Brizo Variants Firmware Manual

GNSS DIRECTIONAL WAVEHEIGHT SENSOR

Version 1.5

September 2020

Shipped From

Contact Us

Email	support@xeostech.com
Phone	(902) 444-7650
Fax	(902) 444-7651
Website	www.xeostech.com

Version History

Version No.	Firmware Ver.	Date	Description
0.7	1.1.4310	Apr 2018	Rough Manual Release
0.8	1.1.4310	July 2018	Combined Firmware/Hardware Manuals
1.0	1.1.4483	July 2018	Document Overhaul
1.1	1.1.4761	Aug 2018	New Serial Commands Added
1.2	1.1.5847	Sept 2019	Change to Wave Height Output Interface, wave height message example, legacy section added for old changes
1.3	1.1.6314	Feb 2020	Add Brizo X and new message types
1.4	1.1.6480	Mar 2020	Add SDI-12 and Polling mode
1.5	1.1.6650	Sep 2020	Hardware version upgrade

This manual is written with respect to the firmware aspects of the Brizo and is meant to be used in combination with the Brizo Hardware User Manual or Brizo-X Hardware Manual.

Table of Contents

Device Theory of Operation

The Brizo utilizes a Global Navigation Satellite System (GNSS) receiver to measure wave statistics. All enabled peripherals, including the GNSS are controlled by a scheduler. The scheduler will enable and disable the GNSS at the commanded interval. After collecting enough data for the measurement period, the device will calculate the wave coefficients and output them over the enabled communication interfaces.

GNSS Receiver Theory of Operation

The GNSS receiver schedule is controlled by the configured wave height sample rate. The GNSS outputs configurable Septentrio Binary Format (SBF) blocks, which the processor uses to calculate the wave coefficients. Raw GNSS measurements are recorded to the SD Cards. The GNSS will power on and off to save power if an ongoing measurement period is not occurring.

After power up, the GNSS receiver waits to receive GNSS signal before starting calculations. GNSS signals are not available indoors. This means the device will not attempt to transmit messages over Iridium, Cell or Serial until it has GNSS signal.

USB, Serial 1 and 2 Theory of Operation

The Brizo has two independent serial ports, serial 1 and serial 2. Serial 1 is designated for diagnostics and command and control. The baud rate of this port is 115.2 kbaud. The USB port output is the duplicative of Serial 1. For more information on USB and Serial 1 see the section on Device Configuration. Serial 2 is designated for outputting calculated wave height messages (configurable). Messages can be output in RS232 as Binary or ASCII message types. This can be configured through the **$waveheightsetresultinterface** commands autonomously or polled.

SDI-12 Theory of Operation

When SDI-12 is enabled data can be collected on the SDI-12 interface at 1200 baud. Data is transmitted in **ASCII** format and communication occurs over a single data line in half-duplex.
SDI-12 follows a master-slave configuration whereby a data logger (or other data acquisition device) requests data from the Brizo. The Wave Height message is broken into 3 parts to fit into the SDI-12 protocol. The interface only works on a polled methodology. The SDI-12 interface must be enabled using **$SDI12SetEnabled T** before using the interface.

SBD Theory of Operation

SBDs are a low cost, low power method of communicating remotely with a unit if an Iridium modem is installed. SBDs are the Iridium equivalent of a text message (SMS).

If the Iridium interface is enabled using the **$waveheightsetresultinterface** command, an SBD message will be automatically scheduled and sent to XeosOnline with the calculated wave coefficients. This is timed to avoid GNSS interference. At this point the modem will also check for messages waiting at the Iridium gateway. Additional SBD message checks can be scheduled, but may interfere with the GNSS if the antenna placement is not properly managed.

Any messages sent by the user between the last message check and the next message check will wait at the Iridium gateway for the unit to check in.

The unit requires an unlock code in order for any command sent to the device to be acted upon. This is to prevent undesired command and control. Unlock codes are sent out to the provisioned addresses of the modem on startup and each SBD after. Unlock codes are automatically included in SBDs appearing in XeosOnline's message log.

XBD Theory of Operation

Note: Cell modems are not currently available on the Brizo X, only the Brizo. Please contact the sales department if this is of interest.

XBDs are a low cost, low power method of communicating remotely with a unit if a cell modem is installed. XBDs are similar to iMessages in that they use data, not SMS. This has the advantage of being able to be recorded by a computer database (Xeos Online).

If the Cellular interface is enabled using the **$waveheightsetresultinterface** command an XBD message will be automatically scheduled and sent to Xeos Online with the calculated wave coefficients. At this point the modem will also check for messages waiting at the server.

The unit requires an unlock code in order for any command sent to the device to be acted upon. This is to prevent undesired command and control. Unlock codes are sent out to the provisioned addresses of the modem on startup and each XBD after. Unlock codes are automatically included in XBDs appearing in XeosOnline's message log.

SD Cards Theory of Operation

All collected data from the GNSS receiver is initially written to an SD Card. While the file is being written to the file that is 'open' on the device, this makes it important to notify the device when the intent is to shut it down.

Proper shut down etiquette is to either send the device a **$resetnow** command or to hold the test button (not available on the Brizo X) for 3+ seconds. Failure to do so will cause the file to be left open. Open files are partially recoverable on a computer provided they are not written over. Files written to the SD card use a naming convention with an ASCII representation of BASE-36 characters with the following name format:

SSFYYMDH.sbf

SS	Two-character representation of device serial number
F	Stream Source (1-4), A (Wave Height Messages))
YY	Two-character representation of year
M	Month of year
D	Day of Month
H	Hour of Day

Example: **2T10FATC.sbf**

Serial #	2T =101
Stream	1 =1
Year	0F = 15 (2015)
Month	A=10 (October)
Day	T=29
Hour	C = 12

The GNSS SBF files follow the format outlined in the **Septentrio SBF Reference Guide**. SBF files can be converted to Rinex or ASCII using the SBF Converter from Septentrio included in the free RxTools software suite linked below:

https://manuals.alertgeo.com/RxTools/

Processed GNSS data (wave height messages) can also be written to the SD cards on a configurable basis. This can be configured through the **$waveheightsetresultinterface** commands

Device Configuration

All of the settings on the Brizo can be configured either by connecting one of the diagnostic ports to a serial port terminal, a USB port, or by sending configuration commands from a remote location over the Iridium Satellite Network or Cellular Data (telemetry options needed). If configuring the unit locally, USB is recommended.

USB Diagnostic Port

Note: The USB Diagnostic port is only available on the Brizo and is not available on the Brizo X

All the settings on the Brizo can be configured by connecting the USB diagnostic port to a USB port on your computer.

If using Windows to connect to the USB diagnostic port, you may need to download a driver for the STM32 USB chip contained in the unit. The driver can be found here:

http://www.st.com/en/development-tools/stsw-stm32102.html

Modern Linux and Mac operating systems already support the driver.

After downloading and installing the driver, connect the provided USB mini cable between the Brizo and your computer. Your computer should assign the Brizo a COM port number (Windows only), that will show up under ports in the Windows Device Manager. If the device is not recognized, check the driver the device is implementing.

To communicate with the Brizo you will need a serial port terminal program. **Xeos Technologies recommends STerm, a Windows-only product**. A free alternative for Linux and Mac is **CoolTerm**. STerm and related configuration files can be downloaded here:

https://manuals.alertgeo.com/STerm

The port settings for the Brizo USB are:

Baud Rate	Any
Data Bits	8
Parity	None
Stop Bits	1
Flow Control	Disabled

Serial Diagnostic Port

The RS 232 serial diagnostic port is available for both the Brizo and the Brizo X. See the hardware manual for the respective pin outs diagrams. The serial port 1 settings for the Brizo USB are:

Baud Rate	115 200
Data Bits	8
Parity	None
Stop Bits	1
Flow Control	Disabled

Using STerm and Sending Commands

FIGURE 1: AN STERM WINDOW

STerm is the officially supported serial port terminal program of Xeos Technologies. STerm can load configuration files to allow the commands for Xeos Technologies products to appear as buttons on the right side of the window. This can be done through the FILE>OPEN menu and opening the Brizo Config File.stm file. This is available to download here:

https://manuals.alertgeo.com/STerm

To open a connection to your device, power it on and wait for the computer to recognize the device driver. Select the COM port of your device on the Serial Port Settings menu in the bottom right corner of the screen. Click the **Open Port** button. You should now be connected to the unit. The buttons that appear on the right side of the window can be clicked and will send commands to your unit.

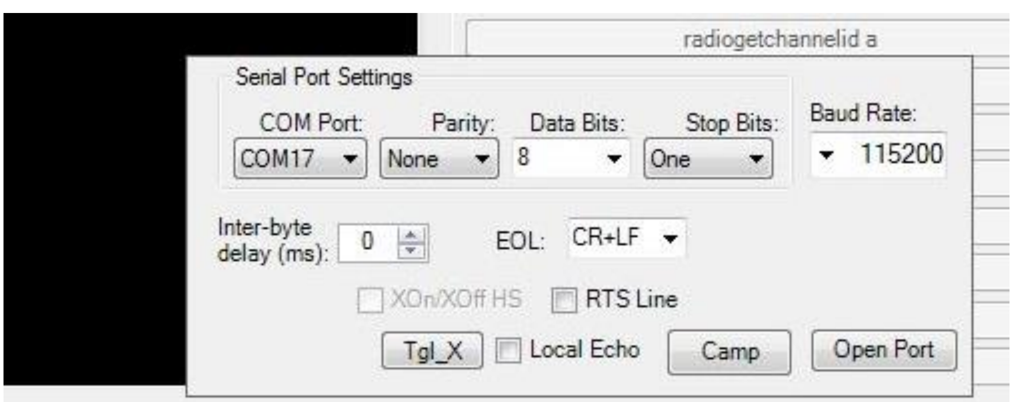


FIGURE 2: STERM SERIAL PORT SETTINGS MENU

The buttons on the right side of the window can be edited to change the input variable you are sending to the device. This is done by hovering over the right side of the button then clicking the green edit button. Type your changes into the button then move your mouse away to end the edit. Remember to save your changes by navigating to **File > Save.**

FIGURE 3: THE EDIT ICON OF THE COMMAND BUTTONS

Configuration Commands

While there are many commands outlined in this section, it is likely that your unit is mostly configured from the factory.

Measurement Methodology

Measurements can be enabled with the following command **"$WaveHeightSetEnabled T**". The operation of the GNSS and the output calculations are all controlled by the onboard processor, with users able to configure the device to operate once, twice or three times and hour. The timers on the device are powered by an onboard battery backed up by a real time clock (RTC) and are generated by the number of messages per hour. The RTC is updated from GNSS time to compensate for drift. The RTC continues to tick while the device is powered off and operates in the UTC timeframe.

All peripherals and calculations can be disabled using **"$WaveHeightSetEnabled F**".

Wave Height Measurement Rate Configuration

The wave height sample rate (per hour) can be configured with the following command:

$WaveheightSetSampleRate [number]

This command sets the number of measurement periods per hour, not the sample rate of the measurement during the measuring period. For instance to configure the measurement rate to be 3 times per hour the following command would be used:

$WaveheightSetSampleRate 3

The only valid options for this command are 1, 2 or 3. The number or periods per hour can be confirmed with the following command:

$WaveheightGetSampleRate

Autonomous Wave Height Measurement Output Interface

Output interfaces can be configured to autonomously output data using

$WaveHeightSetResultInterface [interface] [datatype]

The current autonomous options for interface (the method in which data is output) are:

[1] Iridium SBD *

[2] Output over Serial 2

[3] Cell XBD *

[4] SD Card

Additionally, the options for datatype (the format in which data is output) are:

[0] OFF

[1] Binary

[2] ASCII

To set the unit to output results over Iridium in binary, the following command would be sent.

$WaveheightSetResultInterface 1 1

The Brizo can output data in multiple forms of interfaces simultaneously, however each interface used must be configured using the **$WaveHeightSetResultInterface** command separately.

The autonomous output interfaces can be confirmed with the **$WaveHeightGetResultInterfaces** command without a number for all interfaces, or entering a specific interface for a response indicative to the requested interface.

Iridium SBD and Cell XBD output data in ASCII is not supported, only Binary.
Automous SDI-12 output is not supported.

Serial-2 Configuration

The Serial 2 interface port settings are as follows:

Baud Rate	Configurable
Data Bits	8
Parity	None
Stop Bits	1
Flow Control	Disabled

The serial 2 baud rate can be set with the command (on one of the diagnostic interfaces (i.e. Serial 1) **$Serial2SetBaud [Baud].** For example, to set the baud rate to 115200 the following command would be issued. **$Serial2SetBaud 115200** . To retrieve the current baud rate the command **$Serial2GetBaud** . Additionally the Serial 2 interface can be tested with the command **$Serial2Test** (on the diagnostic port) . The line “Serial 2 test FROM DIAGNOSTIC PORT” will print on the diagnostic port and on the Serial 2 interface.

No configuration beyond the baud rate is required to poll data from the Serial 2 port. If you wish for the data to be delivered autonomously, the serial 2 autonomous wave height result interface must be set to be enabled, and deliver either ASCII or binary data as outlined in the Autonomous Wave Height Measurement Output Interface section

SDI-12 Configuration

The SDI-12 interface must be enabled as it is not powered by default. It can be enabled using **$SDI12SetEnabled T**. A device reset is required after enabling the interface. After being enabled, the enabled state holds through future resets. The status of the interface can be requested with **$SDI12GetEnabled.** Commands for retrieving data over the SDI-12 Interface are covered in the Retrieving Data section of the manual.

Wave Height Message Type

The Brizo waveheight message type can be set by sending the following commands:

$WaveheightSetMessageType [type]

Where [type] is an integer. A reset is required after updating the message type. A list of message types is available in Appendix B. The current message type can be recovered with

$WaveheightGetMessageType

SBD and XBD Configuration

If your unit is equipped with cellular and/or Iridium modems, then it is possible to use SBD (Iridium) and/or SMS (cellular) to interrogate or configure the unit.

Additional SBD and SMS message checks (outside of the sent messages) can be configured individually using the interval and offset timer commands. Messages with the calculated data will be automatically configured based on the set sample rate. The commands to configure the timers are below and operate on the same principals outlined in detail in the GNSS section:

$XBDSetInterval *[time]*
$XBDSetOffset *[time]*
$SBDSetInterval *[time]*
$SBDSetOffset *[time]*

SBD and XBD capable units will, when configured send a power-up message when the unit first turns on. This is useful for confirming when deploying in the field, and notifying users of remote stations of unexpected power loss.

Sending SBDs or XBDs Using XeosOnline

After logging into XeosOnline, click on "**FILE>SEND**"**.** A new window will appear in the browser.

The left side of the window will have a list of all the units in your XeosOnline organization in the form of the Serial Number or other identification name followed by an (IMEI). IMEI's are identifying numbers of Iridium or Cellular modems.

Select which units you want to send messages to (you can send messages to multiple units at once) by moving them into the right hand selected column by using the arrow keys, or dragging the name of the device to the right side of the window.

Type out the commands you wish to send to your unit in the commands column at the bottom of the window.

Include the **$** ahead of the commands, leave white space or return carriages after commands.

Click **Send** to distribute the messages to the gateway, to wait for the unit to check in. When the unit receives the command(s) it will respond to the command(s) via SBD/XBD.

Only single line responses are sent via SBD/XBD. Any command that responds in a multiline list will not reply with the list. Both Outgoing and incoming messages can be viewed in the **messages tab** on XeosOnline.

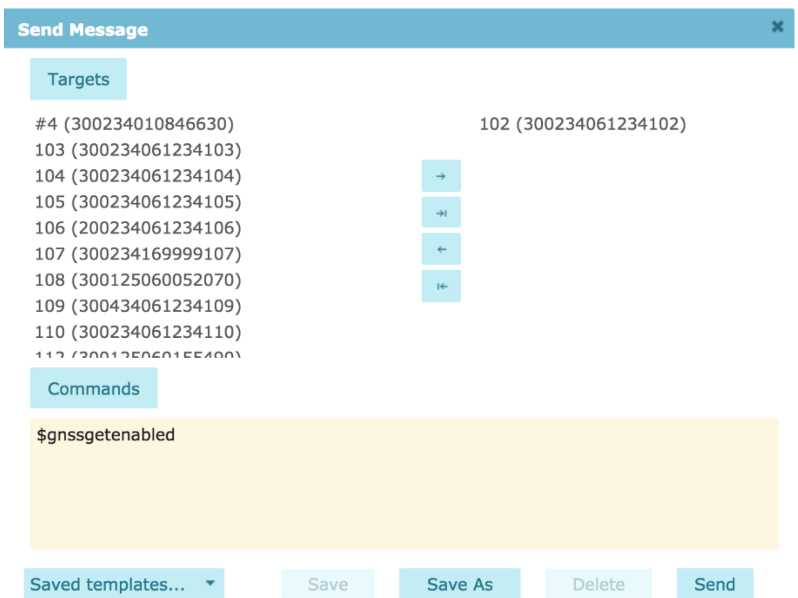


FIGURE 4: XEOSONLINE SEND MESSAGE WINDOW

XeosOnline

Contact Xeos Technologies to set up an organization in XeosOnline. XeosOnline can be used to view the state of the unit's Tunnel Connection, Status of Health and for easily sending and receiving SBD messages to units configured with the correct modems. Units with Iridium SBD can also view the wave parameters on XeosOnline under the device tab.

Wave Height Data

After each wave height session, data is stored in the Brizo for that measurement period in preparation for transmission (if applicable). An example of the data sent from a wave height session is below (message Type 2). For all data messages see Appendix B

```
Timestamp: 2018-07-28T05:30:00.000Z, BatteryV: 10.20, Temperature: 22,
Significant Wave Height (m): 0.58, Peak Wave Period(s): 7.52, Maximum Wave Height(m): 1.00,
Wave Direction (deg): 147, Wave Spread (deg): 34, Sample Quality: 7
```

Variable Descriptions:

Field	Units	Description
Timestamp	N/A	Time at the start of the wave height session
BatteryV	Volts	Last voltage read by the Brizo before transmission
Temperature	°C	Temperature internally read by the Brizo before transmission
Significant Wave Height	Meters	The average of the highest one-third of waves, as measured from the trough to the crest of the waves
Peak Wave Period	Seconds	Corresponds to the frequency band with the maximum value of spectral density in the non-directional wave spectrum
Maximum Wave Height	Meters	Largest wave measured over the period
Wave Direction	Degrees True North (90° being east)	Direction with the most wave energy in a directional wave spectrum (reported as degrees from true north, 90° being east)
Wave Spread	Degrees	Estimate of directional spread of the dominant wave
Sample Quality	N/A	Statistic that indicates how many segments of a measurement period received good GNSS data (max 7)
Mean Zero Up Crossing Period	Seconds	The mean of the time between zero elevation crossings in the upward direction. Also referred to as Tz.

Retrieving Data

The Brizo includes multiple interfaces to retrieve data. It is important to note the GNSS requires a clear view of the sky to begin calculating wave parameters and no message will be transmitted or output if the unit antenna cannot view the sky or collect the GNSS signals (for example the unit is indoors).

SBD Transmissions

SBD Transmissions can be accessed on XeosOnline. Log in to XeosOnline and click on the Wave Height Tab to display wave parameters and status of health information. Tabular data can be downloaded as CSV or excel files or displayed graphically on XeosOnline.

XBD Transmissions

XBD Transmissions can be accessed on XeosOnline. Log in to XeosOnline and click on the Wave Height Tab to display wave parameters and status of health information. Tabular data can be downloaded as CSV or excel files or displayed graphically on XeosOnline.

Serial (RS-232)

Serial data can be collected on the 'Serial 2' RS-232 port. Data is transmitted in a binary or ASCII form outlined in Appendix B. Sample messages, parsed code, and matlab parsing code are also available to help with integration here. Serial data can be acquired through two modes, Autonomous mode where periodically receive wave height data once with each cycle of wave height calculation (sample) and Polling mode where the latest wave height data is received whenever polled.

Data could be polled in two formats; ASCII or Binary depending on the following two commands that should be run from data logger (or other data acquisition device) on the Serial 2 port.

$waveheightgetdataascii

$waveheightgetdatabin

Note: No data will be available if Brizo is just turned on or reset until the first measurement period.

Xeos supplies a serial cable with the Brizo when no other telemetry packages are selected. See the hardware manual for cable drawings.

Wiring of the serial 2 connection can be confirmed by sending a **$serial2test** command over either USB or Serial 1. The device will output the ASCII text "Serial 2 test FROM DIAGNOSTIC PORT" over both the diagnostic port, and serial 2 port. Note: the device will not output wave data until it has a valid GNSS fix with greater than 8 satellites (initial start condition only), this means it must be outdoors to send wave data.

Xeos also supplies a Serial Cable with the Brizo X. See the Brizo X hardware manual for the Brizo X Pinout options.

SDI-12

Serial data can be collected on the SDI-12 interface. Data is transmitted in **ASCII** format and communication occurs over a single data line in half-duplex.
SDI-12 follows a master-slave configuration whereby a data logger (or other data acquisition device) requests data from the Brizo. The Wave Height message is broken into 3 parts to fit into the SDI-12 protocol as follows:

- Data part 0: Type, Battery Voltage, Temperature
- Data part 1: Significant Wave Height, Maximum Wave Height, Peak Wave Period, Peak Wave Direction
- Data Part 2: Peak Wave Spread, Quality, Unix Time, Mean Zero Crossing Period

The following commands are for testing SDI-12 connection through a serial port terminal program by using SDI-12 USB Adapter

Note: Carriage Return and Line Feed are represented with <CR> and <LF>. The output results depend upon the message type selected.

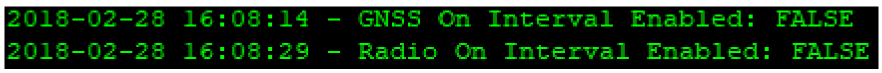
Command	Response example (With data available-message type 4)	Response (No data Available)	Notes
0!	0<CR><LF>	0<CR><LF>	Confirm communication
0C!	000011<CR><LF>	000000<CR><LF>	Check for data, 0 if none, 11 if some available. This is required before the 0D commands
0D0!	0+4+19.39+27<CR><LF>		Data part 0
0D1!	0+0.00+0.00+15.05+56<CR><LF>		Data part 1
0D2!	0+38+4+1581619440+8.28<CR><LF>		Data part 2

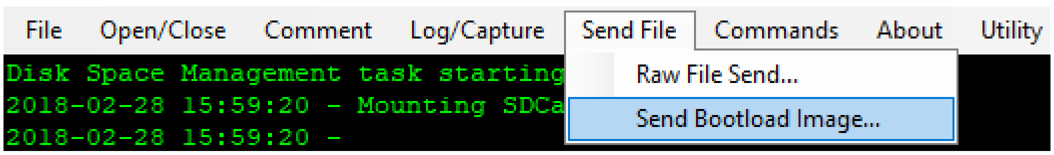
Upgrading Firmware

Connect to the Brizo with a USB or Serial cable and start STerm. Open a connection at 115200 baud.

Send the following commands to turn off wave height calculation:

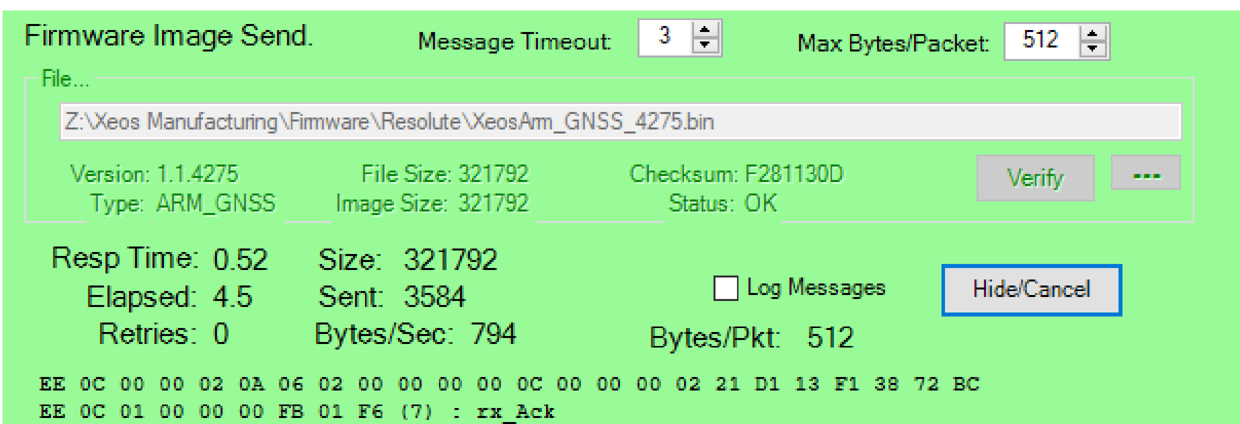
$waveheightsetenabled f





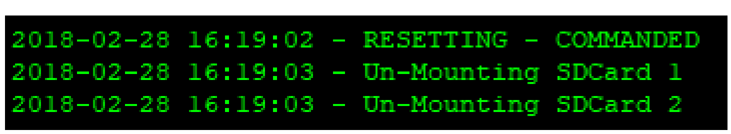
On the top menu bar, navigate to *Send File* and click on *Send Bootload Image*.

The *Firmware Image Send* window will open and you will need to select the new version of firmware by clicking the … and opening the .bin file. Once the *Sent* number matches the *Size* number, the upload is complete.



Reset the unit by sending the following command and wait for the unit to restart:

$resetnow



Check the firmware version to confirm the installation was successful by entering the **$ver** command and comparing it to the number on the .bin file that was uploaded:

$ver

```
2018-02-28 16:20:34 - Ver: 1.1.4275  Boot: 1.1.4097
```

If the new version wasn't properly uploaded, resend the image.

Delete the old firmware using the **$imglst** command to view all the firmware images on the Brizo.

```
2018-02-28 16:21:44 - Sctr Errs     DateTime      Size  Type    Version   Pri/Count
2018-02-28 16:21:44 -    2    0   1519833755    320512 x3000  1.01.04262  1/011
2018-02-28 16:21:44 -   81    0   1519834264    321792 x3000  1.01.04275  1/015
```

In this example, firmware version 4275 was uploaded so 4262 must be deleted. Enter **$imgdelete** followed by the version number, omitting the zeroes and replacing periods with a space. For this example, the following command would be sent to delete the old firmware version **1.01.04262**:

$imgdelete 1 1 4262

Be careful to not delete the current version of firmware by accident.

Finally, re-enable the wave height measurements by entering this command:

$waveheightsetenabled t

Appendix A: Command Index

Command	Variables	Description	Sample
$resetnow	N/A	Resets the unit	$resetnow
$schedule	N/A	Shows schedule of all enabled peripherals	$schedule
$ver	N/A	Shows the firmware version loaded	$ver
$getserialnumber	N/A	Shows the device serial nubmer	$getserialnumber
$waveheightsetenabled	T,F	T (true) enables the measurement, F (false) disables the measurement	$waveheightsetenabled t
$waveheightgetenabled	N/A	Returns the enabled state of the measurment	$waveheightgetenabled
$waveheightsetsamplerate	[1-3]	Set the number of measurement periods per hour	$waveheightsetsamplerate 1
$waveheightgetsamplerate	N/A	Gets the number of measurement periods per hour	$waveheightgetsamplerate
$waveheightsetresultinterface	[1-4] [0-2]	Sets the result interfaces and message modes (ASCII or Binary)	$waveheightsetresultinterface 1 1
$waveheightgetresultinterface	[1-4]	Returns the status of a specific result interface	$waveheightgetresultinterface 4
$waveheightgetresultinterfaces	N/A	Returns all result interfaces	$waveheightgetresultinterfaces
$WaveheightSetMessageType	[2,4]	Sets Message Type	$WaveheightSetMessageType 4
$WaveheightGetMessageType	N/A	Returns active message type	$waveheightgetmessagetype
$irdsetdod	T,F	T (true) enables DISA gateway, F (false) disables DISA gateway	$irdsetdod t

$irdgetdod	N/A	Returns enabled state of DISA gateway	$irdgetdod
$sbdsetenabled	T,F	T (true) enables SBD, F (false) disables SBD	$sbdsetenabled t
$sbdgetenabled	N/A	Returns the enabled state of SBD message checks	$sbdgetenabled
$sbdsetinterval	s,m,h,d	Set the interval of SBD message checks	$sbdsetinterval 4h
$sbdgetinterval	N/A	Returns the interval of the SBD message checks	$sbdgetinterval
$sbdsetoffset	s,m,h,d	Set the interval of the SBD message checks	$sbdsetoffset 0
$sbdgetoffset	N/A	Returns the offset of the SBD message checks	$sbdgetoffset
$serial2test	N/A	Outputs "Serial 2 test FROM DIAGNOSTIC PORT" in ASCII format over diagnostic and Serial 2 ports	$serial2test
$SDI12SetEnabled	T/F	Turn on the SDI12 bus	$sdi12SetEnabled t
$SDI12GetEnabled	N/A	Returns whether SDI-12 is enabled	$sdi12GetEnabled
$waveheightgetdataascii	N/A	Request Data in ASCII format (Serial 2 only)	$waveheightgetdataascii
$waveheightgetdatabin	N/A	Request Data in Binary format (Serial 2 only)	$waveheightgetdatabin
0!	N/A	SDI-12 Command to confirm communication	0!
0C!	N/A	SDI-12 Command	0C!

		to confirm data available	
0D0!	N/A	SDI-12 Command to request data message part 0	0D0!
0D1!	N/A	SDI-12 Command to request data message part 1	0D1!
0D2!	N/A	SDI-12 Command to request data message part 2	0D2!

Appendix B: Message Formats

Message Type 2: Binary "First 5" Coefficients Message – Total 19 Bytes

Data Type	Name	Value /Unit
uint8_t	Start Byte	0x6f
uint8_t	Type	0x02
uint8_t	Battery Voltage	0.1 V
int8_t	Temperature	deg C
uint16_t	Significant Wave Height	cm
uint16_t	Maximum Wave Height	cm
uint16_t	Peak Wave Period	cs (or 0.01 s)
uint16_t	Peak Wave Direction	Degrees from North
uint16_t	Peak Wave Spread	Degrees
uint8_t	Quality	Integer
uint32_t	Unix Time	Unix Minutes

Message Type 2: ASCII "First 5" Coefficient Message (Comma Delimited) – Total varies

Type, Battery Voltage (V), Temperature (degrees C), Significant Wave Height (m), Max Wave Height(m), Peak Wave Period(s), Wave Direction (Degrees), Wave Spread (Degrees), Sample Quality, Unix Time (seconds)

Message Type 4: Binary Extended Wave Coefficients Message – Total 21 Bytes

uint8_t	Start Byte	0x6f
uint8_t	Type	0x04
uint8_t	Battery Voltage	0.1 V
int8_t	Temperature	deg C
uint16_t	Significant Wave Height	cm
uint16_t	Maximum Wave Height	cm
uint16_t	Peak Wave Period	cs (or 0.01 s)
uint16_t	Peak Wave Direction	DDegrees from North
uint16_t	Peak Wave Spread	Degrees
uint8_t	Quality	Integer
uint32_t	Unix Time	Unix Minutes
uint16_t	Mean Zero Crossing Period	cs (or 0.01s)

Message Type 4: ASCII Extended Wave Coefficient Message (Comma Delimited) – Total varies

Type, Battery Voltage (V), Temperature (degrees C), Significant Wave Height (meters), Max Wave Height (meters), Peak Wave Period (seconds), Wave Direction (Degrees), Wave Spread (Degrees), Sample Quality, Unix Time (seconds), Mean Zero Crossing Period (seconds)

Appendix C: RxTools Download

Download the RxTools software and manual from https://manuals.alertgeo.com/RxTools/
RX Tools contains the SBF Converter to convert SBF blocks to ASCII or RINEX. SBF Block info can be found in the SBF Reference Guide also in the link above.

Appendix D: Legacy Features

Wave Height Measurement Output Interface (Pre-Build 5847)

The output interface can currently be configured using

$WaveHeightSetResultInterface [bitmap]

The current options for the bitmap are:

[1] Iridium SBD

[2] Output over serial

Other telemetry options available upon request. To set the unit to output results over Iridium for example the following command would be sent.

$WaveheightSetResultInterface 1

The output interface can be confirmed with the **$WaveHeightGetResultInterface** command without a number.

Warranty, Support and Limited Liability

Xeos Technologies Inc. warranties the Brizo to be free of defects in material or manufacturing for a period of one year following delivery. Liability is limited to repair or replacement of the defective part and will be done free of charge.

LIMITED WARRANTY: Xeos Technologies Inc. warrants that the product will perform substantially in accordance with the accompanying written materials for a period of one year from the date of receipt.

CUSTOMER REMEDIES: Xeos Technologies Inc. entire liability and your exclusive remedy shall be at Xeos Technologies Inc. option, either (a) return of the price paid or (b) repair or replacement of the product that does not meet Xeos Technologies Inc. Limited Warranty and that is returned to Xeos Technologies Inc. with a copy of your receipt. This Limited Warranty is void if failure of the product has resulted from accident, abuse, or misapplication. Any replacement product will be accepted for the remainder of the original warranty period or ninety (90) days, whichever is longer.

NO OTHER WARRANTIES: Xeos Technologies Inc. disclaims all other warranties, either expressed or implied, including but not limited to implied warranties of merchantability and fitness for a purpose, with respect to the product or the accompanying written materials. This limited warranty gives you specific legal rights. You may have others, which vary from state to state.

NO LIABILITY FOR CONSEQUENTIAL DAMAGES: In no event shall Xeos Technologies Inc. or its suppliers be liable for any damages whatsoever (including, without limitation, damages for loss of equipment, for loss of business profits, business interruption, loss of business information, or other pecuniary loss) arising out of the use of or inability to use this Xeos Technologies Inc. product, even if Xeos Technologies Inc. has been advised of the possibility of such damages.