

# **Standard Payload Interface (SPI) Specification**

**Version 1.12a (draft 2/9/15)**

**Bluefin Robotics Corporation  
553 South Street  
Quincy, MA 02169  
(+1) 617.715.7000  
[www.bluefinrobotics.com](http://www.bluefinrobotics.com)**

## Table of Contents

Table of Contents.....	2
1 Limitations .....	4
2 Timekeeping .....	4
3 Data Logging .....	4
4 Session Details.....	5
4.1 Session Initialization and Configuration.....	5
4.2 Message Format .....	6
4.3 Message Initiation and Synchronization.....	7
4.4 Payload Command Acknowledgements .....	7
5 Command Messages from Bluefin Vehicle to Payload .....	8
5.1 \$BFMSC - Payload Mission Command .....	8
5.2 \$BFSHT - Payload Shutdown.....	8
5.3 \$BFBDL - Begin Data Logging.....	8
5.4 \$BFSDL - Stop Data Logging .....	8
5.5 \$BFTOP - Topside Message (Not Implemented) .....	9
5.6 \$BFDVT - Begin/End DVL External Triggering .....	9
5.7 \$BFVER - Vehicle Interface Version .....	9
6 Status Messages from Bluefin Vehicle to Payload .....	10
6.1 \$BFNVG - Navigation Update .....	10
6.2 \$BFNVR - Velocity and Rate Update .....	10
6.3 \$BFTEL - Telemetry Status (Not Implemented) .....	11
6.4 \$BFSVS - Sound Velocity .....	11
6.5 \$BFRCM - Raw Compass Data.....	11
6.6 \$BFRDP - Raw Depth Sensor Data .....	12
6.7 \$BFRVL - Raw Vehicle Speed.....	12
6.8 \$BFRBS - Battery Voltage .....	12
6.9 \$BFMBS - Begin New Behavior .....	13
6.10 \$BFMBE - End Behavior.....	13
6.11 \$BFMIS - Mission Status .....	14
6.12 \$BFERC - Elevator and Rudder Data.....	14
6.13 \$BFDVL - Raw DVL Data .....	15
6.14 \$BFDV2 - Raw DVL Data, Extended .....	15
6.15 \$BFIMU - Raw IMU Data.....	16
6.16 \$BFCTD - Raw CTD Sensor Data.....	16
6.17 \$BFRNV - Relative Navigation Position.....	16
6.18 \$BFPIT - Pitch Servo Positions .....	17

6.19	\$BFCNV - Cartesian Relative Navigation Position.....	17
6.20	\$BFPLN - Mission Plan Element .....	18
6.21	\$BFACK - Message Acknowledgement.....	19
6.22	\$BFTRM - Trim Status.....	20
6.23	\$BFBOY - Buoyancy Status.....	21
6.24	\$BFCTL - Backseat Control .....	22
7	Messages From Payload to Bluefin Vehicle .....	22
7.1	\$BPLOG - Logging Control .....	22
7.2	\$BPSTS - Payload Status Message.....	23
7.3	\$BPTOP - Request to Send Data Topside .....	23
7.4	\$BPDVR - Request to Change DVL Triggering Method.....	24
7.5	\$BPRTK - Request Additional Trackline .....	25
7.6	\$BPRTC - Request Additional Trackcircle .....	26
7.7	\$BPRGP - Request Additional GPS Hits.....	28
7.8	\$BPRCN - Cancel Requested Behavior.....	28
7.9	\$BPRCA - Cancel All Requested Behaviors .....	29
7.10	\$BPRCB - Cancel Current Behavior .....	29
7.11	\$BPRCE – Cancel Current Mission Element .....	29
7.12	\$BPRMB - Modify Current Behavior.....	29
7.13	\$BPEMB - End Behavior Modify .....	30
7.14	\$BPTMR - Topside Message Relay (Not Available on Most Vehicles).....	30
7.15	\$BPCTD - Raw CTD Sensor Data.....	31
7.16	\$BPABT - Abort Mission.....	31
7.17	\$BPKIL - Kill Mission.....	32
7.18	\$BPMSG - Log Message .....	32
7.19	\$BPRMP - Request Mission Plan.....	32
7.20	\$BPSEM - Start Empty Mission (Not Implemented) .....	32
7.21	\$BPNPU - Navigation Position Update.....	33
7.22	\$BPSIL - Silent Mode.....	33
7.23	\$BPTRM - Request Trim Adjustment Behavior .....	34
7.24	\$BPBOY - Request Buoyancy Adjustment Behavior .....	35
7.25	\$BPVER - Payload Interface Version.....	35
8	Messages Specific to Hovering Vehicles: From Payload to Bluefin Vehicle .....	36
8.1	\$BPSUS - Suspend Mission.....	36
8.2	\$BPCON - Continue .....	36
8.3	\$BPRES - Resume Mission .....	36
8.4	\$BPSPD - Hull Relative Speed Limit.....	36
8.5	\$BPSAN - Set Sonar Angle .....	37
8.6	\$BPGHP - Go To Hull Position.....	37
8.7	\$BPGBP - Go to Bottom Position .....	37
8.8	\$BPRNS - Reset Relative Navigation.....	38
8.9	\$BPRBO - Hull Relative Bearing Offset.....	38

8.10 \$BPPTZ - Pan Tilt Zoom Camera Control .....	38
8.11 \$BPLIT - Strobe Light Control.....	39
9 Advanced Messages Specific to Low Level Acoustic Communications.....	39
9.1 \$BFCMA - Communications Medium Access .....	39
9.2 \$BFCPS - Communications Packet Sent .....	40
9.3 \$BFCPR - Communications Packet Received Data .....	41
9.4 \$BPCPD - Communications Packet Data .....	42

## Software Interface

This specification describes the communications interface between the Bluefin Main Vehicle Computer (MVC) and an attached payload.

### 1 Limitations

The following limitations apply:

- This specification applies only to vehicles running the “BehaviorControl” application. Older vehicles running “LayeredControl” do not provide large portions of the described interface.
- Portions of this interface are specific to certain vehicle classes as noted. In general, this specification applies to the following vehicle classes: 9”, 12”, 21”, and HAUV. It does not apply to Bluefin Gliders.
- Some features of this specification are intended for future implementation or are deprecated. They remain in the document for completeness, and are noted appropriately.
- The software interfaces defined by this specification are likely to change over time. Bluefin will attempt to maintain backwards compatibility as the specification evolves, but to avoid uncertainty, clients of this interface are advised to ensure that the version of the specification they are referencing corresponds to the version of the software they are using.

### 2 Timekeeping

The Bluefin vehicle keeps the reference time for all mission navigation log data. The vehicle will be disciplined from GPS hits when locked, and optionally by a high-accuracy crystal oscillator when submerged. Payloads can synchronize their local time to the main vehicle clock in several ways. They are listed below in order of increasing accuracy:

- Each message sent from the Bluefin vehicle will have as the first data field a timestamp indicating when the message was created. The timestamps are guaranteed to include precision at least to the millisecond. However, network delays and operating system latencies will reduce the quality of synchronization based on timestamps. Some

messages will contain an additional timestamp at the end of the message indicating when the data in the message were generated or received by the vehicle.

- The MVC will operate as a Network Time Protocol server (NTP). Payloads using this method can expect time synchronization to within several hundred microseconds.
- As an option, the Bluefin vehicle can provide a pulse-per-second (PPS) signal derived from the local oscillator. This method can achieve synchronization to within microseconds.

### **3 Data Logging**

The payload is expected to store all necessary output data internally. Data analysis can be performed after mission completion using the Ethernet connection. Small amounts (a few bytes) of data or data validity indicators can be sent during the mission through the Bluefin payload interface. Access to the payload directly by acoustic modem or RF modem is generally quite limited, and depending on other aspects of the vehicle's communications architecture may not be possible at all.

The validity/status indicators provided by the payload via the payload interface are utilized in fixed safety rules to help maintain the safety of the vehicle and payload. If the specifics of these rules are of interest, please discuss with a Bluefin representative.

## **4 Session Details**

### **4.1 Session Initialization and Configuration**

All navigation, sensor, and control information between the payload and MVC will be sent using a TCP connection over the onboard Ethernet. The MVC (Main Vehicle Computer) will act as a TCP server listening on port 29500. The payload should connect to this port after booting.

The payload driver is able to support multiple clients/payloads, all of which can connect to and disconnect from the server whenever and as often as they wish without causing errors.

The payload driver can be configured with the expected IP address of an expected client, along with a symbolic name for that client. Associating names with particular payload clients allows Bluefin vehicle applications and topside tools to look for and act on messages from specific named entities.

As a future enhancement, it may be possible to connect a single client to the payload driver via an RS232 serial connection.

Bluefin will perform the necessary configuration of payload names(s) and address(es). The additional information below is provided for completeness.

The `-iprole` command-line option to the payload driver is used to associate specific role names with client payloads. It may be used multiple times.

- Command line parameter: `iprole=IPADDRESS:PORT:ROLE`  
where

- IPADDRESS is the IP of the machine from which the client is connecting, (in x.x.x.x format)
  - PORT is the port the client is using to connect (specifying port number is optional)
  - ROLE is the Soma sub-role to be associated with the connection. Commands to the specific payload are sent to "payload" as "ROLE.command".
- Examples
- iprole="127.0.0.1:12345:foo" says when a connection from a client at 127.0.0.1 using port 12345 comes in, associate it with the Soma role "foo". Status about the client appears as "status.foo" under the "payload" role. Commands are sent to "foo.<command>" c/o the "payload" role.

iprole="127.0.0.1::foo"  
says that when a client at 127.0.0.1 (using any port) connects, associate the sub-role "foo" with it.

iprole="127.0.0.1::foo,127.0.0.1:12345:bar"  
if a client connects using port 12345, it gets the role "bar", otherwise it gets the role "foo"; or, two clients may be connected, one with the role "bar", one with the role "foo".

Multiple definitions for the same role may exist, but Roles may not be duplicated so that **the driver can command specific payloads using Soma.**

iprole="127.0.0.1:12345:foo,192.168.10.10::foo"  
when a client connection fits one of these patterns, give it the role "foo".

The above example only allows a single connection, since only one role is defined. But, if one payload disconnects, a different one can connect and use the now released role.

## 4.2 Message Format

All messages sent between the payload and MVC take the form of NMEA messages. Many messages have common field types which are described in Table 1. In addition, each message sent by the Bluefin vehicle includes a standard NMEA checksum. Messages sent by the payload may or may not include the checksum. Line termination is the NMEA standard carriage return followed by line feed. As per the NMEA spec, comma (',') and asterisk (\*) characters are reserved characters for use only as field and checksum delimiters respectively. All messages defined in this spec are considered to be case-sensitive.

Future versions of this interface are likely to define additional messages, and perhaps to add new fields to existing messages. Payload client software should be written to ignore unexpected messages received from the MVC. Also, it is recommended that client software ignore unexpected extra fields at the end of messages. (That is, if new fields are ever added to the end

of existing messages, they will always be optional, and safe to ignore.)

*Table 1: Field Types*

<i>Field Type</i>	<i>Format</i>	<i>Definition</i>
Latitude	llll.l1	Fixed/variable length standard NMEA latitude. The first two digits are degrees, the remaining digits are variable precision minutes. When providing data, the Bluefin vehicle will send the maximum amount of precision available.
Longitude	yyyyy.yy	Fixed/variable length standard NMEA longitude. The first 3 digits are decimal degrees. The remaining digits are variable precision minutes.
Timestamp	hhmmss.ss	Hours, minutes, and variable-length seconds of the current day in UTC time. The seconds will be transmitted with a minimum precision to the nearest millisecond, but may be more precise.
Integer Numeric	x xx	A variable-length integer numeric field. Optional leading and trailing zeros. If negative, the first non-whitespace character will be the minus sign.
Floating-point Numeric	x.x	A variable-length, variable-precision numeric field. The decimal point may be omitted. Optional leading and trailing zeros. If negative, the first non-whitespace character will be the minus sign.
Variable Text	c--c	Variable length character field. Any non-NMEA reserved character may be included.

### 4.3 Message Initiation and Synchronization

Messages are sent both from the payload to the Bluefin MVC, and from the MVC to the payload. In general, messages can be sent independently and asynchronously in both directions. There is not a rigid command/response model.

The client payload may send any payload message to the Bluefin MVC at any time. Payload messages are described in section 7.

By default, only command messages are sent from the vehicle to the payload. On connect, the payload can control what status messages are sent to it via the BPLOG message (see section 7.1). Command messages are described in section 5 and status messages are described in section 6.

A future revision to this specification may define additional mechanisms to allow predefinition of Bluefin status messages to be sent.

#### 4.4 Payload Command Acknowledgements

Optionally, the payload can request that the Bluefin MVC send acknowledgements for messages sent from the payload to the MVC. If requested, acknowledgements are sent immediately after the acknowledged payload message is received. However, since other Bluefin messages are being sent asynchronously and might be received by the payload at any time, the payload should not assume that the first message it receives after sending a payload message will necessarily be the Bluefin acknowledgement of that payload message.

The acknowledgement lets the payload know (a) that the message was received; (b) whether or not it was interpreted correctly; and, if the message requested an action on the part of Bluefin, (c) whether that action completed successfully or not.

Some payload messages request that Bluefin perform complex actions (such as adding additional mission elements) for which a success/failure determination cannot be made immediately. In these cases, the time necessary before success or failure can be determined is at odds with the goal of providing an immediate acknowledgement. Therefore, it is possible for Bluefin to send two acknowledgements. The first, immediate acknowledgement contains a success/failure indication of “pending”. The second acknowledgement, which will be returned in at most 2 seconds, will contain the actual success/failure indication.

The descriptions of the various payload commands below contain additional information about what “success” or “failure” of these commands mean. Most commands always return a success/failure status immediately. Those commands which might return an initial “pending” acknowledgement are explicitly noted.

See section 6.21 “\$BFACK - Message Acknowledgement” below for more information about message acknowledgements.

### 5 Command Messages from Bluefin Vehicle to Payload

These messages can be sent from the Bluefin MVC to a connected payload at any time.

#### 5.1 \$BFMSC - Payload Mission Command

This command will be sent by the Bluefin vehicle when executing user-defined behaviors in the mission file.

\$BFMSC, hhmmss.ss, m--m*hh	
hhmmss.ss	Timestamp
m--m	Arbitrary textual message. Semantics determined by the payload.

## 5.2 \$BFSHT - Payload Shutdown

This message is sent by the Bluefin vehicle at least 1 minute before the payload will be electrically shut down. The payload can use this time to perform normal cleanup operations.

\$BFSHT, hhmmss.ss*hh	
hhmmss.ss	Timestamp

## 5.3 \$BFBDL - Begin Data Logging

This message will be sent when beginning the lead-in trackline as generated by the mission planner.

\$BFBDL, hhmmss.ss,m--m*hh	
hhmmss.ss	Timestamp
m--m	Textual message associated with the current mission step as entered by the mission planner.

## 5.4 \$BFSDL - Stop Data Logging

This message will be sent after the lead-out trackline has completed.

\$BFSDL, hhmmss.ss,m--m*hh	
hhmmss.ss	Timestamp
m--m	Textual message associated with the current mission step as entered by the mission planner.

## 5.5 \$BFTOP - Topside Message (Not Implemented)

Delivery of a message sent from the topside.

\$BFTOP, hhmmss.ss,m,d--d*hh	
hhmmss.ss	Timestamp
m	Method of transport 0 = Ethernet 1 = 802.11 (future implementation) 2 = RF Modem 3 = Acoustic Modem
d--d	Textual message sent from the topside.

## 5.6 \$BFDVT - Begin/End DVL External Triggering

If the hardware portion of the payload interface is equipped with the optional DVL triggering lines and the payload is equipped to perform DVL triggering, then the payload *must* utilize and adhere to this message and the BPDVR message.

See section 7.4 for a complete description of how this message will be used.

\$BFDVT, hhmmss. ss, a*hh	
hhmmss. ss	Timestamp
a	0 = The payload should cease external triggering 1 = The payload may begin external triggering

## 5.7 \$BFVER - Vehicle Interface Version

This message contains the Bluefin Standard Payload Interface version that the vehicle supports. The vehicle will send this message at the beginning of communications with the payload.

\$BFVER, hhmmss. ss, v-v*hh	
hhmmss. ss	Timestamp
v-v	Version number

# 6 Status Messages from Bluefin Vehicle to Payload

These messages are sent from the Bluefin MVC to a connected payload only if explicitly requested by the payload (see section 7.1).

## 6.1 \$BFNVG - Navigation Update

\$BFNVG, hhmmss. ss, llll. ll, a, yyyy. yy, b, g, c, c, d, d, h, h, r, r, p, p, hhmmss. ss*hh	
hhmmss. ss	Timestamp
llll. ll	Latitude
a	Hemisphere (N/S)
yyyy. yy	Longitude
b	Hemisphere (E/W)

\$BFNVG, hhmmss. ss, llll. ll, a, yyyy. yy, b, g, c. c, d. d, h. h, r. r, p. p, hhmmss. ss*hh	
g	Quality of position estimate, if no GPS is available this will be 0. When the position solution includes GPS, the value will be 1.
c. c	Altitude (m)
d. d	Depth (m)
h. h	Heading (deg True North in current UTM coordinate space, i.e. has the UTM convergence factor applied)
r. r	Roll (deg)
p. p	Pitch (deg)
hhmmss. ss	Timestamp (of computed navigation solution)

**Example:**

\$BFNVG, 081025. 987, 4221. 81092, N, 07106. 84603, W, 1, 45. 3, 20. 1, 203. 1, -3. 4, 4. 5, 081025. 980\*5b

## 6.2 \$BFNVR - Velocity and Rate Update

\$BFNVR, hhmmss. ss, e. e, n. n, v. v, s. s, t. t, g. g*hh	
hhmmss. ss	Timestamp
e. e	East velocity (m/s)
n. n	North velocity (m/s)
v. v	Down velocity (m/s)
s. s	Pitch rate (deg/s)
t. t	Roll rate (deg/s)
q. q	Yaw rate (deg/s)

## 6.3 \$BFTEL - Telemetry Status (Not Implemented)

\$BFTEL, hhmmss. ss, a. a, b. b, c. c, d. d, r, e*hh	
hhmmss. ss	Timestamp

\$BFTEL, hhmmss. ss, a, a, b, b, c, c, d, d, r, e*hh	
a. a	Estimated number of kilobytes of remaining acoustic modem uplink capacity available in the upcoming minute.
b. b	Total number of kilobytes successfully uploaded over last minute.
c. c	Estimated kB of remaining acoustic modem download capacity in upcoming minute.
d. d	Total kB successfully downloaded over last minute over acoustic modem.
r	RF modem status. When the RF modem is connected, this field will be 1, otherwise it will be 0.
e	Ethernet status. 1 indicates available, 0 otherwise.

## 6.4 \$BFSVS - Sound Velocity

If the Bluefin vehicle is equipped with a sound velocity sensor, this message will provide the raw output of that sensor. If not, then an estimated value will be provided.

\$BFSVS, hhmmss. ss, x. x*hh	
hhmmss. ss	Timestamp
x. x	Variable precision sound velocity (m/s)

## 6.5 \$BFRCM - Raw Compass Data

If the Bluefin vehicle is equipped with at least one compass, this message will provide the raw output of that sensor. In case of multiple compasses in the vehicle, a separate message will be sent for each compass.

\$BFRCM, hhmmss. ss, n, h, h, p, p, r, r, hhmmss. ss*hh	
hhmmss. ss	Timestamp
n	Integer compass number. Compasses are numbered consecutively starting at 1.
h. h	Variable precision heading (degrees)
p. p	Variable precision pitch (degrees)

\$BFRDM, hhmmss. ss, n, h, h, p, p, r, r, hhmmss. ss*hh	
r. r	Variable precision roll (degrees)
hhmmss. ss	Timestamp (when compass data was received)

## 6.6 \$BFRDP - Raw Depth Sensor Data

\$BFRDP, hhmmss. ss, p. p*hh	
hhmmss. ss	Timestamp
p. p	Variable precision pressure (kPa)

## 6.7 \$BFRVL - Raw Vehicle Speed

This message is available only on vehicles with a tailcone.

\$BFRVL, hhmmss. ss, r. r, v. v*hh	
hhmmss. ss	Timestamp
r. r	Variable precision thruster rotation (RPM)
v. v	Vehicle velocity through water as estimated from thruster RPM, may be empty if no lookup table is implemented (m/s)

## 6.8 \$BFRBS - Battery Voltage

When one or more batteries are installed in the Bluefin vehicle, this message will provide voltage information from the batteries. A separate message will be sent for each battery. A special pseudo battery 0 will be sent for systems with multiple batteries. It reports summary information averaged or summed across all batteries.

\$BFRBS, hhmmss. ss, n, v. v, a. a, b. b, t. t, c. c, p. p*hh	
hhmmss. ss	Timestamp
n	Integer battery number. Batteries are numbered consecutively starting at 1. Special battery 0 contains overall battery system information.
v. v	Variable precision total battery stack voltage (V)
a. a	Variable precision minimum cell voltage (V)
b. b	Variable precision maximum cell voltage (V)

\$BFRBS, hhmmss.ss, n, v, v, a, a, b, b, t, t, c, c, p, p*hh	
t, t	Variable precision hottest cell temperature in degrees Celsius.
c, c	Variable precision battery capacity (kWh)
p, p	Percent State of Charge (%)

## 6.9 \$BFMBS - Begin New Behavior

This message is sent when the Bluefin vehicle is just beginning a new behavior in the current mission. It can be used by payloads for record-keeping or to synchronize actions with the current mission. Use of the (d--d) dive file field is considered deprecated in favor of getting the same information from BFMIS. See also the BFPLN message below.

\$BFMBS, hhmmss.ss, d--d, n, iiii, t--t*hh	
hhmmss.ss	Timestamp
d--d	Current dive file. If current behavior has been requested by the payload, this field will be empty.
n	Behavior number in dive file. If the current behavior has been requested by the payload, this field will be empty. Use of this field is deprecated.
iiii	Payload-requested behavior identifier. If this field is non-empty, it signifies that the Bluefin vehicle is just beginning the payload-requested behavior with the given number.
t--t	Textual string giving behavior type. Currently defined behavior types include, but are not limited to “trackline”, “waypoint”, and “setpoint”.

## 6.10 \$BFMBE - End Behavior

Use of the (d--d) dive file field is considered deprecated in favor of getting the same information from BFMIS.

\$BFMBE, hhmmss.ss, d--d, n, iiii, t--t*hh	
hhmmss.ss	Timestamp
d--d	Current dive file. If current behavior has been requested by the payload, this field will be empty.

\$BFMBE, hhmmss.ss, d--d, n, iiii, t--t*hh	
n	Behavior number in dive file. If the current behavior has been requested by the payload, this field will be empty. Use of this field is deprecated.
iiii	Payload-requested behavior identifier. If this field is non-empty, it signifies that the Bluefin vehicle is ending the payload-requested behavior with the given number.
t--t	Behavior type as in BFMBBS message.

## 6.11 \$BFMIS - Mission Status

This message is sent when the mission status changes, such as when a mission starts or stops.

\$BFMIS, hhmmss.ss, d--d, t-t, a--a*hh	
hhmmss.ss	Timestamp
d--d	Current dive file. If current behavior has not been requested by the mission plan, this field will be empty.
t--t	Textual string for mission status. Currently defined mission status types include, but are not limited to “Offline”, “Waiting”, “Invalid Dive File”, “Invalid Wrapup File”, “Ready”, “Running”, “WrappingUp”, “Aborting”, “Completed”, “Aborted”, and “Stopped”. The status string starting with “Running” may be followed by the logging directory for that mission.
a--a	Additional mission status details such as error information for an abort. The text will be human readable for debug and logging purposes.

## 6.12 \$BFERC - Elevator and Rudder Data

This message provides the current tailcone elevator and rudder angles as well the target angles to which the tailcone has been commanded. It is only provided on vehicles with tailcones.

\$BFERC, hhmmss.ss, e.e, r.r, f.f, s.s*hh	
hhmmss.ss	Timestamp
e.e	Variable precision current elevator angle (degrees). Pitch up is positive.

\$BFERC, hhmmss.ss, e.e, r.r, f.f, s.s*hh	
r.r	Variable precision current rudder angle (degrees). Pitch starboard is positive.
f.f	Variable precision target elevator angle (degrees). Pitch up is positive.
s.s	Variable precision target rudder angle (degrees). Pitch starboard is positive.

### 6.13 \$BFDVL - Raw DVL Data

\$BFDVL, hhmmss.ss, x.x, y.y, z.z, r1, r2, r3, r4, t.t, hhmmss.ss*hh	
hhmmss.ss	Timestamp (when message is sent)
x.x	X velocity (m/s) positive is forward
y.y	Y velocity (m/s) positive is to starboard
z.z	Z velocity (m/s) positive is down
r1-r4	Beam ranges (m/s)
t.t	Temperature (C)
hhmmss.ss	Timestamp (when DVL data was received)

### 6.14 \$BFDV2 - Raw DVL Data, Extended

This message is useful for vehicles which use the DVL in a non-standard mode that produces beam velocities, not vehicle velocities. Currently this is the HAUV only. For other vehicles the values reported here as beam velocities will actually be X,Y, and Z velocities and the fourth beam velocity will be the DVL reported error velocity term. Non-HAUV customers are encouraged to use the previous message instead.

\$BFDV2, hhmmss.ss, v1.x, v2.x, v3.x, v4.x, r1, r2, r3, r4, t.t, hhmmss.ss*hh	
hhmmss.ss	Timestamp (when message is sent)
v1.x	Beam 1 velocity (m/s)
v2.x	Beam 2 velocity (m/s)
v3.x	Beam 3 velocity (m/s)

\$BFDV2, hhmmss. ss, v1. x, v2. x, v3. x, v4. x, r1, r2, r3, r4, t. t, hhmmss. ss*hh	
v4. x	Beam 4 velocity (m/s)
r1-r4	Beam ranges (m/s)
t. t	Temperature (C)
hhmmss. ss	Timestamp (when DVL data was received)

## 6.15 \$BFIMU - Raw IMU Data

\$BFIMU, hhmmss. ss, ax. ax, ay. ay, az. az, lx. lx, ly. ly, lz. lz, hhmmss. ss*hh	
hhmmss. ss	Timestamp (when message is sent)
ax. ax	Angular rate about X axis (deg/s) right-hand rule
ay. ay	Angular rate about Y axis (deg/s) right-hand rule
az. az	Angular rate about Z axis (deg/s) right-hand rule
lx. lx	Acceleration in X direction (m/s <sup>2</sup> ) positive is forward
ly. ly	Acceleration in Y direction (m/s <sup>2</sup> ) positive is starboard
lz. lz	Acceleration in Z direction (m/s <sup>2</sup> ) positive is down
hhmmss. ss	Timestamp (when IMU data was received)

## 6.16 \$BFCTD - Raw CTD Sensor Data

If the Bluefin vehicle is equipped with a CTD sensor, this message will provide the raw output of that sensor. If not, then no output will be provided.

\$BFCTD, hhmmss. ss, c. c, t. t, p. p, hhmmss. ss*hh	
hhmmss. ss	Timestamp (when message is sent)
c. c	Variable precision conductivity (uSiemens/cm)
t. t	Variable precision temperature (degrees Celsius)
p. p	Variable precision pressure (kPa)
hhmmss. ss	Timestamp (when CTD data was received)

## 6.17 \$BFRNV - Relative Navigation Position

Vehicles that allow relative navigation (currently the HAUV) can emit this message to indicate position relative to an initial position.

\$BFRNV, hhmmss. ss, a. aa, b. bb, d. dd, h. h, d. dd, h. h, r. rr, p. pp, tttttt. ttt*hh	
hhmmss. ss	Timestamp (when message is sent)
a. aa	Horizontal position, in meters
b. bb	Vertical position in meters
d. dd	Distance from target, in meters
h. h	Heading, in degrees (zero means directly towards target)
d. dd	Depth, in meters
h. h	Absolute heading, in degrees (not relative to target, but still not compass heading)
r. rr	Roll, in degrees
p. pp	Pitch, in degrees
tttttt. ttt	Timestamp of the navigation data itself

## 6.18 \$BFPIT - Pitch Servo Positions

Vehicles that have movable DVLs and sonars (currently the HAUV) can emit this message to indicate the position of their DVL and sonar.

\$BFPIT, hhmmss. ss, d. d, s. s*hh	
hhmmss. ss	Timestamp (when message is sent)
d. d	DVL position, in degrees
s. s	Sonar position, in degrees

## 6.19 \$BFCNV - Cartesian Relative Navigation Position

Vehicles that allow relative navigation (currently the HAUV) can emit this message to indicate the vehicle's three-dimensional position in an inertial frame of reference.

\$BFCNV, hhmmss. sss, x. xx, y. yy, z. zz, h. hh*hh
---

\$BFCNV, hhmmss. sss, x. xx, y. yy, z. zz, h. hh*hh	
hhmmss. sss	Timestamp (when message is sent)
x. xx	X coordinate, in meters
y. yy	Y coordinate, in meters
z. zz	Z coordinate, in meters
h. hh	Heading, in degrees (zero means directly towards target)

## 6.20 \$BFPLN - Mission Plan Element

This message communicates an element of the vehicle's current mission plan to the payload.

Depending on the behavior, it may be necessary for the vehicle to transmit multiple BFPLN messages to describe a single behavior. In particular, tracklines are transmitted as a start point and an end point, and a trackcircle is sent as a series of waypoints approximating the planned circle. When multiple BFPLN messages describe a single behavior, the *m* field increments by 1 for each message. But, ignoring the *m* and *e* fields, it is always the case that the sequence of waypoints contained in a series of BFPLN messages will accurately describe a mission's planned course.

Depending on the payload's requirements, it might be desirable to receive mission plan information using one of two different schemes.

It might be desirable to have the planned details of a behavior transmitted in real time, as that behavior begins executing. (See also BFMBS in section 6.9 above.) This scheme would be requested by using BPLOG (section 7.1) to request BFMBS messages. This scheme is not implemented in this release of this specification.

Alternatively, it might be desirable to receive the details of all behaviors making up a complete mission, all at once. This functionality is requested using the BPRMP message (see section **Error! Reference source not found.**). This scheme is implemented in this release of this specification.

\$BFPLN, hhmmss. ss, n, iiii, m, e, llll. ll, a, yyyyy. yy, b, x. x, c, d*hh	
hhmmss. ss	Timestamp (of this message, <i>not</i> of behavior)
n	Behavior number in dive file. (Same as BFMBS.)
iiii	Payload-requested behavior identifier. (Same as BFMBS.)

\$BFPLN, hhmmss.ss, n, iiii, m, e, llll.ll, a, yyyyy.yy, b, x, x, c, d*hh	
m	Sequential index, if multiple BFPLN messages are required to describe a single plan element
e	Code indicating element type: 0=waypoint 1=trackline start 2=trackline end 3=trackcircle
llll.ll	Latitude
a	Hemisphere (N/S)
yyyyy.yy	Longitude
b	Hemisphere (E/W)
x, x	Future expansion (not specified)
c	Future expansion (not specified)
d	Future expansion (not specified)

## 6.21 \$BFAK - Message Acknowledgement

Optionally, the Bluefin vehicle will respond to each payload command with an acknowledgement.

Broadly, there are four different acknowledgements. A command can be rejected as invalid (an unrecognized command name, or one with an incorrect number or types of parameters). A command can be recognized, processing of the command can be initiated, but the processing can fail. A command can be recognized and successfully processed. Finally, it may not always be immediately possible for the Bluefin vehicle to determine whether a command has succeeded or failed, in which case a pending acknowledgement may be issued.

When a pending acknowledgement is returned, the payload knows that its request was at least syntactically valid, and that the vehicle is working on processing the request. At some later time (generally within 2 seconds), the vehicle will transmit an actual success or failure acknowledgement corresponding to the same payload request. Within this specification, those payload commands which have the possibility to return these “pending” acknowledgements are explicitly noted. Otherwise, commands can be assumed to always return requested acknowledgements immediately.

The acknowledgement also contains a message string which may contain additional details about the failure, or certain extra status information in case of success. The format of these message

strings is not specified. Except by prearrangement with Bluefin Robotics, payload software should not attempt to interpret the contents of the message string. Message strings in acknowledgements are intended for debugging and human consumption, not machine parsing.

By default, these acknowledgements are *not* sent. The payload can request that acknowledgements be sent by using the BPLOG command (just as it requests other status messages; see section 7.1 7.1 ).

\$BFACK, hhmmss.ss, c-c, hhmmss.ss, iiii, s, c, m-m, *hh									
hhmmss.ss	Timestamp								
c-c	Name of payload command being acknowledged								
hhmmss.ss iiii	Timestamp of payload command being acknowledged  If the command being acknowledged is a new behavior insertion (e.g. BPRTK), the payload-assigned identifier for that behavior								
s	<i>Acknowledgement Status Code</i>								
	<table border="1"> <tr> <td>0</td> <td>Invalid Request (unknown message, or improper parameters)</td> </tr> <tr> <td>1</td> <td>Request unsuccessfully processed</td> </tr> <tr> <td>2</td> <td>Request successfully processed</td> </tr> <tr> <td>3</td> <td>Pending: request received and processing initiated</td> </tr> </table>	0	Invalid Request (unknown message, or improper parameters)	1	Request unsuccessfully processed	2	Request successfully processed	3	Pending: request received and processing initiated
0	Invalid Request (unknown message, or improper parameters)								
1	Request unsuccessfully processed								
2	Request successfully processed								
3	Pending: request received and processing initiated								
c	Future use; always 0 in this revision of this specification								
m-m	String describing reason for failure (or acknowledgement of success) in possibly more detail								

## 6.22 \$BFTRM - Trim Status

This status message contains vehicle trim estimator information and can be sent any time. It also notifies the payload when the Trim Adjustment behavior (\$BPTRM section **Error! Reference source not found.**) is being executed and the status of the operation upon completion.

By default, these status messages are *not* sent. The payload can request that they be sent by using the BPLOG command (see section 7.1 ).

\$BFTRM, hhmmss.ss, s, c, m-m, p. p, r. r*hh		
hhmmss.ss	Timestamp	
s	<i>Status Code</i>	
	0	Idle
	1	In Progress
	2	Completed Successfully
	3	Failed
c	<i>Error Code</i>	
	0	No error
	1	Trim operation failed
	2	Trim operation canceled
	3	Trim at limit / zero flow
4	Idle Error	
m-m	String describing status (including errors, for logging purposes)	
p. p	Pitch trim estimator, in degrees	
r. r	Roll trim estimator, in degrees	

## 6.23 \$BFBOY - Buoyancy Status

This status message contains vehicle buoyancy estimator information and can be sent any time. It notifies the payload when the Buoyancy Adjustment behavior (\$BPBOY section **Error! Reference source not found.**) is being executed, status of the operation upon completion, and when no adjustment is in progress.

By default, these status messages are *not* sent. The payload can request that they be sent by using the BPLOG command (see section 7.1).

\$BFBOY, hhmmss.ss, s, c, m-m, b. b\*hh

\$BFBOY, hhmmss.ss, s, c, m-m, b. b*hh		
hhmmss.ss	Timestamp	
s	<i>Status Code</i>	
	0	Idle
	1	In Progress
	2	Completed Successfully
	3	Failed
c	<i>Error Code</i>	
	0	No error
	1	Buoyancy operation failed
	2	Buoyancy operation canceled
	3	Buoyancy at limit / zero flow
4	Idle Error	
m-m	String describing status (including errors, for logging purposes)	
b. b	Buoyancy Estimator, in Newtons (1 Newton = 0.22481 pound-force)	

## 6.24 \$BFCTL - Backseat Control

This message explicitly hands over control from Bluefin (frontseat) to the payload (backseat). A value of 1 (true) will be sent to the backseat indicating that it now has control over the vehicle. This will be sent at a time in the mission when the internal value “allowDynamicRequests” transitions to true. It must be programmed as such in the mission plan. The value of 0 (false) will be sent to the backseat when the internal value “allowDynamicRequests” transitions to false, or when the mission state transitions out of “Running”. Alternatively, the vehicle can be configured to hand control to the backseat when the mission starts and returned to the frontseat at mission stop.

\$BFVER, hhmmss.ss, c\*hh

\$BFVER, hhmmss.ss, c*hh	
hhmmss.ss	Timestamp
c	Control (boolean true = 1, false = 0)

## 7 Messages From Payload to Bluefin Vehicle

### 7.1 \$BPLOG - Logging Control

When a payload initially connects to the vehicle, all status messages are turned off. The BPLOG command is used to turn ON or OFF a specific message type. The special keyword ALL can also be used to turn on or off all messages (with the exception of BFACTION, which must always be requested explicitly).

If this message is acknowledged (see section 6.21), a 'success' acknowledgement indicates that the requested logging action (ON or OFF) was performed.

\$BPLOG, rrr, a-a*hh	
rrr	Requested 3-letter message identifier (e.g. NVG to turn on/off logging of navigation data to this payload), or the keyword ALL to turn on/off logging of all data to this payload
a-a	ON or OFF

### 7.2 \$BPSTS - Payload Status Message

\$BPSTS, hhmmss.ss, f, m--m*hh	
hhmmss.ss	Timestamp
f	Status flag. 0 = Failed, 1 = All OK.
m--m	Variable length textual status message.

### 7.3 \$BPTOP - Request to Send Data Topside

This command allows the payload to send an arbitrary message to the topside. The message will be transmitted using the best communications link currently available. The message will be logged, both on the vehicle and the topside, and will be displayed to the operator in an appropriate manner.

The payload may wish to advise the system that a message is not appropriate to be transferred over slower communications links. For example, it might be the case that a certain high-volume or low-priority message should be sent over a high-speed Ethernet or RF link, if available, but

should not be sent over a low-speed acoustic link. The “Slowest method of transport to use” field allows the payload to make this specification. This field is advisory-only, and may be ignored. (Currently, this field is unimplemented, such that all topside messages will be sent using any communications link.)

The actual handling of the data once it reaches the topside is not constrained by this specification. There are several possibilities:

- display immediately in a field on the main operator display screen
- display on a subsidiary screen which the operator can open to review payload messages
- display in a pop-up alert dialog
- log to a file
- retransmit via TCP or UDP over topside network

The actual specification of the topside display action will be made in the context of the particular topside operator control software being used (typically Bluefin’s Dashboard product).

If this message is acknowledged (see section 6.21 ), a ‘success’ acknowledgement indicates that the requested message was successfully queued for transmission by an available communications link. It does *not* indicate that the message was actually received on the topside.

Since processing of this command requires potentially complex interactions with the vehicle’s communications subsystem, it is possible that an immediate, initial “pending” acknowledgement will be returned prior to the actual success/failure acknowledgement.

\$BPTOP, hhmmss.ss, f, m--m, d*hh	
hhmmss.ss	Timestamp
f	Recommended slowest method of transport to use (advisory). 0 = Ethernet 1 = 802.11 (future implementation) 2 = RF modem 3 = Acoustic Modem
m--m	Variable length textual message. The Bluefin vehicle will make efforts to compress this data if being transmitted over the acoustic modem. At a minimum, if the string contains nothing but valid uppercase hexadecimal characters, it will be sent as the equivalent binary data. Other compression techniques may be employed. Note that the WHOI acoustic modem sends only a few 26-byte packets per minute.
d	Reserved for future specification. Must be empty or “0”.

## 7.4 \$BPDVR - Request to Change DVL Triggering Method

Using this command the payload can request to place the on-board DVL into external trigger mode or indicate that it has stopped or started asserting the external trigger line. This message *must* be sent any time either of its fields would change value.

If the payload wants to start external triggering, it should first send this message with the request field set to 1 and the currently triggering field set to 0. When it is safe to begin external triggering, the Bluefin vehicle will send a BFDVT message to the payload telling it to commence triggering. Once the payload has stopped triggering, it may send either a request=0, triggering=1 message, or a request=0, triggering=0 message. The former will elicit a stop triggering command from the Bluefin vehicle after which the payload should respond with a request=0, triggering=0 message.

At any time when in external triggering mode the Bluefin vehicle may initiate a stop triggering command. The payload is expected to stop triggering and respond with a BPDVR command indicating so. The payload may indicate that it still wishes to be in external trigger mode with this response. In this case, the vehicle will re-enter external triggering mode as soon as is possible, sending the appropriate BFDVT command.

If the payload does not trigger the DVL for a set amount of time while in external trigger mode (5 seconds by default), the DVL will revert to internal trigger mode. [(Not Implemented) The vehicle will send a BFDVT message and immediately switch to internal mode without waiting for a response (the payload should still send a response if possible).] The timeout can be configured by Bluefin if desired.

\$BPDVR, hhmmss.ss, a, b*hh	
hhmmss.ss	Timestamp
a	0 = Request to end external triggering 1 = Request to begin external triggering
b	0 = Payload is not currently triggering the DVL 1 = Payload is currently triggering the DVL

## 7.5 \$BPRTK - Request Additional Trackline

Request that the Bluefin vehicle perform a trackline maneuver. The maneuver is a straight survey line between the specified start and end points with the payloads activated and any default leadin/leadout behavior. By default, the new trackline will be inserted at the end of the current planned (not dynamic) survey leg or trackcircle or before the beginning of the next one.

Optionally, the newly-requested behavior can interrupt the currently-running one. If so, then after the dynamically-inserted behavior completes, the vehicle will proceed to the next behavior following to the interrupted behavior. (That is, the vehicle will not attempt to resume the interrupted behavior.)

Future versions of this specification may allow finer-grained control over the behavior at the end of a dynamically-inserted, interrupting behavior.

If several calls to BPRTK (or BPRTC) are made in a row, subsequent dynamically-inserted behaviors will be appended at the end of the set. If the first dynamically-inserted behavior in the set was specified as interrupting, the remaining ones will be completed as well, before finally returning to the interrupted behavior.

If a dynamically-inserted behavior has interrupted the previous behavior and is still running, it is an error to attempt to insert another dynamic behavior that interrupts. (That is, it is not possible to interrupt an interruption.)

The ability of the payload to modify the running mission by inserting new behaviors is a complex feature that may negatively impact mission reliability. The vehicle’s willingness to accept dynamic mission plan requests is configurable. Dynamic requests are enabled by setting the onboard “allowDynamicRequests” parameter to true.

If this message is acknowledged (see section 6.21 ), a ‘success’ acknowledgement indicates that the new behavior was successfully added to the currently-executing mission plan. It does *not* indicate that the requested behavior has begun executing, or has completed successfully.

Since processing of this command requires potentially complex interactions with the vehicle’s behavior control subsystem, it is possible that an immediate, initial “pending” acknowledgement will be returned prior to the actual success/failure acknowledgement.

\$BPRTK, hhmmss.ss, iii, llll.ll, a, yyyy.yy, b, mmmm.mm, c, wwwww.ww, d, o, v, v, s, s, e, f*hh	
hhmmss.ss	Timestamp
iii	Numeric identifier to be used in future communication regarding this request
llll.ll	Start latitude
a	Start latitude hemisphere (N/S)
yyyy.yy	Start longitude
b	Start longitude hemisphere (E/W)
mmmm.mm	End latitude
c	End latitude hemisphere (N/S)
wwww.ww	End longitude
d	End longitude hemisphere (E/W)

\$BPRTK, hhmmss.ss, iiiii, llll.ll, a, yyyyy.yy, b, mmmm.mm, c, wwwwww.ww, d, o, v. v, s. s, e, f*hh	
o	Vertical mode 0 = depth 1 = altitude
v. v	Depth or altitude (m)
s. s	Desired speed in m/s or rpm
e	Speed mode 0 = m/s 1 = rpm
f	Interrupt code. 0=new behavior runs after current (default) 4=new behavior interrupts current (Other interrupt values may be specified in future versions of this document.)

## 7.6 \$BPRTC - Request Additional Trackcircle

This message requests that the vehicle perform an additional trackcircle behavior. It is very similar to the “Request Additional Trackline” message described in section 7.5 just above.

If this message is acknowledged (see section 6.21 ), a ‘success’ acknowledgement indicates that the new behavior was successfully added to the currently-executing mission plan. It does *not* indicate that the requested behavior has begun executing, or has completed successfully.

Since processing of this command requires potentially complex interactions with the vehicle’s behavior control subsystem, it is possible that an immediate, initial “pending” acknowledgement will be returned prior to the actual success/failure acknowledgement.

\$BPRTC, hhmmss.ss, iiiii, llll.ll, a, yyyyy.yy, b, r. r, n. n, d, v. v, s. s, e, f*hh	
hhmmss.ss	Timestamp
iiii	Numeric identifier to be used in future communication regarding this request
llll.ll	Center latitude
a	Center latitude hemisphere (N/S)

\$BPRTC, hhmmss.ss, iii, llll.ll, a, yyyyy.yy, b, r.r, n.n, d, v.v, s.s, e, f*hh	
yyyyy.yy	Center longitude
b	Center longitude hemisphere (E/W)
r.r	Radius of circle (m)
n.n	Number of orbits. Positive indicates clockwise traversal, negative indicates counterclockwise.
d	Vertical mode 0 = depth 1 = altitude
v.v	Depth or altitude (m)
s.s	Desired speed in m/s or rpm
e	Speed mode 0 = m/s 1 = rpm
j.j	Entry Angle (degrees). For example, 0.0 would be the northmost point on the circle, and 180.0 would be the southmost
f	Interrupt code. 0=new behavior runs after current (default) 4=new behavior interrupts current (Other interrupt values may be specified in future versions of this document.)

## 7.7 \$BPRGP - Request Additional GPS Hits

\$BPRGP, hhmmss.ss, iii, llll.ll, a, yyyyy.yy, b, r.r, nnnn, s*hh	
hhmmss.ss	Timestamp
iii	Numeric identifier to be used in future communication regarding this request
llll.ll	Desired surface latitude

\$BPRGP, hhmmss.ss, iii, llll.ll, a, yyyy.yy, b, r.r, nnn, s*hh	
a	Latitude hemisphere (N/S)
yyyy.yy	Desired surface longitude
b	Longitude hemisphere (E/W)
t.t	Ignored. Future: Maximum time to surface (s); currently uses vehicle default.
nnn	Ignored. Future: Number of GPS hits to acquire; currently uses vehicle default.
s	Surface mode 0 = straight line 1 = spiral (future implementation) 2 = manual (no automatic surfacing)

## 7.8 \$BPRCN - Cancel Requested Behavior

This request cancels (deletes) a dynamically-inserted behavior inserted by a preceding BPRTK or BPRTC request.

If the behavior has already begun executing, it is not possible to cancel it in this way.

If this message is acknowledged (see section 6.21 ), a ‘success’ acknowledgement indicates that the requested behavior was successfully removed from the dive plan.

Since processing of this command requires potentially complex interactions with the vehicle’s behavior control subsystem, it is possible that an immediate, initial “pending” acknowledgement will be returned prior to the actual success/failure acknowledgement.

\$BPRCN, hhmmss.ss, iii*hh	
hhmmss.ss	Timestamp
iii	Numeric identifier of behavior to cancel.

## 7.9 \$BPRCA - Cancel All Requested Behaviors

\$BPRCA, hhmmss.ss*hh	
hhmmss.ss	Timestamp

## 7.10 \$BPRCB - Cancel Current Behavior

This message causes premature termination of the currently-executing behavior. Control passes immediately to the next behavior in the mission.

The ability of the payload to modify the running mission by canceling running behaviors is a complex feature that may negatively impact mission reliability. The vehicle’s willingness to accept dynamic mission plan requests is configurable. Dynamic requests are enabled by setting the onboard “allowDynamicRequests” parameter to true.

\$BPRCB, hhmmss.ss, n*hh	
hhmmss.ss	Timestamp
n	Reserved for future specification. Must be empty or “0”.

## 7.11 \$BPRCE – Cancel Current Mission Element

This message causes premature termination of the currently-executing Planner-level mission element. Control passes immediately to the next element in the mission.

The ability of the payload to modify the running mission by canceling running elements is a complex feature that may negatively impact mission reliability. The vehicle’s willingness to accept dynamic mission plan requests is configurable. Dynamic requests are enabled by setting the onboard “allowDynamicRequests” parameter to true.

\$BPRCE, hhmmss.ss, n*hh	
hhmmss.ss	Timestamp
n	Reserved for future specification. Must be empty or “0”.

## 7.12 \$BPRMB - Modify Current Behavior

This message may be used to provide the Bluefin vehicle with a temporary alternate heading, depth, and/or speed to traverse along. The current behavior will not be terminated immediately, but may be at some future point if the total time duration or deviation from original behavior is too large. Not all of the fields are required; any empty fields do not modify that aspect of the current behavior.

If the payload wishes to continue modifying the current behavior this message must be sent at least every 5 seconds or the Bluefin vehicle will resume control of the mission.

Note that this message can not be used to command the vehicle beyond its safe operating parameters.

The ability of the payload to modify the vehicle’s behavior is a complex feature that may

negatively impact mission reliability. The vehicle’s willingness to accept dynamic mission plan requests is configurable. Dynamic requests are enabled by setting the onboard “allowDynamicRequests” parameter to true. If a \$BPRMB request is received during a mission or a phase of a mission where “allowDynamicRequests” is not enabled, the request will be ignored, or acknowledged with a failure response if acknowledgements are enabled.

\$BPRMB, hhmmss.ss, h, h, d, d, dm, v, v, vm, hm*hh	
hhmmss.ss	Timestamp
h, h	Variable precision heading in degrees. Empty means use the current behavior’s value.
d, d	Variable precision depth or altitude in meters. May be empty.
dm	Depth mode. 0 signifies the previous field is a depth; 1 signifies the previous field is an altitude; 2 signifies the previous field is a pitch; 3 signifies the previous field is an elevator adjustment. May be empty.
v, v	RPM to command the thruster or speed in m/s to command the vehicle. May be empty.
vm	Speed mode. 0 signifies the previous field is a thruster RPM; 1 signifies the previous field is a speed in meters per second. May be empty.
hm	Horizontal mode. 0 signifies the first field is a heading; 1 signifies a rudder adjustment.

### 7.13 \$BPEMB - End Behavior Modify

The payload may send this message to signify that it no longer intends to modify the current behavior.

\$BPEMB, hhmmss.ss*hh	
hhmmss.ss	Timestamp

### 7.14 \$BPTMR - Topside Message Relay (Not Available on Most Vehicles)

In some vehicles, the acoustic modem is part of the payload. Certain vehicle control messages are sent acoustically. These messages need to be relayed from the payload to the vehicle. This is the interface used to relay the messages. Transport method has been retained to mirror the BFTOP message.

\$BPTMR, hhmmss.ss,m,d--d*hh	
hhmmss.ss	Timestamp
M	Method of transport 0 = Ethernet 1 = 802.11 (future implementation) 2 = RF Modem 3 = Acoustic Modem
d--d	Textual message sent from the topside.

## 7.15 \$BPCTD - Raw CTD Sensor Data

Some vehicles have a CTD sensor in the payload. Navigation can benefit from using this CTD data. This message is provided so that the payload can send CTD data to the vehicle to aid in navigation.

\$BPCTD, hhmmss.ss,c.c,t.t,p.p, hhmmss.ss*hh	
hhmmss.ss	Timestamp (when message is sent)
c.c	Variable precision conductivity (uSiemens/cm)
t.t	Variable precision temperature in degrees Celsius
p.p	Variable precision pressure (kPa)
hhmmss.ss	Timestamp of measurement

## 7.16 \$BPABT - Abort Mission

This command causes the current mission to end, and its wrapup behavior executed (such as a powered ascent to the surface).

\$BPABT, hhmmss.ss,c--c,a*hh	
hhmmss.ss	Timestamp (when message is sent)
c--c	Message. This may be displayed to the user to explain the abort.

\$BPABT, hhmmss. ss, c--c, a*hh	
a	Abort reason (0 = successful mission, 1 = abort with errors). Abort with errors (1) is the default.

## 7.17 \$BPKIL - Kill Mission

This command causes the current mission to end, and all vehicle movement to stop. The vehicle will drift.

\$BPKIL, hhmmss. ss, c--c*hh	
hhmmss. ss	Timestamp (when message is sent)
c--c	Message. This may be displayed to the user to explain the abort.

## 7.18 \$BPMSG - Log Message

This command causes the vehicle to log a text message in its logs. This can be useful for diagnostics.

\$BPMSG, hhmmss. ss, c--c*hh	
hhmmss. ss	Timestamp (when message is sent)
c--c	Message. This will be placed in both the text and binary vehicle logs.

## 7.19 \$BPRMP - Request Mission Plan

This message requests that the Bluefin MVC transmit the currently-executing mission plan. The plan will be transmitted using a series of BFPLN messages (see section 6.20).

If \$BPRMP is used to request the mission plan in this way, it is not necessary for the payload to use \$BPLOG to explicitly request delivery of \$BFPLN messages.

If this message is acknowledged (see section 6.21), a ‘success’ acknowledgement indicates that the description of the entire mission has been successfully transmitted.

Since processing of this command requires interaction with the vehicle’s behavior control subsystem followed by transmission of a potentially large number of BFPLN messages, this command will always return an immediate, initial “pending” acknowledgement, followed by the

BFPLN messages, followed by the final success/failure acknowledgement.

\$BPRMP, hhmmss.ss*hh	
hhmmss.ss	Timestamp (when message is sent)

## 7.20 \$BPSEM - Start Empty Mission (Not Implemented)

This command causes the vehicle to start an empty mission. An empty mission starts all of the processes and communications for behaviors to be performed. After a mission is started, the payload can use the modify current behavior command to tell vehicle to perform actions. Must be finalized with end, abort, or kill mission on completion of all behaviors. The vehicle must be specially configured to allow empty missions to be started by the payload.

\$BPSEM, hhmmss.ss, t. t, a*hh	
hhmmss.ss	Timestamp (when message is sent)
t. t	Maximum mission duration (hours, minutes or seconds)
A	Mission duration mode (hours = 0, minutes = 1, seconds = 2)

## 7.21 \$BPNU - Navigation Position Update

This command provides the vehicle with an additional navigation fix. This is used for passing usbl updates to the vehicle when the acoustic modem is not under the control of Bluefin software.

\$BPNU, hhmmss.ss, 1111.11, a, yyyy.yy, b, e. e, d. d, f, hhmmss.ss*hh	
hhmmss.ss	Timestamp (when message is sent)
1111.11	Latitude
A	Hemisphere (N/S)
yyyyyy.yy	Longitude
B	Hemisphere (E/W)
e. e	Circular error probability for this fix. 50% odds that the target is actually within this position.
d. d	Depth (m)
F	Force update (true = 1, false = 0).
hhmmss.ss	Timestamp (of computed navigation solution)

## 7.22 \$BPSIL - Silent Mode

This command tells the vehicle to enter and exit silent mode. Silent mode quiets any devices that may interfere with sensitive acoustic devices within the payload. These devices include acoustic modems, the tailcone (must already be stopped), the variable buoyancy system (if equipped), and the trim adjustment system (if equipped). The DVL and INS will not be silenced in order to keep the navigation accurate.

In order to monitor the status of the silent mode, Payload Command Acknowledgements must be enabled (section 6.21 ). An immediate acknowledgement on the receipt of the command will be sent to the payload, followed by a deferred acknowledgement within 2 seconds indicating completion. Before silent mode is entered, the target state of the tailcone is checked to determine if it is stopping spinning. Before the deferred acknowledgement is sent, the current state of the tailcone is checked to see if it has stopped. If it has not, the deferred acknowledgement will be held off for another two seconds. If the tailcone is still spinning after that, the deferred acknowledgement will be an error.

The silent mode will time out after a configurable amount of time (default 5 minutes) if it is not explicitly set back to normal. The vehicle will resume as normal with no feedback to the payload. If the command to enter silent mode is sent again while in silent mode, it will not act to extend it beyond the time limit. The command will only be accepted if the vehicle is either not currently in a mission, or if a mission is executing but the tailcone is not spinning. To change the configurable timeout, please consult Bluefin.

\$BPSIL, hhmmss.ss, m*hh	
hhmmss.ss	Timestamp (when message is sent)
m	Mode (normal = 0, silent/quiet = 1)

## 7.23 \$BPTRM - Request Trim Adjustment Behavior

This command tells the vehicle to perform the trim adjustment behavior which levels the vehicle in the pitch and/or roll axes. This operation is completed within the vehicle's specified tolerance and only if the hardware to adjust trim is present. If it is detected that no adjustment is necessary, the behavior will immediately complete successfully. The implementation of the trim adjustment system is vehicle-specific; please consult with Bluefin for greater detail.

The status of a requested trim adjustment is communicated in two ways: 1. If acknowledgements have been enabled, a successful acknowledgement to the \$BPTRM message indicates that the trim adjustment behavior has begun, while a failure acknowledgement indicates that the trim-adjustment functionality is not available or otherwise could not be deployed. 2. If \$BFTRM messages have been requested, they are sent to the payload periodically while the trim adjustment is in progress, with the final \$BFTRM message indicating the final success or failure of the trimming operation. Additionally, \$BFMBS and \$BFMBE messages will be transmitted at the beginning and end of the trim-adjustment behavior.

\$BFTRM must be registered using the \$BPLOG command before requesting the trim behavior.

The order (interleaving) of the \$BFMBS and \$BFMBE and the other messages is unspecified.

The behavior will return an error status and end if unable to complete trimming operation within a specified time, typically 20 minutes. The operation can be canceled with the Cancel Current Behavior command (see section 7.10 ).

The command will only be accepted if the vehicle is either not currently in a mission, or if a mission is executing but the tailcone is not spinning. The vehicle must also not be on the surface. The implemented trim algorithm will adjust the trim after it is detected that the vehicle has stopped moving forward.

It is possible that the vehicle may perform implicit trim adjustments at times other than explicitly requested by this command. Any implicit adjustments, however, will be immediately terminated (will not continue) if the vehicle is commanded to go silent (via the \$BPSIL command, section 7.22 ).

\$BPTRM, hhmmss.ss*hh	
hhmmss.ss	Timestamp (when message is sent)

## 7.24 \$BPBOY - Request Buoyancy Adjustment Behavior

This command tells the vehicle to perform the buoyancy adjustment behavior using the variable buoyancy system. This operation is completed within the vehicle's specified tolerance and only if the hardware to adjust buoyancy is present. If it is detected that no adjustment is necessary, the behavior will immediately complete successfully. The implementation of the variable buoyancy system is vehicle-specific; please consult with Bluefin for greater detail.

The status of a requested buoyancy adjustment is communicated in two ways: 1. If acknowledgements have been enabled, a successful acknowledgement to the \$BPBOY message indicates that the buoyancy adjustment behavior has begun, while a failure acknowledgement indicates that the buoyancy-adjustment functionality is not available or otherwise could not be deployed. 2. If \$BFBOY messages have been requested, they are sent to the payload periodically, a \$BFBOY message will indicate the final success or failure of the buoyancy adjusting operation. Additionally, \$BFMBS and \$BFMBE messages will be transmitted at the beginning and end of the buoyancy-adjustment behavior.

\$BFBOY must be registered using the \$BPLOG command before requesting the buoyancy behavior. The order (interleaving) of the \$BFMBS and \$BFMBE and the other messages is unspecified.

The behavior will return an error status and end if unable to complete buoyancy adjusting operation within a specified time, typically 20 minutes. The operation can be canceled with the Cancel Current Behavior command (see section 7.10 ).

The command will only be accepted if the vehicle is either not currently in a mission, or if a mission is executing but the tailcone is not spinning. The vehicle must also not be on the

surface. The Buoyancy adjustment may cause the vehicle to no longer be trimmed. If the vehicle is equipped with a trim adjustment system, the Request Trim Adjustment Behavior (\$BPTRM section **Error! Reference source not found.**) should be sent afterward. The implemented buoyancy algorithm will adjust the buoyancy after it is detected that the vehicle has stopped moving forward.

It is possible that the vehicle may perform implicit buoyancy adjustments at times other than explicitly requested by this command. Any implicit adjustments, however, will be immediately terminated (will not continue) if the vehicle is commanded to go silent (via the \$BPSIL command, section 7.22).

\$BPBOY, hhmmss.ss*hh	
hhmmss.ss	Timestamp (when message is sent)

## 7.25 \$BPVER - Payload Interface Version

This message reports the version of the Bluefin Standard Payload Interface that the payload supports. The payload must send this message at the beginning of communications.

\$BPVER, hhmmss.ss, v-v*hh	
hhmmss.ss	Timestamp
v-v	Version number

## 8 Messages Specific to Hovering Vehicles: From Payload to Bluefin Vehicle

### 8.1 \$BPSUS - Suspend Mission

This command is used to suspend any running mission and take control of the vehicle. The vehicle will await other commands until a Resume Mission command is received or a two second timeout expires. If the vehicle is already suspending a mission, this command has no effect (other than to reset the timeout).

\$BPSUS, hhmmss.ss*hh	
hhmmss.ss	Timestamp (when message is sent)

### 8.2 \$BPCON - Continue

This command does nothing except reset the two second mission suspension timeout. A payload can send this command, or equivalently the Suspend Mission command, in order to keep the vehicle under payload control.

\$BPCON, hhmmss.ss*hh	
hhmmss.ss	Timestamp (when message is sent)

### 8.3 \$BPRES - Resume Mission

This command causes the vehicle to resume the mission that was running when a Suspend Mission command was received. Depending on the behavior that was running when the mission was suspended, the vehicle may first return to the point at which the mission was suspended or may continue from the current location.

\$BPRES, hhmmss.ss*hh	
hhmmss.ss	Timestamp (when message is sent)

### 8.4 \$BPSPD - Hull Relative Speed Limit

This command sets a maximum desired speed for payload commanded hull relative movement. The vehicle will not go faster than its best controllable speed no matter what this is set to, but setting this to 0.5 m/s can be useful if a slow survey is desired. 1.5 is the normal default.

Note that only hull relative vehicles (such as the HAUV) can perform hull relative movement.

\$BPSPD, hhmmss.ss, x.x*hh	
hhmmss.ss	Timestamp (when message is sent)
x.x	Speed, in meters per second

### 8.5 \$BPSAN - Set Sonar Angle

For vehicles that have a movable sonar (currently the HAUV only), this command adjusts the angle the sonar is pointed to.

\$BPSAN, hhmmss.ss, x.x*hh	
hhmmss.ss	Timestamp (when message is sent)
x.x	Sonar angle, in degrees

## 8.6 \$BPGHP - Go To Hull Position

For vehicles capable of Hull relative movement, this command, if given while a mission is suspended, will cause the vehicle to travel to the indicated position on the hull. Otherwise it is ignored. This command also resets the two second watchdog timer mentioned in Suspend Mission.

\$BPGHP, hhmmss. ss, h. h, v. v, d. d*hh	
hhmmss. ss	Timestamp (when message is sent)
h. h	Horizontal position, in meters
v. v	Vertical position, in meters
d. d	Distance from the hull, in meters

## 8.7 \$BPGBP - Go to Bottom Position

For vehicles capable of relative navigation (currently the HAUV), this command, if given while a mission is suspended, will cause the vehicle to travel to the indicated position. Otherwise it is ignored. This command also resets the two second watchdog timer mentioned in Suspend Mission

\$BPGBP, hhmmss. ss, x. x, y. y, c, d. d*hh	
hhmmss. ss	Timestamp (when message is sent)
x. x	X coordinate, in meters
y. y	Y coordinate, in meters
c	Interpretation of the next field. 'D' means depth, 'A' means altitude
d. d	Depth or Altitude, in meters

## 8.8 \$BPRNS - Reset Relative Navigation

For vehicles capable of relative navigation (currently the HAUV), this command will update the

vehicle position. This is intended to be used to improve the accuracy of the navigation by using external information.

\$BPRNS, hhmmss. ss, x. x, y. y, b. b*hh	
hhmmss. ss	Timestamp (when message is sent)
x. x	Updated X coordinate, in meters
y. y	Updated Y coordinate, in meters
b. b	Updated bearing

## 8.9 \$BPRBO - Hull Relative Bearing Offset

For vehicles capable of relative navigation (currently the HAUV), this command will add an additional offset to the inertial bearing of the vehicle.

\$BPRBO, hhmmss. ss, x. x*hh	
hhmmss. ss	Timestamp (when message is sent)
x. x	Bearing offset, in degrees

## 8.10 \$BPPTZ - Pan Tilt Zoom Camera Control

For vehicles with a Pan Tilt Zoom Camera installed (HAUV), this command will control the camera position. Each of these values ranges from -9 to 9, measured in 1/10ths of a second worth of movement.

\$BPPTZ, hhmmss. ss, p, t, z	
hhmmss. ss	Timestamp (when message is sent)
p	Pan Movement
t	Tilt Movement
z	Zoom Movement

## 8.11 \$BPLIT - Strobe Light Control

For vehicles with a Strobe Light installed (HAUV), this command will control the light. Each field (excepting the timestamp) is an integer.

\$BPPTZ, hhmmss.ss, l, o	
hhmmss.ss	Timestamp (when message is sent)
l	Which light to use (ignored for now)
o	Light On/Off. 0 corresponds to Off, 1 to On.

## 9 Advanced Messages Specific to Low Level Acoustic Communications

The following messages: Communications Medium Access (\$BFCMA section 9.1 ), Communications Packet Sent (\$BFCPS section 9.2 ), Communications Packet Received (\$BFCPR section 9.3 ), and Communications Packet Data (\$BPCPD section **Error! Reference source not found.** provide the payload low level access to the underwater acoustic communications hardware. These messages are supported only on certain vehicles, and not necessarily in all versions of this specification, ask Bluefin for more details.

### 9.1 \$BFCMA - Communications Medium Access

This message along with Communications Packet Sent (\$BFCPS section 9.2 ), Communications Packet Received (\$BFCPR section 9.3 ), and Communications Packet Data (\$BPCPD section **Error! Reference source not found.** provide the payload low level access to the underwater acoustic communications hardware. These messages are supported only on certain vehicles, and not necessarily in all versions of this specification, ask Bluefin for more details.

This message notifies the payload of a timing window available to it for sending data over the vehicle's communication devices. These devices include acoustic modems and possibly Iridium SBD in the future. The message contains a timestamp signifying the end of the timing window. Using this, the payload can calculate the total length of time of the window by subtracting the message timestamp from it. The modem address is supplied as well as the type of device that is being allocated for the payload's use. The timing windows allocated for the payload are given as larger chunks of time instead of more frequent short windows (time for only a single message). This allows the payload freedom to send multiple messages during a timing window and allows for enough time for the payload and vehicle's communications to not interfere with each other. During the timing window, the payload can send messages to the vehicle using the \$BPCPD message (section **Error! Reference source not found.**).

By default, these status messages are *not* sent. The payload can request that they be sent by

using the BPLOG command (see section 7.1 ). Registering to receive these messages will not automatically give the payload communications time windows. The schedule and window length must first be worked out with Bluefin.

\$BFCMA, hhmmss. ss, hhmmss. ss, a, d-d, *hh	
hhmmss. ss	Timestamp
hhmmss. ss	End time of timing window
a	Modem address
d-d	Device type (“Sonardyne6G”, “WHOI”, “WHOIPSK”)

## 9.2 \$BFCPS - Communications Packet Sent

This message along with Communications Medium Access (\$BFCMA section 9.1 ), Communications Packet Received (\$BFCPR section 9.3 ), and Communications Packet Data (\$BPCPD section **Error! Reference source not found.** provide the payload low level access to the underwater acoustic communications hardware. These messages are supported only on certain vehicles, and not necessarily in all versions of this specification, ask Bluefin for more details.

This message is sent to the payload notifying that the specified packet (Request ID) has been sent. The Request ID is a unique identifier supplied by the payload to the vehicle when the message is originally requested to be sent using the \$BPCPD command (see section **Error! Reference source not found.**). The message supplies the number of frames that were sent, followed by the status for each frame. The status for each frame will be either unset (if no frame was specified), unacknowledged (if no ack was requested), or acked (if an ack was requested). In the event that the modem is not properly functioning, a status message may not be generated for every request to send data. The message will also be sent for every packet that the vehicle has originated. It is important to work out a unique identifier creation scheme with Bluefin that will not conflict.

By default, these status messages are *not* sent. The payload can request that they be sent by using the BPLOG command (see section 7.1 ).

\$BFCPS, hhmmss. ss, hhmmss. ss, r, n, fs0, fs1, fs2, fs3, fs4, fs5, fs6, fs7*hh	
hhmmss. ss	Timestamp (when message is sent)
hhmmss. ss	Time Sent (when packet was sent)
r	Request ID (unique identifier for the message, as specified in corresponding \$BPCPD request)
n	Number of Frames sent (1-8)

\$BFCPS, hhmmss. ss, hhmmss. ss, r, n, fs0, fs1, fs2, fs3, fs4, fs5, fs6, fs7*hh	
fs0	Frame 0 Status (Unset = 0, Unacknowledged = 1, Acked = 2, FailedToSend = 3)
...	
fs7	Frame 7 Status (Unset = 0, Unacknowledged = 1, Acked = 2, FailedToSend = 3)

### 9.3 \$BFCPR - Communications Packet Received Data

This message along with Communications Medium Access (\$BFCMA section 9.1 ), Communications Packet Sent (\$BFCPS section 9.2 ), and Communications Packet Data (\$BPCPD section **Error! Reference source not found.** provide the payload low level access to the underwater acoustic communications hardware. These messages are supported only on certain vehicles, and not necessarily in all versions of this specification, ask Bluefin for more details.

This message is sent to the payload when a message is received by the vehicle. The arrival time of the message as well as the source and destination (note: destination may NOT be this vehicle) modem identifiers are provided. The telemetry mode used (communications hardware specific) is available (InvalidMode = -1 if not known) as well as the total number of frames received. The frame data with a status for each is immediately following. If only two frames were received, the number of frames will list two, and only the first two frames will have data and statuses. The remaining slots for frames will be blank for the data, and unset (0) for the status.

By default, these status messages are *not* sent. The payload can request that they be sent by using the BPLOG command (see section 7.1 ).

Base 16 (hex) encoding uses the following mapping: 0-9 A-F (digits, followed by the uppercase letters A-F). The encoding allows for 4 bits to be represented per printed character (as defined by RFC2045 / RFC3548 / RFC4648).

\$BFCPR, hhmmss. ss, hhmmss. ss, s, d, t, n, fs0, f0, fs1, f1, fs2, f2, fs3, f3, fs4, f4, fs5, f5, fs6, f6, fs7, f7*hh	
hhmmss. ss	Timestamp (when message is sent)
hhmmss. ss	Arrival Time
s	Source (identifier for the source modem ID)
d	Destination (identifier for the destination modem ID)
t	Telemetry Mode (communications hardware specific, InvalidMode = -1 if unknown)
n	Number of Frames being sent (1-8)

\$BFCPR, hhmmss.ss, hhmmss.ss, s, d, t, n, fs0, f0, fs1, f1, fs2, f2, fs3, f3, fs4, f4, fs5, f5, fs6, f6, fs7, f7*hh	
fs0	Frame 0 Status (Unset = 0, Good = 1, Corrupted = 6, Undecodable = 7)
f0	Frame 0 Data (hex encoded)
...	
fs7	Frame 7 Status (Unset = 0, Good = 1, Corrupted = 6, Undecodable = 7)
f7	Frame 7 Data (hex encoded)

## 9.4 \$BPCPD - Communications Packet Data

This message along with Communications Medium Access (\$BFCMA section 9.1), Communications Packet Sent (\$BFCPS section 9.2), and Communications Packet Received (\$BFCPR section 9.3) provide the payload low level access to the underwater acoustic communications hardware. These messages are supported only on certain vehicles, and not necessarily in all versions of this specification, ask Bluefin for more details.

This command tells the vehicle to transmit the included frame data through the communications device to the destination provided. Anywhere from one to eight frames can be provided to the vehicle if the currently used device supports multiple frames as specified through the telemetry mode. The number of frames must also be provided to the vehicle through this interface along with the data for each. The request ID is a unique identifier that the payload must provide. This allows the vehicle to provide status on the sending of the packet (see \$BFCPS section 9.2). Since the status message will also be sent for every packet originated with the vehicle, it is important to work out a unique identifier creation scheme with Bluefin that will not conflict. If the payload desires an ack'd response to the packet being sent, set the flag to 1, and the vehicle will wait to send the \$BFCPS message until the ack has been received. This ack is different than the Payload Interface Message Acknowledgement (\$BFACK) described in section 6.21. A \$BFACK will be sent to the payload if enabled via the \$BFLOG command (see section 7.1).

The payload can only send messages through the communications devices when provided a window with the \$BFCMA (see section 9.1). Any message sent to the vehicle outside of the provided windows will be ignored.

Base 16 (hex) encoding uses the following mapping: 0-9 A-F (digits, followed by the uppercase letters A-F). The encoding allows for 4 bits to be represented per printed character (as defined by RFC2045 / RFC3548 / RFC4648).

\$BPCPD, hhmmss.ss, r, d, t, a, n, f0, f1, f2, f3, f4, f5, f6, f7*hh	
hhmmss.ss	Timestamp (when message is sent)
r	Request ID (unique identifier for the message, generated by the payload)

\$BPCPD, hhmmss.ss, r, d, t, a, n, f0, f1, f2, f3, f4, f5, f6, f7*hh	
d	Destination (identifier for the destination modem ID)
t	Telemetry Mode (communications hardware specific)
a	Ack Requested (boolean false = 0, true = 1)
n	Number of Frames being sent (1-8, dependent on Telemetry Mode)
f0	Frame 0 Data (hex encoded)
...	
f7	Frame 7 Data (hex encoded)