beam 1, Beam 2, Beam 3, Beam 4 b) Instrument Coordinates: 1->2, 4->3, toward face, error c) Ship Coordinates: Starboard, Fwd, Upward, Error d) Earth Coordinates: East, North, Upward, Error
27-28 / 2	BEAM#2 BT VEL	
29-30 / 2	BEAM#3 BT VEL	
31-32 / 2	BEAM#4 BT VEL	
33 / 1	BEAM#1 BT CORR.	Contains the correlation magnitude in relation to the sea bottom (or surface) as determined by each beam. Bottom-track correlation magnitudes have the same format and scale factor as water-profiling magnitudes (Table 5).
34 / 1	BEAM#2 BT CORR.	
35 / 1	BEAM#3 BT CORR.	

36 / 1	BEAM#4 BT CORR.	
37 / 1	BEAM#1 EVAL AMP	Contains the evaluation amplitude of the matching filter used in determining the strength of the bottom echo. Scaling: LSD = 1 count; Range = 0 to 255 counts
38 / 1	BEAM#2 EVAL AMP	
39 / 1	BEAM#3 EVAL AMP	
40 / 1	BEAM#4 EVAL AMP	
41 / 1	BEAM#1 BT %GOOD	Contains bottom-track percent-good data for each beam, which indicate the reliability of bottom-track data. It is the percentage of bottom-track pings that have passed the ADCP's bottom-track validity algorithm during an ensemble. Scaling: LSD = 1 percent; Range = 0 to 100 percent
42 / 1	BEAM#2 BT %GOOD	
43 / 1	BEAM#3 BT %GOOD	
44 / 1	BEAM#4 BT %GOOD	
45-46 / 2	REF LAYER MIN {BL}	Stores the minimum layer size, the near boundary, and the far boundary of the BT water-reference layer (BL-command). Scaling (minimum layer size): LSD = 1 dm; Range = 0-999 dm Scaling (near/far boundaries): LSD = 1 dm; Range = 0-9999 dm
47-48 / 2	REF LAYER NEAR {BL}	
49-50 / 2	REF LAYER FAR {BL}	
51-52 / 2	BEAM#1 REF LAYER VEL	Contains velocity data for the water reference layer for each beam. Reference layer velocities have the same format and scale factor as water-profiling velocities (Table 34, page 139). The BL-command explains the water reference layer.
53-54 / 2	BEAM #2 REF LAYER VEL	
55-56 / 2	BEAM #3 REF LAYER VEL	
57-58 / 2	BEAM #4 REF LAYER VEL	
59 / 1	BM#1 REF CORR	Contains correlation magnitude data for the water reference layer for each beam. Reference layer correlation magnitudes have the same format and scale factor as water-profiling magnitudes (Table 5).
60 / 1	BM#2 REF CORR	
61 / 1	BM#3 REF CORR	
62 / 1	BM#4 REF CORR	
63 / 1	BM#1 REF INT	Contains echo intensity data for the reference layer for each beam. Reference layer intensities have the same format and scale factor as water-profiling intensities.
64 / 1	BM#2 REF INT	
65 / 1	BM#3 REF INT	
66 / 1	BM#4 REF INT	
67 / 1	BM#1 REF %GOOD	Contains percent-good data for the water reference layer for each beam. They indicate the reliability of reference layer data. It is the percentage of bottom-track pings that have passed a reference layer validity algorithm during an ensemble. Scaling: LSD = 1 percent; Range = 0 to 100 percent
68 / 1	BM#2 REF %GOOD	
69 / 1	BM#3 REF %GOOD	
70 / 1	BM#4 REF %GOOD	
71-72 / 2	BT MAX. DEPTH {BX}	Stores the maximum tracking depth value (BX-command). Scaling: LSD = 1 decimeter; Range = 80 to 9999 decimeters
73 / 1	BM#1 RSSI AMP	Contains the Receiver Signal Strength Indicator (RSSI) value in the center of the bottom echo as determined by each beam. Scaling: LSD ≈ 0.45 dB per count; Range = 0 to 255 counts
74 / 1	BM#2 RSSI AMP	
75 / 1	BM#3 RSSI AMP	
76 / 1	BM#4 RSSI AMP	
77 / 1	GAIN	Contains the Gain level for shallow water. See WJ-command.

78 / 1	(*SEE BYTE 17)	Contains the most significant byte of the vertical range from the ADCP to the sea bottom (or surface) as determined by each beam. This vertical range does not consider the effects of pitch and roll. When bottom detections are bad, BT Range=0. See bytes 17 through 24 for LSB description and scaling. Scaling: LSD = 65,536 cm, Range = 65,536 to 16,777,215 cm
79 / 1	(*SEE BYTE 19)	
80 / 1	(*SEE BYTE 21)	
81 / 1	(*SEE BYTE 23)	
82-85 / 4	RESERVED	Reserved

16.1.10 Checksum Data Format

Byte#	Field name	Notes
1-2 / 2	CHECKSUM DATA	This field contains a modulo 65535 checksum. The Workhorse computes the checksum by summing all the bytes in the output buffer excluding the checksum.

16.1.11 Notes

- The ADCP packs velocity data into a two-byte, two's-complement integer [-32768, 32767] with the LSB sent first. The ADCP scales velocity data in millimetres per second (mm/s). A value of –32768 (8000h) indicates a bad velocity.
- In order to log PD0 messages and no other DVL message types, the message name 'DVLPD0' can be used

.

Velocity Convention

For PD0 output with Coordinate Transformation configured as EX01010, the velocity conventions are as follows:

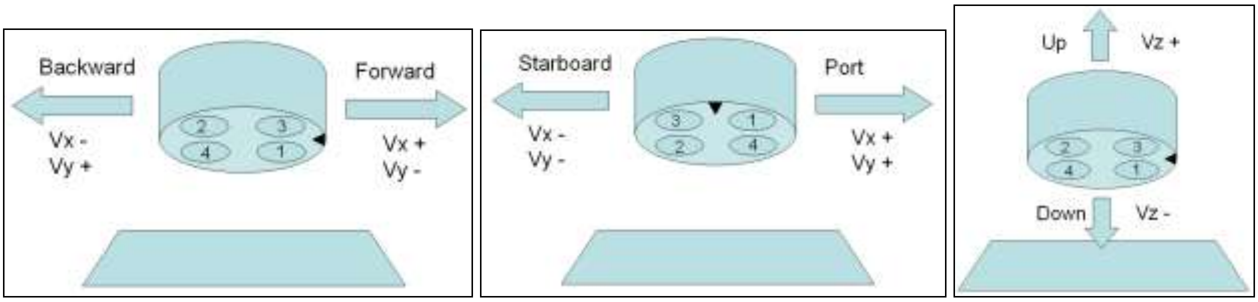


Figure 16-1 PD0 velocity conventions

Conversion from PD0 to PD4

The X, Y and Z velocities for the Bottom and Reference layer are negated internally before using their values in the same way as for a PD4 input.
However, if the original value is 0x8000 (-32768) then the value is left alone and not negated as this value indicates a problem.
Make use of the offsets in the header to find the start of a section and then check the ID to determine which section it is.
Note, the below uses byte numbering from 1.
If there is no Bottom-Track Data then the Vel-Btm and Vel Ref values are set to 0x8000 and the Bm range values are set to 0. As a Bm range value uses 3 bytes in a PD0 and only 2 in a PD4, if the MSB is non-zero in the PD0 then the Bm range value is set to 0 and thus the observation is not used in the INS solution as per the SDS Kalman doc.

Byte 1 = 0x7D
Byte 2 = 0
Byte 3 = 45
Byte 4 = 0
Byte 5 = (Fixed Leader Data's SystemConfig, byte[5] & 0x7) |
(Fixed Leader Data's CoordinateTransformEX, byte[26] & 0x1E)<<3
Byte 6 = Bottom-Track Data, byte[25] // Vel-Btm
Byte 7 = Bottom-Track Data, byte[26]
Byte 8 = Bottom-Track Data, byte[27]

Byte 9 = Bottom-Track Data, byte[28]
Byte 10 = Bottom-Track Data, byte[29]
Byte 11 = Bottom-Track Data, byte[30]
Byte 12 = Bottom-Track Data, byte[31]
Byte 13 = Bottom-Track Data, byte[32]
Byte 14 = Bottom-Track Data, byte[17] // Bm
Byte 15 = Bottom-Track Data, byte[18]
Byte 16 = Bottom-Track Data, byte[19]
Byte 17 = Bottom-Track Data, byte[20]
Byte 18 = Bottom-Track Data, byte[21]
Byte 19 = Bottom-Track Data, byte[22]
Byte 20 = Bottom-Track Data, byte[23]
Byte 21 = Bottom-Track Data, byte[24]
Byte 22 = Sets bit0 if (Bottom-Track Data, byte[33] < Bottom-Track Data, byte[7])
Sets bit2 if (Bottom-Track Data, byte[34] < Bottom-Track Data, byte[7])
Sets bit4 if (Bottom-Track Data, byte[35] < Bottom-Track Data, byte[7])
Sets bit6 if (Bottom-Track Data, byte[36] < Bottom-Track Data, byte[7])
Sets bit1 if (Bottom-Track Data, byte[37] < Bottom-Track Data, byte[8])
Sets bit3 if (Bottom-Track Data, byte[38] < Bottom-Track Data, byte[8])
Sets bit5 if (Bottom-Track Data, byte[39] < Bottom-Track Data, byte[8])
Sets bit7 if (Bottom-Track Data, byte[40] < Bottom-Track Data, byte[8])
Byte 23 = Bottom-Track Data, byte[51] // Vel Ref Layer
Byte 24 = Bottom-Track Data, byte[52]
Byte 25 = Bottom-Track Data, byte[53]
Byte 26 = Bottom-Track Data, byte[54]
Byte 27 = Bottom-Track Data, byte[55]
Byte 28 = Bottom-Track Data, byte[56]
Byte 29 = Bottom-Track Data, byte[57]
Byte 30 = Bottom-Track Data, byte[58]
Byte 31 = Bottom-Track Data, byte[47] // Ref layer start
Byte 32 = Bottom-Track Data, byte[48]
Byte 33 = Bottom-Track Data, byte[49] // Ref layer end
Byte 34 = Bottom-Track Data, byte[50]
Byte 35 = 0xFF // Ref layer status
Byte 36 = Variable Leader Data's RTC Hour[TS], byte[8]
Byte 37 = Variable Leader Data's RTC Minute[TS], byte[7]
Byte 38 = Variable Leader Data's RTC Second[TS], byte[9]
Byte 39 = Variable Leader Data's RTC Hundredths[TS], byte[10]
Byte 40 = Variable Leader Data's BIT Result, byte[13]
Byte 41 = Variable Leader Data's BIT Result, byte[14]
Byte 42 = Variable Leader Data's Speed of Sound, byte[15]
Byte 43 = Variable Leader Data's Speed of Sound, byte[16]
Byte 44 = Variable Leader Data's Temperature, byte[27]
Byte 45 = Variable Leader Data's Temperature, byte[28]
Byte 46 = Checksum LSB
Byte 47 = Checksum MSB

16.2 PD4/PD5

Name	PD4/PD5
Type	Binary – Proprietary
Input / Output / Log	Input
Length	Variable
Multiplex Message ID	140

16.2.1 Description

This proprietary message is generated by the Teledyne RDI DVL (e.g. Workhorse Navigator) when it has been configured using the PD4 or PD5 command. See "WorkHorse Commands and Output Data Format" for details. Note: the encoding for this message can be binary or hex-ASCII.

The PD4/5 output data format assumes the bottom is stationary and the instrument is moving, the PD0 assumes the opposite.

16.2.2 PD4 Format

Byte#	Field name	Units	Notes
1 / 1	DVL DATA ID 7Dh	-	Stores the DVL (speed log) identification word (7Dh).
2 / 1	DATA STRUCTURE	-	Identifies which data pattern will follow based on the Pdcommand. 0 = PD4 = Bytes 1 through 47. 1 = PD5 = Bytes 1 through 45 and bytes 46 through 88 from PD5 table
3-4 / 2	NO. OF BYTES	-	Contains the number of bytes sent in this data structure, not including the final checksum.
5 / 1	SYSTEM CONFIG	-	Defines the DVL hardware/firmware configuration. Convert to binary and interpret as follows. BIT 76543210 00xxxxxx BEAM-COORDINATE VELOCITIES 01xxxxxx INSTRUMENT-COORDINATE VELOCITIES 10xxxxxx SHIP-COORDINATE VELOCITIES 11xxxxxx EARTH-COORDINATE VELOCITIES xx0xxxxx TILT INFORMATION NOT USED IN CALCULATIONS xx1xxxxx TILT INFORMATION USED IN CALCULATIONS xxx0xxxx 3-BEAM SOLUTIONS NOT COMPUTED xxx1xxxx 3-BEAM SOLUTIONS COMPUTED xxxxx010 300-kHz DVL xxxxx011 600-kHz DVL xxxxx100 1200-kHz DVL
6-7 / 2	X-VEL BTM	mm/s	These fields contain the velocity of the vessel in relation to the bottom in mm/s. Positive values indicate vessel motion to east (X), north (Y), and up (Z). LSD = 1 mm/s (Note 1).
8-9 / 2	Y-VEL BTM		
10-11 / 2	Z-VEL BTM		
12-13 / 2	E-VEL BTM		
14-15 / 2	BM1 RNG TO BTM	cm	These fields contain the vertical range from the ADCP to the bottom as determined by each beam. This vertical range does not compensate for the effects of pitch and roll. When a bottom detection is bad, the field is set to zero. Scaling: LSD = 1 centimetre; Range = 0 to 65535 cm
16-17 / 2	BM2 RNG TO BTM		
18-19 / 2	BM3 RNG TO BTM		
20-21 / 2	BM4 RNG TO BTM		
22 / 1	BOTTOM STATUS	-	This field shows the status of bottom-referenced correlation and echo amplitude data. Convert to binary and interpret as follows. A zero code indicates status is OK. BIT 76543210 1xxxxxxx BEAM 4 LOW ECHO AMPLITUDE x1xxxxxx BEAM 4 LOW CORRELATION xx1xxxxx BEAM 3 LOW ECHO AMPLITUDE xxx1xxxx BEAM 3 LOW CORRELATION xxxx1xxx BEAM 2 LOW ECHO AMPLITUDE xxxxx1xx BEAM 2 LOW CORRELATION xxxxxx1x BEAM 1 LOW ECHO AMPLITUDE xxxxxxx1 BEAM 1 LOW CORRELATION
23-24 / 2	X-VEL REF LAYER	mm/s	These fields contain the velocity of the vessel in relation to the water-mass reference layer in mm/s. Positive values indicate vessel motion to east (X), north (Y), and up (Z). LSD = 1 mm/s (Note 2)
25-26 / 2	Y-VEL REF LAYER		
27-28 / 2	Z-VEL REF LAYER		

29-30 / 2	E-VEL REF LAYER		
31-32 / 2	REF LAYER START	0.1m	These fields contain the starting boundary (near surface) and the ending boundary (near bottom) of the water-mass reference layer (BL-command). If the minimum size field is zero, the ADCP does not calculate reference-layer data. Scaling: LSD = 1 dm; Range = 0-9999 dm
33-34 / 2	REF LAYER END		

35 / 1	REF LAYER STATUS	-	This field shows the status of reference layer depth and correlation data. Convert to binary and interpret as follows. A zero code indicates status is OK.
			`BIT 76543210`
			`xxx1xxxx ALTITUDE IS TOO SHALLOW`
			`xxxx1xxx BEAM 4 LOW CORRELATION`
			`xxxxx1xx BEAM 3 LOW CORRELATION`
			`xxxxxx1x BEAM 2 LOW CORRELATION`
			`xxxxxxx1 BEAM 1 LOW CORRELATION`

36 / 1	TOFP-HOUR	Hr	These fields contain the time of the first ping of the current ensemble.
37 / 1	TOFP-MINUTE	Min	
38 / 1	TOFP-SECOND	Sec	
39 / 1	TOFP-HUNDREDTHS	0.01 sec	

40-41 / 2	BIT RESULTS	-	BIT Results These fields contain the results of the ADCP's Built-in Test function.
			A zero code indicates a successful BIT result.
			(BYTE 41 RESERVED FOR FUTURE USE)
			`BYTE 40  BYTE 41`
			`1xxxxxxx xxxxxxxx = RESERVED`
			`x1xxxxxx xxxxxxxx = RESERVED`
			`xx1xxxxx xxxxxxxx = RESERVED`
			`xxx1xxxx xxxxxxxx = DEMOD 1 ERROR`
			`xxxx1xxx xxxxxxxx = DEMOD 0 ERROR`
			`xxxxx1xx xxxxxxxx = RESERVED`
			`xxxxxx1x xxxxxxxx = DSP ERROR`
			`xxxxxxx1 xxxxxxxx = RESERVED`

42-43 / 2	SPEED OF SOUND	m/s	Contains either manual or calculated speed of sound information (EC-command). Scaling: LSD = 1 metres per second; Range = 1400 to 1600 m/s
44-45 / 2	TEMPERATURE	0.01°C	Temperature Contains the temperature of the water at the transducer head. Scaling: LSD = 0.01 C; Range = -5.00 to +40.00 C
46-47 / 2	CHECKSUM	-	This field contains a modulo 65536 checksum. The ADCP computes the checksum by summing all the bytes in the output buffer excluding the checksum. NOTE: This field contains the checksum only when the PD4-command is used.

16.2.3 PD5 Format

Byte#	Field name
46 / 1	SALINITY
47-48 / 2	DEPTH
49-50 / 2	PITCH
51-52 / 2	ROLL
53-54 / 2	HEADING
55-58 / 4	DISTANCE MADE GOOD/BTM (EAST)
59-62 / 4	DISTANCE MADE GOOD/BTM (NORTH)
63-66 / 4	DISTANCE MADE GOOD/BTM (UP)
67-70 / 4	DISTANCE MADE GOOD/BTM (ERROR)
71-74 / 4	DISTANCE MADE GOOD/REF (EAST)
75-78 / 4	DISTANCE MADE GOOD/REF (NORTH)
79-82 / 4	DISTANCE MADE GOOD/REF (UP)
83-86 / 4	DISTANCE MADE GOOD/REF (ERROR)

87-88 / 2	CHECKSUM

16.2.4 Notes

- All Data is transmitted LSB first
- In order to log PD4/5 messages and no other DVL message types, the message name ‘DVLPD5PD5’ can be used

Note 1 –

- The ADCP packs velocity data into a two-byte, two’s-complement integer [-32768, 32767] with the LSB sent first. The ADCP scales velocity data in millimetres per second (mm/s). A value of –32768 (8000h) indicates a bad velocity.
- Bottom or reference-layer velocities will be all valid or all invalid. That is, if the X-velocity is valid then the Y and Z-velocities are valid; if X is not valid, Y and Z are not valid.
- The ADCP allows 3-beam transformations when the fourth beam is invalid.
- Indication of a 3-beam transformation for bottom-track is 1. valid bottom velocities and 2. one and only one beam’s range to bottom is marked bad (zero).

Note 2 – There is no indication that a 3-beam transformation was performed for water reference layer velocity data.

Notes concerning LinkQuest NavQuest 600 Micro DVL

PD4 message output from the LinkQuest NavQuest 600 Micro DVL has the following differences from the standard (RDI DVL) message output:

- System configuration field in PD4 message is zero for Instrument co-ordinates velocities
- Value of Vertical range to bottom from Beam N (n = 1,2,3,4) can be negative or zero. If value is negative, it means that bottom lock is lost by Beam N. There are four beams in NavQuest system, but only three beams are required for the solution
- Speed of Sound field is always zero

Velocity Convention

For PD4/5 output with Coordinate Transformation configured as EX01010, the velocity conventions are as follows:

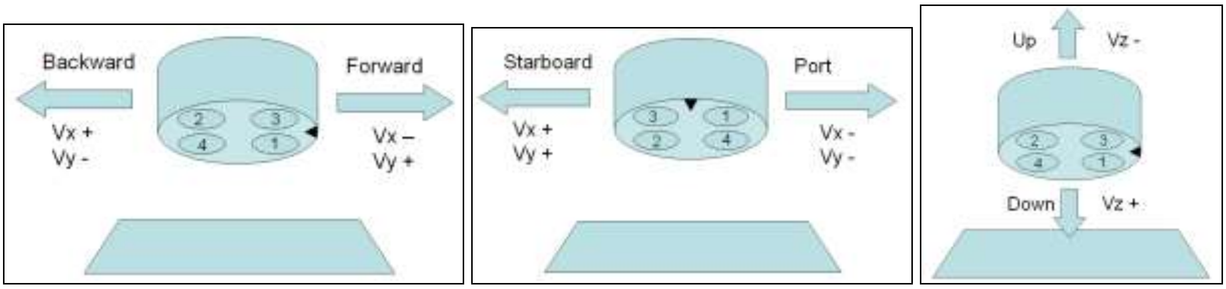


Figure 16-2 PD4/5 velocity conventions

Logging

When this message is logged, it will be prepended with a 48 bit system time stamp (LSB = 10^{-6} sec) indicating the **time of arrival** at the INS of the DVL measurement, followed by an additional 48 bit timestamp (LSB = 10^{-6} sec) indicating the **trigger time** of the DVL measurement. If the DVL is not triggered then the trigger time is set to 0.