DRX Interface Control Document

Version 2.77

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
Matthew Taylor	19/02/14	Initial Version	1.0
Gavin Sentch	25/02/14	First Pass At Messages and Content	1.01
Gavin Sentch	27/02/14	Modifications after discussion.	1.02
Gavin Sentch	28/02/14	Further modifications after R&D discussion. Current version for development	1.1
Gavin Sentch	5/03/14	Integrated Generic ICD fields into this document to separated them.	1.11
Matthew Taylor	10/04/14	Updated IQ data packet description	1.12
Keith Fletcher	04/07/2014	Updated sensor updates message Added FPGA Direct Memory Access protocol	1.13
Gavin Sentch	16/7/2014	Added in message for FPGA replay	1.14
Gavin Sentch	17/7/2014	Added replay type to ping request, updated FPGA replay description	1.15
Keith Fletcher	23/07/2014	Added “Proposed Future” FPGA Direct Memory Access protocol	1.16
Keith Fletcher	21/08/2014	Modified the “On Connection SetUp” section	1.17
Keith Fletcher	01/09/2014	Created Annex for ENL Development protocols	1.18
Keith Fletcher	28/10/2014	Modified the SENS_SET message to support Id’s	1.19
Gavin Sentch	5/12/2014	Added mapping tile messages	1.20
Gavin Sentch	11/12/2014	Modified mapping messages	1.21
Gavin Sentch	12/12/2014	Removed mapping messages, moved to Map Server Spec for now as not really a DRX message	1.22
Gavin Sentch	12/01/2015	Added note on how ack/nack messages will be implemented.	1.23
Gavin Sentch	14/01/2015	Removed onConnection process Moved sea floor reference from sensor setup to processing options Added tide options to processing options	1.24
Matthew Taylor	20/01/2015	Modified title, added MSG_REQ_, added flags for all messages, updated flags within messages to cover all non-block variables in the message	1.25
Matthew Taylor	21/01/2015	Various additions and modifications across all messages. Added examples of ICD use. Modified System Message Flags.	1.26
Matthew Taylor	22/01/2015	Added packet type to section headers. Modified SERIALST for CAN, Ethernet and to support multiple packet responses	1.27
Ee Chen	23/01/2015	Changed SERIALST to PORT_SET because it applies to many more ports than just serial, and cleared up some flags/fields. Added section 3.5 on data order.	1.28
Ee Chen	23/01/2015	Added 64-bit timestamp to common header	1.29
Matthew Taylor	23/01/2015	Re reviewed – modified SENSSET examples and packet definition, tidy up of examples and overview after changes.	1.30
Ee Chen	23/01/2015	Modified FISH_SET flags to be more consistent.	1.31
Matthew Taylor	27/01/2015	Added command type field to MSG_REQ	1.32
Matthew Taylor	28/01/2015	SENSUPDAT date rejig, PRO_OPTN TVG typo	1.33
Joshua Williams	28/01/2015	Document tidy-ups and consistency changes	1.34
Matthew Taylor	30/01/2015	Added note on unexpected data due to multiple clients	1.35
Matthew Taylor	02/02/2015	Fixed SENS_SET reserved field and flags	1.36

Matthew Taylor	09/02/2015	Removed beam index from SONARDISP – removed note regarding fixed width BATHYCOR. Fixed example MSG_REQ_ lengths.	1.37
Keith Fletcher	25/02/2015	Modified the SENS_SET and PORT_SET interpretation of port, device and protocol to be consistent.	1.38
Ee Chen	27/02/2015	Adding sound speed/velocity to PRO_OPTN	1.39
Ee Chen	2/03/2015	Modified PRO_OPTN by adding water temperature, and moved the water salinity field next to water temperature/sound velocity field. Updated reference to discovery process document, and internal links to proper svn server.	1.40
Ee Chen	2/03/2015	Added comment clarifying what happens if sound velocity, water temperature, and water salinity were all valid/invalid in the PRO_OPTN message.	1.41
Keith Fletcher	2/03/2015	Clarified the output of SENUPDAT to be 10Hz, non-polled.	1.42
Gavin Sentch	4/03/2015	Added beam width to SONADISP as this is required to allow rendering of all beams.	1.43
Joshua Williams	12/03/2015	Updates to SYS_INFO to offer more comprehensive information.	1.44
Ee Chen	24/3/2015	Added replay sample rate to PING_REQ, which is used only for replay data.	1.45
Ee Chen	24/3/2015	Added replay sample count to PING_REQ – used only for replay data.	1.46
Matthew Taylor	26/3/2015	Updated Spreading Loss field in PRO_OPTN	1.47
Matthew Taylor	1/4/2015	Updated Ping State description in SONASTAT	1.48
Stefan	2/4/2014	Corrected description, PING_REG range not depth	1.49
Matthew Taylor	21/04/15	Various clarifications to field descriptions through the entire document PRO_OPTN – Updated fields, Expanded TVG	1.50
Gavin Sentch	28/04/15	Added BATHYOPT, RAW_SENS and BATHYRAW. Cleaned up all messages, rearranged flags to be sequential. Some message content has been re-arranged.	2.00
Matthew Taylor	29/04/15	Further clean up – redefined timestamp fields, clarified sample timing	2.01
Gavin Sentch	29/04/15	Added max beams and detection type to BATHYCOR and BATHYRAW Added setting detection type to BATHYOPT	2.02
Gavin Sentch	30/04/15	Corrected errors and added required fields for BATHYRAW to handle multiple detection points.	2.03
Matthew Taylor	30/04/15	Changes to FISH_SET, added FISHSTAT Removed add/delete from FISH_SET, described threshold.	2.04
Gavin Sentch	4/05/15	SONASTAT – changed msg flags as was missing for tx bandwidth BATHYCOR – Max Beams and N changed from U16 to U32 FISHSTAT – Sample Type move below M field (flags changed) WCT_SETT – Modified flags as WC Seafloor flag missing RAW_SENS – Removed Reserved field and changed protocol identifier from U8 to U16. (flags changed)	2.05
Keith Fletcher	8/05/15	SENS_SET – typos on examples SENS_SET – add section **Error! Reference source not found. Error! Reference source not found.**	2.06
Keith Fletcher	8/05/15	SENS_SET – add comments to clarify sending of NACK messages.	2.07
Keith Fletcher	8/05/15	BATHYOPT Added units and limits to some of parameters	2.08
Gavin Sentch	26/05/15	SONADISP Removed the beam width field, added information on how to provide correct output for wedge display.	2.09
Stefan	23/6/15	Corrected RAW_SENS packet Time field to read nanoseconds rather than milliseconds	2.10

Matthew Taylor	2/7/15	Updated Appendix D port protocol information	2.11
Stefan	7/7/15	PRO_OPTN update for Gain Mode, Spreadling loss and Absorption Loss fields	2.12
Gavin Sentch	15/7/15	Changed the Quality field for the BATHYCOR message to be two U8 values, one for detection point and one for backscatter	2.13
Matthew Taylor	16/7/15	Sonar data fields updated to Q7 (/128 instead of /256 to allow for less internal conversions and greater dynamic range).	2.14
Stefan	22/7/15	Corrected F32 to U32 on PRO_OPTN::Absorption Loss, WGT-1443 Corrected descriptions for spreading and absorption losses Updated PING_REG Range Mode and Power Mode with placeholder modes (not yet supported) Updated description on Tide Level	2.15
Stefan	22/7/15	Moved sensor setup to Annex	2.16
Stefan	20/8/15	Added comments to FISH_SET and WCT_SETT Remeved FS Mode undefined from FISH_SET Removed WC Mode undefined from WCT_SETT Removed TideC Mode undefined from BATHYOPT Added FS Seafloor Distance to FISH_SET Removed Ping Number from SENUPDAT (replaced with Reserved)	2.17
Gavin Sentch	21/08/15	Added latitude, longitude and heading to SIDESCAN Added new command to MSG_REQ_ to get current requested messages from the connected DRX.	2.18
Richard Maxwell	4/9/15	Added Supervisor 128 bit serial number to SYS_INFO	2.19
Stefan	15/9/15	Removed comments regarding future modes in PING_REQ and PRO_OPTN	2.20
Richard Maxwell	29/9/15	Added Network Link speed to SONASTAT message.	2.21
Matthew Taylor	6/10/15	BATHYRAW: Fixed duplication and quality matches BATHYCOR	2.22
Stefan	8/10/15	FISHDATA: Update to include lat, lon, bearing and beam angle, width	2.23
Matthew Taylor	12/11/15	BATHYRAW: Explicitly specified beam angle field signs.	2.24
Matthew Taylor	30/11/15	FISH_SET: Made explicit use of beam width to zero to remove a beam and how this must be done.	2.25
Matthew Taylor	10/12/15	SONADATA V2, FISHSTAT: Added position and bearing fields. SIDESCAN – fixed flag order.	2.26
Keith Fletcher	13/12/15	Change SeaLevelRef to Transducer Draft in BATHYOPT	2.27
Matthew Taylor	15/12/15	Added automatic detection type to BATHYCOR	2.28
Matthew Taylor	12/01/16	Added PPSCNFIG and PPSSTATS messages	2.29
Gavin Sentch	13/01/16	Added in current version numbers for ICD messages	2.30
Matthew Taylor	17/02/16	Removed PPS control from BATHYOPT	2.31
Matthew Taylor	5/04/16	Updated/Fixed Rx Time Zero diagram	2.32
Matthew Taylor	7/04/16	Added new message LIC_FEAT to report DRX current license configuration	2.33
Matthew Taylor	20/4/16	Description change on BATHYCOR and WCT_DATA packet, added tide to WCT_DATA (no size change).	2.34
Matthew Taylor	2/5/16	Added Clutter value to PRO_OPTN	2.35
Matthew Taylor	11/05/16	Changed IF in PRO_OPTN to one UINT32 – replaced threshold with Aeration filter control.	2.36

Keith Fletcher	07/06/16	Added flags to BATHYCOR DPS backscatter and Fish	2.37
Matthew Taylor	13/06.16	BATHYOPT Start/End angle descriptions fixed.	2.38
Keith Fletcher	16/06/16	Added IMS_OPTN message	2.39
Keith Fletcher	17/06/16	Modified IMS_OPTN message	2.40
Matthew Taylor	30/06/16	Added BATHYOPT Glitch Filter Control	2.41
Tian Gong	30/06/16	Added new sensor models to Appendix A	2.42
Gavin Sentch	11/07/16	Modified Header and Footer values to better represent little endin U32	2.43
Matthew Taylor	2/8/16	Updated aeration filter limits in PRO_OPTN	2.44
Matthew Taylor	17/8/16	Improved descriptions of thresholding for FISHSET and WCT_SET	2.45
Keith Fletcher	06/09/2016	WGT-3269 Match ICD for the new LIC_FEAT and SYS_INFO changes required for 80kHz	2.46
Keith Fletcher	07/09/2016	Upgrade the SYS_INFO and LIC_FEAT version numbers + put in method of identifying if transducer data has been read from the transducer or default file.	2.46
Tian Gong	9/09/2016	Add water temperature to SENUPDAT	2.47
Keith Fletcher	27/09/2016	Add Backscatter progress to SONASTAT	2.48
Keith Fletcher	27/09/2016	Modify Backscatter descriptions in BATHYCOR and BATHYRAW.	2.49
Matthew Taylor	12/10/2016	BATHYCOR 2D Fish field description elaborated	2.50
Matthew Taylor	20/10/2016	Made U32 packed field byte order more explicit WCT_DATA changed to S16Q7 like every other TS value WCT_SETT changed Threshold mode and description	2.51
Tian Gong	3/11/2016	WCT_SETT added mode for seafloor gating WGT-3403	2.52
Keith Fletcher	15/11/2016	Add all undocumented Range Limits	2.53
Matthew Taylor	16/11/2016	Updated LIC_FEAT with full license list	2.54
Stefan and Matthew Taylor	18/11/2016	Description updates; IMS_OPT, LIC_FEAT proposed and implemented Typical values and range limits reviewed	2.55
Matthew Taylor	24/11/2016	Second pass updating Typical values	2.56
Richard Maxwell	15/03/2017	Added IMS_OPTN	2.57
Richard Maxwell	24/03/2017	Updated SONASTAT to add flag for existence of calibration file	2.58
Richard Maxwell	30/03/2017	Updated IMS_OPTN to enable correct reporting of manual values	2.59
Matthew Taylor	30/03/2017	SENUPDAT now using Version 3 – added Transducer Draft	2.60
Matthew Taylor	7/4/2017	Added Maximum Acquisition Range field to LIC_FEAT	2.61
Matthew Taylor	12/4/2017	Improved/updated description for RAW_SENS	2.62
Matthew Taylor	18/4/2017	Added Maximum Acquisition Range flag field to LIC_FEAT	2.63
Matthew Taylor	30/5/2017	Updated KEYPULSE message format Added KP_STATS message proposed format.	2.64
Tian Gong	6/06/2017	Added RTK settings to BATHYOPT and GPS quality output in SENUPDAT	2.65
Matthew Taylor	13/06/2017	KP_STATS became KPSTATUS and modified message content	2.66
Matthew Taylor	27/06/2017	Fixed wording of BATHYRAW to reference fields of its ICD header rather than specifying a fixed spreading loss.	2.67
Matthew Taylor	29/06/2017	KEYPULSE tidy up	2.68
Gavin Sentch	6/7/2017	BATHYCOR and BATHYRAW packet version updated for DRX software version 3.2 due to new backscatter generation process to allow 3rd parties to determine prior backscatter values.	2.69
Richard Maxwell	1/8/2017	Added Transmitter type to SYS_INFO ICD. Marked SYS_INFO Protocol version to be obsolete (as it contains outdated information).	2.70

Matthew Taylor	30/8/2017	Re added Swap Array bit to Corrections flags of BathyOpt.	2.71
Matthew Taylor	4/9/2017	Fixed description of Time (accurate) field of RAW_SENS	2.72
Tian Gong	7/09/2017	Updated SENUPDAT – GPS fix moved to internal GPGGA/PTNLGGK data struct. Replaced with antenna height and geoidal separation values output for RTK read from GGA	2.73
Matthew Taylor	12/9/2017	Updated LIC_FEAT Licences list	2.74
Matthew Taylor	9/10/2017	KPCONFIG Updated	2.75
Matthew Taylor	28/11/2017	LIC_FEAT updated	2.76
Matthew Taylor	22/11/2017	Added RAW_SENS explanatory note	2.77
Richard Maxwell	28/2/2018	Added depth of keel below transducer to BATHYOPT	2.78
Richard Maxwell	7/3/2018	Updated RAW_SENS to version 3, update protocol. Added Valeport to Appendix A.	2.79

1. Overview

1.1 Purpose and Scope

This document specifies the format of data packets sent to and from the G3 processor and a client. The primary purpose of this link is to send high bandwidth processed data in one direction and to allow control information to be sent to the processor.

1.2 References

1.3 Abbreviations / Acronyms

Abbreviation / Acronym	Description
U8	Unsigned 8bits (one byte)
I8	Signed 8bit character (printable)
U16	Unsigned 16bit word
S16	Signed 16bit word
U32	Unsigned 32bit word
S32	Signed 32bit word
F32	Single precision floating point number 32bit (IEEE 754)
U64	Unsigned 64bit word
S64	Signed 64bit word
F64	Double precision floating point number 64bit (IEEE 754)
RMS	Root Mean Square
PM	Processing Module – the device/application which provides processed and raw sonar data streams
UI	User Interface – the device/application which displays processed data and controls sonar operation
TBD	“To be decided”
SNR	Signal to Noise Ratio

Table 1: Abbreviations / Acronyms

2. Packet Types

Description	ID (ASCII)	Brief description
Connection Control Messages		
Message Request	MSG_REQ_	Used to specify data packets to receive from a DRX – send this on first connection and anytime you want to increase or reduce the amount of data received from the DRX.
Dynamic Update Messages		
Sonar Status	SONASTAT	Core sonar sampling information
Sensor updates	SENUPDAT	Report of current ship's sensor information
General Message	GEN_MESG	Reports textual information, warnings and errors
PPS Status	PPSSTATS	Reports PPS related status information
DRX Configuration Messages		
Ping Request	PING_REQ	Used to request the start/stop of acquisition and control ping parameters
Sensor Settings	SENS_SET	Used to request and modify sensor settings
Port Setup	PORT_SET	Allows for data layer configuration of serial ports
Processing Options	PRO_OPTN	Configures core processing options of the DRX
Bathy Options	BATHYOPT	Configures bathy processing options of the DRX
PPS Configuration	PPSCNFIG	Configures PPS time synchronization of the DRX
DRX System Request Messages		
System Information	SYS_INFO	Reports DRX system versions and configures static information such as the transducer serial number
License Information	LIC_FEAT	Reports current DRX system parameters and features currently licensed.
Corrected UI Display Data		
Fish Finder Setting	FISH_SET	Configuration of fish finder beams
Water Column Setting	WCT_SETT	Configuration for water column detection
Sonar Data	SONADISP	Processed display data for the sonar display
Sidescan Data	SIDEDATA	Processed sidescan display data
Corrected Bathymetry (for waterfall and mapping)	BATHYCOR	Processed corrected sea floor points in geographic three dimensional space
Fish Finder Data	FISHDATA	Processed display data for the fish finder display(s)
Fish Statistics	FISHSTAT	Processed count of fish found per beam, binned by target strength (size)
Water column Targets	WCT_DATA	Processed water column targets for display
Uncorrected Data		
Raw Sensor Data	RAW_SENS	Time stamped sensor data as it was received on the DRX ports.
Raw Bathy Data	BATHYRAW	Time stamped uncorrected bathy data.

Table 2: Table of general use messages, packet identifiers and descriptions

2.1 Connection Process

All communications are done over TCP. You are required to create a TCP client and connect to the DRX IP Address and Port. The address and port of active DRX units can be found using a DRX discovery process [6].

A DRX can also be given a fixed IP address. By default, the port number to connect to is 55555 – but this can change.

2.2 Requesting Data

In order to receive any data the client must send the MSG_REQ_ packet outlined in section 4.1.1. Only messages which are required should be set to minimize network bandwidth usage.

2.3 Unexpected data

In normal operation it may be that there are multiple clients active, thus a client can receive an acknowledgement to a command that a different client sent. This allows parameters that are changed by one client to be propagated to all clients in a system.

NOTE: An acknowledgement will not be returned unless the selected message has been requested by the client with a MSG_REQ_ message.

NOTE: MSG_REQ_ is an exception to the above rule in that MSG_REQ acknowledgement messages are only received by the client that initiates them. By default MSG_REQ messages are enabled to all clients to allow response to the MSG_REQ_ message.

2.4 Data Rates and Network Bandwidth

The update rates of sonar ping based messages will be updated at the rate the DRX unit is pinging. This is determined by the acquisition range the DRX unit is set to.

Maximum theoretical data rate for sonar based messages can be calculated from the maximum DRX sample rate and maximum number of samples.

The maximum upstream sample rate used by the DRX unit is 40 kHz.

For the SONADISP packet, the maximum number of samples (M) is 2048. The size of each sample is 2 bytes long times the number of beams (64).

Maximum Packet payload = Samples x sample size = 2048 x 64 x 2 = 262144 bytes

Maximum Rate for Maximum Packet = Maximum Sample Rate (Hz) / Maximum Number of Samples

Which = 40000 / 2048 = 19.53 per second.

Maximum Data Rate = Sample size x Sample Rate = 2 * 64 * 40000 = **5.12 MB/sec.**

Note: These numbers do not account for fixed SONADISP packet data

2.5 Common Packet Header

All packets will have a header with identical formatting to allow processing at both ends with relative ease. The design is to allow:

- Resynchronization
- TCP clients to skip over unknown packets using length field
- Unique human readable message codes
- Changes to the ICD if required
- The same message format to be used for multiple purposes

The start, length, type and version fields should be self-explanatory in both their use and utility. To allow the same messages to be used to send and receive information the indication on the purpose of a message is contained in the <msg flags> field.

The eight least significant bits are used as a code which describes both the message purpose and indicates what the flags in the subsequent bits indicate. The system message codes are constant across all messages. The remaining 24 bits are designated message flag bits, and are used to flag individual fields in the message as valid or not.

System message codes:

System Code	Description
0	**Invalid** – not a valid code – should never occur
1	**Command Message** – This is a message which contains a task to be performed by the recipient which is initiated by the client.
2	**Request Status** - This flag can be used to force a Reported Status message. Useful for when you connect to a system and require information. Sent from client.
3-127	Reserved for requests from client.
128	**Acknowledge** – sent from server, flags identify which fields in the command/request were acknowledged. Acknowledge is used to reply to both Command and Request code messages.
129	**Not Acknowledge** – Once the recipient has processed the message, it uses this flag along with the message flags to indicate which, if any, of the tasks could not be performed. Sent from server. Not Acknowledge messages will still contain the most recent values in each field which has the corresponding message specific flag set.
130-254	Reserved for responses from server.
255	Not Supported - sent from server if the packet type is not recognized.

The common packet header is shown below:

Field designation	Size/Type	Description
<magic start>	U32	D4C3B2A1
<length>	U32	Length of this packet including header
<packet type>	U64 (ASCII readable)	Identification field describing the nature of the data after this header.
<packet version>	U32	Version of this packet type.
<msg flags>	U32	Bit Mask

		<table><tr><td>0000 0000</td><td>0000 0000</td><td>0000 0000</td><td>0000 0000</td></tr><tr><td colspan="3">Message Specific Flags</td><td>System Code</td></tr></table> **System Codes** See system message codes above. **Message Specific** See individual message for these flags, they will be listed in column 4 as a hexadecimal bit mask.
<packet time stamp>	U64	Time message was formed 64-bit ns timestamp. This time is not guaranteed to be synchronized to anything, and is mainly used for development/recording. Most clients to the DRX can leave this to 0. The DRX itself will populate this field for all messages that it sends. The purpose of the field is to determine the time between consecutive messages.

2.6 Common Packet Footer

A footer will also be added to every message.

Field designation	Size/Type	Description
<magic end>	U32	2B3C4D5E

3. ICD Message Usage

There are two types of messages which can be used to communicate with the DRX unit. These are the command message and request status message. These messages are configured using the <msg flags> field in the common packet header.

3.1 Connection Process

To use any ICD message in this document you first need to configure the DRX unit to enable the ICD messages you want the DRX to process. This is usually done once the connection has been established with the DRX Unit.

Once you are connected to the DRX Unit, it is configured to not accept or send any ICD messages. This is done to allow filtering of the ICD messages as clients may not require all messages.

This process uses the MSG_REQ_ (section 4.1.1) which can add and remove which ICD messages will be processed by the DRX.

All following examples require you to have set up the DRX to process the ICD messages used.

3.2 Command Message

This uses a <msg flag> value of 0x1 which is the flag for a command message. Command messages are used to change settings in the DRX unit and can change one or more settings per ICD message.

3.2.1 Fixed Size Message Example

Fixed size messages are ICD messages which do not change size as their contents are a fixed number of bytes.

To create a command message you need to do the following. We will use the PING_REQ ICD message as an example,

1. Setup the common packet header for the message type you are going to use.

Field designation	Setting
<magic start>	Set to **D4C3B2A1**
<length>	This value will be set after other values are set. See step 3.
<packet type>	Set to **PING_REQ**
<packet version>	Set to **0**
<msg flags>	Set initially to **0x1**
<packet time stamp>	Set to **0**

2. Now the rest of the message can be filled in as required. You only set the <msg flags> for the fields you wish to change. For example we will set the DRX unit range and range mode. The range will be changed to 30m and mode changed to auto.

Field designation	Setting	< msg flags >
Ping Mode	Set to **0**	*N/A*
Range	Set to **30.0**	Set bit **0x200**
Range Mode	Set to **2**	Set bit

		0x400
Pulse Type	Set to **0**	*N/A*
Power Mode	Set to **0**	*N/A*
Power Level	Set to **0.0**	*N/A*
Reserved	Set to **0**	*N/A*
<magic end>	Set to **2B3C4D5E**	*N/A*

(N/A indicates you do not set or change the value)

3. Set the <length> field to the size of the complete message from and including <magic start> and <magic end>. For this example the length would be **96**.

3.2.2 Variable Size Message Example

Variable size messages are ICD messages which will change size depending on the message contents.

To create a command message you need to do the following. We will use the MSG_REQ_ ICD message as an example. For this example we will be requesting three ICD message types from the DRX unit: PING_REQ, SONADISP and BATHYCOR.

1. Setup the common packet header for the message type you are going to use.

Field designation	**Setting**
<magic start>	Set to **D4C3B2A1**
<length>	This value will be set after other values are set. See step 3.
<packet type>	Set to **MSG_REQ_**
<packet version>	Set to **0**
<msg flags>	Set initially to **0x1**
< packet time stamp>	Set to **0**

2. Now the rest of the message can be filled in as required. You only set the <msg flags> for the fields you wish to change.

   We will be using no security values.

Field designation	**Setting**	**< msg flags >**
Security	Set all values to **0**	*N/A*
Spare	Set to **0**	*N/A*
Command Type	Set to **1** (Add)	Set bit **0x2000**
Message Types	Set to **0**	*N/A*
N	Set to **3**	Set bit **0x8000**
Requested Messages	Set U64[0] to **PING_REQ** Set U64[1] to **SONADISP** Set U64[2] to **BATHYCOR**	*N/A*
<magic end>	Set to **2B3C4D5E**	*N/A*

(N/A indicates you do not set or change the value)

3. Set the <length> field to the size of the complete message from and including <magic start> and <magic end>. For this example the size of the fixed data fields is 76 bytes, and we are sending 3 'Requested Messages' which total 24 bytes. So the total <length> needs to be set to **100.**

3.3 Request Status Message

This uses a <msg flag> value of 0x2 which is the flag for a request status message. Request status messages are used to get the current settings from the DRX unit. The <msg flags> of the packet fields are not used with the request status message.

To create a request status message you need to do the following. We will use the PING_REQ ICD message as an example.

1. Setup the common packet header for the message type you are going to use.

Field designation	Setting
<magic start>	Set to **D4C3B2A1**
<length>	This is normally set at the end of the message, especially for messages which have a dynamic size.
<packet type>	Set to **PING_REQ**
<packet version>	Set to **0**
<msg flags>	Set to **0x2**
< packet time stamp>	Set to **0**

2. The rest of the message is not required except for the <magic end> value.

Field designation	Setting	< msg flags >
Ping Mode	*N/A*	*N/A*
Range	*N/A*	*N/A*
Range Mode	*N/A*	*N/A*
Pulse Type	*N/A*	*N/A*
Power Mode	*N/A*	*N/A*
Power Level	*N/A*	*N/A*
Reserved	*N/A*	*N/A*
<magic end>	Set to **2B3C4D5E**	*N/A*

(N/A indicates you do not set or change the value)

3. Set the <length> field to the size of the complete message from and including <magic start> and <magic end>. For this example the length would be **96**.

3.4 Acknowledge and Not Acknowledge

After a command message or request status message has been sent you could receive an Acknowledge message, Not Acknowledge message or both.

3.4.1 Acknowledge Message

Acknowledge messages are sent from the DRX unit to the connected client or clients. This message will contain the fields which were successfully processed. All fields which have their <msg flags> set will contain the current DRX unit values.

This is a message which has the <msg flags> system code set to the value of 128.

To process an Acknowledge Message you read the <msg flag> bits, these indicate which fields of the ICD message were successfully processed. The field value will be set to the current value from the DRX unit.

3.4.2 Not Acknowledge Message

Not Acknowledge messages are sent from the DRX unit to the client which initiated the ICD message. This message will contain the fields which were not able to be processed. All fields which have their <msg flags> set will indicate that value could not be processed or the unit cannot process that type of request. The field value will contain the current DRX unit value.

This is a message which has the <msg flags> system code set to the value of 129.

To process a Not Acknowledge Message you read the <msg flag> bits, these indicate which fields of the ICD message were not able to be processed. The field value will be set to the current value from the DRX Unit.

3.5 Data Order

Unless explicitly stated otherwise:

- All integer/floating data is little endian by default
- The magic start/end are both big endian
    - If you treated the U32 Magic Start field as U8[4], then U8[0] = 0xA1, U8[1] = 0xB2 etc.
- Strings (such as packet type) are laid out as you would expect in memory
    - If you treated the U64 field as U8[8], then “MSG_REQ_” would have U8[0] = ‘M’, U8[1] = ‘S’, etc.

4. System Network Messages

4.1 Connection Control Messages

4.1.1 Message Request (MSG_REQ_)

Used once the TCP connection has been established with the DRX. See Appendix B for a usage example.

Field designation	Size/Type	Description	Flags
Common Header		<packet type> = MSG_REQ_	
		<packet version> = 2	
Security	U64[4]	Not Used: DO NOT set the flag bits if unknown! ***Set all to 0***	00000Fxx
Spare	U16	Reserved. ***Set to 0***	000010xx
Command Type	U16	0 = Not specified (Default) 1 = Add Requested Messages by packet type 2 = Delete Requested Messages by packet type 3 = Report currently set Requested Messages ***Used when sending a command message. Set to 0 if sending a request status message.*** ***Command 3 will report the current messages the DRX Unit is set to send on this connection.***	000020xx
Message Types	U16	Used to list a subset of the supported messages 0 = All (modes TBD) ***Set to 0***	000040xx
N	U16	Number of message packet types contained in 'Requested Messages' field designation. ***Set when using command message type 1 and 2, read when using command message 3 or receiving the response to request status message.***	000080xx
Requested Messages	U64[N]	Array of <packet type> e.g. SONADISP, SIDEDATA, BATHYCOR etc.	N/A
<magic end>	U32	2B3C4D5E	N/A

4.2 DRX Dynamic Update Messages

These messages are sent from the DRX to any connected clients which have asked to get them using the process outlined in sections 3.1 and 4.11. All dynamic update messages cannot be used as command message or request status message and will be ignored by the DRX.

4.2.1 Sonar Status (SONASTAT)

This output will be updated on the network at a frequency of 1Hz to any client registered to receive the message. The data will contain the latest received value for each parameter.

Field designation	Size/Type	Description	Flags
Common Header		<packet type> = SONASTAT	
		<packet version> = 4	
System Temp	F32	Temperature of system in degrees Celsius ***(Range: -2147483.648 to 2147483.647; Typical 40; Units Celsius)***	000001xx
Transducer Temp	F32	Temperature of the transducer sensor in degrees Celsius	000002xx

		(Range: -3.4 *10^{38} to 3.4 *10^{38}; Typical -5 to 40; Units Celsius)	
Ping Rate	F32	Current ping rate in pings per second ***(Range: -3.4 *10^{38} to 3.4 *10^{38}; Typical 0 to 80; Units Hz)***	000004xx
Transmission Centre Frequency	F32	Current transmitter center operating frequency in Hz ***(Range: 0.0 to 5120000; Typical 40000 to 160000; Units Hz)***	000008xx
Transmission Bandwidth	F32	Current transmitter bandwidth in Hz ***(Range: 0 to 5120000; Typical 0 to 60,000; Units Hz)***	000010xx
Ping State	U32	Indicates current state of transceiver 0 = None 1 = Standard	000020xx
Sound Velocity	F32	Current sound velocity in meters per second ***(Range: 1400 to 1600; Typical 1500.0; Units m/s)***	000040xx
Tide Value	F32	Current tide value in meters ***(Range: -3.4 *10^{38} to 3.4 *10^{38}; Typical -1 to 10; Units m)***	000080xx
Link Speed	U32	Current Network Link Speed in Mb/s ***(Range: 0 to 2147483647; Typical 1000; Units Mb/s)***	000100xx
Progress and Status	U8	Progress (0- 100%)	000200xx
	U8	Source (If more than one monitoring task is happening at the same time then updates will be interleaved): 0 => No Monitoring 1 => Backscatter Monitoring	
	U16	Status (remains set until either new source of new monitor): 0 => Not Run 1 => Active 2 => Cancelled 3 => Completed: Success 4 => Completed: Failure 0x1000 => Calibration File Exists (This is a flag is set/unset regardless of the source or progress fields)	
Reserved	U32[11]		N/A
<magic end>	U32	2B3C4D5E	N/A

4.2.2 Sensor Updates (SENUPDAT)

This output will be updated on the network at a frequency of 10Hz to any client registered to receive the message. The data will contain the latest received value for each parameter.

Note: Not all parameters will be received at the same time from the same sensor

Field designation	Size/Type	Description	Flags
Common Header		<packet type> = SENUPDAT	
		<packet version> = 3	
Year	U16	Calendar Year, synchronised to GPS time if available	000001xx
Month	U8	Calendar Month, synchronised to GPS time if available	000001xx
Day	U8	Calendar Day, synchronised to GPS time if available	000001xx
Hour	U8	Calendar Hour, synchronised to GPS time if available	000001xx
Minute	U8	Calendar Minute, synchronised to GPS time if available	000001xx
Millisecond	U16	Calendar Millisecond, synchronised to GPS time if available	000001xx
Reserved	U32		000002xx
Latitude	F64	Latitude from GPS sensor in degrees. North is positive. ***(Range: -90.0 to 90.0; Typical -90 to 90; Units Degrees)***	000004xx
Longitude	F64	Longitude from GPS sensor in degrees, East is positive ***(Range: -180.0 to 180.0; Typical -180 to 180; Units Degrees)***	000008xx

Field designation	Size/Type	Description	Flags
Heading	F32	Heading of vessel in degrees ***(Range: 0.0 to 360.0; Typical 0 to 360; Units Degrees)***	000010xx
Roll	F32	Latest vessel roll in degrees. Starboard down positive. ***(Range: -90.0 to 90.0; Typical -90 to 90; Units Degrees).***	000020xx
Pitch	F32	Latest vessel pitch in degrees. Bow up positive. ***(Range: -90.0 to 90.0; Typical -90 to 90; Units Degrees).***	000040xx
Heave	F32	Latest vessel heave value in meters. Up is positive. ***(Range: -3.4 *10^{38} to 3.4 *10^{38}; Typical -10.0 to 10.0; Units m).***	000080xx
SOG	F32	Speed over ground from GPS sensor in knots. ***(Range: -3.4 *10^{38} to 3.4 *10^{38}; Typical 0 to 30.0; Units knots).***	000100xx
COG	F32	Course over ground form GPS sensor in degrees. ***(Range: 0.0 to 360.0; Typical 0.0 to 360.0; Units Degrees).***	000200xx
Nadir Depth	F32	Current depth from transducer face in meters. Roll stabilized. ***(Range: -3.4 *10^{38} to 3.4 *10^{38}; Typical 0.0 to 1000.0; Units m).***	000400xx
Temperature	F32	Water temperature in degrees Celsius. ***(Range: -3.4 *10^{38} to 3.4 *10^{38}; Typical -5.0 to 30.0; Units Deg. Celsius).***	000800xx
Transducer Draft	F32	Current Transducer Draft (as set by BATHYOPT) included for help resolving absolute depth from surface of water. **Note this field is Read only use BATHYOPT to write.** ***(Range: -3.4 *10^{38} to 3.4 *10^{38}; Typical 0.0 to 30.0; Units Metres. m).***	001000xx
Geoidal Height	F32	Geoidal height (separation between Geoid and Ellipsoid) as reported in GGA. Units typically metres	002000xx
Antenna Height	F32	Antenna height above Geoid as reported in GGA. Units typically metres	004000xx
Reserved	U32[0]		N/A
<magic end>	U32	2B3C4D5E	N/A

4.2.3 General Messages (GEN_MESG)

This message is used to report errors as well as normal status reports.

Field designation	Size/Type	Description	Flags
Common Header		<packet type> = GEN_MESG <packet version> = 2	

Msg Level	S32	Used to define level of message. E.g. critical, information. Use the Message Logging levels.
		`None = INT_MAX, //Only really intended for use //when setting output logging //levels - this will mean that //nothing is logged`
		`Critical = 100, //Critical events - much more //important than a simple "error"`
		`Error = 10, //Errors`
		`Warning = 5, //Warnings`
		`Normal = 0, //Just normal every day logging`
		`Debug = -5, //Used only for debugging`
		`Info = -8, //Used only for informational //messages`
		`Diagnostic = -10,//Used for diagnostic messages //that you don't want on most of //the time, but might be handy to //have`
			000001xx

		`All = INT_MIN, //Only really intended for use //when setting logging levels – //this means everything is logged`	
Msg Code	U32		000002xx
M	U16	Message length in bytes.	000004xx
Message Data	U8[M]		N/A
<magic end>	U32	2B3C4D5E	N/A

4.2.4 PPS Configure (PPSSTATS)

PPS Status message is output at 1Hz to inform clients of the PPS synchronsiation status.

Field designation	Size/Type	Description	Flags
Common Header		<packet type> = PPSSTATS	
		<packet version> = 2	
PPS Timestamp	U64	A nanosecond time stamp of last active edge of PPS pulse received. This time is referenced at zero at 000hrs UTC time. This time may jump before synchronisation is achieved. No local time zone correction is applied. This time stamp will be free running if UTC time synchronisation is not available. ***(Range: 0 to 18,446,744,073,709,551,615; Units ns).***	000001xx
Time Timestamp	U64	A nanosecond time stamp of last time sentence received. This time is referenced at zero at 000hrs UTC time. This time may jump before synchronisation is achieved. No local time zone correction is applied. This time stamp will be free running if UTC time synchronisation is not available. ***(Range: 0 to 18,446,744,073,709,551,615; Units ns).***	000002xx
Hour	U8	Calendar Hour, synchronised to GPS time if available	000004xx
Minute	U8	Calendar Minute, synchronised to GPS time if available	
Millisecond	U16	Calendar Millisecond, synchronised to GPS time if available	
PPS Status	U32	0 = Disabled 1 = Untrained – no Time or PPS detected 2 = Untrained – no Time detected 3 = Untrained – no PPS detected 4 = Training in progress 5 = Synchronised – Poor (> 2ms jitter – occurs on initial sync) 6 = Synchronised – OK (< 2ms jitter) 7 = Synchronised – Good (< 100us Jitter) 8 = Synchronised – Excellent (< 10us Jitter) 9 = Synchronised – Perfect (< 1us Jitter)	000008xx
PPS Jitter ms	F32	Milliseconds of jitter detected between internal DRX time keeping and external PPS pulse. ***(Range: -3.4 *10^{38} to 3.4 *10^{38}; Typical -100.0 to 100.0; Units ms).***	000010xx
PPS Frequency detected	F32	Approximate detected frequency of PPS signal in Hz. ***(Range: -3.4 *10^{38} to 3.4 *10^{38}; Typical 1.0; Units Hz).***	000020xx
Reserved	U32[6]		N/A
<magic end>	U32	2B3C4D5E	N/A

4.2.5 Key Pulse Status (KPSTATUS) (In development subject to change)

This message gives status information regarding the Key pulse inputs and other derived signals after internal configuration adjustments.

Field designation	Size/Type	Description	Flags
Common Header		<packet type KPSTATUS>	
		<packet version> = 1	
Reserved	U32[8]		0000FFxx
TX_OUTPUT	KPS	Shows statistics for the actual DRX Transmit envelope.	000100xx
KP_OUTPUT	KPS	Shows statistics for this output after internal inversion and pulse width extensions have been applied.	000200xx
KP_MUTE	KPS	Shows statistics for this signal after holdoff settings have been applied.	000400xx
KP_BLANK	KPS	Shows statistics for this signal after holdoff settings have been applied.	000800xx
KP_SLAVE	KPS	Combined Slave Trigger signal with hold off and skip settings applied. Duty is not relevant as at this point we are only looking for a trigger to start transmission	001000xx
KP_INPUT1	KPS	Shows statistics for this input after internal inversion has been applied (before skip and hold off).	002000xx
KP_INPUT2	KPS	Shows statistics for this input after internal inversion has been applied (before skip and hold off).	004000xx
KP_WIRELESS	KPS	Shows statistics for this input before skip and hold off.	008000xx
Reserved	KPS[8]	Spare space	FF0000xx
<magic end>	U32	2B3C4D5E	N/A

4.2.5.1 KPS Struct

Field designation	Size/Type	Description
Duty	Float	Percentage of the time of which the signal was found to be high over the previous pulse. 100*High Time/(High Time+Low Time) 100% implies signal may be stuck high 0% implies signal may be stuck low ***Activity flag or PPM should be used to detect if there is recent activity.*** ***Measurement of pulses is restricted to 64 us intervals thus a 10kHz+ signal may not register as having any duty.*** ***Range 0-100***
PPM	U16	Pulses that were detected in the last minute of operation pulse rates greater than 4095 will register as 4095. ***Range 0-4095***
FLAGS	U16	Status Flags for this line Bit 0: Activity detected in last period (4 seconds) Bit 1: Current Line State Bit 2: Flag set if this line is currently disabled due to excessive duty – line state will be set to zero until duty reduces. Bit 2-15: Unused

4.3 DRX Configuration Messages

Sonar control messages are used to configure the DRX Unit and to request the current configuration.

The command message flag is used to update the current settings and the request status message is used to request the current settings. See section 3.

4.3.1 Ping Request (PING_REQ)

Ping requests are used to tell the transceiver how to ping. Under normal conditions auto mode will be used and this message only needs to be sent if changing range or ping modes.

Field designation	Size/Type	Description	Flags
Common Header		<packet type> = PING_REQ	
		<packet version> = 2	
Ping Mode	U32	1 = Single Ping 2 = Auto Pinging 3 = Stop Pinging ***On response to a request status message a value of 0 will indicate the system is not pinging.***	000001xx
Range	F32	Range to collect data to in meters. ***(Range: 1.0 to 12000.0; Typical 5.0 to 500.0; Units m).***	000002xx
Range Mode	U32	1 = Manual Range 2 = Auto Range Further modes may be added in future releases	000004xx
Pulse Type	U32	Not defined ***Set to 0***	000008xx
Power Mode	U32	0 = no transmission 1 = Manual 2 = Auto (by Range) 3 = Auto (by Signal) Further modes may be added in future releases	000010xx
Power Level	U32	Power level to use as a percent from 0-100%*	000020xx
Reserved	U32	Not defined, ***Set to 0***	000040xx
Reserved	U32[8]		N/A
<magic end>	U32	2B3C4D5E	N/A

```
* This is translated into a percentage of the available dB range of the
transmitter and as such is non linear (however the visual effect is linear when
viewing logarithmic target strength being the standard displayed form of sonar
data).
For example: if 50% is requested and the dB range of the power amplifier is -
20dBVrms...+40dBVrms then +10dBVrms will be the transmitted power level.
```

4.3.2 Processing Options (PRO_OPTN)

This message will set the available processing options and modes for the DRX you are connected to.

Use a command message to set new values or a request status message with all flags cleared to get the current settings from the DRX. For a command message, the flag values are only set for the fields you wish to change.

Field designation	Size/Type	Description	Flags
Common Header		<packet type> = PRO_OPTN <packet version> = 2	
Side-lobe Mode	U32	1 = Manual (SLR value is used) 2 = Automatic	000001xx
Side-lobe reduction (SLR) threshold	F32	Specifies a threshold in dB. Signals which are this number of dBs less than the peak signal are removed. ***(Range: 3.0 to 100.0; Typical 20.0; Units dB).***	000002xx
IF Control	U32	Ping to Ping Filtering – Number of pings to use 0,1 = Off 2 = Take minimum value from two pings 3 = Take minimum value from three pings 4+ = Take minimum value from four pings ***(Range: 0 to 201; Typical 0 to 4; Units N/A).***	000004xx
Spare	U32	Reserved ***Set to 0***	000008xx
SLI Gain Mode	U32	Which TVG gain mode is used: 0 = Generic 1 = Fish School 2 = Seafloor detection 3 = Fish Further modes may be added in future releases	000010xx
Absorption Loss Mode	U32	Type of Absorption loss: 0 = Default Seawater Further 'Ocean Classification' modes may be added in future releases	000020xx
Spreading Loss Mode	U32	Type of spreading loss to use: 0 = Default 1 = Spherical 2 = Cylindrical	000040xx
Sound Velocity Mode	U32	Used to tell the DRX which mode to use. 1 = Manual 'Sound Velocity' value. 2 = Manual 'Water Salinity' + Manual 'Water Temperature' 3 = Internal 'Water Salinity' + DRX transducer temperature. 4 = External Sound Speed Sensor	000080xx
Sound Velocity	F32	Sound velocity/speed at sea surface in units of meters per second. ***(Range: 1350.0 to 1600.0; Typical 1500.0; Units m/s).*** ***Used only if the 'Sound Velocity Mode' is set to 1***	000100xx
Water Salinity	F32	Water salinity used in sound speed calculations in ppt (parts per thousand). ***(Range: 0.0 to 50.0; Typical 35.0; Units ppt).*** ***Used if the 'Sound Velocity Mode' is set to 2 or 3***	000200xx
Water Temperature for Sound Velocity	F32	Sea surface temperature in degrees Celsius for use in sound velocity computation. ***(Range: 0.0 to 100.0; Typical 20.0; Units Deg. Celsius).*** ***Used only if the 'Sound Velocity Mode' is set to 2***	000400xx
Display Clutter	F32	Sonar and Fish Finder noise threshold in dB from noise floor. ***(Range: -1000.0 to 1000.0; Typical -6.0 to 30; Units dB).*** ***(default = 16)*** A value of 0dB will give 50% noise speckle.	000800xx

		A value of +16dB will give very little speckle.	
Seafloor Aeration Filter Control	U32 U16,U8,U8	Ping to ping energy monitor used to disable seafloor tracking when energy of ping reduces or increases drastically. 0 = Off First U16 (<<16): Threshold percent (Q8 - 100-percent) 0 = Off (Disables Aeration Filter) 1-25089 = Threshold Range ***(Valid Range 0 – 25089)*** ***(Typical Range 0, 20480-25088)*** First U8(<<8): Maximum pings to ignore after last aeration detected (extends duration after detection) ***(Valid Range 0 – 255)*** ***(Typical Range 0 – 10)*** Second U8(<<0): Maximum pings to ignore consecutively (duration) 0 = Off (Disables Aeration filter) ***(Valid Range 0 – 255)*** ***(Typical Range 0 – 100)***	001000xx
Seafloor Tracking Persistence	U32 U16,U16	Control for ping to ping tracking persistence – used for rejecting fish schools as the seafloor also helps stability in noisy conditions First U16(<<16): Threshold percent (Q8 - 100-percent) 0 = Off 256-25599 = Percentage of current seafloor detection range to scan for next seafloor detection 0 = Off; 256 = 1%; 2560 = 10% (Typical); ***(Valid Range 0 or 256 – 25600)*** ***(Typical Range 1280(5%)-12800(50%))*** Second U16(<<0): Maximum pings to track previous ping (duration) 0 = Off (Minimal ping to ping tracking) 1-65535 Pings to keep attempting to find the seafloor in the same area as found last ping (per beam) ***(Valid Range 0 – 65535)*** ***(Typical Range 0 – 100)***	002000XX
Reserved	U32[9]		N/A
<magic end>	U32	2B3C4D5E	N/A

4.3.3 Bathy Options (BATHYOPT)

Used to set the options which relate to the bathymetry.

The amount of detections generated for each beam can be set using the Detection Mode field if the DRX Unit supports it. Normally this is set to single detection per beam. Multiple detection per beam will generate one or more detections per beam.

Field designation	Size/Type	Description	Flags

Common Header		<packet type> = BATHYOPT <packet version> = 4	
Detections Start Angle	S16	Angle in degrees at which to start sea floor detections from. Reference is Nadir at 0 degrees ***Valid range is from -89 to -1 degrees.*** ***Negative values are to port.*** ***This value MUST be less than 'Detections End Angle'***	000001xx
Detections End Angle	S16	Angle in degrees at which to stop sea floor detections from. Reference is Nadir at 0 degrees ***Valid range is from 1 to 89 degrees.*** ***Negative values are to port.*** ***This value MUST be greater than 'Detections Start Angle'***	000002xx
Transducer Draft	F32	Meters from sea level to the face of the transducer. Down is positive. This is used to account for the vessels draft offset. **Valid range is from -100m to 100m**	000004xx
Tide Mode	U16	1 = Tides disabled 2 = Spare, undefined 3 = External Tide Updates 4 = RTK based tides	000008xx
Detection Mode	U16	1 - Default	000010xx
Tide Value	F32	If Tide Mode is external, this is the tide value which will be used. If Tide Mode is not external, this tide value will be ignored. Units in meters ***Valid range is from -100m to 100m***	000020xx
Correction Flags	U32	Tells the DRX Unit which correction data to use when processing new ping data. 0x001 = Pitch Correction Enabled 0x002 = Heave Correction Enabled 0x004 = Roll Correction Enabled 0x008 = Spare 0x010 = Invert Array (180 degree heading rotation of transducer) 0x020 = Invert Roll 0x040 = Induced Heave Enabled 0x080 = SPARE 0x100 = Invert Pitch 0x200 = Invert Heave	000040xx
Backscatter Control Flags	U32	Commands the DRX to perform a task with respect to the Backscatter. Setting a bit high will instigate the command. The bit will then be set low again by the DRX. 0x0001 = Auto Calibrate. Clears the existing calibration parameters. Performs the calibration. Forces the calibration file to be saved on completion. 0x0002 = Manual calibration Clears the existing calibration parameters. Performs the calibration. Calibration file not saved on completion. 0x0003 = Refine calibration Starts with the existing calibration parameters. Performs a fine tuning calibration. Calibration file not saved on completion. 0x0004 = Cancel Calibration 0x0005 = Save calibration file.	000080xx

Glitch Filter Mode	U32	Glitch Filter Mode for filtering BATHYCOR depth output 0 = Default, (Uses Med setting) 1 = Off (No glitch filter applied) 2 = Low (Least points removed) 3 = Med (Moderate setting) 4 = High (Most points will be removed)	000100xx
RTK Window	U32	Time period over which linear regression is applied to the RTK value, in seconds DEFAULT: 300 Valid range: [60,600] inclusive	000200xx
RTK Timeout	U32	If GPS quality drops below (user set) threshold, time to continue using current best fit prediction, in seconds DEFAULT: 300 Valid range: [300,1800] inclusive	000400xx
RTK Offset from Ellipsoid	F32	Manual MSL offset in metres to 2d.p. , valid range -10000.00 to +10000.00 This value is ignored if geoidal separation is being referenced from serial data or internal reference (see RTK Geoid mode)	000800xx
RTK Geoid Mode	U32	0 = User manual offset from ellipsoid 1 = Geoidal separation from serial data 2 = Geoidal separation from selected internal geoid file (NOT IMPLEMENTED)	001000xx
Depth of Keel Below Transducer	F32	Distance in meters below the transducer to the bottom of the keel. The value should be positive if the keel is deeper than the transducer. DEFAULT: 0 **Valid range is from -100m to 100m**	002000xx
Reserved	U32[5]		N/A
<magic end>	U32	2B3C4D5E	N/A

4.3.4 PPS Configure (PPSCNFIG)

PPS Configure message is used to request or change the current PPS synchronization settings.

Field designation	Size/Type	Description	Flags
Common Header		<packet type> = PPSCNFIG	
		<packet version> = 2	
PPS Input Port	U32	0 = PPS synchronisation disabled 1 = Source 1 - KP/PPS (dedicated port) 2 = Source 2 - NMEA0183 port 3 = Source 3 - RS422 port 1	000001xx
PPS Active Edge	U32	0 = PPS synchronisation disabled 1 = Positive (rising) edge used for PPS time synchronisation 2 = Negative (falling) edge used for PPS time synchronisation	000002xx
Reserved	U32[8]		N/A
<magic end>	U32	2B3C4D5E	N/A

4.3.1 Frequency Options (IMS_OPTN)

This message will set the available Frequency Control and Interference Management System (IMS) options and modes for the DRX you are connected to.

Use a command message to set new values or a request status message with all flags cleared to get the current settings from the DRX. For a command message, the flag values are only set for the fields you wish to change.

Field designation	Size/Type	Description	Flags
Common Header		<packet type> = IMS_OPTN	
		<packet version> = 1	
Mode	U32	*IMS (Method)Frequency Source:* ➢ 0: OFF (default values); ➢ 1: AUTO values; ➢ 2: MANUAL values;	000001xx
Reserved	U32 (2)	reserved	N/A
Reserved	U8 (2)	reserved	N/A
Type	U8	*Type of contained candidate values* ➢ 0: MANUAL: - manual mode (default) ➢ 1: AUTO: - result of the IMS calculation	000020xx
Reserved	U8	reserved	N/A
Reserved	U32 (2)	reserved	N/A
Chirp Centre Frequency (Hz)	F32	*Used for: MANUAL VALUES* ➢ Range: 0Hz…500,00Hz (transducer dependent). ➢ WMB-80 :: Typically: 60kHz to 80kHz, Default: 80kHz. ➢ WMB-160 :: Typically: 120kHz to 160kHz, Default: 160kHz.	000040xx
Reserved	U32 (17)	reserved	N/A
<magic end>	U32	2B3C4D5E	N/A

4.3.2 Key Pulse (KEYPULSE) (subject to change)

This message is yet to be implemented in the DRX and thus is subject to change - This message has been created for Key pulse configuration for Slave, Blank/Mute and KP Output control. Further information on the intent is described under KP_Specification.doc.

KP_IMS settings may be rejected or ignored if the appropriate license is not available.

Field designation	Size / Type	Description	Flags
Common Header		<packet type> = KEYPULSE	
		<packet version> = 2	
KP_MODE	U8	Mode: 0 = OFF. Ignore KP inputs, and generate no outputs. 1 = SLAVE. Force ping as defined by KP Configuration. 2 = MASTER. Ignore input KP, but generate an output defined in KP Configuration. 3 = AUTO. Attempt to synchronise with an external sounder if the sounder has a longer period else will perform blanking. ***Default = MASTER***	000001xx

Field designation	Size / Type	Description	Flags
KP_OUT_MODE	U8	Selects what triggers the KP_OUT pulse. Priority encoder highest priority wins (priority shown in brackets 1 = highest priority): 0x00 = OFF 0x01 = KP_IN_1_REPEAT(2) 0x02 = KP_IN_2_REPEAT(3) 0x04 = TX_REPEAT(1) 0x08 = MASTER(5) 0x20 = IMS_KP(4) 0x80 = INVERT_POLARITY ***Default* = TX_REPEAT**	000002xx
KP_RX_MUTE_MODE	U8	**DO NOT USE!** Controls the analogue mute circuit. **NOT RECOMMENDED FOR USE.** Bitwise OR of: 0x00 = OFF 0x01 = KP_IN_1_REPEAT 0x02 = KP_IN_2_REPEAT 0x04 = TX_REPEAT 0x08 = MASTER 0x10 = KP_OUT 0x20 = IMS_KP ***Default = OFF***	000004xx
KP_RX_BLANK_MODE	U8	Controls the digital receive mute process. Bitwise OR of: 0x00 = OFF 0x01 = KP_IN_1_REPEAT 0x02 = KP_IN_2_REPEAT 0x04 = TX_REPEAT 0x08 = MASTER 0x10 = KP_OUT 0x20 = IMS_KP ***Default = OFF***	000008xx
KP_TIMEOUT_MODE	U8	After TRIGGER SOURCE timeout the DRX will return to the following mode: As per KP_MODE enumeration. ***Default = MASTER***	000010xx
KP_OUT_SKIP_COUNT	U8	Used to skip over transmission cycles so slave doesn't always ping at the same time as DRX. Allow rate limiting of an interference source to give us a better quality of sonar overall (some pings being interference free). Integral part of PING_PONG Mode. Bit 7 is used to create inverted operation. The skip counter (bits 6-0) will decrement on each transmission. The KP_OUT will be set high when the skip counter <= 1. ***0, 1,128,129 = disabled;*** ***2-127 = KP_OUT occurs on 1 of value pings;*** ***130-255 = ((value-128)-1) of value-128 pings*** ***Default = 1 (no skip)***	000020xx

Field designation	Size / Type	Description	Flags
KP FUNCTIONS	U8	Bit flags to enable various functions: Bit 7-2 Unused. Bit 1 = Set to 1 to disable free running KP Output when DRX is not pinging. Bit 0 = PING_PONG. Depending on KP_OUT_SKIP DRX only pings when the KP_OUT is not sent and other system pings when KP_OUT is sent (as per normal). ***Default = 0 (All OFF)***	000040xx
Reserved	U8[1]	Possible new modes	000080xx
KP_TRIG_SOURCE	U8	Bitwise OR of: 0x00 = OFF 0x01 = KP_IN_1 0x02 = KP_IN_2 0x20 = IMS_KP ***Default = OFF***	000100xx
KP_TX_JITTER_SLOT_COUNT	U8	See Section **Error! Reference source not found.**. ***Range: 1..10*** ***Default = 4***	000200xx
KP_TX_JITTER_SLOT_SEPARATION_US	U16	See Section **Error! Reference source not found.**. ***0 = chirp-pulse-width;*** ***>0 = chirp-pulse-width + value micro-seconds;*** ***Default = 0***	000400xx
KP_MAX_PRF_HZ	F32	PRF = Pulse Repetition Frequency. Typically inverse of the collection range or less. The DRX will wait at the end of each ping to ensure ping rate can't exceed PRF specified. Ping rate of the DRX will be slower than the PRF specified. ***Range: Positive values*** ***0.01 or less = best effort*** ***Default = 0***	000800xx
MODE_TIMEOUT_US	U32	Microsecond timeout. If this timeout is exceeded before receiving an external key pulse and system is in AUTO or SLAVE mode then will revert to mode defined by KP_TIMEOUT_MODE. System will resume previous state when the external key pulse resumes. ***Range: 0..30,000,000*** ***Default: 5,000,000***	001000xx
KP_OUT_PULSE_LENGTH_US	U32	Pulse length (maximum) of slaved devices (used for Jitter and internal processing. ***0 => Same as DRX*** ***Range: 0..262,143 Microsecond length*** ***Default = 0***	002000xx

Field designation	Size / Type	Description	Flags
KP_OUTPUT_DELAY %	S16Q8	Determines when KP output will occur relative to the DRX Tx. Units % Rx Time. Negative % KP output to occur before DRX_TX. Positive % KP output to occur after DRX_TX. 0: KP Output occurs immediately before DRX Tx (though jitter may delay this further). 1: KP Output occurs immediately after falling edge of DRX Tx (though jitter may delay this further) 25600 Output occurs after the final Rx sample has been collected by the DRX. **Range -25,600 to +25,600;** **Default: 0;**	004000xx
SPARE	U16		008000xx
KP_IMS Specific Data			
KP_IMS_SPARE	U8[3]	Reserved to align KP_IMS with KP1 and KP2 fields	010000xx
KP_IMS_SKIP_COUNT	U8	**Used in SLAVE mode:** Bit 7 is used to create inverted operation. The skip counter (bits 6-0) will decrement on each transmission. The KP_IMS will be set high when the skip counter <= 1. ***0, 1,128,129 = disabled;*** ***2-127 = KP_IN occurs on 1 of value pings;*** ***130-255 = ((value-128)-1) of value-128 pings*** ***Default = 1 (no skip)Default = 1***	
Reserved	U32	Spare	
KP_IMS_SLAVE_HOLDOFF	U32	This value defines an offset from the Key Pulse trigger edge. This offset determines the starting point for the first jitter slot and is the earliest point at which a TX_REQ can be sent. 0x3FFFF = Sync with pulse as close as possible. 0x00000000 = Sync with end of pulse as close as possible. Other values = microsecond delay after end of pulse to wait before first transmission slot. ***Range: 0…262,143*** ***Default = 0***	02000xx
KP_IMS_BLANK_MUTE_HOLD	U32	Extension to the length of the mute trigger pulse. ***Range: 0…100,000*** ***Default = 0***	
Reserved	U32	Spare	
KP_1 Specific Data			
KP1_MODEL	U8	Define the Model of the KP input source. If the model is USER_DEFINED then the following configuration parameters will become active, else defaults (stored in the DRX) will be used for the defined model. ***0 = USER DEFINED*** ***1 = FURUNO*** ***2 = SIMRAD*** ***Default = FURUNO (1)***	040000xx

Field designation	Size / Type	Description	Flags
KP1_TRIG_POLARITY	U8	0 = active-low; 1 = active-high ***Default = active-high (1)***	080000xx
KP1_THRESHOLD	U8	Input voltage trigger threshold for the KP input. ***Range: 0..127*** ***Default = 70***	
KP1_SKIP_COUNT	U8	***As KP_IMS_SKIP_COUNT***	
Reserved	U32	***Spare***	
KP1_SLAVE_MODE_HOLDOFF_US	U32	***As KP_IMS_SLAVE_MODE_HOLDOFF_US***	100000xx
KP1_RX_BLANK_MUTE_HOLD_US	U32	***As KP_IMS_BLANK_MUTE_HOLD_US***	
Reserved	U32	***Spare***	
KP_2 Specific Data			
KP1_MODEL	U8	***As KP1_MODEL***	200000xx
KP2_TRIG_POLARITY	U8	***As KP1_TRIG_POLAIRTY***	400000xx
KP2_THRESHOLD	U8	***As KP1_THRESHOLD***	
KP2_SKIP_COUNT	U8	***As KP_IMS_SKIP_COUNT***	
Reserved	U32	***Spare***	
KP2_SLAVE_MODE_HOLDOFF_US	U32	***As KP_IMS_SLAVE_MODE_HOLDOFF_US***	800000xx
KP2_RX_BLANK_MUTE_HOLD_US	U32	***As KP_IMS_BLANK_MUTE_HOLD_US***	
Reserved	U32	***Spare***	
END KP_IN Specific Data			
<magic end>	U32	2B3C4D5E	N/A
Total (Bytes)	***136***		

Table 3 Proposed DRX ICD message

4.4 DRX System Request Messages

All messages in this section can only be used as a request status message.

Using a command message is invalid and will be ignored by the DRX.

4.4.1 System Information (SYS_INFO)

This message is used to get the DRX system information. This is usually done after a connection and the message request process has been completed.

Field designation	Size/Type	Description	Flags
Common Header		<packet type> = SYS_INFO	
		<packet version> = 4	
Product Type	U32	0 = Undefined 100 = Generic DRX 101 = WMB-1320 200 = Generic DRXe	000001xx
Protocol Version	U32	Protocol Version (read only) This is the version of the DRX Interface Control Document that the systems uses x100 (i.e. a value of 154 equates to ICD v1.54) ***obsolete*** This value cannot be depended on to be accurate.	000002xx
Package Version	U8[4]	[0] = Major Version of the installed package (read only)	000004xx

		[1] = Minor Version of the installed package (read only) [2] = Build LSB of the installed package (read only) [3] = Build MSB of the installed package (read only)	
SW Version	U32[4]	[0] = Major Version (read only) [1] = Minor Version (read only) [2] = Revision (read only) [3] = Build Number (read only)	000008xx
DSP Image Version	U32	DSP Image Version (read only)	000010xx
DRX HW Version	U16	HW Version of the DRX (read only)	000020xx
Tx HW Version	U16	HW Version of the Transmit board (read only)	000040xx
Tx SW Version	U8[2]	[0] = Major Version of the Transmit board SW (read only) [1] = Minor Version of the Transmit board SW (read only)	000080xx
Supervisor Version	U8[2]	[0] = Major Version of the DRX Supervisor micro (read only) [1] = Minor Version of the DRX Supervisor micro (read only)	000100xx
Kernel Version	U8[4]	[0] = Major Version of the Linux Kernel (read only) [1] = Minor Version of the Linux Kernel (read only) [2] = Patch Version of the Linux Kernel (read only) [3] = Spare byte	000200xx
Transmitter Type	U8	0 = None 1 = DTX-25 2 = DTX-32 3 = DRX-46 255 = Unknown	000400xx
Reserved	U8[3]	Reserved for other system versions	00F800xx
Reserved	U32[5]	Reserved for other system versions	00F800xx
Transducer Type	U32	Highest bit to determine source of transducer data: 0x80000000 => Transducer data source mask (TDSM). high => data sourced from smart transducer. Low => data sourced from DRX file. Remaining bits to determine the transducer Type DRX is processing for: 0x0000ffff => Transducer Type mask (TTM) 0 = unknown 1 = standard 80 kHz 2 = standard 160kHz	010000xx
Transducer Serial Number	U8[32]	Serial Number In ASCII	020000xx
Transceiver Serial Number	U8[32]	Serial Number in ASCII	040000xx
Supervisor Serial Number	U32[4]	128bit serial number as reported by the supervisor chip. Treat index 0 as the MSB	080000xx
Reserved	U32	Extra spare area	N/A
<magic end>	U32	2B3C4D5E	N/A

4.4.2 Licensed Features (LIC_FEAT)

This message is used to get the DRX systems current configuration. This is usually done after a connection and the message request process has been completed. Allows present licensed features to be determined.

Field designation	Size/Type	Description	Flags
Common Header		<packet type> = LIC_FEAT	
		<packet version> = 6	

Features Licensed/Present	U32	Bit Flags: Bit 0: Allow Slow Clients Bit 1: Console and Profiling Bit 2: Raw Data enabled (BATHYRAW) Bit 3: Sounder Enabled (FISHDATA) Bit 4: Sidescan Enabled (SIDEDATA) Bit 5: Backscatter Enabled (BATHYCOR::Backscatter) Bit 6: Allow Third Party Clients Bit 7: Allow Fish Statistics (FISHSTAT) Bit 8: Allow Water Column Targets (WCT_DATA) Bit 9: Allow Interference Management System Bit 10: Allow RTK Tide compensation Bit 11: Allow Olex G2 Processing Bit 12: Allow Advanced Key Pulse Functionality Bit 13-31: Reserved	000001xx
Reserved	U32[12]	Spare area for expansion	N/A
FISHDATA, SIDEDATA, SONADISP Max Sampling Hz	F32	Maximum sample rate in Hz that sonar data will be sampled and sent for display purposes	000010xx
Fish Finder Beams, Samples	U32 (U16, U16)	U16(<<16): Maximum number of FISHDATA Beams Supported 0 = Not Supported U16(<<0): Maximum number of FISHDATA Samples Supported 0 = Not Supported **e.g. Bytes 01,08,02,00 = Beams = 2, Samples = 2049**	000020xx
Sidescan Beams	U32 (U16, U16)	U16(<<16): Maximum number of SIDEDATA Beams Supported 0 = Not Supported U16(<<0): Maximum number of SIDEDATA Samples Supported 0 = Not Supported **e.g. Bytes 01,08,02,00 = Beams = 2, Samples = 2049**	000040xx
Maximum Sonar Samples	U32	Maximum number of samples that will be sent for SONADISP 0 = Not Supported **Typical: 2048**	000080xx
Operating Frequency Lower	F32	WMB-160 :: **100,000Hz**…160,000Hz WMB-80 :: **40,000Hz**…80,000Hz	000100xx
Operating Frequency Upper	F32	WMB-160 :: 100,000Hz…**160,000Hz** WMB-80 :: 40,000Hz…**80,000Hz**	000200xx
Maximum Chirp Bandwidth	F32	WMB-160 :: **60,000Hz** WMB-80 :: **60,000Hz**	000400xx
Maximum TVG	F32	DRX(AFE5803) = **40dB** DRXe(AD9671) = **45dB**	000800xx
Maximum Acquisition Range	U32	Maximum range in metres (m) that can be selected for manual acquisition depth. Typical values: WMB-160 :: 400 WMB-80 :: 1000	001000xx
Reserved	U32[11]	Spare area for expansion	N/A
<magic end>	U32	2B3C4D5E	N/A

4.5 Corrected UI Display Data

4.5.1 Sonar Data (SONADISP)

Post processed and corrected sonar data used for a wedge type display.

Both command message and request status cannot be used and are invalid.

Field designation	Size/Type	Description	Flags
Common Header		<packet type> = SONADISP	
		<packet version> = 2	
Time (accurate)	U64	A nanosecond time stamp of sample zero This time is referenced at zero at 000hrs (UTC time is calculated from the timestamp of the ZDA sentence and or a PPS signal when available) No local time zone correction is applied. This time stamp will be free running if UTC time synchronisation is not available. See Appendix E: Sample Zero Time. ***(Range: 0 to 18,446,744,073,709,551,615; Units ns).***	000001xx
Ping Number	U32	Ping Number ID. Used to retrieve data from other messages which relate to the same ping.	000002xx
Latitude	F64	Latitude at transducer in degrees. North is positive. ***(Range: -90.0 to 90.0; Typical -90 to 90; Units Degrees).***	000004xx
Longitude	F64	Longitude at transducer in degrees. East is positive. ***(Range: -180.0 to 180.0; Typical -180 to 180; Units Degrees)***	000008xx
Bearing	F32	Bearing/Heading of vessel in degrees ***(Range: 0.0 to 360.0; Typical 0 to 360; Units Degrees)***	000010xx
Sample Rate	F32	Sample rate in Hz for sonar data. ***(Range: 0 to 3.4 *10^{38}; Typical 2500 to 40000; Units Hz).***	000020xx
Sound Velocity	F32	Sound Velocity in meters per second to use for range calculations. ***(Range: 1350.0 to 1600.0; Typical 1500.0; Units m/s).***	000040xx
Absorption Loss	F32	Absorption loss used in dB/km – 1 way. ***(Range: 0 to 3.4 *10^{38}; Typical 5-70.0; Units dB/km).***	000080xx
Spreading Loss	F32	Spreading loss used in dB/decade – 2 way ***(30 or 40; Typical 40.0; Units dB/decade).***	000100xx
N	U32	Number of beams of sonar data. ***(Range: 0 to 256; Typical 64; Units N/A).*** **Default is 64 beams.**	000200xx
M	U32	Number of samples (per beam) of sonar data. ***(Range: 0 to* 2048*; Typical 0 to* 2048*; Units N/A).*** **Maximum value is 2048**	000400xx
Tx Power Level	F32	Voltage (volts) RMS applied to transmitter in dB (referenced to 1V RMS). ***(Range: -3.4 *10^{38} to 3.4 *10^{38}; Typical -50 to +60; Units dB Vrms).***	000800xx
Pulse Width	U32	Effective pulse width in nano-seconds ***(Range: 0 to 4,294,967,295; Typical 40000000; Units ns).***	001000xx
Sample Type	U32	0 = un-calibrated 1 = calibrated	002000xx
Sample Offset	U32	The number of samples that the first sample is offset by from time zero (transmit time)	004000xx
Reserved	U32[3]	Reserved for header	N/A
Reserved	U32[N]	Reserved for per-beam	N/A
Detection Point	U32[N]	Sample number which most closely matches seafloor detection. 0 = not valid	N/A
Beam Angle	F32[N]	Center beam angle in degrees for this beam. Negative values are port beams. ***(Range: -90.0 to 90.0; Typical -60.0 to 60.0; Units degrees).***	N/A

Sonar Data	S16[N][M]	Sonar values in dB x 128 (S16Q7). **(Range: -256.0dB to 255.992dB; Units See Note).** **Note: To get to dB convert the value into F32 and then divide by 128.**	N/A
<magic end>	U32	2B3C4D5E	N/A

The geometry for the wedge display can be calculated from the Sound Velocity, Sample Rate and M.

Range = (Sound Velocity / Sample Rate) * M

Notes on rendering a wedge display:

The **Sonar Data** values are the values for the specified **Beam Angle.** As the beam width is not available for each **Beam Angle,** it is recommended to interpolate between each **Sonar Data** value across the wedge display.

4.5.2 Sidescan Data (SIDEDATA)

Post processed and corrected sidescan data used to produce a standard waterfall display.

Both command message and request status cannot be used and are invalid.

Field designation	Size/Type	Description	Flags
Common Header		<packet type> = SIDEDATA	
		<packet version> = 2	
Time (accurate)	U64	A nanosecond time stamp of sample zero This time is referenced at zero at 000hrs (UTC time is calculated from the timestamp of the ZDA sentence and or a PPS signal when available) No local time zone correction is applied. This time stamp will be free running if UTC time synchronisation is not available. See Appendix E: Sample Zero Time. ***(Range: 0 to 18,446,744,073,709,551,615; Units ns).***	000001xx
Ping Number	U32	Ping Number ID. Used to retrieve data from other messages which relate to the same ping.	000002xx
Latitude	F64	Latitude at transducer in degrees. North is positive. ***(Range: -90.0 to 90.0; Typical -90 to 90; Units Degrees).***	000004xx
Longitude	F64	Longitude at transducer in degrees. East is positive. ***(Range: -180.0 to 180.0; Typical -180 to 180; Units Degrees)***	000008xx
Bearing	F32	Bearing/Heading of vessel in degrees ***(Range: 0.0 to 360.0; Typical 0 to 360; Units Degrees)***	000010xx
Sample Rate	F32	Sample rate in Hz for sonar data. ***(Range: 0 to 3.4 *10^{38}; Typical 2500 to 40000; Units Hz).***	000020xx
Sound Velocity	F32	Sound Velocity in meters per second to use for range calculations ***(Range: 1350.0 to 1600.0; Typical 1500.0; Units m/s).***	000040xx
Absorption Loss	F32	Absorption loss used in dB/km – 1 way. ***(Range: -3.4 *10^{38} to 3.4 *10^{38}; Typical 5 to 70; Units dB/km).***	000080xx
Spreading Loss	F32	Spreading loss used in dB/decade – 2 way ***(30 or 40; Typical 40.0; Units dB/decade).***	000100xx
N	U32	Number of sidescan beams. ***(Range: 0 to 256; Typical 2; Units N/A).*** **This is currently set to 2**	000200xx
M	U32	Number of samples of data per beam. ***(Range: 0 to 4096; Typical 0 to 2048; Units N/A).***	000400xx

		Maximum is 4096	
Sample Type	U32	0 if un-calibrated, 1 if calibrated	000800xx
Reserved	U32[4]	Reserved for future use	N/A
Sidescan Data	S16[N][M]	Sonar values in dB x 128 (S16Q7) **(Range: -256.0dB to 255.992dB; Units See Note).** **Note: To get to dB convert the value into F32 and then divide by 128. First M samples is beam 1, second is beam 2…**	N/A
<magic end>	U32	2B3C4D5E	N/A

4.5.3 Correct Bathy Data (BATHYCOR)

Post processed and corrected bathymetry data containing the current ping sea floor points used to generate bathymetric maps.

This message supports the ability to send more than one detection point per beam. This option is set in the Bathy Options message.

Both command message and request status cannot be used and are invalid.

Field designation	Size/Type	Description	Flag
Common Header		<packet type> = BATHYCOR	
		<packet version> = 3	
Time (accurate)	U64	A nanosecond time stamp of sample zero This time is referenced at zero at 000hrs (UTC time is calculated from the timestamp of the ZDA sentence and or a PPS signal when available) No local time zone correction is applied. This time stamp will be free running if UTC time synchronisation is not available. See Appendix E: Sample Zero Time. ***(Range: 0 to 18,446,744,073,709,551,615; Units ns).***	000001xx
Max Beams	U32	Maximum number of beams generated by the DRX Unit. ***(Range: 0 to 4,294,967,295; Typical 256; Units N/A).***	000002xx
N	U32	Number of points sent in this message. Invalid detection points will not be sent. This number can be smaller or larger than Max Beams depending on if one or more detection points are sent per beam angle. See BATHYOPT.	000004xx
Ping Number	U32	Ping Number ID. Used to retrieve data from other messages which relate to the same ping.	000008xx
Latitude	F64	Latitude from GPS sensor in degrees. North is positive. ***(Range: -90.0 to 90.0; Typical -90 to 90; Units Degrees)***	000010xx
Longitude	F64	Longitude from GPS sensor in degrees, East is positive ***(Range: -180.0 to 180.0; Typical -180 to 180; Units Degrees)***	000020xx
Bearing	F32	Bearing/Heading of vessel in degrees. ***(Range: 0.0 to 360.0; Typical 0 to 360; Units Degrees)***	000040xx
Roll	F32	Roll of vessel at transmission of this ping in degrees. ***(Range: -90.0 to 90.0; Typical -90 to 90; Units Degrees).***	000080xx
Pitch	F32	Pitch of vessel at transmission of this ping in degrees ***(Range: -90.0 to 90.0; Typical -90 to 90; Units Degrees).***	000100xx
Heave	F32	Heave of vessel at transmission of this ping in meters. ***(Range: -3.4 *10^{38} to 3.4 *10^{38}; Typical -10.0 to 10.0; Units m).***	000200xx
Sample Type	U32	0 = un-calibrated 1 = calibrated	000400xx
Tide Applied	F32	Tide adjustment applied to depth values in meters.	000800xx

		(Range: -3.4 *10^{38} to 3.4 *10^{38}; Typical -15.0 to 15.0; Units m).	
Flags	U32	0x00000001 = One detection per beam (Set in BATHYOPT) 0x00000002 = Multiple detections per beam (Set in BATHYOPT)	001000xx
Reserved	U32[2]		N/A
DET POINTS	**DPS**[N]	Detection points for this ping, see DPS struct below: No Fish : No Backscatter With Fish : No Backscatter No Fish : With backscatter With Fish : With backscatter	 000000xx 002000xx 004000xx 006000xx
<magic end>	U32	2B3C4D5E	N/A

4.5.3.1 DPS Struct

Field designation	Size/Type	Description
Beam Index	U32	Beam index number. Used to indicate which beam the detection is from. There can be more than one point for each beam if Flags indicate Multiple Detections. **Range is 0 to (Max Beams – 1)**
X	F32	Displacement from position source (given in header) in meters laterally along the west/east axis. East is positive ***(Range: -3.4 *10^{38} to 3.4 *10^{38}; Typical -1500.0 to 1500.0; Units m).***
Y	F32	Displacement from position source (given in header) in meters laterally along the north/south axis. North is positive. ***(Range: -3.4 *10^{38} to 3.4 *10^{38}; Typical -1500.0 to 1500.0; Units m).***
Z	F32	Corrected depth in meters below the sea level. Corrected for transducer draft, sound speed and tide if available. Negative down. ***(Range: -3.4 *10^{38} to 3.4 *10^{38}; Typical -1500.0 to 10.0; Units m).*** **'Quality' value of 0 indicates an invalid point**
Beam Angle	F32	Angle of the beam used to calculate detection point in degrees. Positive to starboard. ***(Range: -90.0 to 90.0; Typical -60.0 to 60.0; Units degrees).***
Backscatter	F32	Corrected received signal strength value (dB) for this beam, at the detected seafloor line. The value has been corrected for spreading loss, angle of incidence, specular returns and a calibration applied to compensate for transducer and installation artifacts. The values are between the range -64.0dB to 32.0dB. ***(Range: -64.0 to 32.0; Typical -64.0 to 32.0; Units dB).***
Detection Type	U8	Detection process used to define the point as bit flags 0x01 – indicates this detection is the best detection 0x10 – indicates detection used WMT (Weighted Mean Time) 0x20 – indicates detection used SAC (Split Aperture) 0x40 – indicates detection is automatically generated
2D Fish	U8	Fish intensity value for largest fish target vertically above the detection point. Offset dB steps – subtract 192 to get dB target strength value. 0 = no targets 1 = -191dB target strength 2 = -190dB target strength 192 = 0dB target strength 255 = 63dB (max intensity)

Field designation	Size/Type	Description
Detection Quality	U8	0 = invalid 1-100 = confidence of detection as percent
Backscatter Quality	U8	0 = invalid 1-100 = confidence of backscatter result as percent
Reserved	U32	TBD

4.5.4 Fish Finder Data (FISHDATA)

Post processed and corrected fish finder data which relates to a beam angle and beam width which has been configured using the Fish Finder Setting message. (4.5.5) The number of beams in this message is fixed by the product version of the DRX – all beams are sent all the time so history can be collected.

Both command message and request status cannot be used and are invalid.

Field designation	Size/Type	Description	Flag
Common Header		<packet type> = FISHDATA <packet version> = 2	
Time (accurate)	U64	A nanosecond time stamp of sample zero This time is referenced at zero at 000hrs (UTC time is calculated from the timestamp of the ZDA sentence and or a PPS signal when available) No local time zone correction is applied. This time stamp will be free running if UTC time synchronisation is not available. See Appendix E: Sample Zero Time. ***(Range: 0 to 18,446,744,073,709,551,615; Units ns).***	000001xx
Ping Number	U32	Ping Number ID. Used to retrieve data from other messages which relate to the same ping.	000002xx
Latitude	F64	Latitude at transducer in degrees. North is positive. ***(Range: -90.0 to 90.0; Typical -90 to 90; Units Degrees).***	000004xx
Longitude	F64	Longitude at transducer in degrees. East is positive. ***(Range: -180.0 to 180.0; Typical -180 to 180; Units Degrees)***	000008xx
Bearing	F32	Bearing/Heading of vessel in degrees ***(Range: 0.0 to 360.0; Typical 0 to 360; Units Degrees)***	000010xx
Sample Rate	F32	Sample rate in Hz for this data. ***(Range: 0 to 3.4 *10^{38}; Typical 2500 to 40000; Units Hz).***	000020xx
Sound Velocity	F32	Sound velocity in meters per second to use for range calculations. ***(Range: 1350.0 to 1600.0; Typical 1500.0; Units m/s).***	000040xx
Absorption Loss	F32	Absorption loss used in dB/km – 1 way. ***(Range: 0 to 3.4 *10^{38}; Typical 5 to 70.0; Units dB/km).***	000080xx
Spreading Loss	F32	Spreading loss used in dB/decade – 2 way ***(Range: -30 or 40; Typical 40.0; Units dB/decade).***	000100xx
N	U32	Number of fish finder beams. ***(Range: 0 to 4,294,967,295; Typical 1 to 5; Units N/A).***	000200xx
M	U32	Number of samples of data per beam, range and sample rate dependent. ***(Range: 0 to 4,294,967,295; Typical 500 to 8000; Units N/A).***	000400xx
Sample Type	U32	0 if un-calibrated, 1 if calibrated	000800xx
Reserved	U32[4]	Reserved for future use	N/A
White Line	U32[N]	Sample in beam for white line which is used to indicate where the sea floor is. ***(Range: 0 to 4096; Typical 0 to 2048; Units N/A).***	N/A
Beam Angle	F32[N]	Angle of beam in degrees –ve to port	N/A

		(Range: -90.0 to 90.0; Typical -60.0 to 60.0; Units degrees).	
Beam Width	F32[N]	Width of beam in degrees ***(Range 1 to 180; Typical 1 to 180; Units degrees).***	N/A
Fish Finder Data	S16[M][N]	Fish finder data by beam. In fractional dB x 128. First M samples is beam 1, second is beam 2… **(Range: -256.0dB to 255.992dB; Units See Note).** Note: **To get to dB convert the value into F32 and then divide by 128.**	N/A
<magic end>	U32	2B3C4D5E	N/A

4.5.5 Fish Finder Setting (FISH_SET)

Used to configure the fish finder beams on the DRX unit. The number of beams will be determined by the model of DRX unit you are connected to. Currently this is set to five beams. Beam information outside of the sonar swath will not be populated.

If fewer beams are desired than are presently configured set the beam width of any unwanted beams to zero to show that the beam is not active. Inactive beams must follow active beams.

Field designation	Size/Type	Description	Flag
Common Header		<packet type> = FISH_SET	
		<packet version> = 3	
Beam Number	U32	Fish finder beam number for these settings. ***(Range: 0 to 5; Typical 0 to 5; Units N/A).***	000001xx
Beam Angle	F32	Angle of beam in degrees –ve to port. ***(Range: -89.9 to 89.9; Typical : -89.9 to 89.9; Units degrees).***	000002xx
Beam Width	F32	Width of beam in degrees Beam Width + Beam Angle should not exceed beam angle limits. ***(Range 1 to 180; Typical 1 to 180; Units degrees).***	000004xx
FS Start Depth	F32	Start depth in meters for fish statistics (Valid range 0.0 to positive maximum range) ***This value MUST be smaller than 'FS End Depth' else behavior is undefined*** ***Used when FS Mode is set to 1***	000008xx
FS End Depth	F32	End depth in meters for fish statistics If End depth is less than or equal to Start depth no statistics will be derived (Valid range 0.0 to positive maximum range) ***This value MUST be greater than 'FS End Depth' else behavior is undefined*** ***Used when FS Mode is set to 1***	000010xx
FC Seafloor Distance	F32	Distance in meters from seafloor to collect fish statistics to (Valid range 0.1 to positive maximum range) ***Used when FS Mode is set to 2***	000020xx
FS Threshold	F32	SNR Threshold in dB. Only signals with a signal to noise ratio above this threshold value will be considered. (Valid range 0.0dB to 100.0 dB)	000040xx
FS Mode	U32	1 = Between Depths 2 = From Seafloor	000080xx
Noise Filter	U32	0 = Off 1 = Low 2 = Med 3 = High	000100xx
Beam Scale	U32	This is a global setting for all beams. The last set value will be used. 0 = Range 1 = Depth	000200xx
Reserved	U32[6]		N/A
<magic end>	U32	2B3C4D5E	N/A

4.5.6 Fish Statistics (FISHSTAT)

Post processed fish statistics – count of fish per beam computed using fish statistic configuration from the Fish Finder Setting message. (4.5.5)

Field designation	Size/Type	Description	Flag
Common Header		<packet type> = FISHSTAT	
		<packet version> = 2	
Time (accurate)	U64	A nanosecond time stamp of sample zero This time is referenced at zero at 000hrs (UTC time is calculated from the timestamp of the ZDA sentence and or a PPS signal when available) No local time zone correction is applied. This time stamp will be free running if UTC time synchronisation is not available. See Appendix E: Sample Zero Time. ***(Range: 0 to 18,446,744,073,709,551,615; Units ns).***	000001xx
Ping Number	U32	Ping Number ID. Used to retrieve data from other messages which relate to the same ping.	000002xx
Latitude	F64	Latitude at transducer in degrees. North is positive. ***(Range: -90.0 to 90.0; Typical -90 to 90; Units Degrees).***	000004xx
Longitude	F64	Longitude at transducer in degrees. East is positive. ***(Range: -180.0 to 180.0; Typical -180 to 180; Units Degrees)***	000008xx
Bearing	F32	Bearing/Heading of vessel in degrees ***(Range: 0.0 to 360.0; Typical 0 to 360; Units Degrees)***	000010xx
Absorption Loss	F32	Absorption loss used in dB/km – 1 way. ***(Range: 0 to 3.4 *10^{38}; Typical 40.0; Units dB/km).***	000020xx
Spreading Loss	F32	Spreading loss used in dB/decade – 2 way ***(Range: 30 or 40; Typical 40.0; Units dB/decade).***	000040xx
N	U32	Number of fish finder beams. Fixed by DRX version (Typical 1-5)	000080xx
M	U32	Number of bins per beam. Fixed by DRX version (Typical 10)	000100xx
Sample Type	U32	0 if un-calibrated, 1 if calibrated	000200xx
Bin Size	F32	Size in dBs for the range of values in each bin. The first bin maximum is set at 0.0dB. If the Bin Size is 6dB then all targets between 0.0 and -6dB will be shown in the first bin, -6.0 to -12.0 dB targets will be shown in the second bin and so on.	000400xx
Reserved	U32[6]		N/A
Fish Counts	U16[M][N]	Fish counted per bin per beam. First M samples are the count of fish per bin for beam 1.	
<magic end>	U32	2B3C4D5E	N/A

4.5.7 Water Column Targets (WCT_DATA)

Post processed and corrected water column targets. Targets are generated at a finite resolution and will be passed as one continuous array of points. Each point is located as an x,y,z value from a reference latitude and longitude. The z value is the depth from the face of the transducer.

This message will only be received if asked for via MSG_REQ_. Both command message and request status cannot be used and are invalid.

Field designation	Size/Type	Description	Flags
Common Header		<packet type> = WCT_DATA	
		<packet version> = 2	
Time (accurate)	U64	A nanosecond time stamp of sample zero This time is referenced at zero at 000hrs (UTC time is calculated from the timestamp of the ZDA sentence and or a PPS signal when available) No local time zone correction is applied. This time stamp will be free running if UTC time synchronisation is not available. See Appendix E: Sample Zero Time. ***(Range: 0 to 18,446,744,073,709,551,615; Units ns).***	000001xx
Latitude	F64	Latitude at transducer in degrees. North is positive. ***(Range: -90.0 to 90.0; Typical -90 to 90; Units Degrees).***	000002xx
Longitude	F64	Longitude at transducer in degrees. East is positive. ***(Range: -180.0 to 180.0; Typical -180 to 180; Units Degrees)***	000004xx
N	U32	Number of water column targets (Currently maximum 1000 per ping, hardcoded)	000008xx
Bearing	F32	Bearing/Heading of vessel in degrees. ***(Range: 0.0 to 360.0; Typical 0 to 360; Units Degrees)***	000010xx
Ping Number	U32	Ping Number ID. Used to retrieve data from other messages which relate to the same ping.	000020xx
Tide Applied	F32	Tide adjustment applied to depth values in meters. ***(Range: -3.4 *10^{38} to 3.4 *10^{38}; Typical -15.0 to 15.0; Units m).***	000040xx
Reserved	U32[7]		N/A
Target Data	**WCD**[N]	Water column data points. See WCD structure below.	
<magic end>	U32	2B3C4D5E	

4.5.7.1 WCD structure

Field designation	Size/Type	Description
X	F32	Displacement from position source (given in header) in meters laterally along the west/east axis. East is positive ***(Range: -3.4 *10^{38} to 3.4 *10^{38}; Typical -1500.0 to 1500.0; Units m).***
Y	F32	Displacement from position source (given in header) in meters laterally along the north/south axis. North is positive. ***(Range: -3.4 *10^{38} to 3.4 *10^{38}; Typical -1500.0 to 1500.0; Units m).***
Z	F32	Corrected depth in meters below the sea level. Corrected for transducer draft, sound speed and tide if available. Negative down. ***(Range: -3.4 *10^{38} to 3.4 *10^{38}; Typical -1500.0 to 0.0; Units m).***
Target Strength	S16	Intensity value for water column point as dB target strength x 128 (S16Q7) **(Range: -256.0dB to 255.992dB; Units See Note).** **Note: To get to dB convert the value into F32 and then divide by 128.**
Reserved	U16	Reserved for future use, quality and or flag values

4.5.8 Water Column Settings (WCT_SETT)

Used to configure how the water column targets are generated.

Field designation	Size/Type	Description	Flags
Common Header		<packet type> = WCT_SETT	
		<packet version> = 2	
WC Start Depth	F32	Depth in meters to start collecting water column targets Positive value. (Valid range 0.0 to positive maximum range) ***This value MUST be smaller than 'WS End Depth' else behavior is undefined*** ***Used when WC Mode is set to 1***	000001xx
WC End Depth	F32	Depth in meters to stop collection water column targets Positive value. (Valid range 0.0 to positive maximum range) ***This value MUST be greater than 'WS Start Depth' else behavior is undefined*** ***Used when WC Mode is set to 1***	000002xx
WC Seafloor Distance	F32	Distance in meters from seafloor to collect water column targets to (Valid range 0.1 to positive maximum range) ***Used when WC Mode is set to 2***	000004xx
WC Threshold	F32	Target strength minimum for targets **Minimum -200dB (Disables thresholding)** **Maximum 200dB (Disables all detection!)** **Default value -60dB** **Recommended range -96dB to 0dB**	000008xx
WC Mode	U32	1 = Between Depths 2 = From Seafloor	000010xx
WC Seafloor Gate	U32	1 = Above Seafloor 2 = All In Ping (Seafloor treated as normal target candidates)	000020xx
Reserved	U32[6]		N/A
<magic end>	U32	2B3C4D5E	N/A

4.6 Uncorrected Data

4.6.1 Raw Sensor Data (RAW_SENS)

Contains the raw sensor message as it was received on the DRX unit port such as NMEA messages received on NMEA 0183, NMEA2000 or Ethernet.

<table>
<tr><th>Field designation</th><th>Size/Type</th><th>Description</th><th>Flags</th></tr>
<tr><td rowspan="2">Common Header</td><td rowspan="2"></td><td><packet type> = RAW_SENS</td><td rowspan="2"></td></tr>
<tr><td><packet version> = 3</td></tr>
<tr><td>Time (accurate)</td><td>U64</td><td>A nanosecond time stamp of arrival of rising edge of the first character of serial data (or the actual time of stimulus for internal and edge triggered events) in UTC time referenced at zero at 000hrs (UTC time is calculated from the timestamp of the ZDA sentence and or a PPS signal when available) No local time zone correction is applied.
On systems not supporting UTC time this will be system referenced time
(Range: 0 to 18,446,744,073,709,551,615; Units ns).</td><td>000001xx</td></tr>
<tr><td>Port</td><td>U8</td><td>Describes the physical connection/source of the information. This also describes the format of the incoming information.

Expect ports 1-16 and 33-80 to have ASCII message data* – Other messages have various proprietary formats and should be ignored by third parties at this time.

* Whatever is sent in this serial port will be split into messages at the line feed character. These messages are time stamped at the rising edge of the start bit of the first character. If a message is NMEA0183 format it will start with a $ character, if it is TSS1 a : will be the first character. Sound velocity sensors will be as per manufacturer specification. If the data is binary or is missing a Line feed, or messages exceed 100 characters then the message will resemble a fragment of the incoming data like a TCP packet.

E.g. Port 33(RS232): Message: “$GPGGA,xxx.x…<CR><LF>”

<table>
<tr><th>Port range</th><th>Protocol</th></tr>
<tr><td>0</td><td>Unknown / Internal</td></tr>
<tr><td>1-16</td><td>NMEA0183*</td></tr>
<tr><td>17-32</td><td>NMEA2000 (TBD - Ignore)</td></tr>
<tr><td>33-48</td><td>RS232*</td></tr>
<tr><td>49-64</td><td>RS422*</td></tr>
<tr><td>65-80</td><td>RS485*</td></tr>
<tr><td>81-96</td><td>Ethernet (TBD - Ignore)</td></tr>
<tr><td>97-112</td><td>ENL_FPGA_BINARY (Ignore)</td></tr>
<tr><td>113-128</td><td>ENL_FPGA_MISC (Ignore)</td></tr>
</table>
</td><td>000002xx</td></tr>
</table>

Protocol	U8	For packet version 0-2, this is undefined. Otherwise this is the sensor message as defined in Appendix A	000004xx
Reserved	U16	Unused	N/A
N	U32	Number of bytes from beginning of message to the end **(Range: 0 to 1024; Typical 5 to 128 Units N/A).**	000010xx
Message	U8[N]	Message Data	N/A
<magic end>	U32	2B3C4D5E	N/A

4.6.2 Raw Bathy Data (BATHYRAW)

Contains the raw and uncorrected bathy data which can be post processed using this message and the raw sensor data. This is only required if you wish to perform your own post corrections instead of the built in corrections inside the DRX unit.

Field designation	Size/Type	Description	Flag
Common Header		<packet type> = BATHYRAW	
		<packet version> = 3	
Time (accurate)	U64	A nanosecond time stamp of sample zero This time is referenced at zero at 000hrs (UTC time is calculated from the timestamp of the ZDA sentence and or a PPS signal when available) No local time zone correction is applied. This time stamp will be free running if UTC time synchronisation is not available. See Appendix E: Sample Zero Time. ***(Range: 0 to 18,446,744,073,709,551,615; Units ns).***	000001xx
Max Beams	U16	Maximum number of beams generated by the DRX Unit. ***(Range: 0 to 4,294,967,295; Typical 256; Units N/A).***	000002xx
N	U16	Number of points sent in this message. Invalid detection points will not be sent. This number can be smaller or larger than Max Beams depending on if one or more detection points are sent per beam angle. See BATHYOPT.	000004xx
Ping Number	U32	Ping Number ID. Used to retrieve data from other messages which relate to the same ping.	000008xx
Sample Type	U32	0 = un-calibrated 1 = calibrated	000010xx
Sample Rate	F32	Sample rate in Hz ***(Range: 0 to 3.4 *10^{38}; Typical 2500 to 40000; Units Hz).***	000020xx
Sound Velocity	F32	Sound Velocity in meters per second at the transducer. Sound Velocity in meters per second to use for range calculations ***(Range: 1350.0 to 1600.0; Typical 1500.0; Units m/s).***	000040xx
Absorption Loss	F32	Absorption loss used in dB/km – 1 way. ***(Range: 0 to 3.4 *10^{38}; Typical 40.0; Units dB/km).***	000080xx
Spreading Loss	F32	Spreading loss used in dB/decade – 2 way ***(Range: 30 or 40; Typical 40.0; Units dB/decade).***	000100xx
Transmission Centre Frequency	F32	Current transmitter center operating frequency in Hz. ***(Range: 0 to 5120000; Typical 40000.0 to 160000.0; Units Hz).***	000200xx
Transmission Bandwidth	F32	Current transmitted bandwidth in Hz. ***(Range: 0 to 5120000; Typical 0 to 60,000; Units Hz).***	000400xx
Tx Power Level	F32	Voltage (volts) RMS applied to transmitter in dB (referenced to 1V RMS). ***(Range: -3.4 *10^{38} to 3.4 *10^{38}; Typical -50 to +60; Units Vrms).***	000800xx
Flags	U32	0x00000001 = One detection per beam (Set in BATHYOPT) 0x00000002 = Multiple detections per beam (Set in BATHYOPT)	001000xx
Reserved	U32[6]		N/A

DET POINTS	**RPS**[N]	Raw Detection points for this ping, see RPS struct below	N/A
<magic end>	U32	2B3C4D5E	N/A

4.6.2.1 RPS Struct

Field designation	Size/Type	Description
Beam Index	U32	Beam index number. Used to indicate which beam the detection is from. There can be more than one point for each beam if Flags indicate Multiple Detections. **Range is 0 to (Max Beams – 1)**
Detection Point	F32	Non-corrected fractional sample number with the reference to the receiver's acoustic centre with the sample zero at the centre of the transmit pulse. See Appendix E: Sample Zero Time. ***(Range: -3.4 *10^{38} to 3.4 *10^{38}; Typical 0.0 to 8000.0; Units N/A).***
Rx Angle	F32	Beam steering angle with reference to receiver's acoustic centre in the sonar reference frame, at the detection point; in degrees. Positive Starboard, negative Port. ***(Range: -90.0 to 90.0; Typical -60.0 to 60.0; Units degrees).***
Flags	U16	Detection process used to define the point as bit flags 0x01 – indicates this detection is the best detection 0x10 – indicates detection used WMT (Weighted Mean Time 0x20 – indicates detection used SAC (Split Aperture) 0x100 – backscatter value is valid
Detection Quality	U8	0 = invalid 1-100 = confidence of detection as percent
Backscatter Quality	U8	0 = invalid 1-100 = confidence of backscatter result as percent
Backscatter	float	Received signal strength value (dB) for this beam, at the detected seafloor line. Corrections for absorption loss and spreading loss as per the fields above have been applied, to limit the dynamic range. The values are between the range -64.0dB to 32.0dB. ***(Range: -64.0 to 32.0; Typical -64.0 to 32.0; Units dB).***

5. Appendix A: Sensor Sentence and Model Information

ID	Sentence
0	N/A
10	GGA
11	GGL
12	GNS
13	RMC
14	GGK
15	ZDA
20	HDM
21	HDT
22	HDG
23	VTG
31	PSXN
32	PASHR
33	PFEC,ATT
34	PFEC,HVE
35	RCD
36	MTW
100	TSS1
101	Minisense
102	Valeport

ID	Sensor Model
0	Unknown
1	User Defined 1
2	User Defined 2
3	User Defined 3
4	User Defined 4
5	User Defined 5
10	WASSP 160
11	WASSP 80
12	WASSP 160 Vs
13	WASSP 160 Vl
40	Applanix POS MV V4
41	Furuno SC30
42	Furuno SC50
43	JRC JLR20
44	CDL Minisense
45	Kongsburg MRU
46	Teledyne TSS
47	Maretron SSC200
48	SMC IMU-108
49	SMC IMU
50	Furuno SC
51	Hemisphere
52	Spatial

ID	Sensor Model
53	Trimble

6. Appendix B: Connect and Listen Example

Step 1: Look for DRX units on the network

Step 2: Select and establish TCP connection to one of the DRX units

Step 3: Send 'Message Request' with messages client wishes to see. First the client can request which messages are supported by the DRX using the following message. (Optional)

Field designation	Size/Type	Value
<magic start>	U32	D4C3B2A1
<length>	U32	76
<packet type>	U64	"MSG_REQ_"
<packet version>	U32	0
<msg flags>	U32	Binary Bit Mask 0000 0000 | 0000 0000 | 1000 0000 | 0000 0010 Message Specific Flags | System Code System Code = Request Status Message Specific Flags = N
< packet time stamp>	U64	Time message was formed 64-bit ns timestamp. This time is not guaranteed to be synchronized to anything, and is mainly used for development/recording.
Security	U64[4]	0,0,0,0
Spare	U16	0
Command Type	U16	0
Message Types	U16	0
N	U16	0
<magic end>	U32	2B3C4D5E

Response will be something like this:

Field designation	Size/Type	Value
<magic start>	U32	D4C3B2A1
<length>	U32	212
<packet type>	U64	"MSG_REQ_"
<packet version>	U32	1
<msg flags>	U32	Binary Bit Mask 0000 0000 | 0000 0000 | 1000 0000 | 1000 0000 Message Specific Flags | System Code System Code = Acknowledge Message Specific Flags = N
< packet time stamp>	U64	Time message was formed 64-bit ns timestamp. This time is not guaranteed to be synchronized to anything, and is mainly used for development/recording.
Security	U64[4]	0,0,0,0
Spare	U16	0
Command Type	U16	0
Message Types	U16	0
N	U16	17
Requested Message[0]	U64	MSG_REQ_

Requested Message[1]	U64	SONASTAT
Requested Message[2]	U64	PING_REQ
Requested Message[3]	U64	SENS_SET
Requested Message[4]	U64	SYS_INFO
Requested Message[5]	U64	SERIALST
Requested Message[6]	U64	PRO_OPTN
Requested Message[7]	U64	GEN_MESG
Requested Message[8]	U64	SON_DSET
Requested Message[9]	U64	FISH_SET
Requested Message[10]	U64	WCT_SETT
Requested Message[11]	U64	SONADISP
Requested Message[12]	U64	SIDEDATA
Requested Message[13]	U64	BATHYCOR
Requested Message[14]	U64	FISHDATA
Requested Message[15]	U64	WCT_DATA
Requested Message[16]	U64	SENUPDAT
<magic end>	U32	2B3C4D5E

Then a sub set of these messages can be requested by the client, for example if corrected bathy, sensor information and water column targets were desired:

Field designation	**Size/Type**	**Value**
<magic start>	U32	D4C3B2A1
<length>	U32	100
<packet type>	U64	"MSG_REQ_"
<packet version>	U32	0
<msg flags>	U32	Binary Bit Mask 0000 0000 | 0000 0000 | 1010 0000 | 0000 0001 Message Specific Flags | System Code System Code = Command Message Specific Flags = N
< packet time stamp>	U64	Time message was formed 64-bit ns timestamp. This time is not guaranteed to be synchronized to anything, and is mainly used for development/recording.
Security	U64[4]	0,0,0,0
Spare	U16	0
Command Type	U16	1 (Add) – note that the flag for this field set
Message Types	U16	0
N	U16	3 (count of messages) – note that the flag for this field set
Requested Message[0]	U64	BATHYCOR
Requested Message[1]	U64	WCT_DATA
Requested Message[2]	U64	SENUPDAT
<magic end>	U32	2B3C4D5E

Once this message has been acknowledged - every time the DRX creates and sends these three messages this client will receive a copy of them. No other messages sent from the DRX will be received by the client.

7. Appendix C: Ack/Nack Example

If a command or request is completely valid an Acknowledgement message will be sent – thus a reciprocal message with all flagged fields set and the System Code set to 1 = Acknowledge. If all flagged fields are set to values which are considered invalid then a reciprocal message with the same flags will be sent with the System Code set to 129 = Not Acknowledge. In the case where some of the flagged fields are correct and some of them are incorrect both an Acknowledge and a Not Acknowledge will be sent.

In this example we will use the Ping Request message – and set the range to an invalid number and the power level to a valid number.

This is the request from the client:

Field designation	Size/Type	Description	Flagged
<magic start>	U32	D4C3B2A1	
<length>	U32	96	
<packet type>	U64	PING_REQ	
<packet version>	U32	1	
<msg flags>	U32	Bit Mask 0000 0000 | 0000 0000 | 0011 0110 | 0000 0001 Message Specific Flags | System Code Message Specific flags used to indicate if the data in each field is valid. Refer to Flags column.	
<packet time stamp>	U64	Time message was formed 64-bit ns timestamp. This time is not guaranteed to be synchronized to anything, and is mainly used for development/recording.	
Ping Mode	U32	0 (Note this is invalid but the flag is not set so will be ignored)	No
Range	F32	10000.0 (metres)	Yes
Range Mode	U32	0 (Manual Range)	Yes
Ping Mode	U32	0	No
Power Mode	U32	1 = Manual	Yes
Power Level	F32	100.0 (Full power)	Yes
Pulse Width	U32	0	No
Reserved	U32[8]	0,0,0,0,0,0,0,0	No
<magic end>	U32	2B3C4D5E	N/A

The power level is first acknowledged:

Field designation	Size/Type	Description	Flagged
<magic start>	U32	D4C3B2A1	
<length>	U32	96	
<packet type>	U64	PING_REQ	
<packet version>	U32	1	
<msg flags>	U32	Bit Mask 0000 0000 | 0000 0000 | 0011 0110 | 1000 0000 Message Specific Flags | System Code Message Specific flags indicate which fields were Acknowledged	

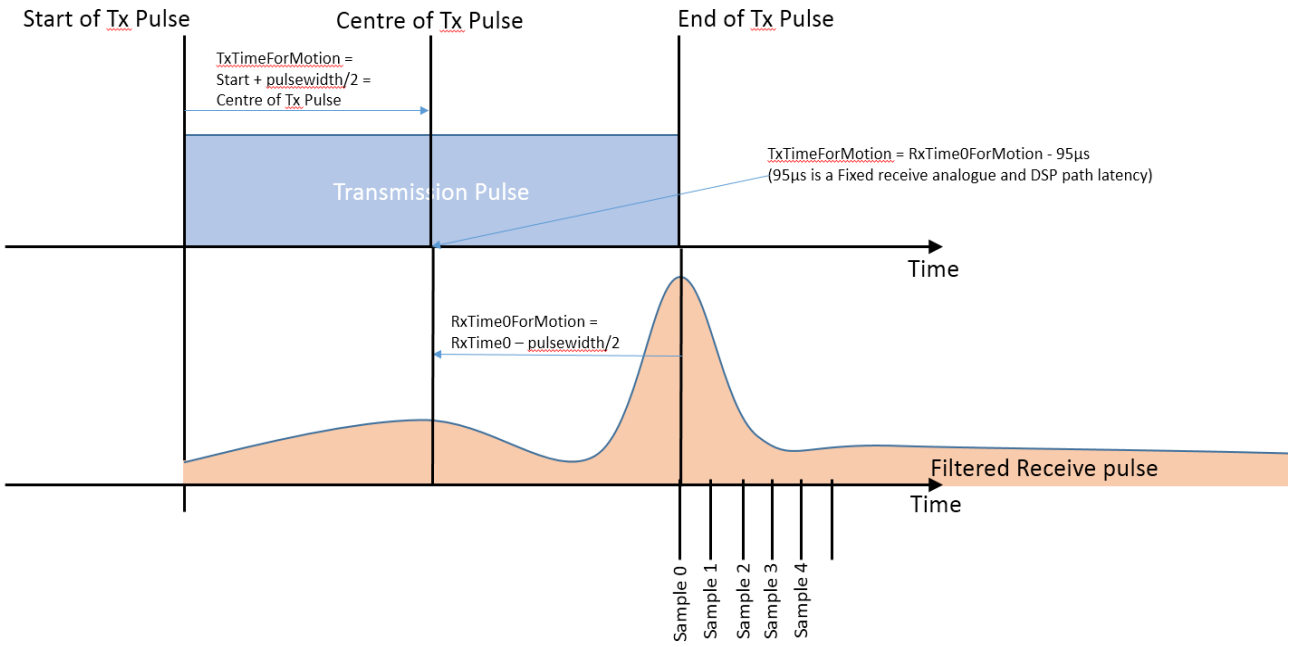
<packet time stamp>	U64	Time message was formed 64-bit ns timestamp. This time is not guaranteed to be synchronized to anything, and is mainly used for development/recording.	
Ping Mode	U32	2 (Current ping mode)	No
Range	F32	50.0 (metres – where this is the current range setting)	No
Range Mode	U32	0 (Manual Range)	Yes
Ping Mode	U32	0	No
Power Mode	U32	1 = Manual	Yes
Power Level	F32	100.0 (Full power)	Yes
Pulse Width	U32	500000 (current pulse width in ns)	No
Reserved	U32[8]	0,0,0,0,0,0,0,0	No
<magic end>	U32	2B3C4D5E	N/A

Then the Range is Not Acknowledged in a separate message

Field designation	Size/Type	Description	Flagged
<magic start>	U32	D4C3B2A1	
<length>	U32	88	
<packet type>	U64	PING_REQ	
<packet version>	U32	1	
<msg flags>	U32	Bit Mask 0000 0000 | 0000 0000 | 0000 0010 | 1000 0001 Message Specific Flags | System Code Message Specific flags used to indicate which field was Not Acknowledged	
<packet time stamp>	U64	Time message was formed 64-bit ns timestamp. This time is not guaranteed to be synchronized to anything, and is mainly used for development/recording.	
Ping Mode	U32	2 (Current ping mode)	No
Range	F32	50.0 (metres – where this is the current range setting)	Yes
Range Mode	U32	0 (Manual Range)	No
Ping Mode	U32	0	No
Power Mode	U32	1 = Manual	No
Power Level	F32	100.0 (Full power)	No
Pulse Width	U32	500000 (current pulse width in ns)	No
Reserved	U32[8]	0,0,0,0,0,0,0,0	No
<magic end>	U32	2B3C4D5E	N/A

8. Appendix E: Sample Zero Time

Sample Zero time is the point where maximum receive signal will occur for a pulse that is received at the same time it is transmitted. Such that would occur if there was perfect reflection at the face of the transducer. Sample Zero time is defined as the centre of the Pulse transmitted in the water as it is received.



To simplify the interface for Third parties the Sample0 Time given is the time of Receive of the centre of the Tx pulse so it can be used for Rx Time zero and Tx Time Zero for motion/position compensation without any modification.

Figure 1: Diagram of Sample Time as if no path delay between Transmitter and Receiver

The time stamp for the rising edge of the transmission pulse is captured and then added to the time offset between the rising edge of the transmission pulse and the centre of the match filtered response from the receiver – this is typically half of the pulse width.

To compute range to any target compute the sample where the peak response from this target occurs and divide this by the sample rate provided for this message and multiply this by the sound velocity – then half this number to account for the two way travel time.

Range = (sound velocity * sample number) / (sample rate * 2)

9. ANNEX A: WASSP DRX Engineering and Development Protocols

This document contains messages using this same ICD format that are for internal development use. See Reference [5].