

## OVERVIEW

**Port configuration:** 115200 baud, no parity, and one stop bit

## COMMAND DESCRIPTIONS

### ? - Help Menus:

?: Display all command groups

x?: Display help/setting for command x

**Break:** ===

## BOTTOM TRACK COMMANDS (defaults are recommended)

bp####: Set # of bottom-track pings to average in each data ensemble (default=bp001)

bx####: Set max tracking depth in bottom-track mode, 10-65535 dm (default=bx00500)

## CONTROL SYSTEM COMMANDS

cs: Start pinging

1. After CS command is sent, no changes to commands can occur until BREAK sent.
2. If RiverPro is set to record data (MR1) & recorder is full, will not start pinging and will return RECORDER NOT READY message.

cstate: Display status of RiverPro

cstop: Stops pinging without resetting RiverPro

cw: Requests most recently stored ensemble for output

cz: Power down

## ENVIRONMENTAL COMMANDS

eb±#####: Corrects for electrical/magnetic bias between RiverPro heading & heading reference

1. A RiverPro/RioPro is receiving heading from its internal compass. A navigation map for the deployment area shows a declination of 10°10'W 1995 (9'E/year). This means the magnetic offset in the year 2001 at this location is  $(- (10+10/60) + (9/60*6)) = -9.26666$  degrees. Set the EB command value to EB-926.

## PERFORMANCE AND TESTING COMMANDS

pa: Diagnostic test

pc0: Help menu

>pc0

PC0 = Help

PC2 = Display Sensor Data

PC4 = Display Voltage Monitor ADC Data

PC5 = Display BlueTooth RSSI Data

PC6 = Display GPS RSSI Data

PC20 = Display Scrolling Sensor Data

PC40 = Display Scrolling Voltage Monitor Data PC50 = Display BlueTooth RSSI Data

PC60 = Display GPS RSSI Data

>

pt0:

## Startup:

Immediately after power is applied to the RiverPro/RioPro, it enters the STANDBY mode. Send a BREAK signal using BBTalk by pressing the End key. Alternately, a software break may be sent by sending either “+++” or “===”. When the RiverPro/RioPro first powers up or receives a BREAK signal, it responds with a wake-up message similar to the one shown below. The RiverPro/RioPro is now ready to accept commands at the “>” prompt from either a terminal or computer program.>break

```
RiverPro
Teledyne RD Instruments (c) 2015
All rights reserved.
Firmware Version: 56.xx
>
```

## Input Commands:

Input commands set RiverPro/RioPro operating parameters, start data collection, run built-in tests (BIT), and asks for output data. All commands are ASCII character(s) and must end with a carriage return (CR). For example,

```
>CR1<CR> [Your input]
```

If the entered command is valid, the RiverPro/RioPro executes the command. If the command is one that does not provide output data, the RiverPro/RioPro sends a carriage return line feed <CR> <LF> and displays a new “>” prompt. Continuing the example,

```
>CR1<CR> [Your original input]
[Parameters set to FACTORY defaults]
> [RiverPro/RioPro response to a valid, no-output command]
```

## DATA DECODING SEQUENCE

1. Locate the header data by locating the header ID number (in the case of PD0 profile data that will be 7F7F).
2. Confirm the correct header ID by:
  - a. Locate the checksum offset (located in the header data). This is the offset to the checksum word in the current ensemble. The size of the ensemble is this offset plus 2 bytes (to account for the checksum word), which is also the offset to the next ensemble.
  - b. Calculate the checksum of total number of bytes in the ensemble excluding the checksum. The checksum is calculated by adding the value of each byte. The 2-byte least significant digits calculated will be the checksum.
  - c. Read the 2-byte checksum word at the end of the ensemble, located by using the checksum offset in the header (determined in step 2-a) and compare this checksum word to the value calculated in step 2-b.
  - d. If the checksums match then this is a valid ensemble. If the checksums do not match then it is not a valid ensemble and should go back to step 1 and search for the next header ID number occurrence.

3. Locate the number of data types (located in the header data). Locate the offset to each data type (located in the header data).
4. Locate the data ID type to decode by using the offset to each data type and confirm the data ID number at that offset matches the ID type looking for.
5. Once the proper ID type has been located, use the Pioneer Technical Manual to understand what each byte represents in that particular data type.

#### Decoding an RiverPro/RioPro Ensemble

Use the following information to help write your own software.

#### Rules for the BroadBand Data Format PDO

1. All data types (i.e. fixed leader, variable leader, velocity, echo intensity, correlation, percent good, etc.) will be given a specific and unique ID number. The table below shows some of the most common IDs.

<b>Table 46: Common Data Format IDs ID</b>	<b>Description</b>
0x7F7F	Header
0x0000	Fixed Leader
0x0080	Variable Leader
0x0100	Velocity Profile Data
0x0200	Correlation Profile Data
0x0300	Echo Intensity Profile Data
0x0400	Percent Good Profile Data
0x0500	Profile Status Data
0x0600	Bottom Track Data