



Project: BP148 – Oculus Sonar

Title: Oculus Data Structure Definitions

Document: NT-148-D00527 Rev: 6

#### Document History...

Rev	Who	Date	Description of Amendment/Modification
1	KW	22/05/2020	Initial issue
2	KW	24/06/2020	Updated for the V2 Simple Fire commands
3	KW	31/03/2021	Document number update
4	KW	16/11/2021	Structure field corrections
5	MR	21/04/2023	Structure updates and new part numbers
6	MR	26/04/2023	Updated to include pinger locator mode

*Last edited on 21/04/2023, Printed on 26/04/2023*

#### Reference Documents...

- QA-001-D00001: Blueprint Quality Manual and specified internal operating procedures
- Specification documents and procedures contained within "Level 3" of the Blueprint Quality System.

**The electronic version of this document is the master document; therefore all printed versions are uncontrolled copies.**

*The information in this document should be considered strictly confidential, and should not be disclosed to any third parties or reproduced without the prior written consent of "Blueprint Design Engineering Ltd" (trading as "Blueprint Subsea"). Please direct any questions relating to the disclosure and distribution of this document to James Barratt (see [www.blueprintsubsea.com](http://www.blueprintsubsea.com) or email [enquiries@blueprintsubsea.com](mailto:enquiries@blueprintsubsea.com)).*

**Blueprint Subsea (Blueprint Design Engineering Ltd)**  
The Clock Tower Business Centre,  
Low Wood, Ulverston, Cumbria,  
LA12 8LY, UK

Phone: +44 (0)1539 531536  
Fax: +44 (0)1539 764227  
Web: [www.blueprintsubsea.com](http://www.blueprintsubsea.com)  
Company Number: 5881532  
VAT Number: GB 895274084

# Contents

1. Introduction .....	3
1.1. UDP Socket .....	3
1.2. TCP Socket .....	3
2. Common Data Structures.....	4
2.1. Message Header .....	4
2.1.1. Message Identifier Field .....	4
3. Messages .....	5
3.1. Status Message .....	5
3.1.1. Device Type Field .....	5
3.1.2. Part Number Field .....	6
3.1.3. Status Field .....	7
3.2. Simple Fire V1 Request.....	9
3.2.1. Master Mode Field .....	9
3.2.2. Ping Rate Field.....	9
3.2.3. Flags Field.....	10
3.2.4. Speed of Sound Field.....	10
3.3. Simple Fire V1 Response.....	11
3.3.1. Data Size Field.....	11
3.4. Simple Fire V2 Request.....	12
3.4.1. Master Mode Field .....	12
3.4.2. Ping Rate Field.....	12
3.4.3. Flags Field.....	12
3.4.4. Speed of Sound Field.....	12
3.4.5. Extended Flags Field.....	13
3.4.6. Beacon Locator Frequency Field .....	13
3.5. Simple Fire V2 Response.....	14
3.5.1. Data Size Field.....	14
3.6. Network Configuration .....	15
3.6.1. Network Mode Field .....	15
4. Operating (Master) Modes.....	16
4.1. Imaging Mode (Low or High Frequency).....	16
4.2. Beacon Locator Mode .....	16
4.2.1. Range.....	16
4.2.2. Beacon Locator Frequency.....	16
4.2.3. Returned Image.....	16

# 1. Introduction

This document details the various types and data structures used when communicating with an Oculus sonar via both UDP and TCP.

## 1.1. UDP Socket

Every Oculus imaging sonar broadcasts a UDP status datagram at regular 1Hz intervals. These datagrams contain information including the IP address and subnet mask of the Oculus on the network (either via DHCP or static allocation) as well as general health information about the unit.

**Port: 52102**

## 1.2. TCP Socket

Control of the sonar and output of image data is done via a TCP socket.

Sending a Simple Fire V1 or V2 message to the sonar will cause it to begin imaging and sending back Simple Fire V1/V2 Response messages. The sonar will continue to ping and return images for a short period of time maintaining a rate according to the pingRate field, but if no further requests are sent the sonar will stop.

An alternative mode of operation can be used by setting the network trigger flag in the request message. This will cause the sonar to ping once only and return a response.

**Port: 52100**

## 2. Common Data Structures

### 2.1. Message Header

Field	Description	Default	Format	Size	Offset
oculusId	Identifier for messages to and from the Oculus Sonar	0x4F53	uint16	2	0
sourceId	Identifier for the source of a message. This will be the serial number for messages from the sonar.	-	uint16	2	2
destinationId	Identifier for the destination of the message	-	uint16	2	4
messageId	Message identifier (section 2.1.1)	-	uint16	2	6
version	Version of message	-	uint16	2	8
payloadSize	Size of the following payload	-	uint32	4	10
partNumber	Blueprint Subsea part number	-	uint16	2	14
Total size					16

#### 2.1.1. Message Identifier Field

The following are the possible values for the message identifier field:

Value	Name	Description
1	messageStatus	Status message
21	messageSimpleFire	Simple fire request
35	messageSimplePingResult	Simple fire response message
34	messagePingResult	Full fire response message
85	messageUserConfig	User configuration message
128	MessageBootInfo	Boot information message
255	messageDummy	Dummy message (keep-alive)

## 3. Messages

### 3.1. Status Message

The status message is broadcast over UDP by each Oculus imaging sonar on the network. The UDP message should be used to obtain the network IP address and subnet mask used when creating and connecting the main TCP communication socket.

Field	Description	Units	Format	Size	Offset
header	Message header	-	(section 2.1)	16	0
deviceId	Oculus serial number	-	uint32	4	16
deviceType	Device type (section 3.1.1)	-	uint16	2	20
partNumber	Part number (section 3.1.2)	-	uint16	2	22
status	Status flags (section 3.1.3)	-	uint32	4	24
version0	ARM0 version information	-	uint32	4	28
date0	ARM0 firmware build date	-	uint32	4	32
version1	ARM1 version information	-	uint32	4	36
date1	ARM1 firmware build date	-	uint32	4	40
version2	Core version information	-	uint32	4	44
date2	Core firmware build date	-	uint32	4	48
ipAddr	Network IP address	-	uint32	4	52
ipMask	Network IP mask	-	uint32	4	56
clientAddr	Currently connected client IP address	-	uint32	4	60
macAddr[6]	Physical network address	-	uint8	6	64
temperatures[8]	Internal temperatures	°C	double	8 x 8	70
pressure	Current pressure rating	bar	double	8	134
Total size					142

#### 3.1.1. Device Type Field

The following are the possible values for the device type field:

Value	Name	Description
0	deviceTypeUndefined	Unknown device type
1	deviceTypeImagingSonar	Oculus Imaging Sonar

### 3.1.2. Part Number Field

The following are the possible values for the part number field:

Value	Description
0	Unknown part
1041	Oculus M370s
2418	Oculus MT370s
1433	Oculus MD370s
1436	Oculus MF370s
1229	Oculus MA370s
1032	Oculus M750d
2419	Oculus MT750d
1434	Oculus MD750d
1134	Oculus MF750d
1135	Oculus MA750d
1042	Oculus M1200d
2420	Oculus MT1200d
1435	Oculus MD1200d
1437	Oculus MF1200d
1228	Oculus MA1200d
2203	Oculus M3000d
2599	Oculus MT3000d
2466	Oculus MF3000d
2924	Oculus MA3000d

### 3.1.3. Status Field

The status field contains a set of flags to indicate the current operational status of the Oculus imaging sonar. Most of the flags are single bit flags, for those that are not, see the following sub sections for descriptions of the possible values.

Bit(s)	Name	Description	Bit Mask	Bit Shift
[2:0]	masterStatus	Master operational status	0x00000007	0
[5:3]	pauseReason	Pause reason during boot	0x00000038	>> 3
6	statusSocket	1 for UDP status socket connection	0x00000040	>> 6
7	mainSocket	1 for TCP main socket connection	0x00000080	>> 7
8	memorySocket	1 for TCP memory socket connection	0x00000100	>> 8
9	diagnosticsSocket	1 for TCP diagnostics socket connection	0x00000200	>> 9
10	arm1	1 for Arm1 up and running	0x00000400	>> 10
11	devloader	1 for devloader present	0x00000800	>> 11
12	noExternalTrigger	1 for external trigger not present	0x00001000	>> 12
13	jitterWarning	1 for jitter warning	0x00002000	>> 13
[15:14]	temperatureStatus	Temperature status	0x0000C000	>> 14
16	transmitFault	1 for critical transmitter fault	0x00010000	>> 16
17	transmitVoltageOversag	1 for transmit voltage over-sag	0x00020000	>> 17
18	magSwitch	1 for presence of reset magnet	0x00040000	>> 18
[25:19]	reserved	NA	0x03F80000	>> 19
26	test1Capable	1 for capable of firmware feature test 1	0x04000000	>> 26
27	test2Capable	1 for capable of firmware feature test 2	0x08000000	>> 27
28	gainBoostCapable	1 for capable of gain boost	0x10000000	>> 28
29	chirpCapable	1 for capable of chirp	0x20000000	>> 29
30	pressureSensorEquipped	1 for equipped with pressure sensor	0x40000000	>> 20
31	ahrsEquipped	1 for equipped with AHRS	0x80000000	>> 31

### Master Status Sub-Field

The following lists the possible values for the master status sub-field:

Value	Name	Description
0	masterStatusSsblBoot	Booting the SSBL
1	masterStatusSsblRun	Running the SSBL
2	masterStatusMainBoot	Booting the main application
3	masterStatusMainRun	Running the main application

### Pause Reason Sub-Field

The following lists the possible values for the pause reason sub-field:

Value	Name	Description
0	pauseMagSwitch	Magnet was present during power-up
1	pauseBootFromMain	Main application rebooted to SSBL
2	pauseFlashError	Flash memory corruption
4	pauseFirmwareError	Previous firmware upgrade attempt failed
5	pauseCompatibilityError	Incompatible firmware upgrade attempted
6	pauseBrownout	Voltage dip occurred during normal operation
7	pauseUndefined	Undefined pause reason

### Temperature Status Sub-Field

The following lists the possible values for the temperature status sub-field:

Value	Name	Description
0	temperatureGood	Temperature is within normal range
1	temperatureOverheat	High temperature, ping rate reduced
2	reserved	NA
3	temperatureOverMax	Temperature beyond maximum, system shutdown



## 3.2. Simple Fire V1 Request

A simple fire request instructs the Oculus imaging sonar to “ping”.

Field	Description	Default	Format	Size	Offset
header	Message header	-	(section 2.1)	16	0
masterMode	Current run mode	-	uint8	1	16
pingRate	Maximum ping rate	-	uint8	1	17
networkSpeed	Maximum network speed	-	uint8	1	18
gamma	Gamma correction value	127	uint8	1	19
flags	Simple fire V1 flags	-	uint8	1	20
range	Demanded range in metres or percent	-	double	8	21
gain	Demanded gain value in percent	-	double	8	29
speedOfSound	Speed-of-sound used	-	double	8	37
salinity	Salinity of the environment	-	double	8	45
Total size					53

### 3.2.1. Master Mode Field

The following are the possible values for the master mode field:

Value	Name	Description
1	LowFrequency	The sonar’s lower frequency
2	HighFrequency	The sonar’s higher frequency

### 3.2.2. Ping Rate Field

The following are the possible values for the ping rate field:

Value	Name	Description
0	pingRateNormal	10Hz maximum ping rate
1	pingRateHigh	15Hz maximum ping rate
2	pingRateHighest	40Hz maximum ping rate
3	pingRateLow	5Hz maximum ping rate
4	pingRateLowest	2Hz maximum ping rate
5	pingRateStandby	Pinging disabled

### 3.2.3. Flags Field

The following lists the available bit flags:

Bit	Name	Description
0	RangeInMetres	Set = range in metres Clear = range as a percentage
1	16BitImg	Set = 16-bit data in image Clear = 8-bit data in image
2	GainSend	Set = The Oculus will return the gain applied at the start of each range line of data. Clear = The Oculus will not return the gain values applied
3	SimpleReturn	Set = Oculus will output simple fire returns Clear = Oculus will output full fire returns (if full fire sent)
4	GainAssist	Set = Gain assist enabled Clear = Gain assist disabled
5	LowPower	Set = Low power mode enabled Clear = Low power mode disabled
6	FullBeams	Set = Use 512 beams Clear = Use 256 beams
7	NetworkTrigger	Set = Oculus only fires when instructed Clear = Oculus fires automatically according to PingRate

### 3.2.4. Speed of Sound Field

The speed of sound field can be set to specify the speed of sound in the working environment manually. Alternatively, setting this field to zero will cause the Oculus to calculate a speed of sound value using the value in the salinity field.

### 3.3. Simple Fire V1 Response

A simple fire response contains the image data for the previous “ping”.

Field	Description	Units	Format	Size	Offset
fireMessage	Simple fire message structure	-	(section 3.2)	53	0
pingId	Ping sequence number	-	uint32	4	53
reserved	Reserved for future use	-	uint32	4	57
frequency	Current actual acoustic frequency	Hz	double	8	61
temperature	External water temperature	°C	double	8	69
pressure	External environment pressure	bar	double	8	77
speedOfSoundUsed	Speed-of-sound used	m/s	double	8	85
pingStartTime	Timestamp of the ping	-	uint32	4	93
dataSize	Data item size	-	uint8	1	97
rangeResolution	Resolution of a single range line	M	double	8	98
rangeCount	Number of range lines in the image	-	uint16	2	106
bearingCount	Number of bearings in the image	-	uint16	2	108
imageOffset	Offset to the image data	-	uint32	4	110
imageSize	Total size of the image	bytes	uint32	4	114
messageSize	Total size of the network payload	bytes	uint32	4	118
bearings	Array of the actual bearings used	-	int16[]	m	122
payload	Image data payload	-	Uint8[]	imageSize	imageOffset
				Total size	122+m+n

#### 3.3.1. Data Size Field

The following are the possible values for the data size field:

Value	Description
0	8-bit image data
1	16-bit image data
2	24-bit image data
3	32-bit image data

## 3.4. Simple Fire V2 Request

The simple fire V2 request is an enhanced version of the original V1 implementation.

Field	Description	Default	Format	Size	Offset
header	Message header	-	struct	16	0
masterMode	Current run mode (section 3.4.1)	-	uint8	1	16
pingRate	Maximum ping rate (section 3.4.2)	-	uint8	1	17
networkSpeed	Maximum network speed	-	uint8	1	18
gamma	Gamma correction value	127	uint8	1	19
flags	Simple fire V1 flags (section 3.4.3)	-	uint8	1	20
range	Demanded range in metres or percent	-	double	8	21
gain	Demanded gain value in percent	-	double	8	29
speedOfSound	Speed-of-sound used	-	double	8	37
salinity	Salinity of the environment	-	double	8	45
extFlags	Extended flags (section 0)	-	uint32	4	53
reserved0[2]	Reserved for future use	-	uint32	4 x 2	57
beaconLocatorFrequency	The frequency of the pinger beacon	-	uint32	4	65
reserved1[5]	Reserved for future use	-	uint32	4 x 5	69
Total size					89

### 3.4.1. Master Mode Field

The possible values for this field are identical to that of the V1 request (see section 3.2.1) with the addition of an extra option:

Value	Name	Description
1	LowFrequency	The sonar's lower frequency
2	HighFrequency	The sonar's higher frequency
3	PingerLocator	Pinger Locator Mode (Available from Firmware v212 onwards)

### 3.4.2. Ping Rate Field

The possible values for this field are identical to that of the V1 request (see section 3.2.2).

### 3.4.3. Flags Field

The flags in a V2 request message are identical to that in a V1 (see section 3.2.3).

### 3.4.4. Speed of Sound Field

The speed of sound field operates in the same way to that in a V1 request (see section 3.2.4)

### 3.4.5. Extended Flags Field

Bit	Name	Description
0	gainReduce6dB	Set = reduce gain by 6dB Clear = do not reduce gain by 6dB
1	gainReduce12dB	Set = reduce gain by 12dB Clear = do not reduce gain by 12dB
2	gainBoost	Set = enable gain boost Clear = disable gain boost
3	chirpEnable	Set = enable chirp Clear = disable chirp

### 3.4.6. Beacon Locator Frequency Field

The beacon locator frequency field has no effect when not using Pinger Locator Mode (master mode field). When this mode is in use, this field should be set to the operating frequency of the pinger beacon in Hz.

## 3.5. Simple Fire V2 Response

The simple fire V2 response is received following a V2 simple fire request.

Field	Description	Units	Format	Size	Offset
fireMessage	Simple fire message struct	-	(section 3.4)	89	0
pingId	Ping sequence number	-	uint32	4	89
reserved	Reserved for future use	-	uint32	4	93
frequency	Current acoustic frequency	Hz	double	8	97
temperature	External water temperature	°C	double	8	105
pressure	External environment pressure	bar	double	8	113
heading	Heading of the sonar	°	double	8	121
Pitch	Pitch of the sonar	°	double	8	129
roll	Roll of the sonar	°	double	8	137
speedOfSound	Speed-of-sound used	m/s	double	8	145
pingStartTime	Timestamp of the ping	-	double	8	153
dataSize	Data item size	-	uint8	1	161
rangeResolution	Resolution of a single range line	M	double	8	162
rangeCount	Number of range lines in the image	-	uint16	2	170
bearingCount	Number of bearings in the image	-	uint16	2	172
reserved[4]	Reserved for future use	-	uint32	4 x 4	174
imageOffset	Offset to the image data	-	uint32	4	190
imageSize	Total size of the image	bytes	uint32	4	194
messageSize	Total size of the network payload	bytes	uint32	4	198
bearings	Array of the actual bearings used	-	int16[]	m	202
payload	Image data payload	-	Uint8[]	n	imageOffset
Total size					202+m+n

### 3.5.1. Data Size Field

The possible values for this field are identical to that of the V1 message (see section 3.3.1).

## 3.6. Network Configuration

A network configuration message allows the user to configure the Oculus imaging sonar to use either a static IP address and subnet mask, or to allow the sonar to obtain an DHCP allocated address.

Field	Description	Default	Format	Size	Offset
header	Message header	-	(section 2.1)	16	0
ipAddr	Static IP address	-	uint32	4	16
ipMask	Static IP mask	-	uint32	4	20
networkMode	The sonar's network mode	-	uint32	4	24
Total size					28

**Note – when DHCP mode is enabled, the ipAddr and ipMask DO NOT reflect the current DHCP assigned addresses. These fields still reflect the static values used when DHCP mode is disabled.**

### 3.6.1. Network Mode Field

The following are the possible values for the network mode field:

Value	Name	Description
0	static	The sonar will use the assigned static IP address
1	DHCP	The sonar will use a DHCP IP address

## 4. Operating (Master) Modes

### 4.1. Imaging Mode (Low or High Frequency)

The standard operation of the Oculus sonar is imaging mode. In this mode, the sonar will perform an acoustic transmit and receive to produce an image.

### 4.2. Beacon Locator Mode

The Oculus sonar also supports another mode called Beacon Locator Mode. This mode is intended for use with simple pinger beacons that repeatedly transmit an acoustic ping at a certain acoustic frequency (commonly a 37kHz ping at a rate of 1Hz). When in this mode, the Oculus will not transmit, but simply receive for a period of time before generating an image to indicate the direction in which the pinger beacon is located. There is some configuration of fields in the Simple Fire V2 message required in order to make use of this mode.

#### 4.2.1. Range

When using beacon locator mode, the range field must be set to a value greater than half of the distance travelled at the current speed of sound in the time between pings from the pinger beacon. This is to ensure that the sonar capture more data than the expected ping interval, therefore guaranteeing that the sonar will see a ping from the beacon. For example, if the pinger beacon is pinging at a rate of 1Hz and the current speed of sound is 1500m/s, the range field must be set to a value greater than half of the distance travelled at 1500m/s in 1 second, so a value greater than 750m.

#### 4.2.2. Beacon Locator Frequency

When using beacon locator mode, the beacon locator frequency field must be set to the operational frequency of the pinger beacon in Hz. For example, if the pinger beacon transmits 37kHz pings, this field should be set to a value of 37000.

#### 4.2.3. Returned Image

When using beacon locator mode, the images returned by the Oculus will be similar but different to those in imaging mode. The images will not contain targets and shadows like they would in imaging mode, instead most of the image will be low level background noise apart from a brighter blob on the image. This blob indicates the direction from which the pinger beacon's ping originated (i.e. the direction to the pinger beacon from the Oculus); note that the position of the blob in the image gives an indication of direction only, and not range. An indication of the range to the pinger can instead be acquired from the intensity level of the blob in the image as this will be proportional to the range.

It is recommended that the gain be set to maximum until the pinger beacon is detected by the Oculus. After the pinger beacon has been detected, it may be necessary to reduce the gain as the Oculus approaches the beacon to avoid the ping saturating the image.