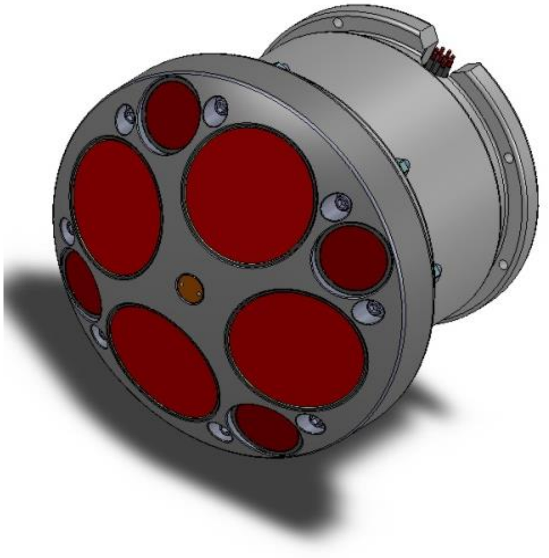
RTI ADCP/DVL

USER GUIDE



ROWE TECHNOLOGIES

Table of Contents

Introduction

Thank you for purchasing the Rowe Technologies Inc. (RTI) ADCP/DVL. This User Guide explains important features and considerations while owning and operating this instrument.

Please spend time in reading and noting the many aspects of this User Guide that relates to your applications.

The RTI ADCP/DVL is capable of DVL Bottom and Water track operations or it can be configured to current profile with or without bottom track enabled. Included in this instrument are a water temperature sensor and a minimum 2 gigabyte (GB) secure digital (SD) memory card. The installed compass/attitude sensor is used primarily for Earth referenced current profiling.

If you have any questions of comments please feel free to contact us at:

Rowe Technologies, Inc.
12655 Danielson Court, Suite 306
Poway, CA 92064
USA

Tel: +1 858 842 3020
Fax: +1 858 842 3021
Email: sales@rowetechinc.com

Chapter 1 Getting Started

Rowe Technologies, Inc. (RTI) develops and manufactures instruments that are pulsed backscatter sonar systems used to transmit and receive echoes from "targets" in the water column and along surface and bottom boundaries. The instruments are used to compute current velocity, vehicle velocity, range to targets based on acoustic travel time and bottom and water column acoustic image characteristics based on echo intensity and range. Instruments are typically divided into two categories Acoustic Doppler Current Profilers (ADCPs) and Doppler Velocity Logs (DVLs)

RTI has four product lines however each instrument can be used as a DVL or ADCP and a direct-reading system can be converted into a self-contained system by integrating a battery pack. Below is a description of the product lines.

ADCP – is a hydro-acoustic instrument or sonar device that is used to measure water velocities in discrete layers over a certain range that is defined by the acoustic frequency. Essentially, the ADCP emits a sound signal of a specific frequency and measures the return frequency that is received from backscatter via particles in the water column.

DVL – is a hydro-acoustic device that is used to bottom-track or process emitted sound signal that return from the seafloor or river bottom. Using a DVL combined with a position fix (via GPS) and a compass the DVL can be used for underwater navigation (i.e., Submarines, AUVs and ROVs).

It is important to note that instruments from RTI have dual functionality, which is to say that the instrument can function as an ADCP or as a DVL. Deployment type is defined when the instrument is being prepared for use and done in the software. Instruments are used in two ways, direct reading or self-contained. Direct reading systems are integrated and power by another system (i.e., DVL) and the data is used real-time. Self-contained or autonomous systems have battery power provided via battery packs and collect data remotely (i.e., bottom mount, mooring etc.). Self-contained systems can have data telemetered to the end user or the data can be downloaded manually upon instrument recovery.

SeaProfiler – are direct reading systems for real-time current profiling applications where a bottom-mounted or surface-deployed configuration is used with hard wired power and communications.

SeaWatch - are battery operated systems, well suited for moored and bottom-mounted applications.

SeaPilot - are battery-operated DVL systems that are ideal for subsea and surface vessel navigation applications. These systems are typically integrated into systems, such as ROVs, AUVs and other underwater navigation systems.

SeaWaves – are battery-operated ADCP systems developed for current measurements as well as wave height, direction and other first-order wave statistics.

All instruments have a measurement range determined by the acoustic frequency used by the system. The table below presents instrument range based acoustic processing technique (Narrowband or Broadband) and functionality, Profiling or Bottom-track.

Frequency (kHz)	**Profiling Broadband Range (m)**	**Profiling Narrowband Range (m)**	**Bottom track Range (m)**
1200	20	30	50
600	50	90	120
300	100	180	350
150	275	425	700
75	455	700	1000

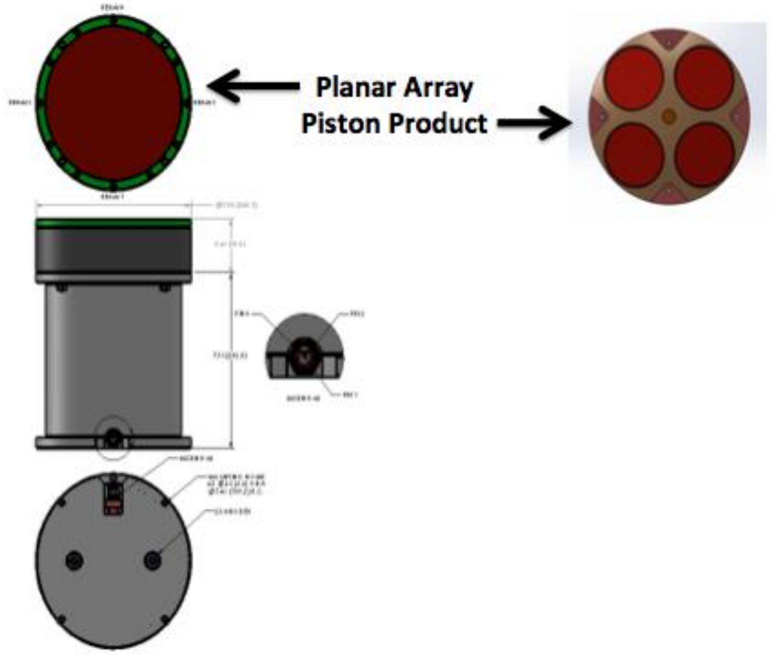
Broadband – is an acoustic signal processing technique that yields highly accurate current velocity measurements, however broadband processing has a decreased profiling range.

Narrowband - – is an acoustic signal processing technique that yields noisy current velocity measurements, however narrowband processing increased profiling range.

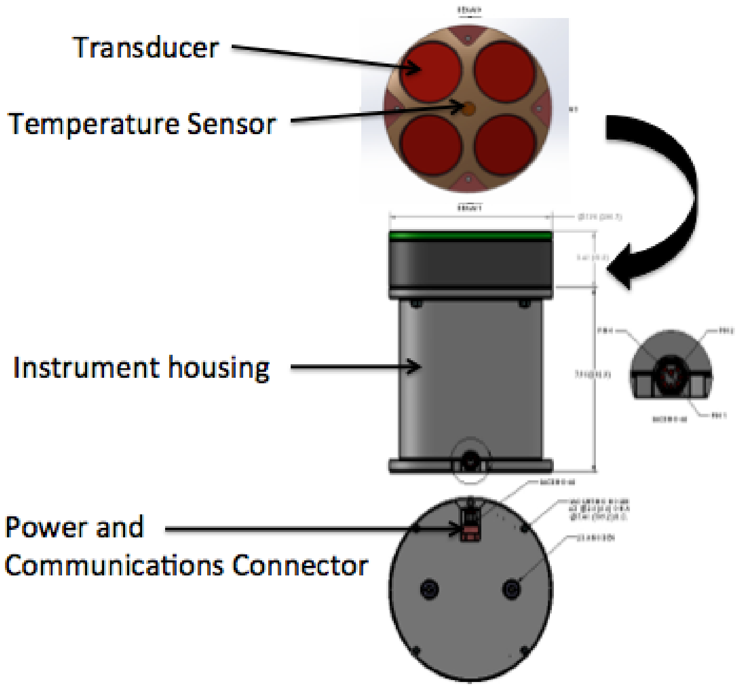
Bottom track - is a method for measuring speed-over-ground in from a floating platform such as a boat. The ADCP emits long acoustic pulses that are scattered by the bottom. Determining the Doppler shift in the returned echo provides a measure of the relative velocity between the ADCP/DVL and the bottom. This information is used to provide motion data required by the ADCP for its water-velocity measurements and navigation data to underwater vehicles.

Chapter 2 System Overview

RTI manufactures two types of ADCPs and DVLs – planar array and piston products. Both systems have similar mechanical housings and connectors are exactly the same, however planar array products have a single transducer, whereas the piston products are a number of individual transducers. The image below shows the difference between the systems.



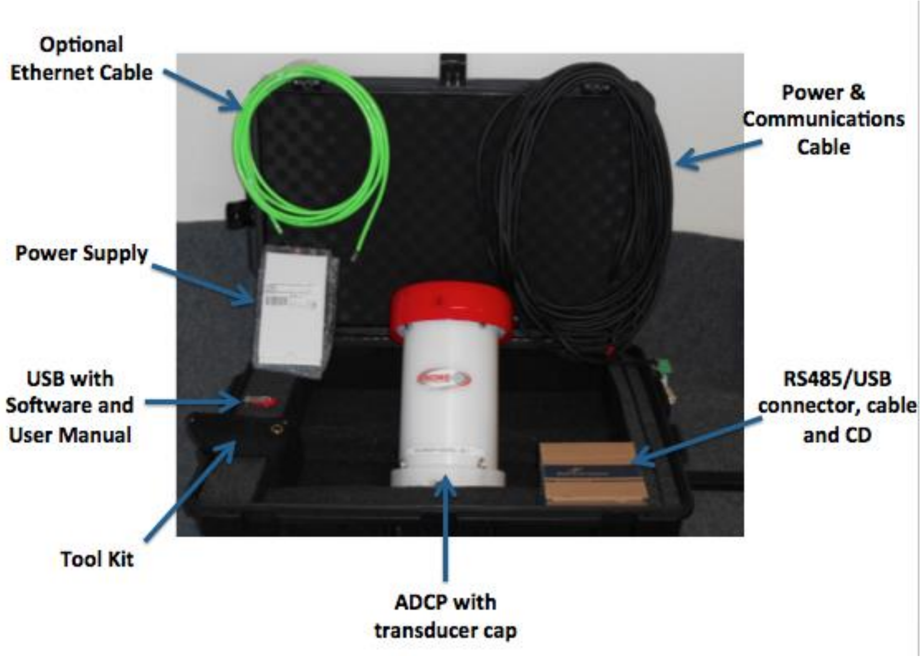
Planar array products form beams based on user inputs. Four beams can be formed at 30° or 20°off the vertical axis. An additional beam can be formed that is a vertical beam or at 0°. Piston systems have fixed transducers that have 20° beam angles. Single frequency systems have 4 transducers while multi-frequency systems have 8 transducers, four transducers with two frequencies. Dual frequency systems provide the benefit of having two instruments in one, allowing for increased resolution velocity profiling and shallow and deep water profiling. The image below presents the important components of a piston ADCP/DVL.



The ADCP is shipped with the accessories presented in the table below. It is important to note the information presented below is for a standard order. Custom orders can have various ADCP housing types to withstand deeper deployment depths (from Delrin to aluminum to titanium), longer cable lengths and a varying number and type of battery packs.

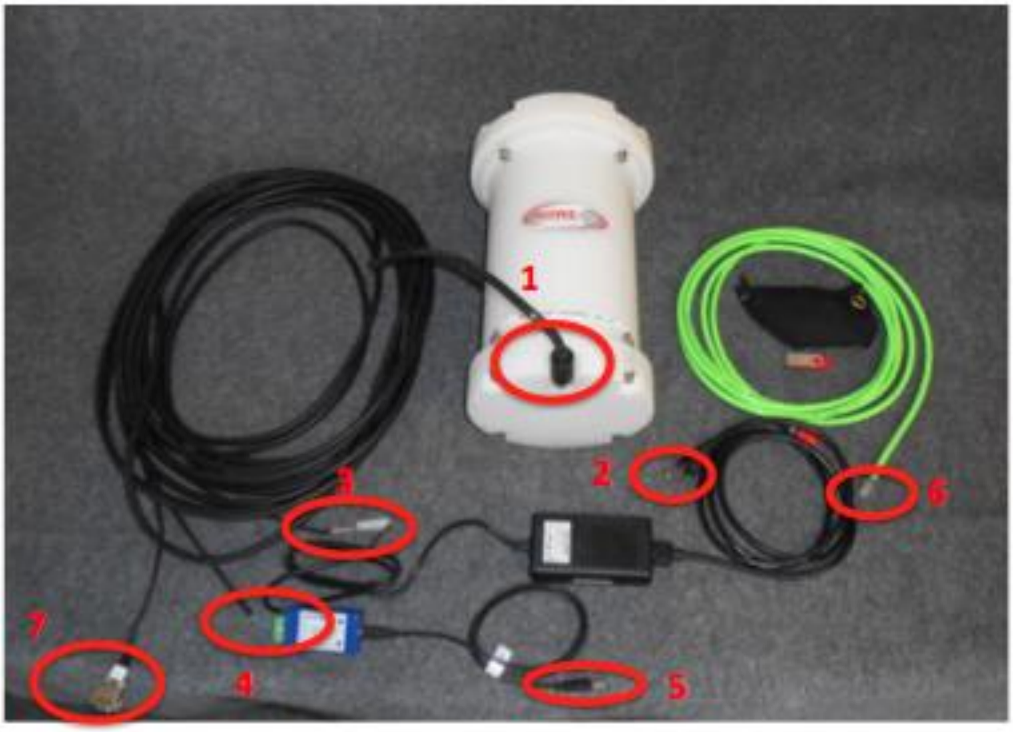
Part Description	Quantity
ADCP with Transducer Cap	1
Power and Communications Cable (10m, standard)	1
Power Supply (100-240V) and cord	1
RS422/485 to USB Converter with CD with driver for USB converter	1
Memory drive with software and documentation	1
Tool kit (wrench and screwdriver, spare hardware, O-rings and dummy plugs)	1
USB drive with software and user manuals	1
Ethernet Cable (3m, standard)	Optional
Battery Pack with D-cell batteries	Used with self-contained systems

Below is a photo of what is included in the Pelican case when the instrument is shipped to the end user.


Optional Ethernet Cable
Power & Communications Cable
Power Supply
USB with Software and User Manual
RS485/USB connector, cable and CD
Tool Kit
ADCP with transducer cap

2.1 Interconnecting the System

The photo below highlights the important cable connections for RTI ADCPs and DVLs. Each connection has an associated number and a corresponding description (included below).



1. ADCP/DVL to Power-Communications Cable – The ADCP/DVL has an 8-pin male wet-mateable connection (same face as the Rowe Technologies sticker) that interfaces with the 8-pin female wet-mateable connector. After mating the connectors, be sure to hand tighten the locking sleeve.

2. AC power connector – prior to connecting to an AC power source be sure that the cable is secure connected to the power adaptor and that connection described in #3 (below) is correctly connected.

3. Keyed 2-pin power connector – make sure that the power supply cable is correctly connected to the Power-Communications cable. The power supply cable has a two-pin female-keyed connector that plugs into a two-pin male connector.

4. The Power and Communications Cable has a male 5-pin connector that plugs into the female RS485 to USB adaptor.

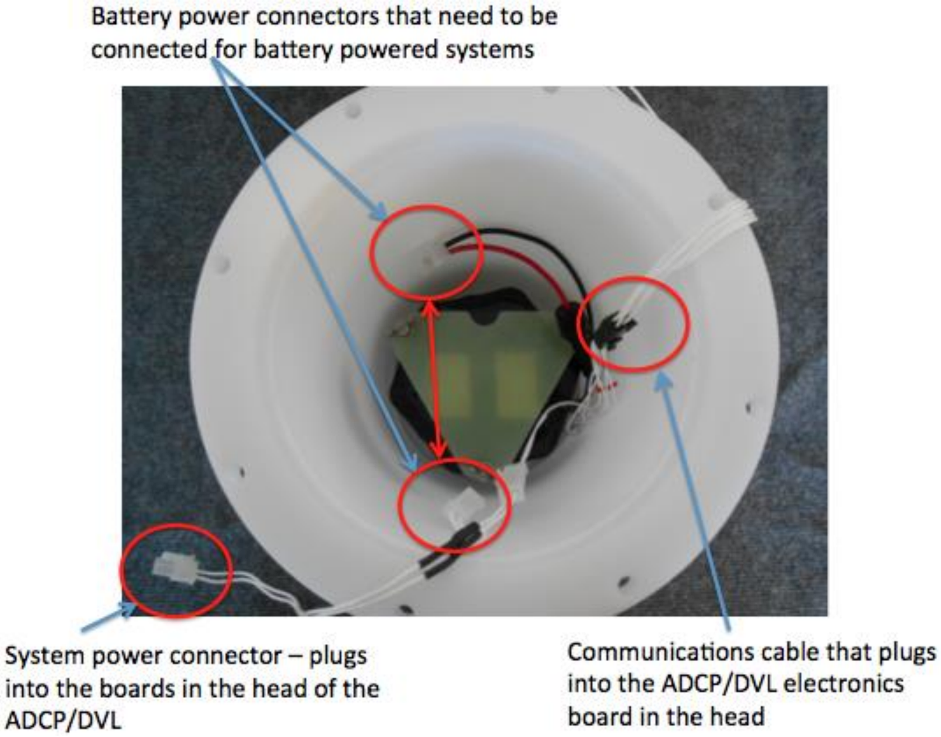
5. The USB connector of the RS484 to USB adaptor plugs directly into the users laptop/desktop computer for communications with the ADCP/DVL.

6. The optional Ethernet connector connects to an 8-pin male wet-mateable (opposite the Power and Communications connector) on the ADCP using an 8-pin wet-mateable female connector with locking sleeve. The other end of the cable connects to a laptop/desktop using the Ethernet female connector. This option will allow for fast downloads of large data files collected by the instrument.

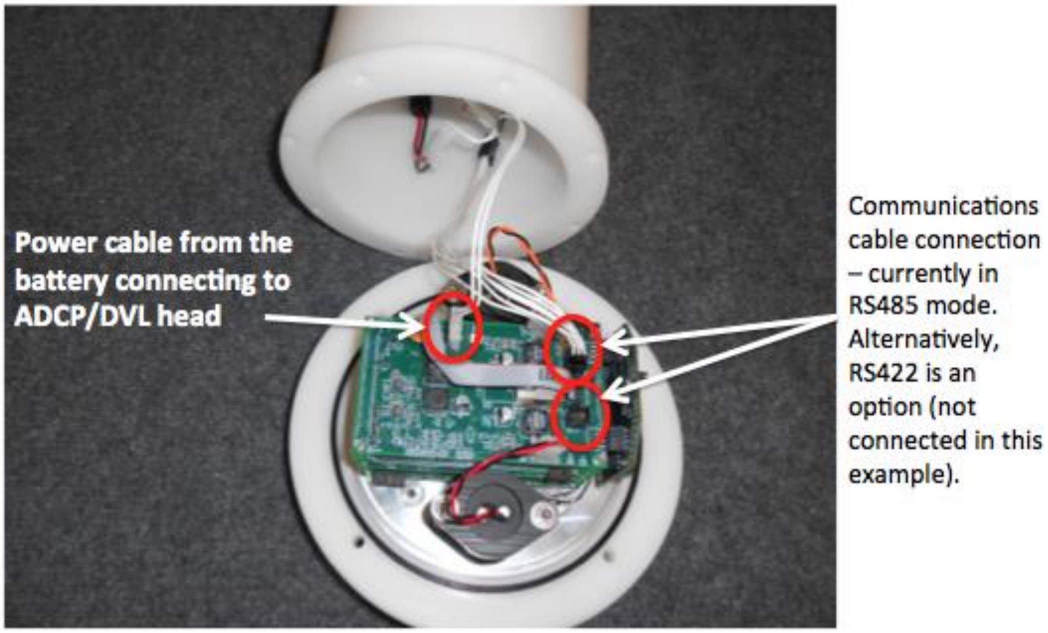
7. RS232 male connector that can be used to interface with a GPS system or connect directly to a female RS232 connector on laptops and desktops.

2.2 Interconnecting a Self-Contained (SC) system

In most cases, self-contained systems will interconnected on inside of the ADCP (unless otherwise requested in the order); with the exception of battery power connection to the ADCP. The internal battery pack will have to be connected for the system to function as a self-contained system. The image below highlights the cables/connectors on the inside of the ADCP.


Battery power connectors that need to be connected for battery powered systems
System power connector – plugs into the boards in the head of the ADCP/DVL
Communications cable that plugs into the ADCP/DVL electronics board in the head

The image below indicates the connections for the cables from the inside of the self-contained system and their connections to the electronic boards on the head of the ADCP. It is important to note that the image below is configured for RS485 communications (for RS422 the communications cable would be removed from the existing connector and placed in the empty connector, indicated below).


Power cable from the battery connecting to ADCP/DVL head
Communications cable connection – currently in RS485 mode. Alternatively, RS422 is an option (not connected in this example).

2.3 Connecting to an ADCP/DVL

Below are the following steps to connect to an ADCP/DVL. It is import to follow each step sequentially. It is important to note that RTI Unities Software is designed for a Windows operating system.

Step1: Install Driver

Prior to connecting to the ADCP/DVL, be sure that you install the driver needed for serial communications between the PC and the ADCP/DVL. The drive can be found on the small CD provided in the shipping case or at the following link:

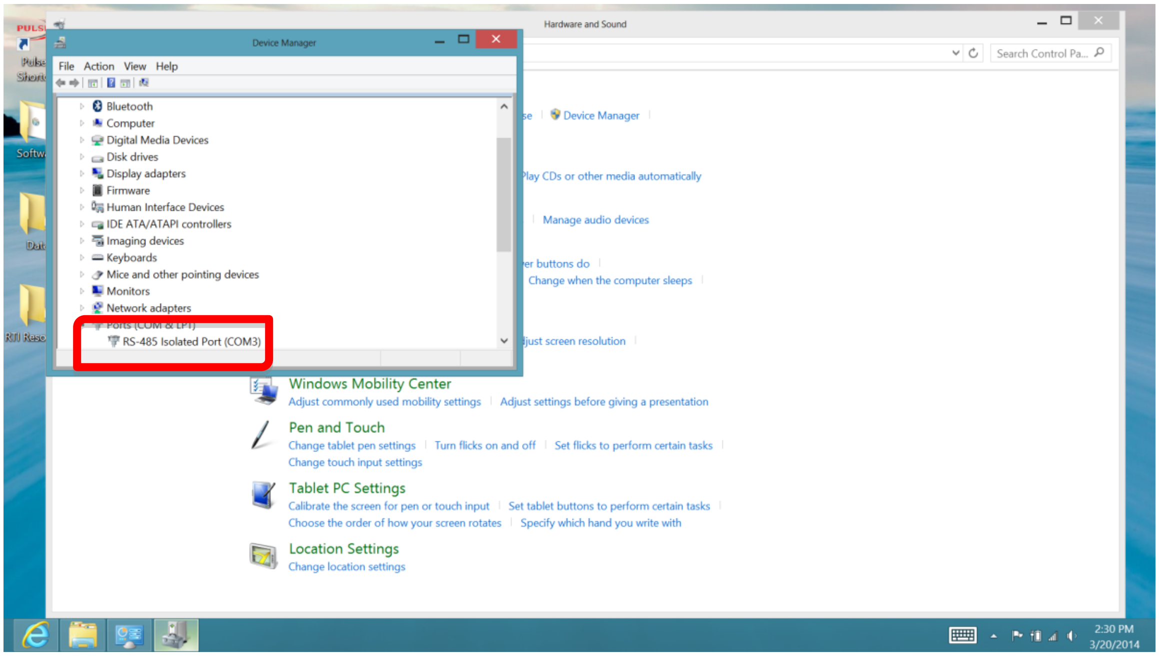
http://www.bb-elec.com/getattachment/c8461811-bebf-456a-8386-6ea1281219b4/USB_Drivers_PKG_v2-08-28.zip.aspx

Follow the instructions provided on screen to install the driver.

Step 2:Verify COM port

To verify that the installation proceeded correctly, insert the USB to Serial Converter into a USB port. Next go to the Control Panel and in the Device Manager menu expand the Ports menu to the COM port. The USB to serial converter should be identified as the following:

RS-485 Isolated Port (COM#)


Device Manager
File Action View Help
Bluetooth
Computer
Digital Media Devices
Disk drives
Display adapters
Firmware
Human Interface Devices
IDE ATA/ATAPI controllers
Imaging devices
Keyboards
Mice and other pointing devices
Monitors
Network adapters
Ports (COM & LPT)
RS-485 Isolated Port (COM3)
Hardware and Sound
Search Control Pa...
Device Manager
Play CDs or other media automatically
Manage audio devices
er buttons do
Change when the computer sleeps
just screen resolution
Windows Mobility Center
Adjust commonly used mobility settings
Adjust settings before giving a presentation
Pen and Touch
Change tablet pen settings
Turn flicks on and off
Set flicks to perform certain tasks
Change touch input settings
Tablet PC Settings
Calibrate the screen for pen or touch input
Set tablet buttons to perform certain tasks
Choose the order of how your screen rotates
Specify which hand you write with
Location Settings
Change location settings
2:30 PM
3/20/2014

The number sign will be replaced with an actual number, such as COM3 Indicated by the red box in the screen shot above). This is the COM port that the users should use to connect to the ADCP/DVL.

Step 3. ADCP/DVL Interconnecting

There are a few hard-connections that are required prior to interfacing to the instrument via software.

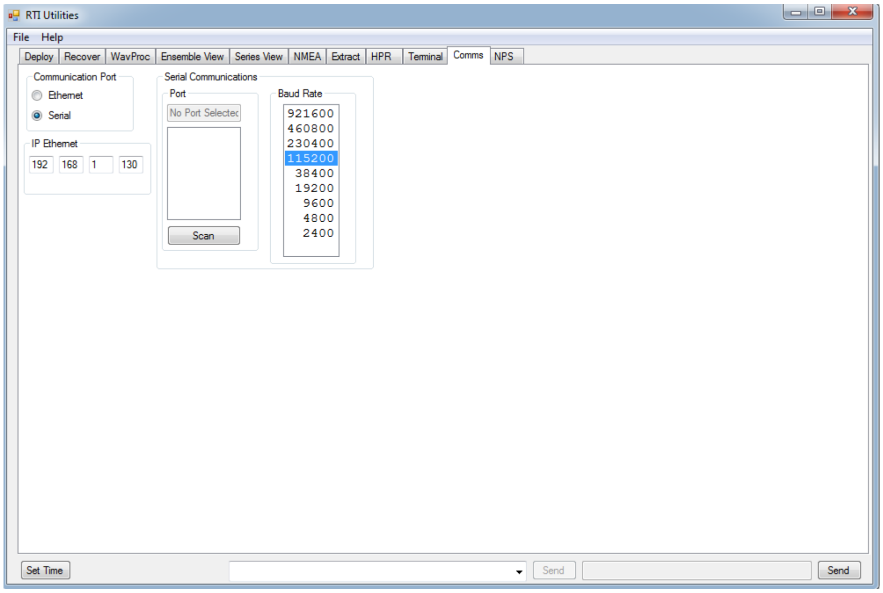
1. Connect the power supply to the instrument cable pig-tail via the keyed two pin connector
2. Make sure that the power supply is plugged into a power source/wall
3. Connect the keyed green five-pin connector on the instrument cable pig-tail to the USB-Serial adaptor
4. Insert the USB connector into a USB port on the PC

Troubleshooting: Once everything is interconnected, look at the ADCP/DVL. If things are powered up correctly, a red and a green light should be visibly pulsating/flickering – you may need to turn the lights off for the best view. If these lights are not visible verify that your connections are correct and that the power sources is actively providing power.

Step 4. Connecting to the Instrument via Software

Once all of the instrument interconnections have taken place, (presented in Step 3) open the RTI Utilities Software to connect/interface with the ADCP.

- Be sure that the Serial Radio Button is selected (as in the image below)
- Click Scan in the Serial Communications, Port Section of the software – this will scan and present available COM Ports.
- Select the COM Port identified in Step 2 above.
- Make sure default baud rate is selected (115200)



Communication Port: ○ Ethernet ● Serial

IP Ethernet: 192 168 1 130

Port: No Port Selectec

Baud Rate: 921600, 460800, 230400, 115200 (selected), 38400, 19200, 9600, 4800, 2400

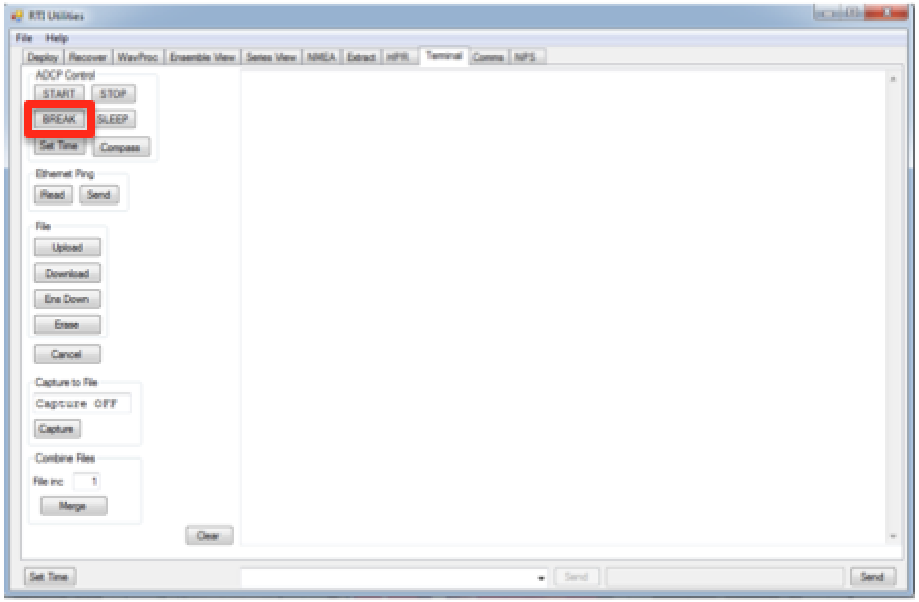
- In order to verify the instruments COMs are setup correctly; go to the Terminal tab (to the immediate left of the Comms Tab, example below).
- Click on the Break button to “wake-up” the instrument, a similar text string will appear in the terminal window:

Copyright (c) 2009-2014 Rowe Technologies Inc. All rights reserved.

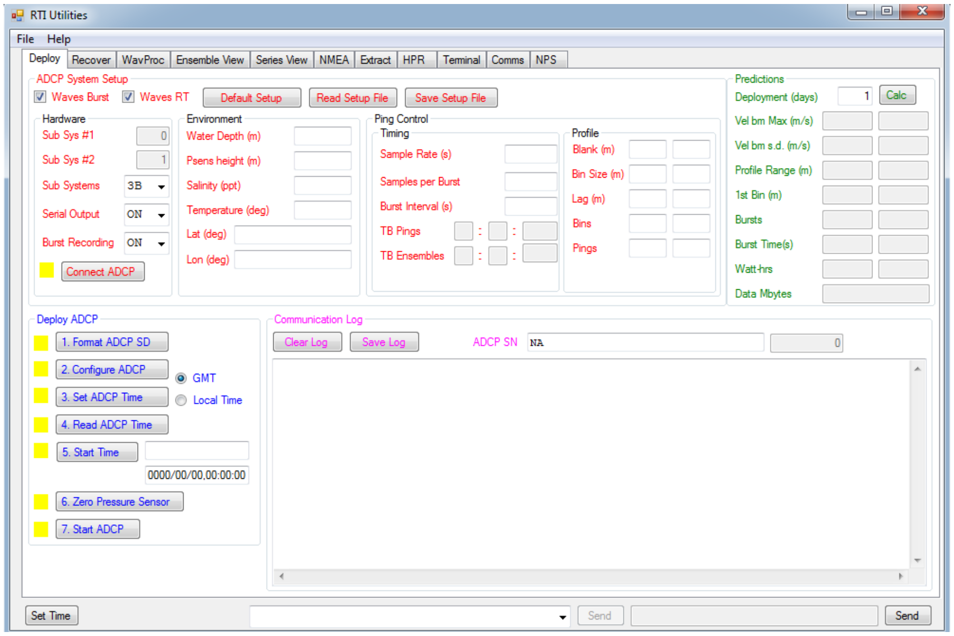
DP600 DP1200

SN: 013A0000000000000000000000000121

FW: 00.02.40 Mar 28 2014 12:24:24



- After the instrument is connected – go to the Deploy Tab to configure the ADCP/DVL (below). Click on the Connect ADCP to connect and configure the ADCP.



☑ Waves Burst ☑ Waves RT

Sub Sys #1: 0
Sub Sys #2: 1
Sub Systems: 3B
Serial Output: ON
Burst Recording: ON

Deployment (days): 1

● GMT ○ Local Time

0000/00/00,00:00:00

ADCP SN: NA 0

Troubleshooting: In the case the system does not connect using the known COM port, please try another baud rate. Although, default set-up baud rate is 155200, the system may have been used in another baud rate – cycle through various baud rates in the know COM to connect to the instrument.

2.4 Connection Diagnostics

This procedure presumes the operator has basic knowledge of the RTI ADCP/DVL and has experience with serial communication interfaces and basic test equipment. In the case that the user needs help contact Rowe Technologies directly.

1. Connect DVL communication port(s) to the Windows PC (outlined above).
2. Open the RTI Utilities software. Select communication port and Baud rate.
3. Select the Terminal screen on the Utilities software and turn on the power supply to the ADCP/DVL and click Break <CR>.
4. If you see the Wake Up message on the Terminal screen:

```
Copyright (c) 2009-2014 Rowe Technologies Inc. All rights reserved.
Doppler Velocity Log
DP1200
SN: 01200000000000000000000000000197
FW: 00.02.47 May 27 2014 12:14:45
```

   a. Proceed to the Utilities DVL Setup page and follow the deployment procedure in the RTI Utilities Manual. Note If the DVL was setup to start pinging on power up it will start pinging approximately 10 seconds after power on.

5. If there is no Wakeup message displayed on the Terminal screen:
   a. Check serial communications:

i. Verify the Baud rate and Communication port setting.

ii. Verify the wiring of the communication port.

1. You can verify RS232 communications on the PC by connecting pins 2 and 3 together on the DB9 connector going into the PC. Then send a command like START to the port. You should see the command/characters echoed on the Terminal screen.
2. Try a NULL MODEM on the RS232.
3. Connect an Oscilloscope to the RS485 data lines. Then send a command like START. You should see the command/characters on the scope display the bit rate should match the PC Baud rate.
4. Try swapping the 2 data lines on the RS485.

iii. Connect an oscilloscope to the ADCP/DVL output data lines on either communication port.

1. Turn OFF the ADCP/DVL power. Wait 10 seconds.
2. Turn ON the ADCP/DVL power supply:

iv. Measure the voltage at the input to the DVL 2 pin power connector. The voltage should be between 12 and 36 VDC depending on the attached power supply specs. If the voltage is out of range find another supply.

v. Measure the power supply current. The power up current should be 0.2 to 0.3 Amps if the system is awake and waiting for a command. If the system is pinging it will be drawing 3 to 4 Amps during transmit capacitor charging and 0.2 to 0.3 Amps the rest of the time.

b. Verify LED operation on the Processor board.

i. Remove the underwater housing keeping the wiring connectors attached.

ii. Check for loose/broken parts and that all of circuit boards and cables are seated in their connectors.

iii. Apply power to the ADCP/DVL and observe the electronic circuit board stack. There should be a Red LED that is on constantly and next to it a Red blinking LED. The blink rate should be once per second. Adjacent to the Red LEDs, on another board, a Green LED should be blinking at a ½ second rate. There should also be a Yellow/Amber LED on another board that is on constantly.

1. If the Red LED is on and no other LED is blinking then the system is powered up but did not boot. Two possibilities for not booting are corrupted system files or failed SD card.

a. Remove the SD card and verify the card has the following files:
BOOT.BIN
RTISYS1.BIN
HELP.TXT
BBCODE.BIN
MAINT.TXT
ENGHELP.TXT
SYSCONF.BIN

b. If there are missing or corrupted file(s). Contact RTI to send you a new set of files. You may also want to replace the SD card with a similar type of SD card.

c. Reinstall the SD card and start this procedure over.

2. If the LEDs are lit correctly there must be a problem with data communications wiring or connectors. Refer to the User guide for wiring information then verify with an oscilloscope the Wakeup message is coming out of the serial port wire(s) when power is applied.
3. If no LEDs are lit then measure/verify the voltage at the 2 wire connector on the circuit board stack.

c. If there is still an issue contact RTI with the results of these tests.

2.5 Preparing for a Deployment

When preparing to deploy a RTI ADCP/DVL there are a few important considerations that users need to be careful of that will help ensure a successful deployment. Below is an informal checklist that should be used to prepare the instrument.

Structural Integrity – Self-contained systems are often opened to change battery packs for deployments, as such it is essential that all O-rings are properly greased and seated in the groove found in the end-cap and the transducer head. In addition, it is important to make sure that when reassembling the system that the nuts and bolts are tightened sufficiently. The structural integrity of cables and connectors are also important, be sure to there are no cuts or cracks in the cable or connectors.

Cable Connections – Be certain that all cable connections are sounds as outlined in Section 2.1. When using a wet-mateable connection, be sure to first mate the connectors fully and then to tighten the locking sleeve, as this is the proper procedure for wet-mateable connections (*Do not* connect the wet-mateable connectors partially and then use the locking sleeve to tighten – this will not provide an adequate seal for the connector).

Power – In addition to making certain that the power connections are sound it is important to use the deployment predictor data presented in the software to ensure that battery power will endure the duration of the deployment. Be sure to evaluate deployment settings and predictions to be sure of the power requirements.

Memory – Like the power, please use the deployment predictor data presented in the software to ensure that there will be sufficient memory for the duration of the deployment. If the memory becomes full, data collection will stop and no new data will be collected.

Dummy Plugs – Dummy plugs are provided with every system in the tool bag. Dummy plugs should *always* be used in self-contained deployments to prevent corrosion damage on the wet-mateable connectors.

Chapter 3 Instrument Care

Below are some general guidelines to taking care of ADCPs and DVLs.

3.1 Guidelines to Instrument Care

Your RTI ADCP/DVL is a robust instrument designed for prolonged use. Please consider the following guidelines to prolong the life of your instrument, decrease the risk of damage and continue its factory tested performance:

1. Please do not open the instrument housing or the deck box enclosure unless you have contacted service at RTI. There is no regular field maintenance to perform on these units.
2. Handle with care; dropping the unit or severe impact could damage the transducer elements, the water tight integrity of the instrument housing and the internal electronics of the system.
3. Avoid leaving the instrument and the deck box in direct sun light for prolong periods of time. If they do need to be placed in direct sun light consider a cover. Try not to store or transport these items in extreme temperatures. When the instrument and deck box are not in use please place them back in the original shipping container.
4. Keep the instrument and the deck box clean and clear of dirt, oils and any chemicals. Dirt may contaminate the o-ring seals and the electrical connection of the underwater connector.
5. The “red” transducer face is made of a special urethane. This urethane has acoustic properties necessary for the instrument to perform. This urethane is also softer than the other exterior plastic that makes up the white housing and the black transducer mounting head. The transducer faces must be given extra care to avoid punctures, cuts, direct sun light, any chemical contact, discoloration from oils and contaminates in the water. The transducer face is soft enough to sustain impressions from anything left on it or if it is placed against or on top of anything.
6. Your instrument has an internal magnetic flux gate compass. The compass may not work correctly if the instrument is near a steel structure or magnetic field.
7. Please do not “ping” your instrument in air for any prolong amounts of time, this could cause damage. Please use a container of water for lab testing.

3.2 Warranty Policy

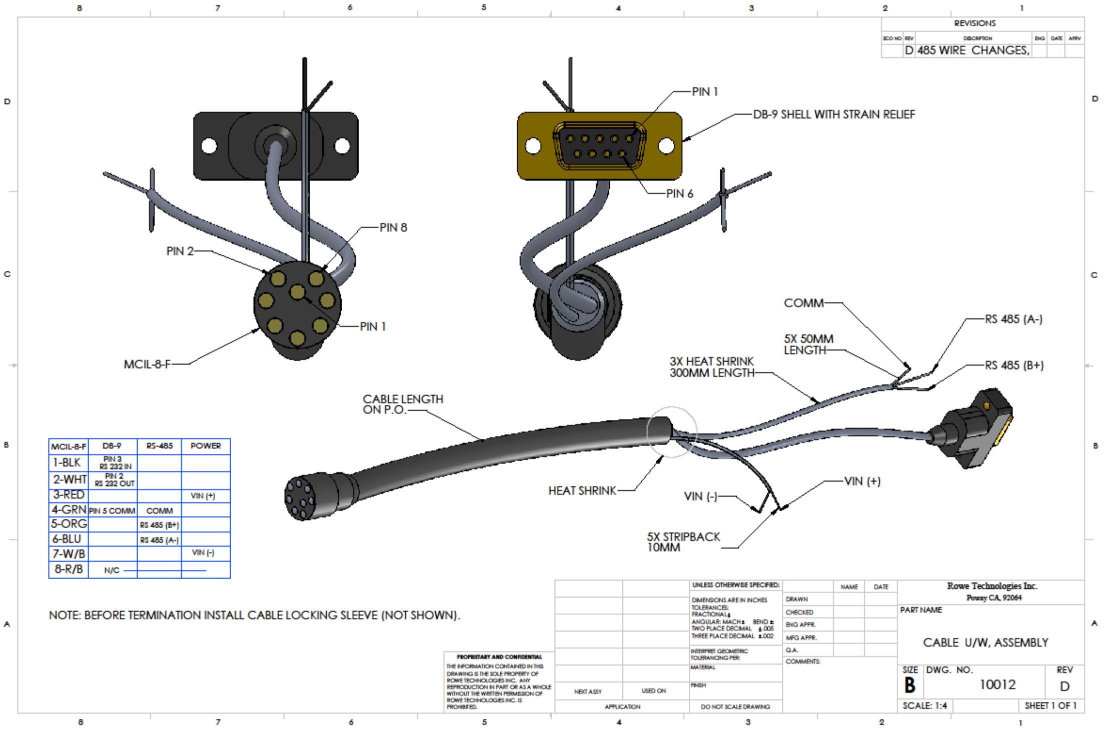
Equipment manufactured by Rowe Technologies Inc, (RTI) is covered under a 12 month, limited warranty, which begins from the date of original shipment. This warranty extends to all parts and labor for any malfunction caused by defects in material or errors in workmanship that occurred during the manufacturing process. Any third party items incorporated into, or included with the equipment will bear the warranty of their manufacturer.

This warranty is based upon proper operation and maintenance of the equipment, as detailed in the User's Guide, and does not apply to goods that have been subject to shipping damage, improper installation, misuse/neglect, alteration, damaged during use, or the like. The warranty does not cover deficiencies with the design of the equipment or any damages that are a result of measurement errors from the equipment.

Upon notification of the nature of the defect, RTI will ask you to either return the equipment to a service facility for repair, or we will ship you replacement parts. If the equipment needs to be returned, RTI will provide you with a return merchandise authorization (RMA) number. The equipment should be shipped in the original packaging, with all delivery costs, including duties, taxes, etc., covered by the customer. RTI reserves the right to refuse receipt of equipment returned without a valid RMA number.

Chapter 4 ADCP/DVL Power and Serial Communications

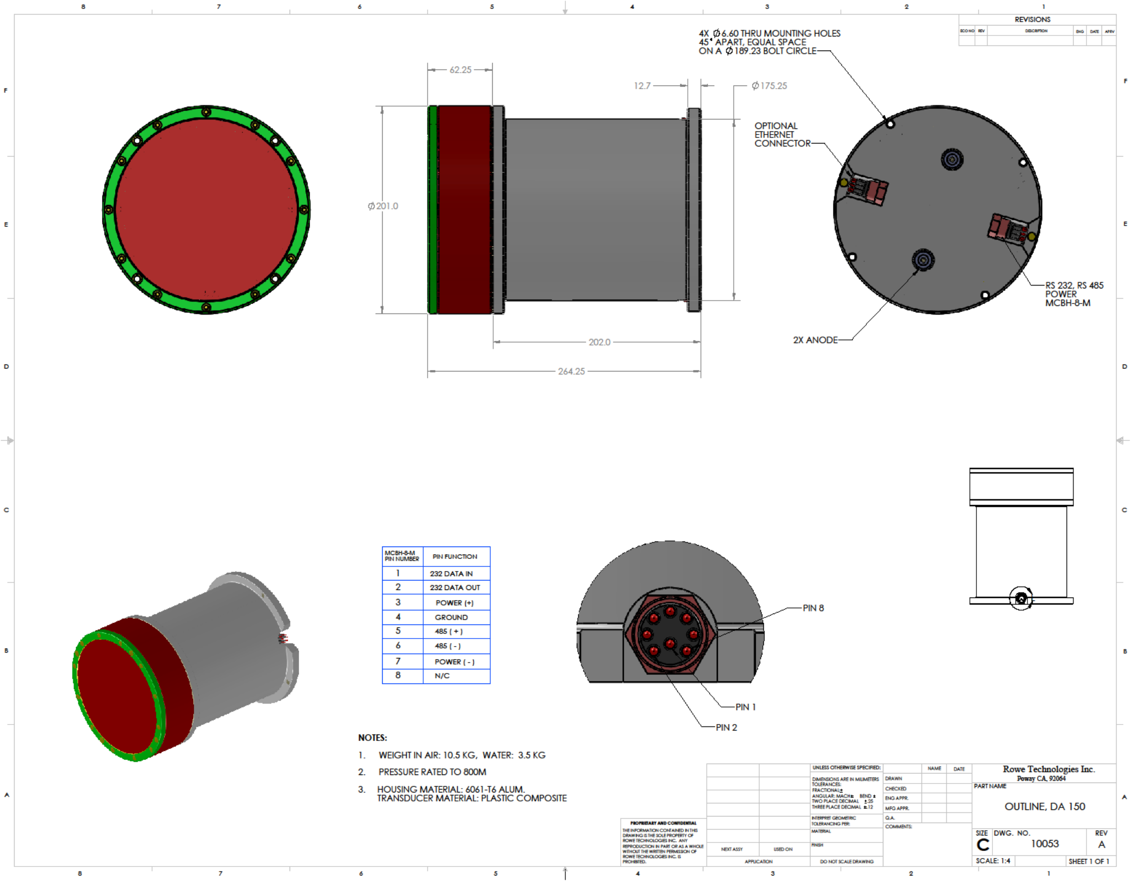
4.1 Pigtail Cable Configuration Diagram



MCIL-8-F	DB-9	RS-485	POWER
1-BLK	PIN 3 RS 232 IN		
2-WHT	PIN 2 RS 232 OUT		
3-RED			VIN (+)
4-GRN	PIN 5 COMM	COMM	
5-ORG		RS 485 (B+)	
6-BLU		RS 485 (A-)	
7-W/B			VIN (-)
8-R/B	N/C		

NOTE: BEFORE TERMINATION INSTALL CABLE LOCKING SLEEVE (NOT SHOWN).

4.2 End Cap Connector



MCBH-8-M PIN NUMBER	PIN FUNCTION
1	232 DATA IN
2	232 DATA OUT
3	POWER (+)
4	GROUND
5	485 (+)
6	485 (-)
7	POWER (-)
8	N/C

NOTES:

1. WEIGHT IN AIR: 10.5 KG, WATER: 3.5 KG
2. PRESSURE RATED TO 800M
3. HOUSING MATERIAL: 6061-T6 ALUM.
   TRANSDUCER MATERIAL: PLASTIC COMPOSITE

4.3 ADCP/DVL Operations with Serial Communications

1. Serial Communications
    a. Electrical Interface
        i. RS-232 (standard)
            1. 8 bit, No Parity, 115200 Baud (can go higher for short distances)
            2. Full Duplex (3 wire, shares return with 485)
            3. Minimal noise immunity
            4. Short distances (100ft)
        ii. RS-485 (standard)
            1. 9 bit, Multi-drop Mode Address/Data Parity, up to 921600 Baud
            2. Half Duplex (3 wire, shares return with 232)
            3. Good noise immunity
            4. Long distances (4000ft)
        iii. RS-422 (precludes 232 and 485)
            1. 8 bit, No Parity, up to 921600 Baud
            2. Full Duplex (requires all 5 communication wires)
            3. Good noise immunity
            4. Long distances (4000ft)
    b. Protocol
        i. ASCII commands are terminated with a carriage return <CR>. Using the example "START<CR>" the firmware inputs each ASCII character one at a time checking for a 0x0d (13) or <CR>. When the <CR> is detected the firmware converts the most resent characters in the input buffer into a string which is then compared to a list of valid commands. The command is echoed (sent back out the data port) to the sender followed by either a 0x06 <ACK> or a 0x15 <NAK>. The <ACK> indicates to the sender that command is valid and will be acted upon. A <NAK> indicates to the sender that the command is not valid and no action will be taken. In the example "START<CR>" the system would recognize the command as valid and echo START<ACK> and then proceed to collect profile data.
2. Windows based PC RS232 Setup
    a. On a Windows based PC install a RS232 to USB convertor.
    b. Note communication port associated with the convertor.
    c. Connect the DB9 RS232 connector to the USB to RS232 convertor.
    d. Connect the 8 pin SEACON MCIL-8-F underwater connector to the ADCP/DVL instrument housing.
    e. Connect the 2 power wires to an appropriate DC power source.
    f. Turn on the DC power source.
3. Windows based PC RS485 Setup
    a. On a Windows based PC install the R485 device driver and the RS485 to USB convertor.
    b. Note communication port associated with the convertor.

   c. Connect the RS485 terminal block to RS485 convertor.
   d. Connect the 8 pin SEACON MCIL-8-F underwater connector to the ADCP/DVL instrument housing.
   e. Connect the 2 power wires to an appropriate DC power source.
   f. Turn on the DC power source.

4. Windows based PC RS485 with Deck box Setup
   a. Install the R485 device driver and the RS485 to USB convertor.
   b. Note communication port associated with the convertor.
   c. Connect the RS485 terminal block to RS485 convertor.
   d. Connect the 4 or 5 pin Switch craft Conxall connector to the RTI deck box.
   e. Connect the 8 pin Switch craft Conxall connector to the RTI deck box.
   f. Connect the 8 pin SEACON MCIL-8-F underwater connector to the DP300 instrument housing.
   g. Connect the 3 pin AC cord to the RTI deck box.
   h. Plug the AC power cord into a conveniently located AC outlet (90 - 230 VAC).
   i. On the RTI deck box turn the power switch located near the AC power cord to ON.
   j. Green LED near the power switch should light up.
   k. The data indicator light on the RTI deck box should illuminate briefly then go out after power is applied to the ADCP.
5. Windows based software setup:
   a. Using the communication port from above refer to the DP-Pro software user guide to enable communications between the ADCP/DVL and the PC serial port.
   b. DP-Pro software can be used to send commands to the DVL.

4.4 Self-Contained ADCP

1. Internal battery pack(s). Internal packs reside within the same pressure housing as the Doppler electronics.
   a. Single 445 W-hr Alkaline pack, RTI PN 10172 Rev E
      i. Refer to Rowe Technologies document number DOC0021 for a description of how to replace/install a single internal battery pack. Also see Appendix D1 through D3.
      ii. After electrically connecting the battery to the system electronics, using the available J7 or J8 connector on the SC wire harness, the system is powered. Additionally, depending on the last state of the ADCP before it was powered down, the ADCP may start pinging. To prevent the battery from discharging prematurely, an external DC supply, such as the 36 VDC unit included with the ADCP, must be connected to an underwater cable connected to the end cap connector.
      iii. The input voltage wire (see Appendix D1 and D2) on the end-cap contains a diode in series with it to prevent DC current flow between an external power supply and the internal battery. The battery pack also has a protection diode

internal to the pack. The DC voltage of a fresh 19 cell pack is approximately 32 Volts.

iv. The battery pack has 5 Amp fuse to protect against sort circuit failures. If you need to change the fuse we recommend that you use:

Littelfuse Inc 0AGW005.V

b. Compass Calibration

i. The battery pack must be degaussed prior to installation.

ii. The internal compass must be recalibrated anytime the internal battery pack is changed.

Chapter 5 User Commands and Data Formats

5.1 User Commands (ASCII)

1. Help Commands
    a. **Help**<CR>, **H**<CR>, or **?**<CR> Returns a list all of the available commands.
2. BREAK
    a. Hardware BREAK
        i. A hardware break can be used on a serial port. A break occurs when the serial port data line(s) is/are held in a non quiescent state for a period of time as long or longer than a single character. The ADCP/DVL hardware detects and wakes up with a break length as short as 10 msec. However, if the system is deployed and set to wake up at a future date/time the break needs to be 0.5 seconds or longer in duration. The long break requirement prevents an accidental break from stopping a deployment. Accidental breaks can occur when the communication cable is disconnected from a battery powered system prior to deploying it underwater. RTI Pulse software plays it safe and uses a 1 second break.
    b. Soft BREAK
        i. **BREAK**<CR> When the ADCP/DVL decode the ASCII command "BREAK" the system stops and outputs the wakeup message.
3. Configuration Commands
    a. Mode:
        i. **CDVL**<CR> Enable DVL Mode. The system, when started, will output NMEA formatted data. Bottom track and a single water tracking cell are supported in this mode.
        ii. **CPROFILE**<CR> Enable Profiling mode. The system, when started, will output binary or text formatted data. Bottom track and multi cell water profiling supported in this mode.
    b. Water Profiling: (used in Profiling mode)

        Note: To control subsystems other than 0 add [c] to the command string, where c is the subsystem number.

        i. **CWPON 1, 2, 3**<CR> Water Profile Pings On. Enables or disables water profile pings.
            1. Enable/Disable
                a. 0 = disable water profiling. When the application requires bottom tracking only disabling profiling allows for more bottom pings per second.
                b. 1 = enable water profiling.
            2. Begin Range
                a. When using multiple sub systems this parameter sets the minimum bottom range at which the profile ping is allowed to

occur. The default is set to 0 (zero) which enables profiling at all ranges beyond zero.

3. End Range
    a. When using multiple sub systems this parameter sets the maximum bottom range at which the profile ping is allowed to occur. The default is set to 0 (zero) which enables profiling at all ranges.

ii. **CWPRT m,n1,n2**<CR>Water Ping Range Tracking. Sets Water ping tracking mode and supporting parameters. When enabled the system locates the highest amplitude signal in the profile amplitude data. The system uses very high resolution sample data to give precise range to the target. The range data is useful for non-directional wave height and bottom depth sounding.
Note: You need to setup the profile to collect more bins than the expected maximum range to the surface.
    1. Mode m
        a. 0 = OFF.
        b. 1 = ON and use n1 and n2 for the first and last bin to look between for a valid amplitude target. Search window is between bins n1 and n2.
        c. 2 = ON and use n1 as a fraction of the range before and after the depth as measured by the pressure sensor. Search window = depth*(1-n1) to depth*(1+n1).
    2. n1, n2 supporting parameters for the tracking mode.

iii. **CWPBB 1, 2, 3, 4, 5**<CR> Water Profile Broad Band. Sets water profile coded pulse transmission and lag.
    1. Transmit Pulse Type and Processing
        a. 0 = Narrowband.
            i. Provides long range profiles at the expense of variance.
            ii. Not recommended for use with bin size less than the default bin size.
        b. 1 = Broadband.
            i. Typically 15% less range than narrow band but has greatly reduced variance (depending on lag length).
            ii. Used in conjunction with CWPBP for small bins.
        c. 2 = Un-coded Broadband (no ambiguity resolver).
            i. Non-coded has slightly higher variance than the coded transmit without the annoying autocorrelation side peaks. Better for small bins.
        d. 3 = Broadband pulse-to-pulse (no ambiguity resolver).
            i. Provides ultra low variance for small bin sizes.
        e. 4 = Non Coded Broadband pulse-to-pulse (no ambiguity resolver).

            i. Provides ultra low variance for small bin sizes.
        f. 5 = Broadband with ambiguity resolver ping.
            i. Used in conjunction with CWPBP averaging.
        g. 6 = Broadband pulse to pulse with pulse to pulse ambiguity resolver ping.
            i. Used in conjunction with CWPAP.
    2. Lag length in vertical meters (m).
        a. Not used with Narrowband.
        b. A longer lag will have lower variance and a lower ambiguity velocity.
    3. Transmit Bandwidth
        a. Not fully implemented
    4. Receive Bandwidth
        a. Not fully implemented
    5. Beam Multiplex
        a. 1 = ping and process each beam one at a time.
        b. 2 = ping and process beam pairs.
        c. 4 = ping and process all four beams together.

iv. **CWPAP 1, 2, 3, 4, 5**<CR> Water Ambiguity Ping (used when CWPBB = 6). Pulse to pulse ping and processing is used for the ambiguity resolver therefore (Blank + Bin Size) < Lag.
    1. 0 to 100 sets the number of pings that will be averaged together.
    2. Lag (meters) sets the length of the lag.
    3. Blank (meters) sets the starting position of the bin.
    4. Bin Size (meters).
    5. Time Between Ambiguity Pings (seconds).

v. **CWPST 1, 2, 3**<CR> Water Profile Screening Thresholds.
    1. Correlation Threshold (0.00 to 1.00).
        a. Used for screening profile beams. A beam with a correlation value less than the threshold will be flagged bad and not included in the bin average. Nominal beam correlation values are dependent on the pulse coding, the number of repeated codes, and whether not the pulse-to-pulse processing is being used. For example:
            i. The pulse-to-pulse nominal correlation is 1.00. A correlation value of 0.50 occurs when the signal is equal to the noise (SNR = 1 or 0dB).
            ii. Broad band correlation is dependent on the number of repeated code sequences in the transmission. If 5 repeats are transmitted the nominal correlation will be 4/5 or 0.80. A correlation value of 0.4, in this case, indicates a signal to noise ratio is 1.

2. Q Velocity Threshold (m/s)
    a. Used for screening transformed profile bins. A bin with a, absolute Q velocity that is higher than the Q threshold will be flagged as bad. Beam coordinate velocity data is not affected.
3. V Velocity Threshold (m/s)
    a. Used for screening transformed profile bins. A bin with a, absolute Vertical velocity that is higher than the V threshold will be flagged as bad. Beam coordinate velocity data is not affected.

vi. **CWPBL n.nn**<CR> Water Profile Blank (meters). n.nn = 0 to 100. Sets the vertical range from the face of the transducer to the first sample of the first bin.

vii. **CWPBS n.nn**<CR> Water Profile Bin Size (meters). n.nn sets the vertical bin size.

viii. **CWPX 1, 2, 3**<CR> Water Profile Transmit (meters).

1. Vertical transmit size (0 to 100m).
    a. A value of 0.00 (default) will cause the system to set transmit to the same length as the bin size.
2. Broadband Transmit Code.
    a. A value of 0 (zero) selects the best match from the default code table. A value of 1 selects single element per lag repeat for lowest transmit power with corresponding reduced maximum range.
3. Beam Enable.
    a. 1111 enables all 4 beams, 0111 disables beam zero and enables beams 1,2,3. 0010 enables beam 2 only.

ix. **CWPBN n**<CR> Water Profile Bin N. n = 0 to 200 sets the number bins that will be processed and output.

x. **CWPP n**<CR> Water Profile Pings. n = 0 to 10,000 sets the number of pings that will be averaged together during the ensemble.

xi. **CWPBP 1, 2**<CR> Water Base Pings (used when CWPBB = 1 or 5). This command should be used when small, non pulse to pulse, bin sizes are required for profiling. The Base Pings are averaged together as a complex number which allows for better correlation screening and averaging.

1. Pings a value of 2 to 100 sets the number of pings that will be averaged together during each CWPP ping. A value of 0 or 1 disables Base Ping averaging.
2. Time Between Base Pings (seconds). Normally it is a small number i.e. 0.01 for the 1200 kHz system.

xii. **CWPTBP n.nn**<CR> Water Profile Time Between Pings. n.nn = 0.00 to 86400.00 seconds (24 hours) sets the time between the last ping, regardless of ping type, and the next profile ping.

c. Ensemble:

   i. **CEI HH:MM:SS.hh**<CR> Ensemble Interval. Sets the time interval that system will output the averaged profile/bottom track data.

      Note: all digits including the space following CEI and the separators must be part of the command or the system will reject it.

   ii. **CBI HH:MM:SS.hh,2,3,4**<CR> Burst Interval. Used when a precise short time interval is required between ensembles followed by a period of sleep.

      1. HH:MM:SS.hh sets the time interval between a series of ensembles.
      2. sets the number of ensembles that are output during each burst. The time between each ensemble is controlled by the CEI command.
      3. Sets the Burst Pair flag. 0 = false, 1 = true; If Burst Pair is set to 1 the next sub system will be interleaved (alternating pings) with the current sub system during the burst. The CBI commands for the pair should be set to the same value(s).
      4. Unused

      Note: Burst sampling typically requires precise timing intervals. If serial data output is enabled during burst sampling the user needs to ensure that each data transfer completes before the next ensemble is output. If the length of the data transfer exceeds the time between samples the data will bottleneck and the desired sample rate will not be maintained. Self-contained ADCP users can disable the serial data output by sending the command **CEOUTPUT 0**<CR>. Of course you need to enable the internal data recording **CERECORD 1**<CR>. During burst sampling, internal data recording only occurs at the end of the burst or when the STOP<CR> command is received by the ADCP. Another option, when real-time data is required, is to increase the data output rate by using the C485B 921600<CR> or C422B 921600<CR> command. Be aware that the RS232 connection may not reliably support the higher data rates. **Note:** Burst pinging recording is limited to 50 MB file size per burst.

   iii. **CEPO cccccccccccccccc**<CR> Ensemble Ping Order. Sets the order in which the various subsystems will be pinged. Note: a space and at least one subsystem code must follow CEPO or the system will reject the command. For example:

      1. CEPO 23<CR> will ping subsystem 2 followed by sub system 3.
      2. CEPO 32<CR> will ping subsystem 3 followed by sub system 2.
      3. CEPO 22<CR> will ping subsystem 2 followed by sub system 2 using a different setup (i.e. bin size).

iv. **CETFP YYYY/MM/DD,HH:mm:SS.hh**<CR> Ensemble Time of First Ping. Sets the time that the system will awaken and start pinging.

> Note: all digits including the space following CETFP and the separators must be part of the command or the system will reject the command.

v. **CERECORD n,m**<CR> n=Ensemble Recording 0=disable, 1=enable. m=SinglePing Recording 0=disable, 1=enable.

1. When ensemble recording is enabled and the ADCP is started (START<CR>) the firmware searches for the next available file number to record to on the SD card. The ensemble file name starts with the letter "A" followed by a 7 digit number and ending with the extension ".ens". For example: The first ensemble file will be named "A0000001.ens". During deployment as each ensemble is completed the data is appended to the current file. The 7 digit number following the "A" is incremented each time the system is (re)started or when the file size exceeds 16Mbytes bytes.
   Note: Internal ensemble data recording during burst sampling only occurs at the end of the burst.

2. When single recording is enabled and the ADCP is started (START<CR>) the firmware searches for the next available file number to record to on the SD card. The single ping file name starts with the letter "S" followed by a 7 digit number and ending with the extension ".ens". For example: The first single ping file will be named "S0000001.ens". During deployment as each ping is completed the data is appended to the current file. The 7 digit number following the "S" is incremented each time the system is (re)started or when the file size exceeds 16Mbytes bytes. Each ping, whether bottom track or profile, is considered to be a single ping.

   Note: No error/ threshold screening or coordinate transformation is performed on the data contained in a single ping file.

vi. **CEOUTPUT n**<CR> Ensemble output type.

1. n = 0 disables serial output. Saves battery energy when recording data to the SD card during a self-contained deployment by reducing extra on time of the system due to data transfer.
2. n = 1 enables the RoweTech binary output data protocol to be sent out the serial port when the system is in "profile" mode. If the system is in

"DVL" mode the $PRTI01 $PRTI02 $PRTI30 $PRTI31 data strings are output.

3. n =2 enables an ASCII text serial output that is dumb terminal compatible to be sent out the serial port when the system is in "profile" mode. If the system is in "DVL" mode the $PRTI01 $PRTI02 $PRTI32 $PRTI33 data strings are output.
4. n = 3 disables all output except for a NMEA status string. Allows the user to verify that the instrument is operating normally while recording data to the internal recorder. Saves power and can improve ping timing.
5. n = 4 enable the special Ocean Server NMEA DVL data output. When CEOUTPUT 4 is selected a second parameter can be sent to select the navigation bin. **CEOUTPUT 4, b**<CR> where b is the profile bin that will be used in the $DVLNAV string.
6. n = 5 If the system is in "DVL" mode the $PRTI03 data string is output.
7. n = 100 selects PD0 binary output. When CEOUTPUT 100 is selected a second parameter can be sent to select the velocity coordinate system. **CEOUTPUT 100, c**<CR> where c = 0 is beam coordinates, c = 1 is instrument (XYZ), c = 2 is Earth (ENU), and c = 3 is Ship (SFM).
8. n = 113 selects PD13 ASCII output.
9. n = 103 selects PD3 binary output. When CEOUTPUT 103 is selected a second parameter can be sent to select the velocity coordinate system. See n = 100.
10. n = 104 selects PD4 binary output. When CEOUTPUT 104 is selected a second parameter can be sent to select the velocity coordinate system. See n = 100.
11. n = 105 selects PD5 binary output. When CEOUTPUT 105 is selected a second parameter can be sent to select the velocity coordinate system. See n = 100.
12. n = 106 selects PD6 binary output.

Important Note: PD output formats are industry standard formats for DVLs and are typically a string of data output that is a subset of the total data available. Below is a summary of the data available in PD specific formats:

> **PD0** - binary output format that includes a header, fixed and variable leader, bottom track, and water profile information. The fixed and variable leader is a recording of time, DVL setup, orientation, heading, pitch, roll, temperature, pressure and self test diagnostic results. The user can select data fields to be output. In the case with RTI instruments users can select the coordinates for the data to be represented.

PD4 - is a binary output format that presents bottom track speed over bottom, speed through water and range to bottom information only.

PD5 – is a superset of PD4 and includes additional information such as, salinity, depth, pitch, roll, heading, and distance made good.

PD6 – is a text-based output format that groups separate sentences containing system attitude data, timing and scaling and speed through water relative to the instrument, vehicle, and earth coordinates. Each data sentence contains a unique starting delimiter and comma delimited fields.

PD13 - is a text output format, like PD6 with the addition of information about range to bottom and raw pressure sensor data.

vii. **C232OUT n**<CR> Same as CEOUTPUT except the data is directed to the RS232 port only.

viii. **C485OUT n**<CR> Same as CEOUTPUT except the data is directed to the RS485 port only.

ix. **C422OUT n**<CR> Same as CEOUTPUT except the data is directed to the RS422 port only.

x. **CUDPOUT n**<CR> Same as CEOUTPUT except the data is directed to the Ethernet UDP port. See the UDPPORT command. Output is limited to the DVL NMEA ASCII strings (non binary data).

d. Bottom Tracking (used in profiling and DVL mode):

Note: To control subsystems other than 0 add [c] to the command string, where c is the subsystem number.

i. **CBTON n**<CR> Bottom Track ON. Enables or disables bottom track pings.

1. n = 0 disable bottom tracking. Allows for more water profile pings per second and saves battery energy during self-contained deployments.
2. n = 1 enable bottom tracking. When enabled a bottom track ping occurs once per ensemble. Or, when profiling is enabled, a bottom track ping occurs at the beginning of the ensemble and then after every 10 profile pings in the ensemble. If there are less than 10 profile pings per ensemble the bottom track ping will only occur once at the beginning of the ensemble.

ii. **CBTBB 1, 2, 3, 4**<CR> Bottom Track BB control.

1. Mode
   a. 0 = Narrowband long range. Maximum velocity 11 m/s 20 degree beam angle, 7 m/s for a 30 degree.

b. 1 = Broadband coded transmit. Maximum velocity 13 m/s 20 degree beam angle, 9 m/s for a 30 degree.

c. 2 = Broadband non-coded transmit. Maximum velocity 22 m/s 20 degree beam angle, 15 m/s for a 30 degree.

d. 3 = NA.

e. 4 = Broadband non-code pulse to pulse. Maximum velocity dependent on lag length.

f. 5 = NA.

g. 6 = NA.

h. 7 = Auto Switch between Mode 0, 2, and 4.

2. Pulse-to-Pulse Lag(m)
   a. Lag length in vertical meters. When enabled bottom track will use pulse-to-pulse transmit and processing at depths less than ½ the lag length. Allows for near bottom ultra low variance velocity measurements.
3. Long Range depth (m).
   a. The range in meters beyond which the bottom track will switch to narrowband long range processing when n = 7.
4. Beam Multiplex
   a. 1 = ping and process each beam one at a time.
   b. 2 = ping and process beam pairs.
   c. 4 = ping and process all four beams together.

iii. **CBTST 1, 2, 3**<CR> Bottom Track Screening Thresholds.

1. Correlation Threshold (0.00 to 1.00).
   a. Used for screening beam data. A beam with a correlation value less than the threshold will be flagged bad and not included in the average. Nominal correlation for bottom tracking is 1.
2. Q Velocity Threshold (m/s)
   a. Used for screening transformed bottom track velocities. An absolute Q velocity that is higher than the Q threshold will be flagged as bad. Beam coordinate velocity data is not affected.
3. V Velocity Threshold (m/s)
   a. Used for screening transformed bottom track velocities. An absolute Vertical velocity that is higher than the V threshold will be flagged as bad. Beam coordinate velocity data is not affected.
   b.

iv. **CBTBL n.nn**<CR> Bottom Track Blank (meters). n.nn = 0 to 10. Sets the vertical distance from the face of the transducer at which the bottom detection algorithm begins searching for the bottom.

v. **CBTMX n.nn**<CR> Bottom Track Max Depth (meters). n = 5 to 10000. Sets the maximum range over which the bottom track algorithm will search for the bottom. A large value will slow acquisition time.

vi. **CBTTBP n.nn**<CR> Bottom Track Time Between Pings. n.nn = 0.00 to 86400.00 seconds (24 hours) sets the time between the last ping, regardless of ping type, and the next bottom ping.

vii. **CBTT a, b, c, d**<CR> Bottom Track Thresholds. Where:

   a. SNR(dB) shallow detection threshold. Lowering the SNR counts "a" and/or "c" will allow to the DVL to detect smaller bottom echo at greater range. The consequence is that DVL may false detect the bottom at the wrong range when the bottom signal is weak.

   b. Depth(m) at which the bottom track switches from using the shallow to the deep SNR. Conditions in shallow water (high backscatter) can be different than deep water so "b" allows for two different SNR settings one for shallow ("a") and one for deep ("c").

   c. SNR(dB) deep detection threshold. Lowering the SNR counts "a" and/or "c" will allow to the DVL to detect smaller bottom echo at greater range. The consequence is that DVL may false detect the bottom at the wrong range when the bottom signal is weak.

   d. Depth(m) at which the bottom track switches from low to high gain receive. The ADCP/DVL has a high power transmitter. In shallow water the bottom echo may saturate the receiver input. While this does not harm the system saturation limits the measurable signal level of the bottom echo which can make it difficult to detect the bottom in a high water backscatter environment. The ADCP/DVL places the receiver in low gain when the depth is below the "d" parameter setting. The change in gain is about 40 dB. If you observe the ADCP/DVL having difficulty detecting the bottom near the "d" setting you may need set "d" to a deeper or more shallow depth depending on the depth where the detection is poor. A good rule to follow is a strong bottom echo requires a larger value in d and a weak bottom echo a smaller value.

viii. **CBTFILT a, b, c, d, e** <CR> Note: The filter will combine Dual Frequency system velocity for output.

   a. Output Low Pass Filter Time Constant (0.00 to 1.00). This parameter controls the bottom track velocity output filter time constant. A value of 0.00 disables the filter and the velocity measurement is passed straight through. A value of 1.00 prevents any new data from entering the filter. A value of 0.80

uses 80% of the previous filter output combined with 20% of the new velocity value. The default value is 0.5.

b. Reference Low Pass Filter Time Constant (0.00 to 1.00). This parameter controls the bottom track velocity reference filter time constant. A value of 0.00 disables the filter and the velocity measurement is passed straight through. A value of 1.00 prevents any new data from entering the filter. A value of 0.80 uses 80% of the previous filter output combined with 20% of the new velocity value. The default value is 0.95.
c. Reference Outlier Count (0 to 9999). When a new velocity measurement is accepted, the internal outlier count variable is reset to this value and the reference filter is set to the current or next valid velocity measurement. The default value is 10.
d. Reference Outlier Threshold (0.00 to 1000 m/s). The reference low pass filter value is compared to the latest velocity measurement and if the absolute value of the difference exceeds the outlier threshold the output filter is not updated and the outlier count is decremented. When the outlier count reaches 0 the reference filter is reset to the current or next valid velocity measurement. The default threshold is 1.0 m/s.
e. Output Coast Count (0 to 9999). If the new velocity value is invalid the internal coast counter is decremented. When the coast counter reaches 0 the output filter is cleared until a new valid velocity measurement occurs. Anytime the velocity measurement is valid the coast counter is reset to the coast count. The default value is 30.

Note:
Low pass filtering helps reduce velocity measurement noise. Longer time constants produce lower velocity noise. The tradeoff being the output velocity is less responsive to changes in velocity (acceleration). Parameter "d" prevents extreme velocity values (outliers) from entering the output filter. However, rapid acceleration can appear as a series of outliers when you really want these velocity values to enter the filter. Parameter "c" is used to detect acceleration (or a series of outliers) and allows the reference filter to catch up by being set to the latest velocity measurement. This occurs when the number of consecutive outliers matches the outlier count parameter. The reference filter will normally be set to a slower time constant than the output filter which makes it better able to detect outliers. Parameter "e" allows the ADCP/DVL to clear all filters when bottom track velocity data is not available for a long period of time. This prevents older velocity data being mixed in with new values. Both filters and the parameters "c" and "d" are updated once per ping which can be as often as 20+ times per second, depending on the range to the bottom and the user selected time between pings

and/or ensembles. Internally there are 6 output and reference filters one each for instrument X,Y,Z and one each for Earth U,V,W.

e. **Water Tracking:** (used in DVL mode)

   Note: To control subsystems other than 0 add [c] to the command string, where c is the subsystem number.

   i. **CWTON n, a, b**<CR> Water Track On.
      1. n = 1 Enables or n = 0 disables water track pings.
      2. a(meters) = minimum range for profiling to occur.
      3. b(meters) = maximum range for profiling to occur.

   ii. **CWTBB n**<CR> Water Track Broadband. Enables or disables water track broadband processing. n = 0 to 1, 0 = disable, 1 = enable.

   iii. **CWTBL n.nn** <CR> Water Track Blank (meters). n.nn = 0 to 100. Sets the vertical range from the face of the transducer to the first sample of the bin.

   iv. **CWTBS n.nn** <CR> Water Track Bin Size (meters). n.nn sets the vertical bin size.

   v. **CWTTBP n.nn** <CR> Water Track Time Between Pings. n.nn = 0.00 to 86400 sets the time between water pings.

f. Environmental (used in all modes):

   i. **CWSSC 1, 2, 3, 4** <CR> Water Speed of Sound Control. 0 = command, 1 = sensor, 2 = internal calculation.
      1. Water Temperature source
      2. Transducer Depth source
      3. Salinity source
      4. Speed of Sound source

   ii. **CWS n.nn**<CR> Water Salinity (ppt). Used in the water speed of sound calculation.

   iii. **CWT n.nn**<CR> Water Temperature (degrees). Used in the water speed of sound calculation if the temperature sensor is not available.

   iv. **CTD n.nn**<CR> Transducer Depth (meters). Used in the water speed of sound calculation.

   v. **CWSS n.nn**<CR> Water Speed of Sound (meters per second). Used when the speed of sound source is set to sensor or command i.e. CWSSCx,x,x,0<CR> or CWSSCx,x,x,1<CR> . During standby and pinging the firmware monitors the communication ports for changes in CWSS and for the NMEA strings: $AML,SVM,SS,SN,nn*cs and/or $DVLSET, SS*cs. These NMEA strings are used on the Ocean Server platform for speed of sound input. When the ADCP/DVL detects a valid speed of sound NMEA string it automatically sets the CWSSC speed of sound source to 0 (command).

   vi. **CHO n.n,m.m,o.o**<CR>
      1. n.n = Heading Offset (+-180 deg) added to the compass or GPS heading prior to being used within the system and then output.
      2. m.m = System to Ship Heading Offset (+-180 deg)

3. o.o = System to PNI Compass Offset (+-180 deg).

vii. **CHS n**<CR> Heading Source. Select the heading source for ENU transformations.

1. n = 0 for no heading
2. n = 1 for internal (PNI) compass
3. n = 2 for GPS HDT string via either serial port.

viii. **CTS n**<CR> Tilt Source. Select the tilt source for ENU transformations.

1. n = 0 for no tilts
2. n = 1 for internal (PNI) compass.

ix. **CVSF n.nn,m.mm**<CR> = Velocity Scale Factor. Scale factor n.nn is applied to all water velocity measurement data. Scale factor m.mm is applied to all bottom velocity measurement data. New Velocity = CVSF * Velocity.

x. **CPZ**<CR> Zero Pressure Sensor. Sets the current pressure reading to the zero point (if pressure sensor is installed).

g. Data Communications:

i. **CEMAC**<CR> temporarily enables Ethernet communication. This command is typically sent via one of the serial ports. The Ethernet port is disabled after power down or sleep. To permanently enable the port a special factory configuration command is required. When the Ethernet port is permanently enabled the system requires an additional 2 seconds after power up to begin accepting commands.

1. One of two responses occurs on the serial port after the command is accepted by the ADCP.

```
CEMAC+
MAC 02:ff:fe:fd:fc:fb
IP  192.168.1.130
Link OK
Speed 1, FullDuplex 1

OR

CEMAC+
MAC 02:ff:fe:fd:fc:fb
IP  192.168.1.130
No Link
```

ii. **IP** n.n.n.n<CR> sets a new Ethernet address in the ADCP. The ADCP response to the IP command will be the same as CEMAC. When sending IP<CR> with no address the system will return the current MAC and IP address.

```
IP+
MAC 02:ff:fe:fd:fc:fb
```

```
IP  192.168.1.130
```

iii. **UDPPORT n**<CR> sets the port number to which the system will send UDP data. The default port is 257.

iv. **C232B 1,2,3,4**<CR> RS232 Serial data control.

v. **C485B 1,2,3,4** <CR> RS485 Serial data control.

vi. **C422B 1,2,3,4** <CR> RS422 Serial data control.

1. Baud Rate (bits per second)
    a. 2400, 4800, 9600, 19200, 38400, 57600, 115200, 230400,460800, 921600.
2. Number of bits 7 or 8.
3. Parity.
    a. 0 = None
    b. 1 = Odd
    c. 2 = Even
4. Stop Bits
    a. 1 = one
    b. 2 = two

vii. **CTRIG n**<CR> External Trigger. Selects which state the external hardware trigger needs to be before pinging. There are 2 types of trigger logic available Edge and Level. Edge requires the trigger line to change state before the ping occurs. For reliable edge detection the minimum width of a pulse should be >= 50 usec. Level just needs the trigger line to be either high or low. There is a 1.4 msec delay before the ping occurs after detection of the trigger.

1. n =1 High level
2. n=2 Low level
3. n=3 Low to high
4. n= 4 High to low
5. n=0 default disabled.

NOTE: The Low Power Regulator Board J7 Pins 2 and 3 on must be jumpered to enable the input trigger. The input trigger line has minimal protection so you need to be careful to not exceed +5.3 and -0.3 Vdc. The threshold for logic high is 3.34 Vdc.

h. System Configuration:

i. **CLOAD**<CR> Loads the file “SYSCONF.BIN”, contained on the SD card, into the system configuration.

ii. **CSAVE**<CR>= Saves a copy of the system configuration to the file “SYSCONF.BIN” on the SD card.

iii. **CSHOW**<CR> Causes the system to display/output the system configuration

```
CSHOW<CR>
```

```
CSHOW
DP1200 System Configuration:
  Mode Profile
  Sim 0
  CEI 00:00:01.00
  CEPO 22
  CETFP 0000/00/00,00:00:00.00
  CERECORD 0
  CEOUTPUT 1
  CWPON[0] 1 [1] 1
  CWPBB[0] 3,0.300 [1] 1,0.050
  CWPST[0] 0.400,1.000,1.000 [1] 0.400,1.000,1.000
  CWPBL[0] 0.10 [1] 0.10
  CWPBS[0] 0.04 [1] 1.00
  CWPX [0] 0.00 [1] 0.00
  CWPBN[1] 20 [1] 40
  CWPP[0] 1 [1] 1
  CWPBP[0] 10,0.00 [1] 1,0.02
  CWPTBP[0] 0.13 [1] 0.13
  CBTON[0] 1 [1] 0
  CBTBB[0] 0, 0.000, 30.00 [1] 0, 0.000, 0.00
  CBTST[0] 0.900,1.000,1.000 [1] 0.000,0.000,0.000
  CBTBL[0] 0.05 [1] 0.00
  CBTMX[0] 75.00 [1] 0.00
  CBTTBP[0] 0.05 [1] 0.00
  CBTT[0] 15.0,25.0,5.0,2.0 [1] 0.0,0.0,0.0,0.0
  CWTON[0] 0 [1] 0
  CWTBB[0] 0 [1] 0
  CWTBL[0] 2.00 [1] 0.00
  CWTBS[0] 2.00 [1] 0.00
  CWTTBP[0] 0.13 [1] 0.00
  CWS 0.00
  CWT 15.00
  CTD 0.00
  CWSS 1500.00
  CHO 0.00
  CHS 1
  CVSF 1.000
  C232B 115200
  C485B 115200
  C422B 115200
```

iv. **CDEFAULT**<CR> = Restores the system configuration to factory defaults

Note: Actual default values will vary depending on system type.

Note: Default values are set for all hardware configured subsystems. When using the CEPO command defaults for repeated subsystems are set to zero (0). You need to set all of the commands when using a single subsystem for multiple setups.

4. Firmware Commands
   a. Information:
      i. **FMSHOW**<CR> Show Firmware Version and creation date.

```
FMSHOW
Copyright (c) 2009-2012 Rowe Technologies Inc. All rights reserved.
System Firmware Version: 00.02.05 Mar 21 2012 06:44:56
```

b. Update: (required in firmware versions 0.0.xx and 0.4.xx. Version 0.2.xx and 0.3.xx boot from the SD card)
    i. **FMCopyB**<CR> Copy the boot firmware “rtiboota.bin” from the secure digital storage to internal Flash.
    ii. **FMCopyS**<CR> Copy the system firmware “rtisys.bin” from the secure digital storage to internal Flash.

5. System Deployment Commands:
    a. System Control:
        i. **START**<CR> Start pinging continuously. Once started the system will only respond the STOP, STIME, and CSHOW commands.
        ii. **STARTHPR**<CR> Start outputting the PNI compass data using the $PRTI34 string. This command will STOP the system from pinging. Sending a START command to the instrument will halt the output of the $PRTI34 string and normal pinging will begin. Sending a STOP command to the instrument will halt the output of HPR data.
        iii. **STARTME**<CR>Start Manual Ensemble pinging for 1 ensemble at the end of ensemble the system will output the data and go to sleep.
        iv. **STOP**<CR> Stop pinging and return to the command input mode.
        v. **SLEEP**<CR> Power down system.
        vi. **SLEEPSECONDS**<CR> Display the sleep countdown time in seconds.
        vii. **SLEEPSECONDS n**<CR> n = the countdown time until the system automatically goes to sleep. The purpose of the sleep countdown is to prevent needless power consumption when a system is stopped (not pinging).
            1. The automatic sleep countdown begins when sleep seconds is set to a value greater than 4 seconds and the system has received a valid STOP<CR> command.
            2. Each time a valid command is received by the system the countdown starts over.
            3. When power is applied to a previously stopped system and SLEEPSECONDS is set to a value greater than 4, the countdown begins immediately.
            4. A value of zero seconds disables the countdown timer.
            5. The default value is 60 seconds.
            6. You must send CSAVE<CR> to the system before the countdown timer reaches zero or any unsaved parameters will be lost when the system goes to sleep.
    b. Geo Position:
        i. **SPOS**<CR> Display system geo position
        ii. **SPOS** Lat(deg),Lon(deg),Depth(m)<CR> Set system geo position and water depth
    c. Real Time Clock:
        i. **STIME**<CR> Display system time.
        ii. **STIME YYYY/MM/DD,HH:MM:SS**<CR> Set system Time.

            Note: all digits including the space following STIME and the separators must be part of the command or the system will reject it.
6. Data Storage Commands:
    a. File Transfer:

i. **DSXRfilename.abc**<CR> Data Storage XMODEM Receive (upload). This command is used to transfer a file, via the serial communication link, from an external device to the SD card contained within the RTI system. File names are limited to a maximum of 8 characters before the extension.

ii. **DSXTfilename.abc**<CR> Data Storage XMODEM Transmit (download). This command is used to transfer a file, via the serial communication link, from the SD card contained within the RTI system to an external device. File names are limited to a maximum of 8 characters before the extension.

b. Secure Digital:

i. **DSFORMAT**<CR> Data Storage Format. Stores the system files in FLASH then completely erases the SD card after which rewrites the system files to the SD card.

ii. **DSDIR**<CR> Data Storage Directory. Show a directory of stored files.

iii. **DSSHOW**<CR> Data Storage Show. Shows SD card capacity and usage.

7. Diagnostic Commands:

a. Compass:

i. **DIAGCPT**<CR> Diagnostic Compass Pass Thru. Allows an external device to connect directly to the internal compass via a serial communication link. To disconnect the pass-thru the external device must send 16 consecutive X's (XXXXXXXXXXXXXXXX) to the system. The pass thru allows external software to directly access and calibrate the compass. The compass Baud rate is 38400 and should never be changed. The ADCP default Baud rate is 115200 which can be changed if desired.

ii. **DIAGPRESSURE**<CR> Shows the Keller30 pressure sensor info. The first part of the response shows the serial number and the version of firmware. The next portion of the response contains the pressure sensor "Coefficients". The third line of which indicates the maximum pressure the sensor can measure. The last portion of the response contains the pressure sensor "Configuration". These values can be programmed using Keller software when the pressure sensor is disconnected from the ADCP.

```
DIAGPRESSURE+

PS SN: 157896
Address: 250
Hardware Ver: 5.20
Firmware Ver: 5.50
Buffer Size: 10
Status: 1

Keller30 Read Coefficients:
64, offset   -1.01360321, bar,  0
80, min       0.20000000, bar,  0
81, max       3.00000000, bar,  0
84, min     -10.00000000, C,  0
85, max      80.00000000, C,  0
```

```
Keller30 Read Configuration:
Nr, Conf, err
 0,  02,  0
 1,  10,  0
 2,  00,  0
 3,  02,  0
 4,  05,  0
 5,  00,  0
 6,  02,  0
 7,  35,  0
 8,  13,  0
 9,  00,  0
10,  00,  0
11,  35,  0
12,  00,  0
13,  FF,  0
```

iii. **DIAGRUB n**<CR> shows the results of the beam continuity test where the user rubs each beam in sequence to determine whether the transducer cup is functional. The test collects statistics for 10 samples then prompts the user to rub the selected beam. PASS will be displayed when the test detects the correct amplitude change associated with rubbing a transducer. Up to 4 beams will be tested per test. For systems with additional beams use the CEPO command to select additional sub systems. To test a single beam send DIAGRUB n where n is the beam number to be tested. You must use the CEPO command to select additional sub systems.

```
DIAGRUB+
DP600 Rub Test:
 10  9  8  7  6  5  4  3  2  1 Vigorously Rub Beam 0 ...........
Beam 0 PASS 58 - 41 >= 3
 10  9  8  7  6  5  4  3  2  1 Vigorously Rub Beam 1 ..
Beam 1 PASS 46 - 43 >= 3
 10  9  8  7  6  5  4  3  2  1 Vigorously Rub Beam 2 ...
Beam 2 PASS 64 - 42 >= 3
 10  9  8  7  6  5  4  3  2  1 Vigorously Rub Beam 3 ..
Beam 3 PASS 62 - 43 >= 3
DP600 Rub Test Complete
```

iv. **DIAGBOOST**<CR> tests the transmit boost circuitry for proper operation. The maximum charge voltage is displayed after which the storage capacitors are discharged to less than 2 volts. The time constant to charge the capacitors is measured and displayed followed by the system being pinged using a system specific length transmit pulse to determine whether the capacitive storage is adequate. Failure occurs if the voltage drops below 3dB of the maximum Voltage.

```
DIAGBOOST+
Boost Test:
        Positve Negative
max(V)  +23.07   -22.76 .......
min(V)   +1.86    -1.70 PASS
TC(s)     0.04     0.03
Xmt(V)  +11.37   -11.06 FAIL
```

v. **DIAGSD**<CR>shows the Secure Digital memory card manufacturer data.

```
DIAGSD+
SD Test:
Manufacturer ID: 3
OEM/Application ID: SDS
Product name: SS08G
Product revision: 8.0
Product serial number: 1345936820
Manufacturing date: 0E7 July 2014
CRC7 checksum: 60
```

vi. **DIAGPNI n**<CR> Shows compass/tilt sensor data.

1. n = 0 shows the current PNI configuration:

```
DIAGPNI 0+
Type: PRME,Revision: 0109
Declination: 0.000
True North: FALSE
Big Endian: TRUE
Mounting Ref: Standard -180
User Cal Stable Check: TRUE
User Cal Num Points: 0
User Cal Auto Sampling: FALSE
Baud Rate: 38400
Polling Mode: FALSE
Flush Filter: FALSE
Sensor Acq Time: 0.00
Interval Rsp Time: 0.00
Filter: Taps = 4, 0.0467086577, 0.4532913423, 0.4532913423,
0.0467086577
```

2. n = 1 shows the current Heading/Pitch/Roll values:

```
DIAGPNI 1+
H= 90.37, P= -0.17, R=  0.14
```

3. n = 2 continuously shows the current Heading/Pitch/Roll values:

```
H= 90.37, P= -0.17, R=  0.14
```

vii. **DIAGSAMP**<CR> Diagnostic Ping. See an example of this command in the System Bench test procedure.

viii. **DIAGSPECTRUM n**<CR> shows a spectrum analyzer type output of frequency vs. amplitude for each of up to 4 beams. The n sets the number of samples averaged together for each data point. If n is not set a default value of 2000 samples is used. In the example below there is an interference term at 622562.48 Hertz on beam 0. This command is used when the system amplitude or correlation data is higher than expected. Typically the user will capture this data and send a copy to Rowe customer service. For systems with more than 4 beams the CEPO command must be used to select the desires sub system.

```
DIAGSPECTRUM+
Noise Spectrum Test, fo = 622562.48, fs = 51880.21, fc
= 622.56, steps = 84, samples = 2000
step,        Hz,   bm0,   bm1,   bm2,   bm3
   0, 596414.86,  8.21,  6.99, 13.47, -3.01
   1, 597037.42, -5.99, -8.29, -5.27, -8.25
   2, 597659.98, -4.91, -6.82, -3.62, -7.36
   3, 598282.55, -2.56, -5.40, -2.26, -5.12
   4, 598905.11, -0.65, -3.67, -0.45, -4.36
   5, 599527.67,  3.11, -2.18,  3.66, -1.30
   6, 600150.23,  8.45, -0.90, 11.70, -0.45
   7, 600772.80,  5.05, -0.13,  7.41,  1.53
   8, 601395.36,  3.24,  1.23,  6.29,  1.12
   9, 602017.92,  2.85,  2.14,  6.56,  1.93
  10, 602640.48,  3.82,  2.25,  6.40,  2.37
  11, 603263.05,  3.75,  4.03,  5.46,  3.11
  12, 603885.61,  4.05,  3.55,  6.21,  5.48
  13, 604508.17,  3.74,  4.06,  6.09,  2.54
  14, 605130.73,  4.97,  4.43,  9.01,  4.65
  15, 605753.30,  5.41,  4.79,  7.41,  3.73
  16, 606375.86,  4.58,  4.66,  8.26,  5.33
  17, 606998.42,  6.15,  4.14,  8.18,  4.46
  18, 607620.98,  5.84,  5.70,  8.05,  4.08
  19, 608243.55,  4.11,  5.39,  6.67,  3.40
  20, 608866.11,  4.75,  4.78,  6.25,  4.75
  21, 609488.67,  4.00,  4.27,  6.95,  4.51
  22, 610111.23,  5.01,  3.88,  6.03,  3.35
  23, 610733.80,  3.90,  3.72,  7.08,  3.81
  24, 611356.36,  3.67,  4.65,  5.78,  4.11
  25, 611978.92,  5.76,  4.75,  7.07,  3.95
  26, 612601.48,  3.53,  3.45,  5.65,  2.98
  27, 613224.05,  4.90,  4.05,  7.26,  3.96
  28, 613846.61,  4.30,  4.63,  7.34,  4.97
  29, 614469.17,  4.87,  4.39,  7.43,  6.08
  30, 615091.73,  5.74,  5.43,  6.78,  3.74
  31, 615714.30,  4.70,  5.07,  6.90,  4.05
  32, 616336.86,  6.66,  6.14,  6.75,  5.66
  33, 616959.42,  5.04,  4.40,  6.55,  5.43
  34, 617581.98,  5.38,  4.73,  7.06,  4.74
  35, 618204.55,  5.26,  4.56,  7.08,  4.81
  36, 618827.11,  4.74,  4.12,  7.25,  5.18
  37, 619449.67,  5.99,  4.91,  7.35,  5.58
  38, 620072.23,  4.85,  5.13,  7.21,  4.31
```

```
39, 620694.80,  5.53,  3.83,  7.48,  5.91
40, 621317.36,  6.22,  4.55,  6.24,  4.60
41, 621939.92,  6.78,  5.30,  6.85,  4.32
42, 622562.48, 11.57,  4.64,  8.73,  5.55
43, 623185.05,  6.18,  4.28,  6.73,  4.50
44, 623807.61,  4.63,  4.33,  7.14,  5.33
45, 624430.17,  5.38,  4.95,  7.05,  4.52
46, 625052.73,  3.76,  4.54,  6.39,  5.01
47, 625675.30,  4.94,  5.09,  7.19,  5.46
48, 626297.86,  4.62,  4.25,  6.50,  3.98
49, 626920.42,  5.41,  4.69,  6.93,  4.81
50, 627542.98,  5.81,  4.50,  6.06,  5.46
51, 628165.55,  6.84,  4.98,  6.63,  4.68
52, 628788.11,  4.59,  5.21,  7.11,  4.74
53, 629410.67,  4.65,  4.56,  6.82,  5.48
54, 630033.23,  4.75,  4.33,  6.79,  4.40
55, 630655.80,  4.68,  4.70,  6.23,  5.33
56, 631278.36,  4.54,  5.12,  5.94,  4.32
57, 631900.92,  4.95,  3.58,  5.58,  4.15
58, 632523.48,  5.83,  3.76,  4.86,  5.13
59, 633146.05,  5.10,  4.38,  5.83,  4.03
60, 633768.61,  4.37,  3.98,  5.27,  3.17
61, 634391.17,  5.47,  4.14,  5.94,  3.91
62, 635013.73,  4.19,  4.78,  5.45,  3.84
63, 635636.30,  5.20,  3.90,  4.70,  4.71
64, 636258.86,  4.69,  5.72,  5.19,  4.38
65, 636881.42,  4.56,  5.08,  5.73,  4.15
66, 637503.98,  4.44,  4.46,  5.10,  5.14
67, 638126.55,  5.69,  4.40,  5.85,  5.44
68, 638749.11,  4.66,  4.88,  6.36,  5.72
69, 639371.67,  5.26,  4.53,  7.72,  4.46
70, 639994.23,  4.85,  4.37,  8.85,  4.99
71, 640616.80,  4.44,  4.53,  6.08,  4.18
72, 641239.36,  5.69,  3.90,  5.95,  2.77
73, 641861.92,  2.98,  4.09,  4.38,  3.84
74, 642484.48,  3.60,  3.95,  4.90,  3.75
75, 643107.05,  2.68,  3.16,  4.18,  1.95
76, 643729.61,  1.41,  0.57,  2.81,  1.67
77, 644352.17,  0.55, -0.02,  1.87,  0.30
78, 644974.73, -1.39, -0.75, -0.19, -1.14
79, 645597.30, -2.26, -2.66, -0.55, -3.13
80, 646219.86, -3.48, -4.56, -1.99, -4.13
81, 646842.42, -5.65, -5.78, -3.30, -5.36
82, 647464.98, -6.09, -7.18, -4.62, -6.76
83, 648087.55, -7.34, -8.19, -5.50, -7.95
```

ix. **DIAGRTC n**<CR> shows the Real Time Clock registers, time, and temperature.

1. n = 0 or 1 for single output.
2. n = 2 for continuous output.

DIAGRTC +

```
time: 15/02/17 2 07:50:26
alm1: 80 10 40 05
alm2: 00 00 00
cntr: 01h
cntr: 00h
stat: 00h
temperature: 22.500 C
```

x. **DIAGRCV n**<CR> shows the contents of the receiver registers and verifies communication with the high speed SPI bus.

1. n = 0 or 1 runs the test once and displays the results.

```
DIAGRCV 0+
RVCR,01
01 40 01 06 00 FF 0F A0 7F 00 00 80 20 00 40 00 60 00 50 07 03 00 10 00 00 27 27
Receiver data transfer test = PASS
```

2. n = 2 runs the test continuously and displays the accumulated results.

```
DIAGRCV 0+
RVCR,01
01 40 01 06 00 FF 0F A0 7F 00 00 80 20 00 40 00 60 00 50 07 03 00 10 00 00 27 27
Receiver data transfer test pass:fail:timeout 21:0:0
```

xi. **DIAGXMT n**<CR> shows the results of the transmit/receive test.

1. n = 0 or 1 performs low gain, high gain, transmit test. Boost voltage and input voltage are also checked. Up to 4 beams are checked at a time. You need to set the CEPO command to select additional sub systems. LG stands for Low Gain, HG for High Gain, XM for transmit, VIN for Voltage Input, BP for Boost Positive, and BN for Boost Negative. RefL and RefH are the values to which the measure values are compared. The unit for the LG, HG, XM are dB and VIN, BP, BN are Volts.

```
DIAGXMT 0+
DP600 Transmit Test:
  beam    LG    HG    XM    RG   VIN    BP    BN
     0  34.0  35.8  87.3  87.5  19.4  23.2 -22.7 PASS
     1  33.0  37.1  85.6  87.6  19.4  23.1 -22.8 PASS
     2  33.3  36.5  87.4  87.6  18.8  23.1 -22.8 PASS
     3  33.3  38.4  84.5  84.5  18.6  23.0 -22.8 PASS
RefL    20.0  30.0  75.0  75.0  11.0  20.0 -19.0
RefH    40.0  50.0  95.0  95.0  40.0  25.0 -24.0
```

2. n = 2 performs all of the tests continuously and accumulates the results.

```
DIAGXMT 0+
DP600 Transmit Test:
 beam   Pass   Fail Timeout
    0     11      0       0
    1     11      0       0
    2     11      0       0
    3     11      0       0
```

5.2 User Command Examples

Example 1: The following is optimized for a 150 kHz DVL. This list of commands will instruct a RTI DVL to collect a broadband bottom track and 16 meter broadband water bin.

1. CDEFAULT (Clear out all previously entered commands)
2. CDVL (Place the system in the DVL mode)
3. CWTON1 (Turns on the water track bin)
4. CWTBS16 (Selects a 16 meter bin for water track)
5. CWTTBP 1 (Sets 1 second between water track pings)
6. CWTBB 1 (Selects Broadband water track processing)
7. CBTON 1 (Turns Bottom track on)
8. CBTBB 1 (Selects Broadband bottom track processing)
9. CSAVE (Save the command setup to the SD card for power up reset)
10. START (Begin pinging and output NMEA data to serial port)

Example 2: The following is optimized for a 300 kHz self-contained ADCP. The following list of commands will instruct a RTI ADCP to collect a 120 meter Broadband profile with bottom track disabled.

1. CDEFAULT (Clears out all previously entered commands)
2. CERECORD 1 (turn on internal SD recording)
3. CEOUTPUT 0 (turn off serial communication ensemble output)
4. CPROFILE (Places the DP300 in the profile mode)
5. CWPBS 4 (Sets the bin size to 4 meters)
6. CWPBN 30 (Selects 30 bins for processing and output)
7. CEI 01:00:00.00 (Sets a 1 hour averaging interval)
8. CWPTBP 720 (Sets 12 minutes between profile pings)
9. CWPBB 1, 1.0 (Selects Broadband profile processing with a 1 meter lag)
10. CWPP 5 (Sets a 5 pings per ensemble)
11. CBTON 0 (Turns Bottom track off)
12. CSAVE (Save the command setup to the SD card for power up reset)
13. START(Begin pinging and output Binary data to serial port)

Example 3: The following is optimized for a dual frequency 300/1200 kHz 20 degree piston ADCP. The following list of commands will instruct the 300 kHz RTI ADCP to collect a 120 meter Narrowband profile with bottom track enabled and the 1200 kHz system to collect a 5 meter Broadband profile. The system SN is 01420000000000000000000000000015

1. CDEFAULT (Clears out all previously entered commands)
2. CPROFILE (Places the DP300/1200 in the profile mode)
3. CEI 00:15:00.00 (Sets a 15 minute ensemble output rate)
4. CEPO 10 (select subsystem 1 to ping first followed by 0)
5. CWPBS[0] 4 (Sets the bin size to 4 meters)
6. CWPBN[0] 30 (Selects 30 bins for processing and output)
7. CWPP[0] 10 (Sets a 10 profile pings per ensemble)
8. CWPTBP[0] 0.5 (Sets 0.5 second between profile pings)
9. CWPBB[0] 0,X (Selects Narrowband profile processing)

10. CBTON[0] 1 (Turns Bottom track on)
11. CBTBB[0] 1 (Selects Broadband bottom track processing)
12. CWPBS[1] 0.5 (Sets the bin size to 0.5 meters)
13. CWPBN[1] 10 (Selects 10 bins for processing and output)
14. CWPP[1] 10 (Sets a 10 ping per ensemble)
15. CWPTBP[1] 0.25 (Sets 0.25 second between profile pings)
16. CWPBB[1] 1,0.1 (Selects Broadband profile processing with a 0.1 meter lag)
17. CBTON[1] 0 (Turns Bottom track off)
18. CSAVE (Save the command setup to the SD card for power up reset)
19. START(Begin pinging and output Binary data to serial port)

Example 4: The following is optimized for a 600 kHz self-contained ADCP. The following list of commands will instruct a RTI ADCP to collect a 2 bins of Broadband velocity data with bottom track disabled.

1. CDEFAULT (Clears out all previously entered commands)
2. CERECORD 1 (turn on internal SD recording)
3. CEOUTPUT 0 (turn off serial communication ensemble output)
4. CPROFILE (Places the DP600 in the profile mode)
5. CWPBS 0.5 (Sets the bin size to 0.5 meter)
6. CWPBN 2 (Selects 2 bins for processing and output)
7. CEI 00:00:00.25 (Sets a 0.25 second ensemble interval)
8. CBI 00:15:00.00,1024 (Sets a 15 minute burst interval containing 1024 ensembles)
9. CWPP 1 (Sets 1 profile ping per ensemble)
10. CWPBB 1,0.25 (Selects Broadband profile processing with a 0.25 meter lag)
11. CBTON 0 (Turns Bottom track off)
12. CSAVE (Save the command setup to the SD card for power up reset)
13. START(Begin pinging and save data to the SD card)

5.3 NMEA Data

1. The standard RTI DVL output is a NMEA type string that starts with a $ and ends with a linefeed. The data strings are RS232/RS485 serial ASCII at 115200 Baud, 8 bit, No Parity.
2. NMEA strings have the following format:
    a. Leading dollar sign ($).
    b. Identity string, followed by a comma (,).
    c. Data fields delimited by commas (final data field has no trailing comma).
    d. An asterisk (*).
    e. Two digit hexadecimal checksum. Checksum is computed from all characters between the leading $ and trailing * characters. Value of the checksum is the exclusive-or (XOR) of those characters.
    f. Carriage return followed by a line feed <CR><LF>
3. The RTI proprietary strings when CEOUTPUT is set to 1:

```
$PRTI01,sssssshh,nnnnnn,TTTT,XXXX,YYYY,ZZZZ,DDDD,xxxx,yyyy,zzzz,dddd,ABCD,S,N*FF<CR><LF>
Where:
PRTI01 = Identity string.
sssssshh = Start time of this sample in hundreds of seconds since power up or user reset.
nnnnnn = Sample number
TTTT = Temperature in hundreds of degrees Celsius. Measured at the transducer
XXXX = Bottom track X velocity component mm/s.    -99999 indicates no valid velocity
YYYY = Bottom track Y velocity component mm/s.    -99999 indicates no valid velocity
ZZZZ = Bottom track Z velocity component mm/s.    -99999 indicates no valid velocity
DDDD = Depth below transducer in mm.              Range to the bottom, 0 = no detection
xxxx = Water mass X velocity component mm/s.      -99999 indicates no valid velocity
yyyy = Water mass Y velocity component mm/s.      -99999 indicates no valid velocity
zzzz = Water mass Z velocity component mm/s.      -99999 indicates no valid velocity
dddd = Depth of water mass measurement in mm.     Position of bin in front of the transducer
ABCD = Built in test and status bits in hexadecimal0000 = OK.
S    = Sub System. See appendix B5 for description.
N    = Sub System index.

$PRTI02,sssssshh,nnnnnn,TTTT,EEEE,NNNN,UUUU,DDDD,eeee,nnnn,uuuu,dddd,ABCD,S,N*FF<CR><LF>
Where:
PRTI02 = Identity string.
sssssshh = Start time of this sample in hundreds of seconds since power up or user reset.
nnnnnn = Sample number
TTTT = Temperature in hundreds of degrees Celsius. Measured at the transducer
EEEE = Bottom track East velocity component mm/s.  -99999 indicates no valid velocity
NNNN = Bottom track North velocity component mm/s. -99999 indicates no valid velocity
UUUU = Bottom track Up velocity component mm/s.    -99999 indicates no valid velocity
DDDD = Depth below transducer in mm.               Range to the bottom, 0 = no detection
eeee = Water mass East velocity component mm/s.    -99999 indicates no valid velocity
nnnn = Water mass North velocity component mm/s.   -99999 indicates no valid velocity
uuuu = Water mass UP velocity component mm/s.      -99999 indicates no valid velocity
dddd = Depth of water mass measurement in mm.      Position of bin in front of the transducer
ABCD = Built in test and status bits in hexadecimal 0000 = OK. See appendix B6 for description.
S    = Sub System. See appendix B5 for description.
N    = Sub System index.


$PRTI30,h.hhh,p.ppp,r.rrr,S,N*15
Where:
PRTI30 = Identity string.
h.hhh = Heading used during the bottom track ping.
p.ppp = Pitch used during the bottom track ping.
r.rrr = Roll used during the bottom track ping.
S    = Sub System. See appendix B5 for description.
N    = Sub System index.


$PRTI31,h.hhh,p.ppp,r.rrr,S,N*15
Where:
PRTI31 = Identity string.
```

```
h.hhh = Heading used during the water mass ping.
p.ppp = Pitch used during the water mass ping.
r.rrr = Roll used during the water mass ping.
S    = Sub System. See appendix B5 for description.
N    = Sub System index.
```

4. The RTI proprietary strings when CEOUTPUT is set to 2:

```
$PRTI01,sssssshh,nnnnnn,TTTT,XXXX,YYYY,ZZZZ,DDDD,xxxx,yyyy,zzzz,dddd,ABCD,S,N*FF<CR><LF>
Where:
PRTI01 = Identity string.
sssssshh = Start time of this sample in hundreds of seconds since power up or user reset.
nnnnnn = Sample number
TTTT = Temperature in hundreds of degrees Celsius. Measured at the transducer
XXXX = Bottom track X velocity component mm/s.    -99999 indicates no valid velocity
YYYY = Bottom track Y velocity component mm/s.    -99999 indicates no valid velocity
ZZZZ = Bottom track Z velocity component mm/s.    -99999 indicates no valid velocity
DDDD = Depth below transducer in mm.              Range to the bottom, 0 = no detection
xxxx = Water mass X velocity component mm/s.      -99999 indicates no valid velocity
yyyy = Water mass Y velocity component mm/s.      -99999 indicates no valid velocity
zzzz = Water mass Z velocity component mm/s.      -99999 indicates no valid velocity
dddd = Depth of water mass measurement in mm.     Position of bin in front of the transducer
ABCD = Built in test and status bits in hexadecimal0000 = OK.
S    = Sub System. See appendix B5 for description.
N    = Sub System index.
```

```
$PRTI02,sssssshh,nnnnnn,TTTT,EEEE,NNNN,UUUU,DDDD,eeee,nnnn,uuuu,dddd,ABCD,S,N*FF<CR><LF>
Where:
PRTI02 = Identity string.
sssssshh = Start time of this sample in hundreds of seconds since power up or user reset.
nnnnnn = Sample number
TTTT = Temperature in hundreds of degrees Celsius. Measured at the transducer
EEEE = Bottom track East velocity component mm/s.  -99999 indicates no valid velocity
NNNN = Bottom track North velocity component mm/s. -99999 indicates no valid velocity
UUUU = Bottom track Up velocity component mm/s.    -99999 indicates no valid velocity
DDDD = Depth below transducer in mm.               Range to the bottom, 0 = no detection
eeee = Water mass East velocity component mm/s.    -99999 indicates no valid velocity
nnnn = Water mass North velocity component mm/s.   -99999 indicates no valid velocity
uuuu = Water mass UP velocity component mm/s.      -99999 indicates no valid velocity
dddd = Depth of water mass measurement in mm.      Position of bin in front of the transducer
ABCD = Built in test and status bits in hexadecimal 0000 = OK. See appendix B6 for description.
S    = Sub System. See appendix B5 for description.
N    = Sub System index.
```

```
$PRTI32,h.hhh,p.ppp,r.rrr,P.PPPPPPPP,t.ttt,S,N*15
Where:
PRTI30 = Identity string.
h.hhh = Heading used during the bottom track ping.
p.ppp = Pitch used during the bottom track ping.
r.rrr = Roll used during the bottom track ping.
P.PPPPPPPP = Pressure in BAR.
t.ttt = Water Temperature in degrees C.
S    = Sub System. See appendix B5 for description.
N    = Sub System index.
```

The RTI proprietary strings when CEOUTPUT is set to 2:

```
$PRTI33,h.hhh,p.ppp,r.rrr,P.PPPPPPPP,t.ttt,S,N*15
Where:
PRTI31 = Identity string.
h.hhh = Heading used during the water mass ping.
p.ppp = Pitch used during the water mass ping.
r.rrr = Roll used during the water mass ping.
P.PPPPPPPP = Pressure in BAR.
t.ttt = Water Temperature in degrees C.
S    = Sub System. See appendix B5 for description.
N    = Sub System index.
```

5. The RTI proprietary string when CEOUTPUT is set to 5:

```
$PRTI03,sssssshh,nnnnnn,TTTT,XXXX,YYYY,ZZZZ,QQQQ,DDDD,xxxx,yyyy,zzzz,dddd,qqqq,ABCD,S,N*FF<CR><LF>
Where:
PRTI03 = Identity string.
sssssshh = Start time of this sample in hundreds of seconds since power up or user reset.
nnnnnn = Sample number
TTTT = Temperature in hundreds of degrees Celsius. Measured at the transducer
XXXX = Bottom track X velocity component mm/s.    -99999 indicates no valid velocity
YYYY = Bottom track Y velocity component mm/s.    -99999 indicates no valid velocity
ZZZZ = Bottom track Z velocity component mm/s.    -99999 indicates no valid velocity
QQQQ = Bottom track Q velocity component mm/s.    -99999 indicates no valid velocity
DDDD = Depth below transducer in mm.              Range to bottom, 0 = no detection
xxxx = Water mass X velocity component mm/s.      -99999 indicates no valid velocity
yyyy = Water mass Y velocity component mm/s.      -99999 indicates no valid velocity
zzzz = Water mass Z velocity component mm/s.      -99999 indicates no valid velocity
qqqq = Water mass Q velocity component mm/s.      -99999 indicates no valid velocity
dddd = Depth of water mass measurement in mm.     Position of bin in front of the transducer
ABCD = Built in test and status bits in hexadecimal 0000 = OK.
S    = Sub System. See appendix B5 for description.
N    = Sub System index.
```

6. The RTI proprietary strings when STARTHPR is used to output PNI compass data:

```
$PRTI34,h.hhh,p.ppp,r.rrr*15
Where:
PRTI34 = Identity string.
h.hhh = Heading.
p.ppp = Pitch.
r.rrr = Roll.
```

5.4 RTI Binary Data

1. RTI Binary Data Packet consists of 3 sections
   a. 32 byte Header
      i. 16 bytes containing 0x80
      ii. 4 byte Ensemble number
      iii. 4 byte Ensemble number ones compliment
      iv. 4 byte payload size (in bytes)
      v. 4 byte payload size ones compliment
   b. MAT-File Payload
      i. Version 4 MAT-File Format (see matlab4.pdf for details)
      ii. A MAT-file may contain one or more matrices. The matrices are written sequentially to a file, with the bytes forming a continuous stream. Each matrix starts with a fixed-length 20-byte header that contains information describing certain attributes of the Matrix. The 20-byte header consists of five long (4-byte) integers.
      iii. ADCP Velocity, Amplitude, and Statistical data are contained in bins x 4 arrays (row x column).
      iv. ADCP Ancillary data such as Time, Heading, Pitch, Roll and Temperature are contained in an n x 1 array. Where n is the number of rows.
   c. CRC-16 Checksum (see appendix B for example)
      i. 4 byte checksum of all the bytes in the payload

1. First 2 bytes are always 0
2. The second 2 bytes contain the CRC-16 value
    a. CCITT 16 bit algorithm (X^16 + X^12 + X^5 + 1)
    b. CRC seed = 0

Note:
If you remove the RTI header for each ensemble Matlab is able to directly read in the ensemble.

2. RTI matrix names are contained in the payload (see Appendix B1 for details)

3. Binary Ensemble decoding (see Appendix B2 for a C# example)

4. This is a hex capture of the first few bytes in a binary file:

```
53 54 41 52 54 06 0d 0a 80 80 80 80 80 80 80 80
80 80 80 80 80 80 80 80 01 00 00 00 fe ff ff ff
88 0b 00 00 77 f4 ff ff 0a 00 00 00 14 00 00 00
04 00 00 00 00 00 00 00 08 00 00 00 45 30 30 30
30 30 31 00 5a 9a ee bd 96 7a f7 bd 5b 27 db bd
4e 8a 27 be b0 90 43 be a8 b8 5a be 8d 56 82 be
a5 d7 16 be 84 48 03 be a8 c6 b1 42 50 f9 04 be
c5 ca 8e be 43 9b c5 be a8 c6 b1 42 a8 c6 b1 42
a8 c6 b1 42 a8 c6 b1 42 a8 c6 b1 42 a8 c6 b1 42
a8 c6 b1 42 72 17 16 bf 99 88 28 bf 1c 48 f6 be
bc 9f 16 bf 2c 67 1a bf 72 28 27 bf 56 ad 3a bf
68 24 43 bf 16 6b 44 bf b4 47 2a bf 6a 40 43 bf
cf 99 2e bf 1f f4 46 bf a8 c6 b1 42 a8 c6 b1 42
```

a. Yellow is the word START followed by a CRLF. This was captured when the user started the ADCP.
b. Green is the RTI 32 byte Header.
c. Cyan is the matlab version 4 matrix-header containing 4 integer values are that are set to 10,20,4, and 8.
    i. The 10 indicates the matrix contains 4 byte floating point numbers.
    ii. 20 is the number of rows in the matrix which is number bins in the ADCP profile.
    iii. 4 is the number of columns which is the number of ADCP profile beams.
    iv. 8 is the number of bytes in the matrix name.
d. Pink is the matrix name or "E000001".
e. Following that there is 4*20 = 80 floating point numbers in the matrix (not all shown here).
f. The next matrix "E000002" starts after the last floating point number... and so on until all of the matrices in the ensemble are read in.
g. The end of the ensemble has 4 more bytes that are used as a checksum.
h. The next ensemble starts with a 32 byte header just like the previous one and contains the same matrix names repeated but with new data.
i. If you remove the RTI header for each ensemble Matlab is able to directly read in the ensemble.

5.5 Ocean-Server NMEA DVL Data Output

1. The CEOUTPUT 3<CR> command enables this output from the DVL.

    a. Navigation String, DVL -> HOST

$DVLNAV,<#>,<FT>, <FQ>,<Vx>,<Vy>,<Vz>,<Xdist>,<Ydist>,<Zdist>, <R1>,<R2>,<R3>, <R4>,<T>*cc

i. Units Meters, Meters/second, Celsius
ii. # = sample number
iii. Fix type (FT), bottom lock or water lock [0,1]
iv. Fix quality (FQ) [0-9]. 0 = worst quality, 9 = best quality
v. Vx, Vy, Vz are rotated from DVL instrument to Ocean Server vessel coordinates before output.
vi. Xdist, Ydist, Zdist begin accumulating once bottom lock occurs.
vii. R1-R4 is the vertical range to the bottom for each of the 4 beams.
viii. T is water temperature as measure by the DVL. A value of -99999.0 indicates a bad temperature reading.

Notes:
Beam 0 is assumed to be facing 45 degrees outward between Forward and Port.
Vz+ is up and towards the reader of this page.

```
Down facing coordinate system

                Vessel Forward
                      Vx+
                       |
               bm0(X+) | bm2(Y+)
Port     Vy+  --------+---------
               bm3(Y-) | bm1(X-)
                       |
```

b. Setup String, HOST-> DVL

$DVLSET,<SoundSpeed(m/s)>,<CTRIG>*cc

i. Sound Speed is required for proper operation of the DVL.
ii. See CTRIG for input trigger operation.

c. Profile Data UP and DOWN, DVL -> HOST

$DVLPDN,<sample#>,<cell#>,<Vx>,<Vy>,<Vz>,<Ve>,<A1>,<A2>,<A3>,<A4>*cc

i. Bin 0 is near the vehicle and grows out.
ii. Sentence is repeated for each bin in the set. 30 bins you get 30 sentences with the same sample#.

iii. The Vx, Vy, Vz data in m/s.
iv. A1-A4 are bin amplitudes in dB ref 0.

5.6 Profile ASCII Data Output

1. The CEOUTPUT 2<CR> command enables this output from the ADCP.
   a. Enables ASCII output of profile data in the following format:

```
2013/12/20 04:56:19.00    22    15.34    0.13    0.71   22.40
bin    cm    V0    V1    V2    V3     E     N     U     Q  C0  C1  C2  C3   A0   A1   A2   A3
  1   224  -400 88888 88888 88888 88888 88888 88888 88888  28  14   2   7   61   46   44   30
  2   424 88888 88888 88888 88888 88888 88888 88888 88888  13   3  11   2   29   29   30   30
  3   624 88888 88888 88888 88888 88888 88888 88888 88888   9  15  12   9   30   29   30   29
Bot       88888 88888 88888 88888 88888 88888 88888 88888   0   0   0   0    0    0    0    0
Bot Range     0     0     0     0
Bot   SNR     0     0     0     0
```

   b. The first character in the output string is the binary value 0x0C (new page). After that all characters are printable ASCII with a CRLF (0x0D, 0x0A) at the end of each line.
   c. The 1st line contains time and ancillary data:
      i. Year/month/day
      ii. hour:minute:second.hsec
      iii. Ensemble Number
      iv. Heading (deg)
      v. Pitch (deg)
      vi. Roll (deg)
      vii. Water Temperature (deg C)
   d. The 2nd line contains the column titles:
      i. Bin Number
      ii. Bin Position (cm)
      iii. Beam 0 Velocity (mm/s)
      iv. Beam 1 Velocity (mm/s)
      v. Beam 2 Velocity (mm/s)
      vi. Beam 3 Velocity (mm/s)
      vii. East Velocity (mm/s)
      viii. North Velocity (mm/s)
      ix. Up Velocity (mm/s)
      x. Error Velocity (mm/s)
      xi. Beam 0 Correlation (%)
      xii. Beam 1 Correlation (%)
      xiii. Beam 2 Correlation (%)
      xiv. Beam 3 Correlation (%)
      xv. Beam 0 Amplitude (dB)
      xvi. Beam 1 Amplitude (dB)
      xvii. Beam 2 Amplitude (dB)
      xviii. Beam 3 Amplitude (dB)
   e. The 3rd line etc. contains Bin 1 data. There will one line for each bin.

      NOTE: A value of `88888` indicates a “bad” velocity.
   f. The last 3 lines contain bottom track data.

- i. The 1st of those contains velocity data.
- ii. The 2nd contains bottom depth for each of the 4 beams.
- iii. And the last line contains the bottom echo signal to noise in (dB).

Chapter 6 System Checkout and Diagnostics

6.1 System Bench Test

1. Power On Test:
    - a. Apply power to the DVL/ADCP via the deck box and observe that the wakeup message contains no error messages.
    - b. Possible error messages are:
        - i. Hardware Timeout!
            1. Indicates that a hardware problem occurred during power on. Refer to the diagnostic procedure to troubleshoot this problem.
2. PNI Prime Magnetic Compass Test:
    - a. Enable the compass diagnostic mode by sending the command: DIAGCPT<CR>.
    - b. Start the PNI software “StudioPrime_V1p7.exe”.
    - c. On the Connection tab establish communication with the compass. The Computer Baud Rate should be 115200. Do not change the Module Baud Rate.
    - d. On the Configuration tab:
        - i. Mounting Options should be Standard or Standard 180 depending on system type.
        - ii. Acquisition Settings should be Push.
        - iii. Calibration Settings should be set to Mag Only Calibration and Automatic Sampling should not be checked.
        - iv. If you have changed any settings click Save.
    - e. On the Calibration tab: (refer to section 6.2 in the Prime user manual for the complete procedure).
        - i. Click Take Sample each time the DA150 is positioned per 6.2.1 in the Prime user manual.
        - ii. Verify that the Mag Score and the Accel Score are within the Prime manual limits.
        - iii. Click Save Current User Cal when finished.
    - f. On the Test tab:
        - i. Click Go.
            1. Orient the DVL/ADCP so that the beams are facing down or up and beam 0 is pointed to the North.
            2. Verify that the heading is correct for the direction in which beam 0 is facing.
            3. Verify that the helicopter is upside down if the system is facing downward.
            4. Raise beam 0 higher than beam 1 and verify that the pitch is positive.
            5. Raise beam 1 above beam 0 and verify that the pitch is negative.
            6. Raise beam 2 above beam 3 and verify that roll is positive. Note: ±180 degree is the same as 0 degrees for a downward facing system.

        7. Raise beam 3 above beam 2 and verify that the Roll is negative.
        8. Click Stop.
    g. Exit the Studio Prime software. To reestablish communication with the DVL/ADCP send a string of 16 X’s to the system XXXXXXXXXXXXXXXX<CR>. Or cycle the DVL/ADCP power.

3. Low Signal Level Test:
    a. Using a dumb terminal program send the command DIAGSAMP<CR>. Note: this test must run in RF “clean” environment.
    b. Observe that the following data are displayed:

```
DIAGSAMP

Auto Gain:
amp     20.00,   30.00,   30.00,   30.00
Hz    xxxx.xx, xxxx.xx, xxxx.xx, xxxx.xx
cor      0.10,    0.09,    0.09,    0.13

Water      22.45 degrees
BackPlane  28.87 degrees
Pressure  0.000000 bar (1Pa = 10^-5 bar)
Battery    19.50 Volts
Boost Cap   1.96 Volts
Leak 65535 counts
```

    c. “amp” should be between 20 and 40.
    d. All of the “cor” values should be less than 0.35.
    e. Water degrees should match within 1 degree of an independent measurement of the ambient water or air temperature where the DVL/ADCP is located.
    f. Backplane degrees should be within 4 degrees of ambient but will be higher if the system has been recently pinging.

4. High Signal Level Test:
    a. Connect a signal generator with 10vpp sine wave output to a hoop of wire that is laying flat on the face of the transducer.
    b. Set the frequency of the signal generator to 1000 Hz above the operating frequency of the DVL/ADCP.

        150 kHz = 155640 Hz
        300 kHz = 311281 Hz
        600 kHz = 622562 Hz

    c. Using a dumb terminal program send the command DIAGSAMP<CR>.
    d. Observe that the following data are displayed:

```
DIAGSAMP

Auto Gain:
amp    110.00,  110.00,  110.00,  110.00
Hz    1000.00, 1000.00, 1000.00, 1000.00
cor      0.99,    0.99,    1.00,    0.99

Water      22.45 degrees
BackPlane  28.87 degrees
Pressure  0.000000 bar (1Pa = 10^-5 bar)
```

    e. “amp” should be significantly higher than in the Low Level Test.
    f. All of the “Hz” values should be near 1000.
    g. All of the “cor” values should be near 1.00.

5. Power Test:
    a. Make sure the DVL/ADCP transducer is submerged in water.

b. Measure the AC Power going into the deck box. It should be less than 3 watts.
c. Send the following commands to the DVL/ADCP:
    i. CBTON1<CR>
    ii. CWTON0<CR>
    iii. CWPON0<CR>
    iv. ENGBTDEEP<CR>
        1. Verify the response is:

```
ENGBTDEEP
BT DEEP ON
```

        2. This command configures the system for worst case long duration ping.
d. Start the DVL/ADCP pinging by sending START<CR>.
e. Measure the Peak AC Power. It should be greater than 100 watts and less than 1000 Watts.
f. Send STOP<CR> to stop the DVL/ADCP pinging.

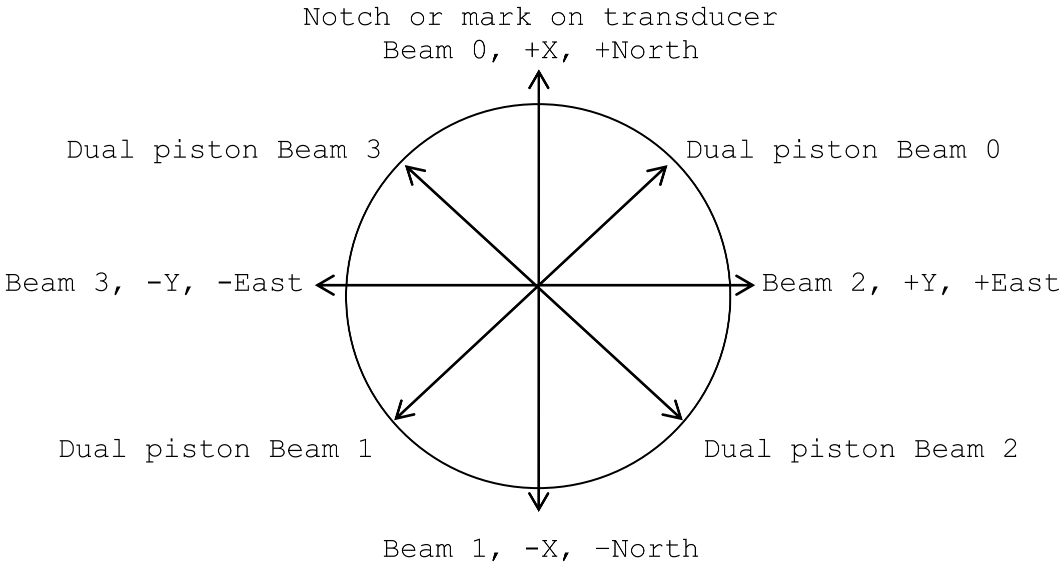
Appendix A

A1 Coordinate systems

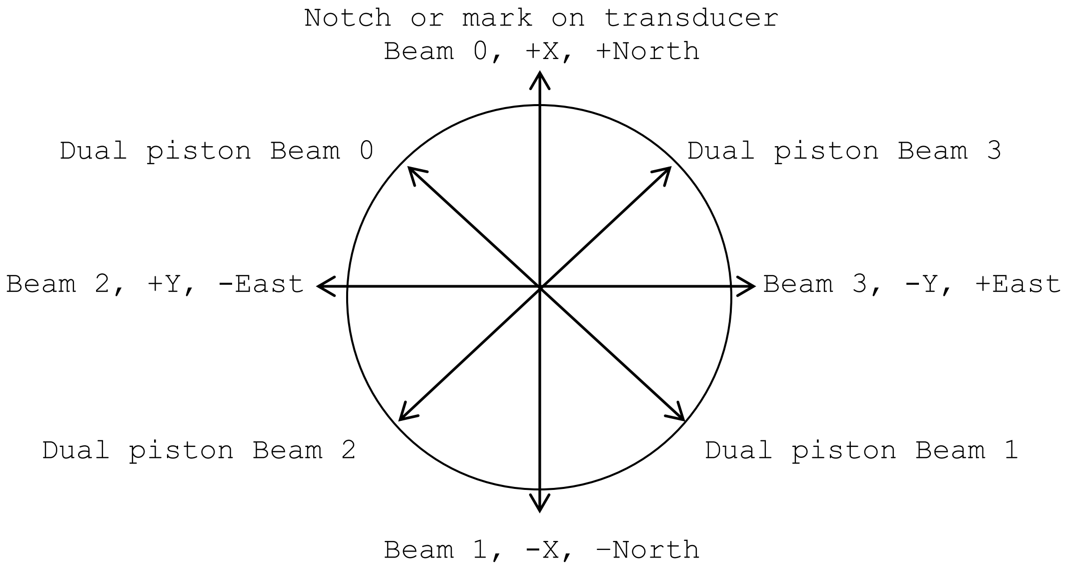
RTI uses three different coordinate systems.

1. Beam
    a. The velocity vector is in the direction each beam points.
    b. Beams are located on azimuth angles of 0, 90, 180, and 270 degrees around transducer.
    c. Beam elevation angles can be 15, 20, or 30 degrees from vertical.
    d. A positive velocity measurement occurs when a beam and an object move closer together.
    e. Beam velocity data is useful for beam diagnostics.
    f. Beam velocity data may require further processing to be useful for navigation or scientific work.
2. Instrument
    a. Beam velocities are rotated to a 3 axis velocity vector X, Y, and Z. The X axis is along a line drawn between beams 0 and 1. The Y axis is along a line drawn between beams 3 and 2. The Z axis is orthogonal to both the X and Y axes.
    b. The instrument coordinate system follows the Right Hand Rule convention.
3. Earth
    a. With the addition of a Heading, Pitch, Roll (HPR) sensor the beam velocities are “bin mapped” to a common horizontal plane and then rotated to the Earth referenced velocity vector East, North, and Up (ENU).

Note: Transformed bottom and water track velocity data are reported as transducer motion. Transformed water profiling bin velocities are reported as water motion. In other words, bottom track velocities have an opposite sign from profile bins if the measured velocity is only due to transducer motion.



Downward facing Transducer (face away from observer)



Upward facing Transducer (face towards observer)

A2 GPS Compass

When using a GPS compass to rotate from the ADCP/DVL XYZ coordinate system to the ENU system use the following:

1. Align beam 0 with the GPS compass North. Beam 0 is located by a small notch on the side of the transducer.
2. Extract the GPS heading from the NMEA HDT string.
3. Process the ADCP XYZ velocity data using the following C# code:

```
double theta = GPS_HDT / 180.0 * Math.PI; //convert from degrees to radians
double CosT = Math.Cos(theta);
double SinT = Math.Sin(theta);
double NorthVelocity = VelocityX * CosT - VelocityY * SinT;
double EastVElocity  = VelocityX * SinT + VelocityY * CosT;
```

4. Repeat the processing for all bins and bottom velocities.

Appendix B

B1 Payload Matrix Contents

1. **E000001**
    a. Single Precision Floating Point
    b. Bins x Beams
        1. Beam Velocity in m/s
            a. Contains the Beam coordinate velocity profile data as measured along each beam. The data is useful for diagnostic purposes and for when the user wants to perform their own transformation. It is arranged beam 0 to 3 for all bins.

2. **E000002**
    a. Single Precision Floating Point
    b. Bins x Beams
        1. Instrument Velocity in m/s
            a. Contains the Instrument coordinate velocity profile. It is arranged X, Y, Z, Q for all bins where:

```
X =  (BmVel[1] - BmVel[0])/(2*sine(BeamAngle))
Y =  (BmVel[3] - BmVel[2])/(2*sine(BeamAngle))
Z = -(BmVel[0] + BmVel[1] + BmVel[2] + BmVel[3])/(4*cosine(BeamAngle))
Q =  (BmVel[0] + BmVel[1] - BmVel[2] - BmVel[3])/(4)
```

3. **E000003**
    a. Single Precision Floating Point
    b. Bins x Beams
        i. Earth Velocity in m/s
            1. Contains the Earth coordinate velocity profile. It is arranged E, N, U, Q for all bins where:

```
East  = X * (SH * CP) - Y * (CH * CR + SH * SR * SP) + Z * (CH * SR - SH * CR * SP);
North = X * (CH * CP) + Y * (SH * CR - CH * SR * SP) - Z * (SH * SR + CH * SP * CR);
Up    = X * (SP)      + Y * (SR * CP)                + Z * (CP * CR);
Q     = (BmVel[0] + BmVel[1] - BmVel[2] - BmVel[3])/(4)
```

Note: X, Y, Z are “bin mapped” prior to the Earth transformation. To bin map the beam data use the following code:

```
CP  = cosine(pitch)
CR  = cosine(roll)
SP  = sine(pitch)
SR  = sine(roll)
SH  = sine(heading)
CH  = cosine(heading)
CBA = cosine(BeamAngle)
SBA = sine(BeamAngle)
```

```
BeamAngle = 15, 20, or 30 degrees

//generate range scale factor for each beam
RS[0] = fabs(CBA / (CP * CR * CBA + SP * SBA));
RS[1] = fabs(CBA / (CP * CR * CBA - SP * SBA));
RS[2] = fabs(CBA / (CP * CR * CBA + SR * CP * SBA));
RS[3] = fabs(CBA / (CP * CR * CBA - SR * CP * SBA));

//generate the map
for(j=0;j<Bins;j++)
{
    for(i=0;i<4;i++)//beams
            BinMap[i][j] = (int)(j * RS[i]);
}

//map the velocity, correlation, and amplitude data
for(j=0;j<Bins;j++)
{
    for(i=0;i<4;i++)
    {
        if(BinMap[i][j] < Bins && BinMap[i][j] >= 0)
        {
            mvel[i][j] = Velocity[i][BinMap[i][j]];   // matrix E000001
            mamp[i][j] = Amplitude[i][BinMap[i][j]];  // matrix E000004
            mcor[i][j] = Correlation[i][BinMap[i][j]];// matrix E000005
        }
        else
        {
            RWP->mvel[i][j] = 0.0;
            RWP->mamp[i][j] = 0.0;
            RWP->mcor[i][j] = 0.0;
       }
    }
}

//transform mapped beam data to XYZ. Repeat for each bin
X =  (mvel[1] - mvel[0])/(2*sine(BeamAngle))
Y =  (mvel[3] - mvel[2])/(2*sine(BeamAngle))
Z = -(mvel[0] + mvel[1] + mvel[2] + mvel[3])/(4*cosine(BeamAngle))
Q =  (mvel[0] + mvel[1] - mvel[2] - mvel[3])/(4)
```

4. **E000004**
    a. Single Precision Floating Point
    b. Bins x Beams
        i. Amplitude in dB
            1. Contains the Beam Amplitude profile. It is arranged beam 0 to 3 for all bins.

5. **E000005**
    a. Single Precision Floating Point
    b. Bins x Beams
        i. Correlation where 1.0 = 100% and 0.5 = 50% correlation.
            1. Contains the Beam Correlation profile. It is arranged beam 0 to 3 for all bins.

6. **E000006**
    a. 32 bit Integer
    b. beams x bins
            1. Good Beam Pings

a. Contains the number of good pings for each bin/beam in the beam velocity data matrix E000001. It is arranged beam 0 to 3 for all bins.

7. **E000007**
    - a. 32 bit Integer
    - b. Bins x Beams
        1. Good Earth Pings
            - a. Contains the number of good pings for each bin/beam in the ENUQ velocity data matrix E000003. It is arranged NG3, NG3, NG3, NG4 for each bin. Where: NG3 = the Number of Good 3 beam solutions and NG4 = the Number of Good 4 beam solutions in the bin.

Note: RTI systems can use 3 or 4 beams to solve for the Earth coordinate system. If 4 beam velocities are available then all 4 are used. If only 3 beams are available, perhaps due to a beam failure or a signal fade, the 4th beam velocity is calculated from the other 3 before calling the Earth transformation function (shown above). To calculate a 3 beam solution it is assumed that the beams are configured as a 4 beam Janus transducer and that the Q velocity on average is equal to 0.0 m/s.

Since

```
Q = (BM0 + BM1 - Bm2 - BM3) = 0.0
```

The “missing” beam solution is:

```
BM0 = -BM1 + BM2 + BM3;
BM1 = -BM0 + BM2 + BM3;
BM2 =  BM0 + BM1 - BM3;
BM3 =  BM0 + BM1 - BM2;
```

8. **E000008**
    - a. 32 bit Integer
    - b. 22 x 1
        - i. Ensemble data
            1. Ensemble Number
            2. Bins
            3. Beams
            4. Pings in Ensemble (desired)
            5. Ping Count (actual)
            6. Status (see appendix B6 for description)
            7. Year
            8. Month
            9. Day
            10. Hour
            11. Minute
            12. Seconds
            13. 1/100 Seconds
            14. System Serial Number A (see appendix B5 for a description of the SN)

            15. System Serial Number B
            16. System Serial Number C
            17. System Serial Number D
            18. System Serial Number E
            19. System Serial Number F
            20. System Serial Number G
            21. System Serial Number H
            22. System Type and Firmware Version
                a. Most significant byte contains a code that describes the hardware Subsystem for this data set. (See description of this code in appendix B5).
                b. Next byte contains the Major firmware version.
                c. Next byte contains the Minor firmware version.
                d. Least significant byte contains the firmware Revision.
            23. Subsystem Configuration
                a. Most significant byte contains a number that identifies which command setup is being used for the current subsystem. Use this byte along with the System Type byte to separate the ensembles for each setup/subsystem.

9. **E000009**
    a. Single Precision Floating Point
    b. 13 x 1
        i. Ancillary data (Contains additional Ensemble data)
            1. Range of First Bin from the transducer in meters
            2. Bin Size in meters
            3. First Profile Ping Time in seconds
            4. Last Profile Ping Time in seconds
            5. Heading in degrees
            6. Pitch in degrees
            7. Roll degrees
            8. Water Temperature in degrees (used in SOS)
            9. System Temperature in degrees
            10. Salinity in parts per thousand (ppt) (used in SOS)
            11. Pressure in bar
            12. Depth in meters (used in SOS)
            13. Speed Of Sound in m/s

18. **E000010**
    a. Single Precision Floating Point
    b. 54 x 1
        i. Bottom Track data
            1. First Bottom Track Ping Time in seconds
            2. Last Bottom Track Ping Time in seconds
            3. Heading in degrees
            4. Pitch in degrees
            5. Roll in degrees
            6. Water Temperature in degrees (used in SOS)
            7. System Temperature in degrees
            8. Salinity in parts per thousand (ppt) (used SOS)

9. Pressure in bar
10. Depth in meters (used in SOS)
11. Speed Of Sound in m/s
12. Status (final value is logical OR of each bit below)
    i. Good Status 0x0000
        a. Water Track 3 Beam solution 0x0001 (DVL only)
            i. 3 of 4 beams have a valid signal
        b. Bottom Track 3 Beam Solution 0x0002
            i. 3 of 4 beams located the bottom
        c. Bottom Track Hold (not searching yet) 0x0004
            i. Holding the search to last known depth
        d. Bottom Track Searching 0x0008
            i. Actively searching for the bottom
        e. Hardware Timeout 0x8000
            i. The hardware did not respond to the ping request
13. Number of Beams
14. Ping Count
15. Range_0 (Beam 0 vertical range in meters)
16. Range_1 (Beam 1 vertical range in meters)
17. Range_2 (Beam 2 vertical range in meters)
18. Range_3 (Beam 3 vertical range in meters)
19. SNR_0 (Beam 0 Signal to Noise in dB)
20. SNR_1 (Beam 1 Signal to Noise in dB)
21. SNR_2 (Beam 2 Signal to Noise in dB)
22. SNR_3 (Beam 3 Signal to Noise in dB)
23. Amplitude_0 (Beam 0 Bottom Amplitude in dB)
24. Amplitude_1 (Beam 1 Bottom Amplitude in dB)
25. Amplitude_2 (Beam 2 Bottom Amplitude in dB)
26. Amplitude_3 (Beam 3 Bottom Amplitude in dB)
27. Correlation_0 (Beam 0 Correlation (0.5 = 50%))
28. Correlation_1 (Beam 1 Correlation (0.5 = 50%))
29. Correlation_2 (Beam 2 Correlation (0.5 = 50%))
30. Correlation_3 (Beam 3 Correlation (0.5 = 50%))
31. Velocity_0 (Beam 0 Velocity in m/s)
32. Velocity_1 (Beam 1 Velocity in m/s)
33. Velocity_2 (Beam 2 Velocity in m/s)
34. Velocity_3 (Beam 3 Velocity in m/s)
35. BeamN_0 (Beam 0 Number of pings averaged)
36. BeamN_1 (Beam 1 Number of pings averaged)
37. BeamN_2 (Beam 2 Number of pings averaged)
38. BeamN_3 (Beam 3 Number of pings averaged)
39. Instrument_0 (X velocity in m/s)
40. Instrument_1 (Y velocity in m/s)
41. Instrument_2 (Z velocity in m/s)
42. Instrument_3 (Q velocity in m/s)
43. XfrmN_0 (Number of 3 beam solutions averaged)
44. XfrmN_1 (Number of 3 beam solutions averaged)
45. XfrmN_2 (Number of 3 beam solutions averaged)

46. XfrmN_3 (Number of 4 beam solutions averaged)
47. Earth_0 (East velocity in m/s)
48. Earth_1 (North velocity in m/s)
49. Earth_2 (Up velocity in m/s)
50. Earth_3 (Q velocity in m/s)
51. EarthN_0 (Number of 3 beam solutions averaged)
52. EarthN_1 (Number of 3 beam solutions averaged)
53. EarthN_2 (Number of 3 beam solutions averaged)
54. EarthN_3 (Number of 4 beam solutions averaged)

19. **E000011**
    a. 8 bit Byte
    b. n x 1
        i. External NMEA ASCII collected during ping
            1. Contains captured serial NMEA data that arrived during the ensemble. One 8 bit byte per ASCII character. Maximum number of bytes = 8192.

20. **E000012**
    a. Single Precision Floating Point
    b. 23 x 1
        i. Profile Engineering data contains data that describes the Profile ping setup.
            1. PrePingVel[0]
            2. PrePingVel[1]
            3. PrePingVel[2]
            4. PrePingVel[3]
            5. PrePingCor[0]
            6. PrePingCor[1]
            7. PrePingCor[2]
            8. PrePingCor[3]
            9. PrePingAmp[0]
            10. PrePingAmp[1]
            11. PrePingAmp[2]
            12. PrePingAmp[3]
            13. SamplesPerSecond
            14. SystemFreqHz
            15. LagSamples
            16. CPCE
            17. NCE
            18. RepeatN
            19. PrePingGap
            20. PrePingNCE
            21. PrePingRepeatN
            22. PrePingLagSamples
            23. TRhighGain
21. **E000013**
    a. Single Precision Floating Point
    b. 30 x 1
        i. Bottom Track Engineering data contains data that describes the bottom track ping setup.
            1. SamplesPerSecond
            2. SystemFreqHz
            3. LagSamples
            4. CPCE
            5. NCE
            6. RepeatN
            7. AmbHz[0]
            8. AmbHz[1]
            9. AmbHz[2]
            10. AmbHz[3]
            11. AmbVel[0]
            12. AmbVel[1]
            13. AmbVel[2]
            14. AmbVel[3]
            15. AmbAmp[0]
            16. AmbAmp[1]

17. AmbAmp[2]
18. AmbAmp[3]
19. AmbCor[0]
20. AmbCor[1]
21. AmbCor[2]
22. AmbCor[3]
23. AmbSNR[0]
24. AmbSNR[1]
25. AmbSNR[2]
26. AmbSNR[3]
27. LagUsed[0]
28. LagUsed[1]
29. LagUsed[2]
30. LagUsed[3]

22. **E000014**

a. Single Precision Floating Point

b. 21 x 1

i. System Settings

1. Bottom Track SamplesPerSecond
2. Bottom Track SystemFreqHz
3. Bottom Track CyclesPerCodeElement
4. Bottom Track NumberOfCodeElementsInCode
5. Bottom Track NumberOfRepeatedCodes
6. Water Profile SamplesPerSecond
7. Water Profile SystemFreqHz
8. Water Profile CyclesPerCodeElement
9. Water Profile NumberOfCodeElementsInCode
10. Water Profile NumberOfRepeatedCodes
11. Water Profile NumberOfSamplesInLag
12. System Input Voltage
13. Transmitter Boost Voltage
14. Bottom Track Mode
15. Bottom Track Pulse to Pulse Lag (m)
16. Bottom Track Long Range Switch Depth (m)
17. Bottom Track Beam Multiplex
18. Water Profile Mode
19. Water Profile Lag (m)
20. Water Profile Transmit Bandwidth
21. Water Profile Receive Bandwidth

23. **E000015**

a. Single Precision Floating Point

b. (3 x number of beams + 1) x 1. Note: there can be up to 13 rows in this data structure.

i. Profile Range Tracking

1. Range Tracking Number of Beams
2. Range Tracking SNR for each beam
3. Range Tracking Range for each beam
4. Range Tracking Number of pings for each beam

B2 C# Binary Decoding Example

```
int PacketPointer = 0;

int ByteArrayToInt(byte[] packet)
{
    ByteArrayToNumber.A = packet[PacketPointer++];
    ByteArrayToNumber.B = packet[PacketPointer++];
    ByteArrayToNumber.C = packet[PacketPointer++];
    ByteArrayToNumber.D = packet[PacketPointer++];

    return ByteArrayToNumber.Int;
}
string ByteArrayToString(byte[] packet, int len)
{
    string s = "";
    int i;
    for (i = 0; i < len; i++)
    {
        s += (char)packet[PacketPointer++];
    }
    return s;
}
float ByteArrayToFloat(byte[] packet)
{
    ByteArrayToNumber.A = packet[PacketPointer++];
    ByteArrayToNumber.B = packet[PacketPointer++];
    ByteArrayToNumber.C = packet[PacketPointer++];
    ByteArrayToNumber.D = packet[PacketPointer++];
    return ByteArrayToNumber.Float;
}

string VelocityID     = "E000001\0";
string InstrumentID   = "E000002\0";
string EarthID        = "E000003\0";
string AmplitudeID    = "E000004\0";
string CorrelationID  = "E000005\0";
string BeamNID        = "E000006\0";
string XfrmNID        = "E000007\0";
string EnsembleDataID = "E000008\0";
string AncillaryID    = "E000009\0";
string BottomTrackID  = "E000010\0";
string NMEAID         = "E000011\0";

void DecodeEnsemble(byte[] packet, ArrayClass m)
{
    int i = 0;
    int SizeCount = 0;
    int ArrayCount = 0;

    PacketPointer = HDRLEN;
    m.VelocityAvailable = false;
    m.InstrumentAvailable = false;
    m.EarthAvailable = false;
    m.AmplitudeAvailable = false;
    m.CorrelationAvailable = false;
    m.BeamNAvailable = false;
    m.XfrmNAvailable = false;
    m.EnsembleDataAvailable = false;
    m.AncillaryAvailable = false;
    m.BottomTrackAvailable = false;
    m.NmeaAvailable = false;

    for (i = 0; i < MaxArray; i++)
    {
        m.Type[i]    = ByteArrayToInt(packet);
        m.Bins[i]    = ByteArrayToInt(packet);
```

```
m.Beams[i]   = ByteArrayToInt(packet);
m.Imag[i]    = ByteArrayToInt(packet);
m.NameLen[i] = ByteArrayToInt(packet);
m.Name[i]    = ByteArrayToString(packet, 8);

ArrayCount = m.Bins[i] * m.Beams[i];
SizeCount = PacketPointer;

if (VelocityID.Equals(m.Name[i], StringComparison.Ordinal))
{
    m.VelocityAvailable = true;
    for (int beam = 0; beam < m.Beams[i]; beam++)
    {
        for (int bin = 0; bin < m.Bins[i]; bin++)
        {
            m.Velocity[beam, bin] = ByteArrayToFloat(packet);
        }
    }
}
else
{
if (InstrumentID.Equals(m.Name[i], StringComparison.Ordinal))
{
    m.InstrumentAvailable = true;

    for (int beam = 0; beam < m.Beams[i]; beam++)
    {
        for (int bin = 0; bin < m.Bins[i]; bin++)
        {
            m.Instrument[beam, bin] = ByteArrayToFloat(packet);
        }
    }
}
else
{
if (EarthID.Equals(m.Name[i], StringComparison.Ordinal))
{
    m.EarthAvailable = true;
    for (int beam = 0; beam < m.Beams[i]; beam++)
    {
        for (int bin = 0; bin < m.Bins[i]; bin++)
        {
            m.Earth[beam, bin] = ByteArrayToFloat(packet);
        }
    }
}
else
{
if (AmplitudeID.Equals(m.Name[i], StringComparison.Ordinal))
{
    m.AmplitudeAvailable = true;
    for (int beam = 0; beam < m.Beams[i]; beam++)
    {
        for (int bin = 0; bin < m.Bins[i]; bin++)
        {
            m.Amplitude[beam, bin] = ByteArrayToFloat(packet);
        }
    }
}
else
{
if (CorrelationID.Equals(m.Name[i], StringComparison.Ordinal))
{
    m.CorrelationAvailable = true;
    for (int beam = 0; beam < m.Beams[i]; beam++)
    {
        for (int bin = 0; bin < m.Bins[i]; bin++)
        {
            m.Correlation[beam, bin] = ByteArrayToFloat(packet);
        }
    }
```

```
}
else
{
if (BeamNID.Equals(m.Name[i], StringComparison.Ordinal))
{
    m.BeamNAvailable = true;
    for (int beam = 0; beam < m.Beams[i]; beam++)
    {
        for (int bin = 0; bin < m.Bins[i]; bin++)
        {
            m.BeamN[beam, bin] = ByteArrayToInt(packet);
        }
    }
}
else
{
if (XfrmNID.Equals(m.Name[i], StringComparison.Ordinal))
{
    m.XfrmNAvailable = true;
    for (int beam = 0; beam < m.Beams[i]; beam++)
    {
        for (int bin = 0; bin < m.Bins[i]; bin++)
        {
            m.XfrmN[beam, bin] = ByteArrayToInt(packet);
        }
    }
}
else
{
if (EnsembleDataID.Equals(m.Name[i], StringComparison.Ordinal))
{
    m.EnsembleDataAvailable = true;
    m.E_EnsembleNumber = ByteArrayToInt(packet);
    m.E_Cells = ByteArrayToInt(packet);
    m.E_Beams = ByteArrayToInt(packet);
    m.E_PingsInEnsemble = ByteArrayToInt(packet);
    m.E_PingCount = ByteArrayToInt(packet);
    m.E_Status = ByteArrayToInt(packet);
    m.E_Year = ByteArrayToInt(packet);
    m.E_Month = ByteArrayToInt(packet);
    m.E_Day = ByteArrayToInt(packet);
    m.E_Hour = ByteArrayToInt(packet);
    m.E_Minute = ByteArrayToInt(packet);
    m.E_Second = ByteArrayToInt(packet);
    m.E_Hsec = ByteArrayToInt(packet);
    for (i =0; i<32; i++)
    {
      if (packet[PacketPointer] > 31 && packet[PacketPointer] < 127)
        m.E_SN_Buffer[i] = packet[PacketPointer];
      else
        m.E_SN_Buffer[i] = 63;
      PacketPointer++;
    }
    m.E_SN_Buffer[i] = 0;
    for (i = 0; i < 4; i++)
    {
      m.E_FW_Vers[i] = packet[PacketPointer];
      PacketPointer++;
    }
}
else
{
if (AncillaryID.Equals(m.Name[i], StringComparison.Ordinal))
{
    m.AncillaryAvailable = true;
    m.A_FirstCellDepth = ByteArrayToFloat(packet);
    m.A_CellSize = ByteArrayToFloat(packet);
    m.A_FirstPingSeconds = ByteArrayToFloat(packet);
    m.A_LastPingSeconds = ByteArrayToFloat(packet);

    m.A_Heading = ByteArrayToFloat(packet);
    m.A_Pitch = ByteArrayToFloat(packet);
    m.A_Roll = ByteArrayToFloat(packet);
    m.A_WaterTemperature = ByteArrayToFloat(packet);
    m.A_BoardTemperature = ByteArrayToFloat(packet);
    m.A_Salinity = ByteArrayToFloat(packet);
    m.A_Pressure = ByteArrayToFloat(packet);
    m.A_Depth = ByteArrayToFloat(packet);
    m.A_SpeedOfSound = ByteArrayToFloat(packet);
}
else
{
if (BottomTrackID.Equals(m.Name[i], StringComparison.Ordinal))
{
    m.BottomTrackAvailable = true;
    m.B_FirstPingSeconds = ByteArrayToFloat(packet);
    m.B_LastPingSeconds = ByteArrayToFloat(packet);

    m.B_Heading = ByteArrayToFloat(packet);
    m.B_Pitch = ByteArrayToFloat(packet);
    m.B_Roll = ByteArrayToFloat(packet);
    m.B_WaterTemperature = ByteArrayToFloat(packet);
    m.B_BoardTemperature = ByteArrayToFloat(packet);
    m.B_Salinity = ByteArrayToFloat(packet);
    m.B_Pressure = ByteArrayToFloat(packet);
    m.B_Depth = ByteArrayToFloat(packet);
    m.B_SpeedOfSound = ByteArrayToFloat(packet);

    m.B_Status = ByteArrayToFloat(packet);
    m.B_Beams = ByteArrayToFloat(packet);
    m.B_PingCount = ByteArrayToFloat(packet);

    for (int beam = 0; beam < m.B_Beams; beam++)
        m.B_Range[beam] = ByteArrayToFloat(packet);
    for (int beam = 0; beam < m.B_Beams; beam++)
        m.B_SNR[beam] = ByteArrayToFloat(packet);
    for (int beam = 0; beam < m.B_Beams; beam++)
        m.B_Amplitude[beam] = ByteArrayToFloat(packet);
    for (int beam = 0; beam < m.B_Beams; beam++)
        m.B_Correlation[beam] = ByteArrayToFloat(packet);
    for (int beam = 0; beam < m.B_Beams; beam++)
        m.B_Velocity[beam] = ByteArrayToFloat(packet);
    for (int beam = 0; beam < m.B_Beams; beam++)
        m.B_BeamN[beam] = ByteArrayToFloat(packet);

    for (int beam = 0; beam < m.B_Beams; beam++)
        m.B_Instrument[beam] = ByteArrayToFloat(packet);
    for (int beam = 0; beam < m.B_Beams; beam++)
        m.B_XfrmN[beam] = ByteArrayToFloat(packet);
    for (int beam = 0; beam < m.B_Beams; beam++)
        m.B_Earth[beam] = ByteArrayToFloat(packet);
    for (int beam = 0; beam < m.B_Beams; beam++)
        m.B_EarthN[beam] = ByteArrayToFloat(packet);
}
else
{
if (NMEAID.Equals(m.Name[i], StringComparison.Ordinal))
{
    m.NmeaAvailable = true;
    i=0;
    while(packet[PacketPointer] != 0)
    {
        m.NMEA_Buffer[i++] = packet[PacketPointer++];
        if (i >= 8192)
            break;
    }
    for (int end = i; end < 8192; end++)
    {
        m.NMEA_Buffer[end] = 0;
    }
}
}
```

```
            }
            }
            }
            }
            }
            }
            }
            }
            }
            //correct for changes in array length
            SizeCount = (PacketPointer - SizeCount) / 4;

            if (SizeCount != ArrayCount)
            {
                PacketPointer += 4 * (ArrayCount - SizeCount);
            }
            if (PacketPointer + 4 >= PacketSize)
                break;//no more data
        }
        nArray = i;
        if (m.E_Cells < 1)
            m.E_Cells = 1;
```

B3 C# Checksum Calculation

The following C# code calculates the checksum for the RTI ensemble:

```
const int HDRLEN = 32;
long csum = 0;

long EnsembleSize  = DataBuff [24];
        EnsembleSize += DataBuff [25] << 8;
        EnsembleSize += DataBuff [26] << 16;
        EnsembleSize += DataBuff [27] << 24;

 //CCITT 16 bit algorithm (X^16 + X^12 + X^5 + 1)

 ushort crc = 0; //seed = 0
 for (i = HDRLEN; i < EnsembleSize + HDRLEN; i++)
 {
    crc = (ushort)((byte)(crc >> 8) | (crc << 8));
    crc ^= DataBuff[i];
    crc ^= (byte)((crc & 0xff) >> 4);
    crc ^= (ushort)((crc << 8) << 4);
    crc ^= (ushort)(((crc & 0xff) << 4) << 1);
 }
 ushort csum = crc;
```

B4 C# Binary Data Packet

1. Ensemble Binary Data Packet consists of 3 sections
    a. 32 byte Header
        i. 16 bytes containing 0x80
        ii. 4 byte Ensemble number
        iii. 4 byte Ensemble number ones compliment
        iv. 4 byte payload size (in bytes)
        v. 4 byte payload size ones compliment
    b. MAT-File Payload
        i. Version 4 MAT-File Format (see matlab4.pdf for details)
        ii. A MAT-file may contain one or more matrices. The matrices are written sequentially to a file, with the bytes forming a continuous stream. Each matrix starts with a fixed-length 20-byte header that contains information describing certain attributes of the Matrix. The 20-byte header consists of five long (4-byte) integers.
        iii. Velocity, Amplitude, and Statistical data are contained in bins x 4 arrays (row x column).
        iv. Ancillary data such as Time, Heading, Pitch, Roll and Temperature are contained in an n x 1 array. Where n is the number of rows.
    c. CRC-16 Checksum
        i. 4 byte checksum of all the bytes in the payload
        ii. First 2 bytes are always 0
        iii. The second 2 bytes contain the CRC-16 value
        iv. CCITT 16 bit algorithm (X^16 + X^12 + X^5 + 1)
        v. CRC seed = 0

B5 System Serial Number and Subsystem Codes

1. The system serial number is 32 bytes in length
    a. SNXX0123456789ABCDEabcdefghi123456
        i. The first 2 bytes after the SN (XX) are combined to form an integer number. This number describes the base electronics hardware architecture.
        ii. The next 15 bytes (0 - E) represent the 15 possible different sub systems.
            1. Each of the 15 digits 0 through E contains a code (0 - Z) that describes a subsystem. See list below for a description of the codes.
            2. Subsystem references are made by using the corresponding character 0 – F. For example: the command CWPP[3] 10<CR> sets the number of water track pings for subsystem 3 to 10 pings. Subsystem 3 is the 4th subsystem contained in the SN.
        iii. The next 9 bytes (a – i) are spares.
        iv. The last 6 digits (1 - 6) are the system serial number.
    b. Example 1:
        i. SN01300000000000000000000000000001
            1. A 600 kHz 4 beam 20 degree piston, serial number 1
    c. Example 2:
        i. SN01340000000000000000000000000001
            1. Dual frequency 600/300 kHz 4 beam 20 degree piston, serial number 1
    d. Example 3:
        i. SN01K0000000000000000000000000001
            1. A 150 kHz 4 beam 30 degree array, serial number 1
    e. Example 4:
        i. SN01KOc000000000000000000000000001
            1. A dual 150/600 kHz 4 beam 30 degree array
            2. A 1200 kHz 4 beam 20 degree piston opposite facing
            3. Serial number 1

Subsystem Codes:

code	description
0	spare
1	2 MHz 4 beam 20 degree piston
2	1.2 MHz
3	600 kHz
4	300 kHz
5	2 MHz 4 beam 20 degree piston, 45 degree heading offset
6	1.2 MHz
7	600 kHz
8	300 kHz
9	2 MHz vertical beam piston
A	1.2 MHz
B	600 kHz
C	300 kHz
D	150 kHz 4 beam 20 degree piston
E	75 kHz
F	38 kHz
G	20 kHz
H	spare
I	600 kHz 4 beam 30 deg array
J	300 kHz
K	150 kHz
L	75 kHz
M	38 kHz
N	20 kHz

```
O     600 kHz 4 beam 15 deg array
P     300 kHz
Q     150 kHz
R      75 kHz
S      38 kHz
T      20 kHz
U     600 kHz 1 beam 0 deg array
V     300 kHz
W     150 kHz
X      75 kHz
Y      38 kHz
Z      20 kHz
a     spare
b      2   MHz 4 beam 20 degree piston opposite facing
c      1.2 MHz
d      600 kHz
e      300 kHz
f      150
g       75
h       38
i       20
j       2   MHz 4 beam 30 degree piston
k      1.2 MHz
l      600 kHz
m      300 kHz
n      150
o       75
p       38
q       20
r       2   MHz 4 beam 30 degree piston opposite facing
s      1.2 MHz
t      600 kHz
u      300 kHz
v      150
w       75
x       38
y       20
z     next letter
z0    spare
z1    2 MHz undefined
z2   1.2 MHz undefined
```

etc…

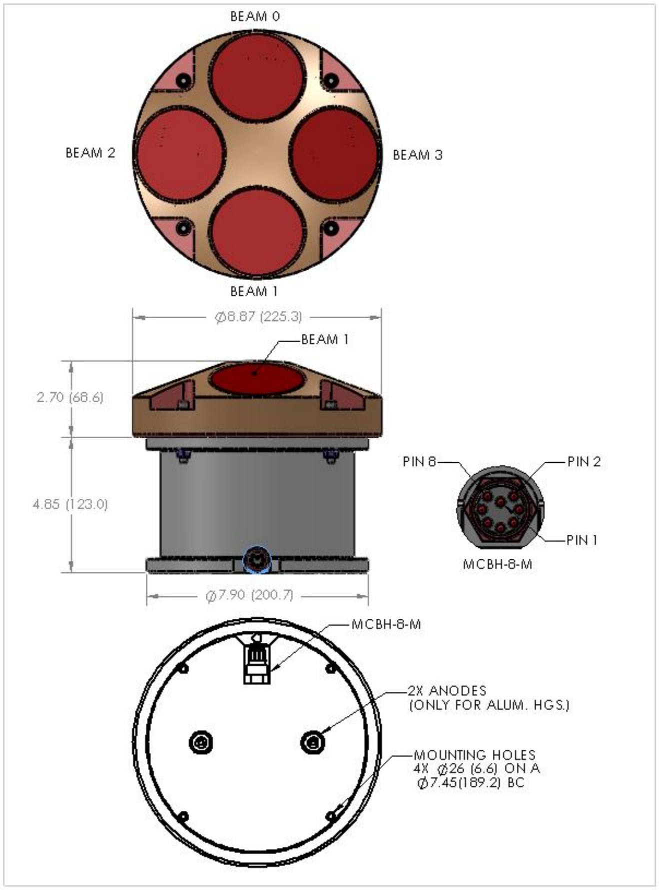
B6 System Status Word

The System Status word is output with both the NMEA and Binary data structures. The final value is the logical OR of each bit shown below:

1. 0x0000 Good Status
2. 0x0001 Bottom track long lag processing is in use
3. 0x0002 Bottom track 3 beam solution
   a. Indicates the bottom track velocity output contains a 3 beam solution.
4. 0x0004 Bottom track hold
   a. Indicates bottom track did not detect the bottom but is still using the last good detection as an estimate of the bottom location.
5. 0x0008 Bottom Track Search
   a. Indicates bottom track is actively changing the ping settings to attempt bottom detection.
6. 0x0010 Bottom Track long range narrow band processing is being used.
   a. Indicates bottom track is using the narrow band processing for long range bottom detection.
7. 0x0020 Bottom track coast.
   a. Indicates the bottom track output filter is in use but no new data is available for output.
8. 0x0040 Bottom track proof.
   a. Indicates bottom track is waiting for the next valid bottom track ping before allowing velocity data to be output.
9. 0x0080 Bottom track low gain.
   a. Indicates bottom track has reduced the receiver gain below the selected switch range.
10. 0x0100 Heading sensor Error
11. 0x0200 Pressure Sensor Error
12. 0x0400 Power Down failure
    a. System power did not shut off between ensembles.
13. 0x0800 Non Volatile data error
    a. Non volatile memory storage checksum failed.
14. 0x1000 Real Time Clock (RTC) error
    a. The RTC did not respond or the time data value contained illegal values i.e. month = 13.
15. 0x2000 Temperature sensor Error
    a. The temperature sensor ADC did not respond or the temperature value was out of range (-30 to 70 C).
16. 0x4000 Receiver data error
    a. The receiver did not output the expected amount of data.
17. 0x8000 Receiver Timeout
    a. The receiver hardware did not respond to the ping request.

Appendix C

C1 DP 300/600 Outline Drawings



Weight in Air: 7.5 kg

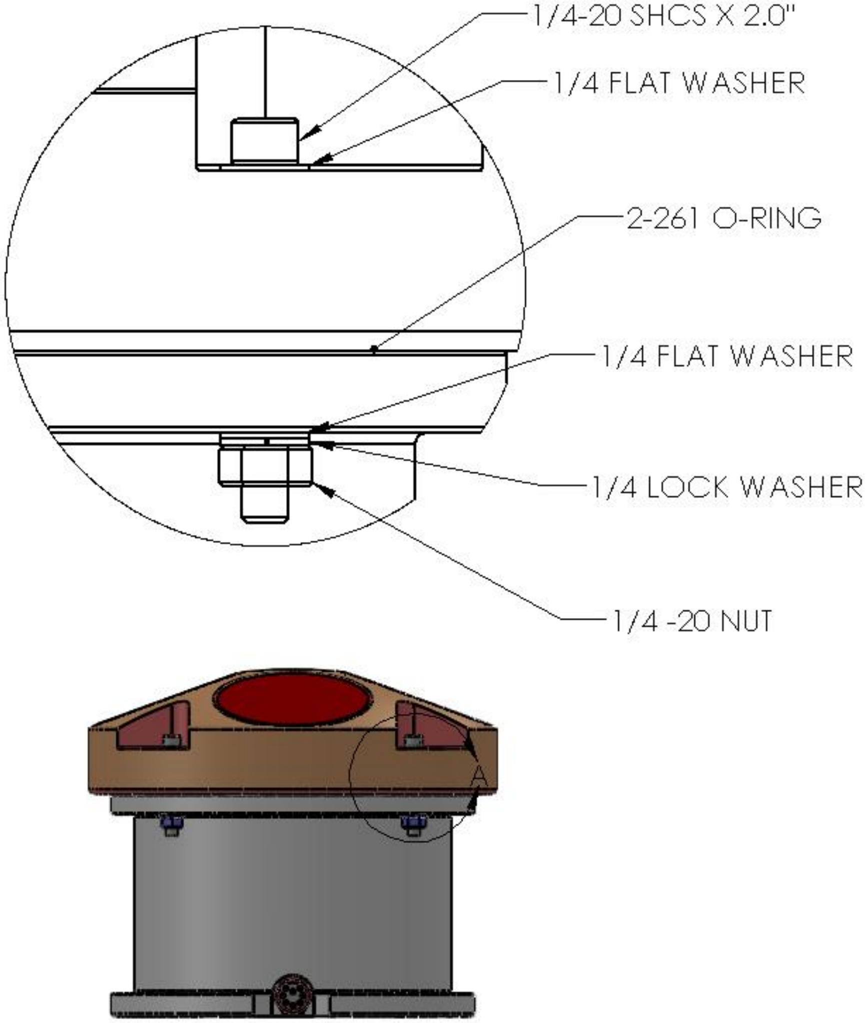
Weight in Water: 1.5 kg

Housing Material: Copolymer Acetal

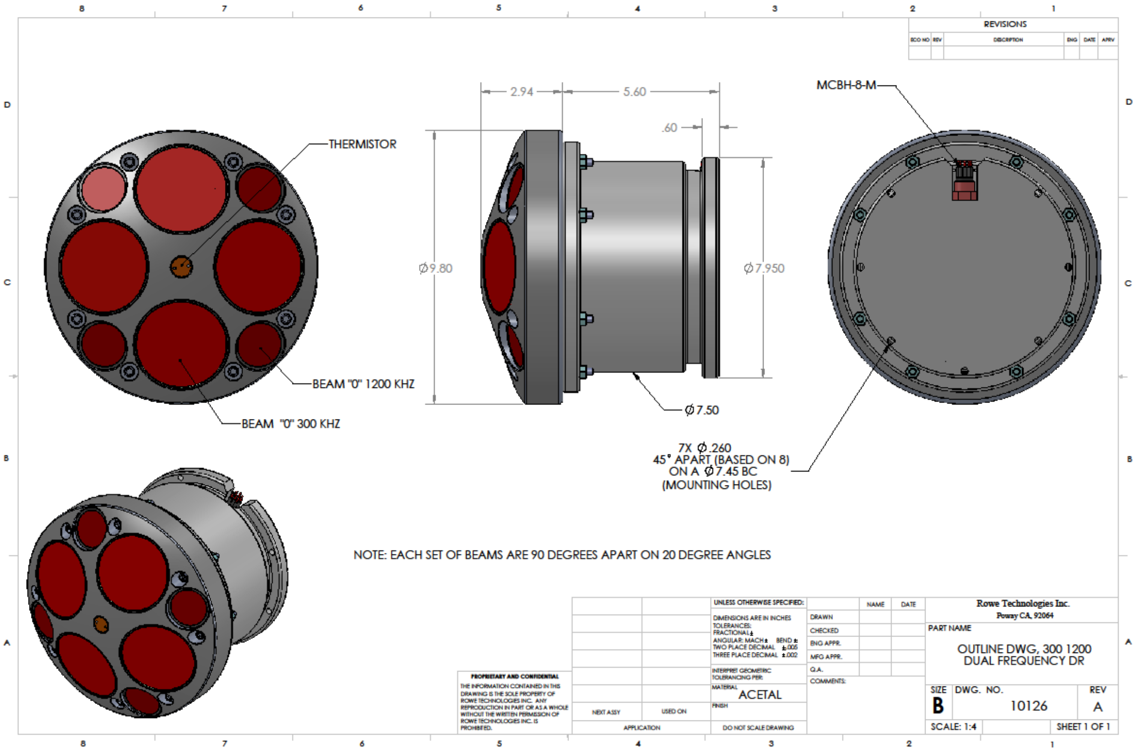
Transducer Face: Urethane

Pressure Rating : 200m

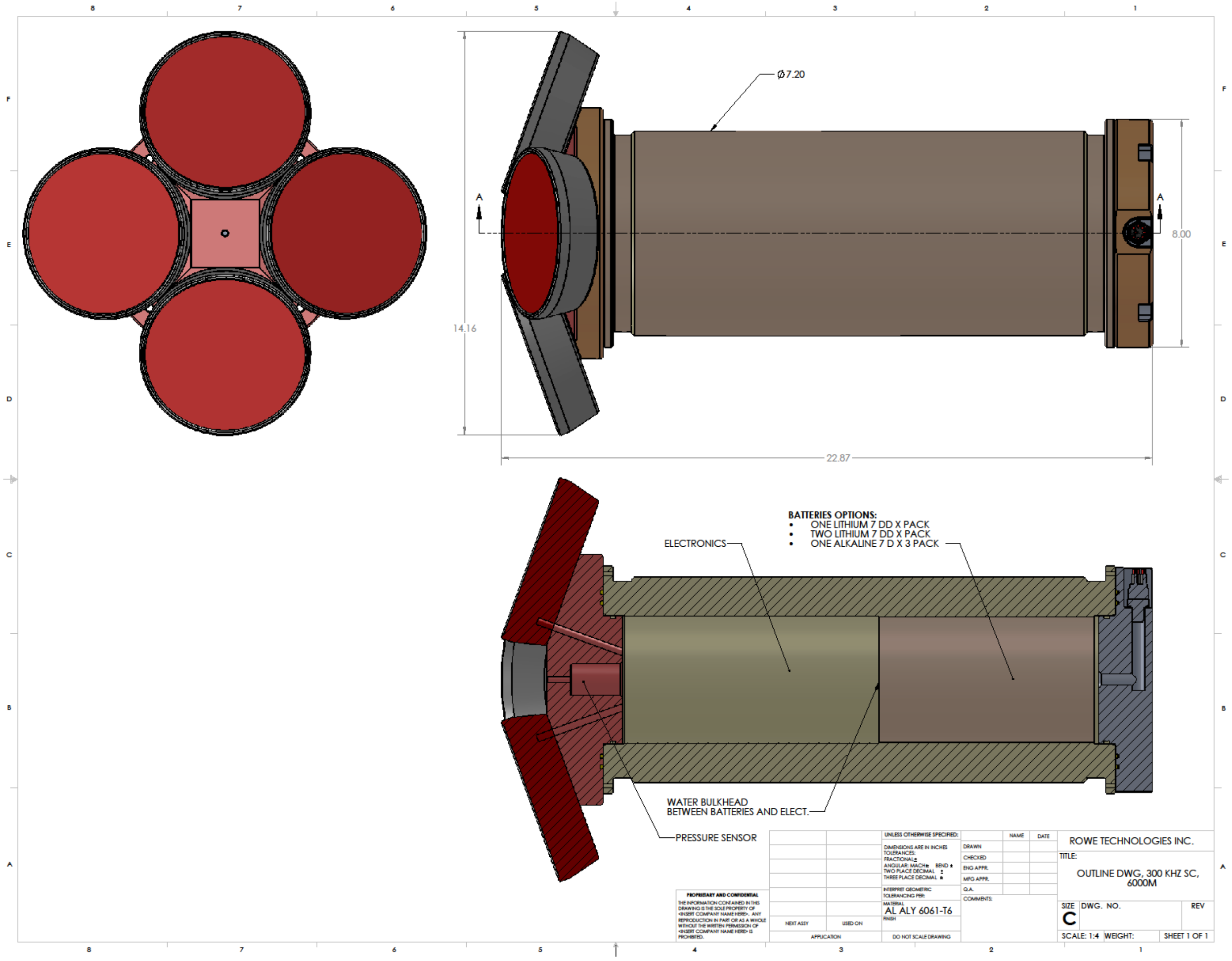
C2 DP 300/600 Close up Hardware



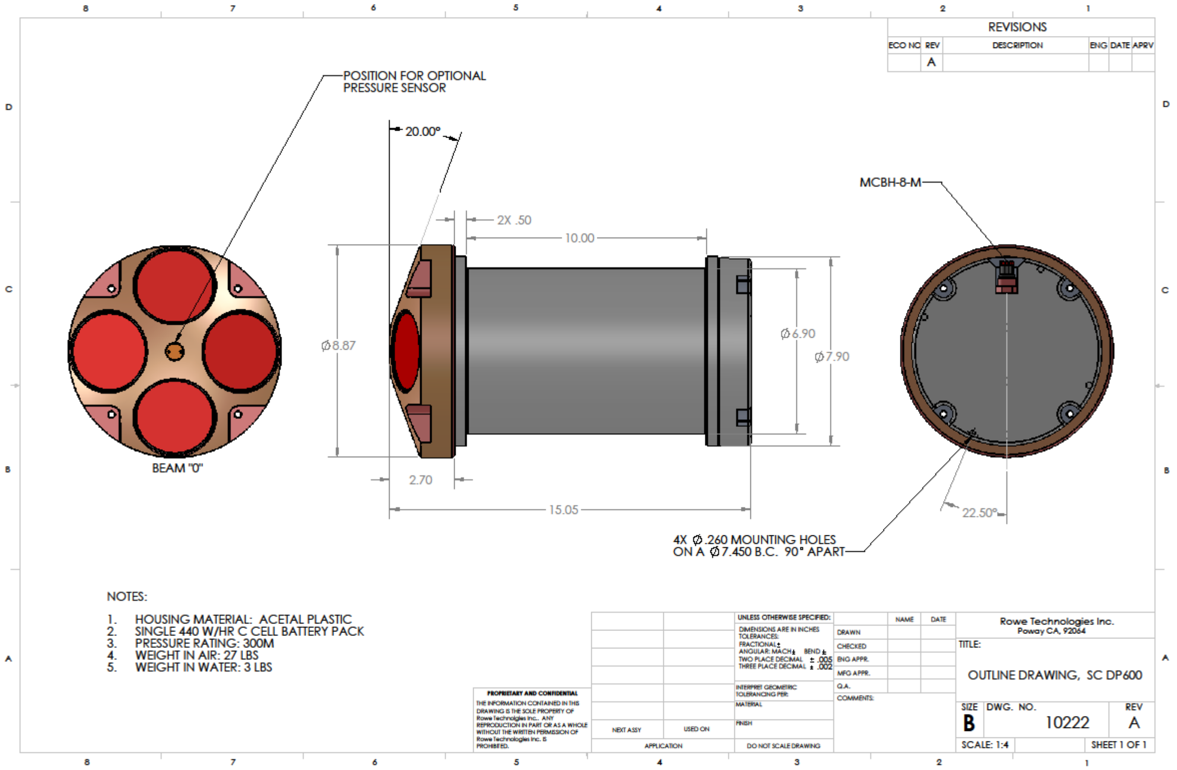
C3 Dual Frequency DP 300/1200 Outline Drawing



C4 300 KHZ SC 6000M ASSY DWG


Ø7.20
A
A
8.00
14.16
22.87
BATTERIES OPTIONS:
• ONE LITHIUM 7 DD X PACK
• TWO LITHIUM 7 DD X PACK
• ONE ALKALINE 7 D X 3 PACK
ELECTRONICS
WATER BULKHEAD
BETWEEN BATTERIES AND ELECT.
PRESSURE SENSOR
PROPRIETARY AND CONFIDENTIAL
THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF <INSERT COMPANY NAME HERE>. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF <INSERT COMPANY NAME HERE> IS PROHIBITED.
NEXT ASSY
USED ON
APPLICATION
UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN INCHES
TOLERANCES:
FRACTIONAL ±
ANGULAR: MACH ± BEND ±
TWO PLACE DECIMAL ±
THREE PLACE DECIMAL ±
INTERPRET GEOMETRIC TOLERANCING PER:
MATERIAL
AL ALY 6061-T6
FINISH
DO NOT SCALE DRAWING
NAME
DATE
DRAWN
CHECKED
ENG APPR.
MFG APPR.
Q.A.
COMMENTS:
ROWE TECHNOLOGIES INC.
TITLE:
OUTLINE DWG, 300 KHZ SC, 6000M
SIZE
C
DWG. NO.
REV
SCALE: 1:4
WEIGHT:
SHEET 1 OF 1
8
7
6
5
4
3
2
1
F
E
D
C
B
A

C5 SC 600 COASTAL ASSY DWG


REVISIONS
ECO NO
REV
DESCRIPTION
ENG
DATE
APRV
A
POSITION FOR OPTIONAL PRESSURE SENSOR
20.00°
2X .50
10.00
MCBH-8-M
Ø8.87
Ø6.90
Ø7.90
BEAM "0"
2.70
15.05
22.50°
4X Ø.260 MOUNTING HOLES ON A Ø7.450 B.C. 90° APART
NOTES:
1. HOUSING MATERIAL: ACETAL PLASTIC
2. SINGLE 440 W/HR C CELL BATTERY PACK
3. PRESSURE RATING: 300M
4. WEIGHT IN AIR: 27 LBS
5. WEIGHT IN WATER: 3 LBS
PROPRIETARY AND CONFIDENTIAL
THE INFORMATION CONTAINED IN THIS DRAWING IS THE SOLE PROPERTY OF Rowe Technolgies Inc.. ANY REPRODUCTION IN PART OR AS A WHOLE WITHOUT THE WRITTEN PERMISSION OF Rowe Technologies Inc. IS PROHIBITED.
NEXT ASSY
USED ON
APPLICATION
UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN INCHES
TOLERANCES:
FRACTIONAL±
ANGULAR: MACH± BEND ±
TWO PLACE DECIMAL ± .005
THREE PLACE DECIMAL ± .002
INTERPRET GEOMETRIC TOLERANCING PER:
MATERIAL
FINISH
DO NOT SCALE DRAWING
NAME
DATE
DRAWN
CHECKED
ENG APPR.
MFG APPR.
Q.A.
COMMENTS:
Rowe Technologies Inc.
Poway CA, 92064
TITLE:
OUTLINE DRAWING, SC DP600
SIZE
B
DWG. NO.
10222
REV
A
SCALE: 1:4
SHEET 1 OF 1
8
7
6
5
4
3
2
1
D
C
B
A

Appendix D

D1 SELF-CONTAINED ENDCAP WIRE DIAGRAM

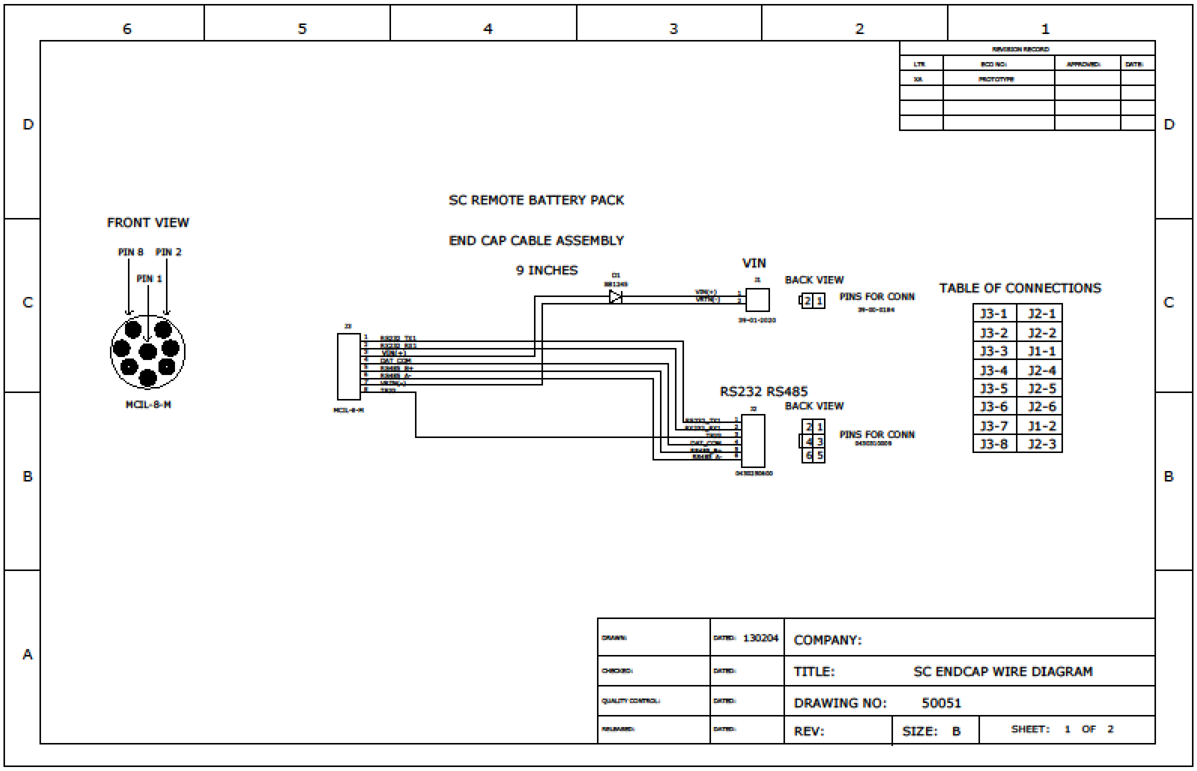
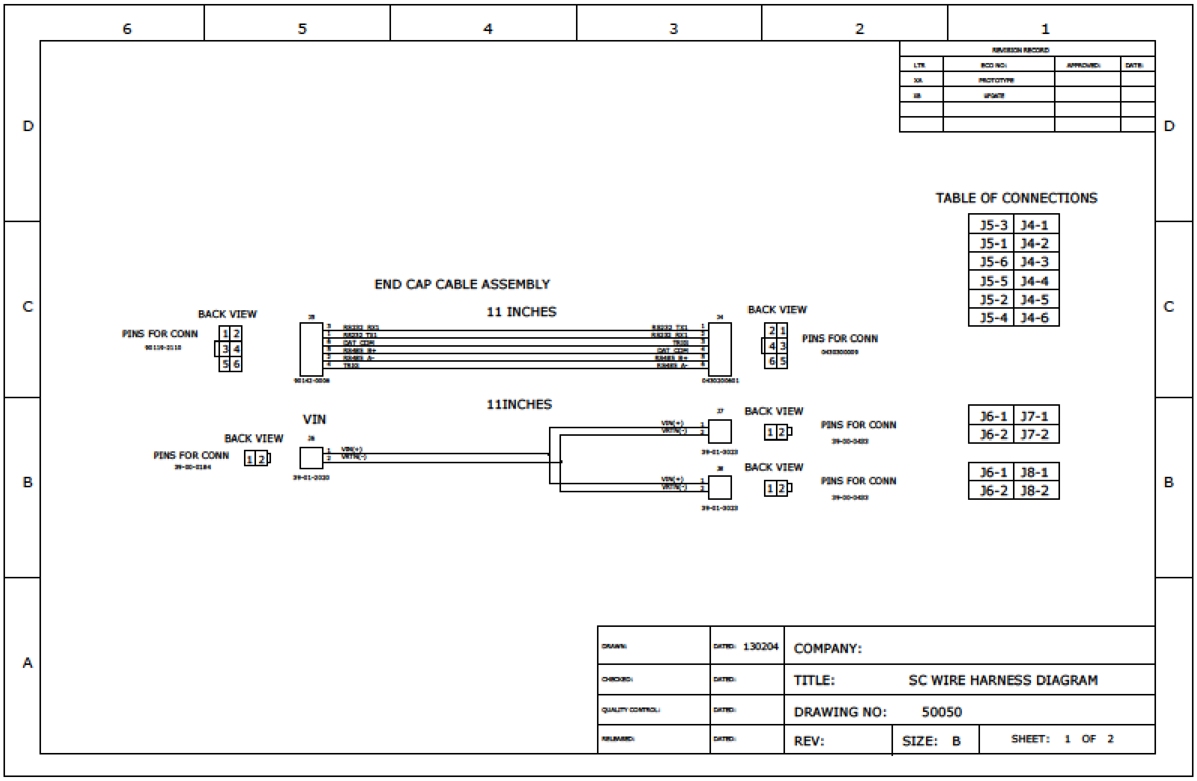


TABLE OF CONNECTIONS

J3-1	J2-1
J3-2	J2-2
J3-3	J1-1
J3-4	J2-4
J3-5	J2-5
J3-6	J2-6
J3-7	J1-2
J3-8	J2-3

DRAWN:	DATED: 130204	COMPANY:	
CHECKED:	DATED:	TITLE: SC ENDCAP WIRE DIAGRAM	
QUALITY CONTROL:	DATED:	DRAWING NO: 50051	
RELEASED:	DATED:	REV:	SIZE: B SHEET: 1 OF 2

D2 SELF-CONTAINED WIRE HARNESS DIAGRAM



REVISION RECORD			
LTR	ECO NO:	APPROVED:	DATE:
XA	PROTOTYPE		
XB	UPDATE		

TABLE OF CONNECTIONS

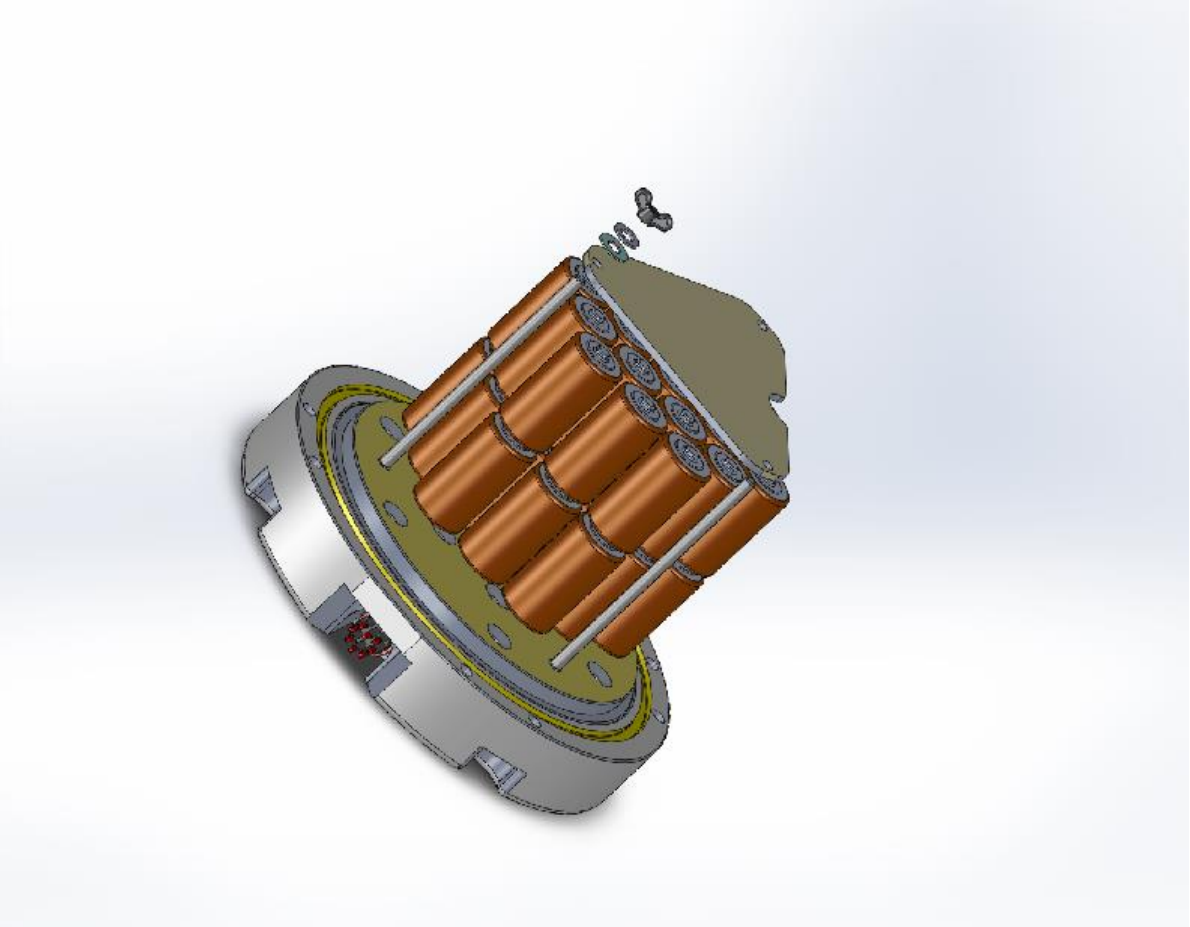
J5-3	J4-1
J5-1	J4-2
J5-6	J4-3
J5-5	J4-4
J5-2	J4-5
J5-4	J4-6

J6-1	J7-1
J6-2	J7-2

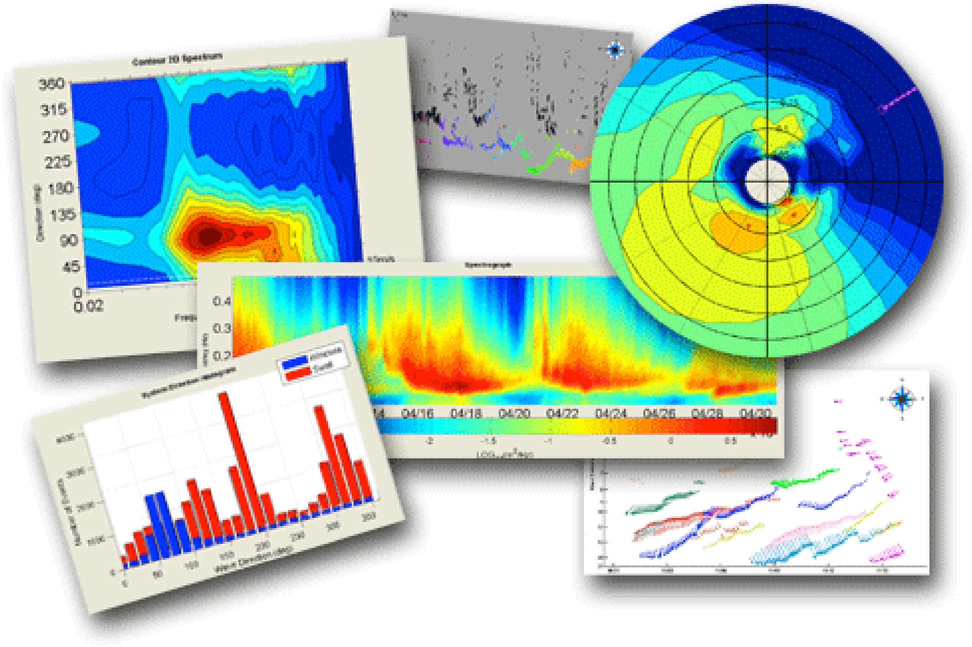
J6-1	J8-1
J6-2	J8-2

DRAWN:	DATED: 130204	COMPANY:	
CHECKED:	DATED:	TITLE: SC WIRE HARNESS DIAGRAM	
QUALITY CONTROL:	DATED:	DRAWING NO: 50050	
RELEASED:	DATED:	REV:	SIZE: B SHEET: 1 OF 2

D3 SELF-CONTAINED ENDCAP BATTERY ASSY



Appendix E



E1 Measuring Waves with RTI SeaWAVES

RTI SeaWAVES equipped ADCP with vertical beam

Rowe Technology Inc. [RTI] (www.rowetechinc.com) ADCP’s are designed with an ultimate built-in diversity. This allows them to be used in many different applications. When used as a stationary device, with proper memory, a pressure sensor and external power, all RTI ADCP’s/DVL’s can measure waves in both near-real time or at the end of an extended deployment.

RTI Pulse software is used to set up the ADCP for deployment. WaveForce Technology's Wavector software is used to process and calculate the wave parameters from the data.

E2 Methodology Used To Measure Waves

ADCP's can be used to collect data that can be processed to obtain the standard Wave parameters, Hs; significant height, Tp: period and direction.

We collect data that we want and we use the PUV processing method. A simplified explanation of the PUV method, is that every ping cycle of the radial beam velocities and surface height is measured using the four orthogonal beams and the pressure sensor. By combining the beam measurements to determine the velocity of the water in the U and V directions and the surface height measured by the pressure sensor, the height, period and direction can be calculated. In every measurement there is uncertainty, and the highest uncertainty determines the ultimate uncertainty in the measurement.

The best method to improve the measurement is to reduce the uncertainty in the pressure-sensor-based height measurement. This can be done by either using the slant beam echo intensity data to find the surface, or by using a vertical beam to directly measure the surface displacement. The industry believes the more reliable method is to directly measure the surface. This measurement typically has an order of magnitude increase in accuracy, thus drastically improving surface measurement. Data is typically collected once an hour, for about a 17-minute period at 2 Hz. This yields the minimum 2048 samples required for FFT processing.

The RTI ADCP/DVL simultaneously measures wave spectra, wave direction, and complete current profiles, for every ping. It can also measure current profiles during the wave burst interval, so you can get full coverage during your deployment.

When the ADCP/DVL is configured for near real time measurements, an underwater cable is attached to a shore-side computer system to process the 2048 samples once an hour- to provide both the wave statistics, as well as the current profile data. When deployed for extended periods, the current profile data from the 4-beam configuration, the fifth beam surface height data and the pressure sensor are processed and stored onboard the ADCP/DVL for wave processing at the deployment period.

E3 System Requirements:

On an RTI ADCP/DVL with a pressure sensor, a Vertical fifth beam is recommended; pressure sensors are not great at directly measuring the true height of the surface, so a fifth vertical beam is recommended.

This beam directly measures the height of the surface above the ADCP with high resolution and high accuracy. This greatly improves the measurement. The RTI ADCDP/DVL can use both a vertical beam and a pressure sensor (0.05%) to accurately measure range to the surface. The vertical beam functions similarly to an inverted echo sounder providing more accurate, higher frequency waves measurements.

A Large Aperture vertical beam -- the RTI ADCDP/DVL uses a full-size transducer [same size as the 4 ADCP beams] for its fifth vertical beam, providing a narrower radiation pattern for more accurate measurements of surface range and vertical water motion.

Data Storage -- the RTI ADCP/DVL can accommodate up to 32GB of internal memory, allowing all raw current profile and pressure sensor data to be stored for more thorough post processing.

Date Interfaces -- In addition to the common serial interfaces that are supported by the RTI ADCP/DVL (RS232, RS485, RS422), it also supports a separate Ethernet interface, which can be used to speed data download. This Ethernet interface is typically used for a cable-based system when collecting and processing data in near time.

RTI also offers a fully integrated ADCP interface and a waves post processing software suite developed by Waveforce Technologies, in their Wavector and XWaves Ocean Wave Data Analysis software packages. (www.waveforcetechnologies.com)

E4 Tips For A Successful Deployment

Care must be taken in choosing a deployment site. Sometimes the best site for measuring waves may not be the best site for deploying an ADCP. You should have an idea of typical wave environmental conditions.

- **Knowledge of typical wave environment:** Having an idea of the expected sea conditions will assist in proper system set up. Attention to the details in the early stages will ensure best data measurements.
- **Bottom conditions:** The system must be mounted stable and as close to plumb as possible. Anchored gimbaled mounts are ideal.
- **Median currents:** Mount must be secure enough to resist ambient currents
- **Theft:** Any object with a surface witness is subject to theft. Diver deployable mounts are preferred.
- **Optimized location:** Care must be taken to locate the best location for deployment. Any location where there are nearby barriers or barrage materials should be avoided to prevent reflected waves from biasing the data. Areas subject to excessive silting, high ambient acoustical noise, or large EMI sources should be avoided.
- **Maintenance recovery:** Local conditions determine the maintenance period. Bio-fouling removal, battery replacement, data download are typically done during the maintenance.

Intervals may be as frequent as 30 days, or as long as 180 days depending on deployment conditions.

- **Barrier to navigation:** System must be mounted so as to not interfere with vessel navigation or harbor operations.
- **Bottom mount type:** There are many different types of commercial bottom mounts available, or you can build your own simple mount. Ferrous materials must be avoided to prevent biasing of the compass. Trawl-resistant mounts should be used in areas with high fishing or recreational traffic. Simple clump weighted fixtures can be used were there is minimal trawling events. Rowe Technologies does not supply the bottom mounts, but our technical staff can assist in the proper selection.
- **Cabling for power/data transmission:** If using a near real-time system, proper sizing and armoring of the cable must be done. Cables in the surf zone should be armored and properly anchored.
- **Data processing:** Once the samples have been collected the data must be sent to a computer for processing into currents and the standard wave parameters.
- **Data archiving:** While your interest may be only in near real-time wave conditions, it is recommended that the raw data be collected and archived for future use. The data you collect today may have future significance many years from now.
- **Permits and licensing:** Operators of scientific equipment in the coastal regions may be subject to local permitting and licensing requirements. It is the responsibility of the operator to ensure that all proper permits and licenses are obtained before deployment.

E5 Waves Performance:

Wave Height Hs, <1% of measured value

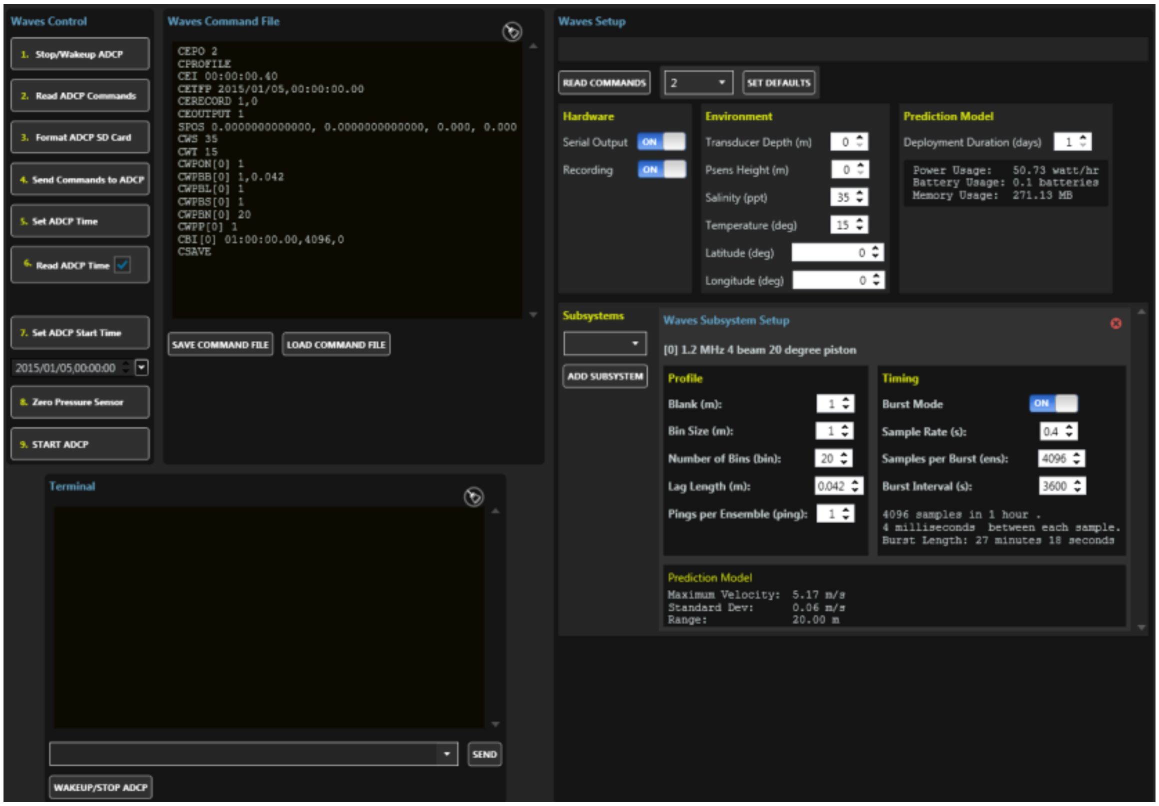
Period Tp, 1 -100s

Direction Dp, accuracy 2 degrees

Sample Rate 2Hz typical, including vertical beam

E6 Steps To Deploy:

1. Prepare ADCP. Perform any necessary maintenance.
2. Test System
3. Set up for Waves
4. Calibrate the System
5. Deploy System
6. Recover Data (near real-time) or recover system (self-contained) to upload data with Pulse Software.
7. Process data with Wavector Software.



2015/01/05,00:00:00

Waves Command File

```
CEPO 2
CPROFILE
CEI 00:00:00.40
CETFP 2015/01/05,00:00:00.00
CERECORD 1,0
CEOUTPUT 1
SPOS 0.0000000000000, 0.0000000000000, 0.000, 0.000
CWS 35
CWT 15
CWPON[0] 1
CWPBB[0] 1,0.042
CWPBL[0] 1
CWPBS[0] 1
CWPBN[0] 20
CWPP[0] 1
CBI[0] 01:00:00.00,4096,0
CSAVE
```

Waves Setup

2

Hardware

Serial Output ON

Recording ON

Environment

Transducer Depth (m) 0

Psens Height (m) 0

Salinity (ppt) 35

Temperature (deg) 15

Latitude (deg) 0

Longitude (deg) 0

Prediction Model

Deployment Duration (days) 1

```
Power Usage:   50.73 watt/hr
Battery Usage: 0.1 batteries
Memory Usage:  271.13 MB
```

Subsystems

Waves Subsystem Setup

[0] 1.2 MHz 4 beam 20 degree piston

Profile

Blank (m): 1

Bin Size (m): 1

Number of Bins (bin): 20

Lag Length (m): 0.042

Pings per Ensemble (ping): 1

Timing

Burst Mode ON

Sample Rate (s): 0.4

Samples per Burst (ens): 4096

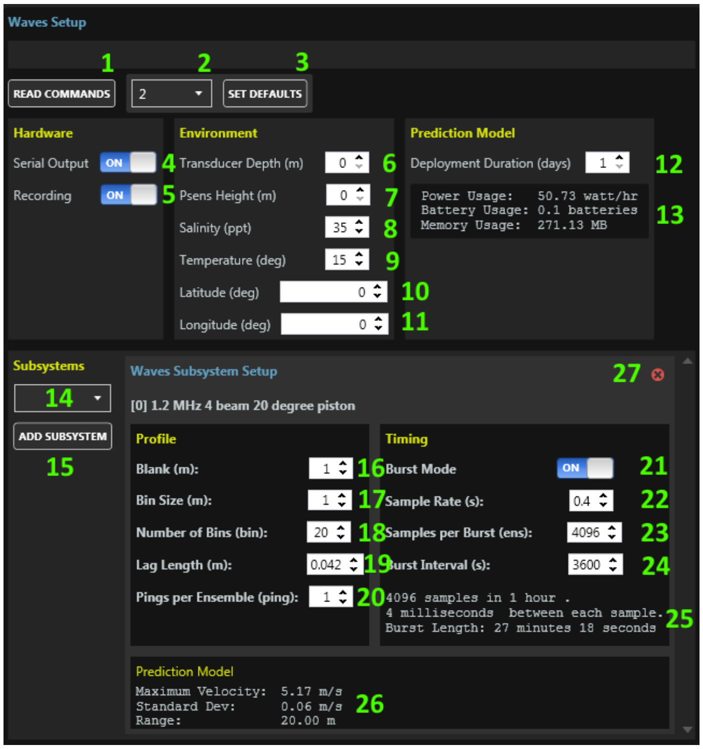
Burst Interval (s): 3600

```
4096 samples in 1 hour .
4 milliseconds  between each sample.
Burst Length: 27 minutes 18 seconds
```

Prediction Model

```
Maximum Velocity:  5.17 m/s
Standard Dev:      0.06 m/s
Range:             20.00 m
```

Sample Pulse Wave setup



2

Serial Output ON

Recording ON

Transducer Depth (m) 0

Psens Height (m) 0

Salinity (ppt) 35

Temperature (deg) 15

Latitude (deg) 0

Longitude (deg) 0

Deployment Duration (days) 1

```
Power Usage:   50.73 watt/hr
Battery Usage: 0.1 batteries
Memory Usage:  271.13 MB
```

[0] 1.2 MHz 4 beam 20 degree piston

Blank (m): 1

Bin Size (m): 1

Number of Bins (bin): 20

Lag Length (m): 0.042

Pings per Ensemble (ping): 1

Burst Mode ON

Sample Rate (s): 0.4

Samples per Burst (ens): 4096

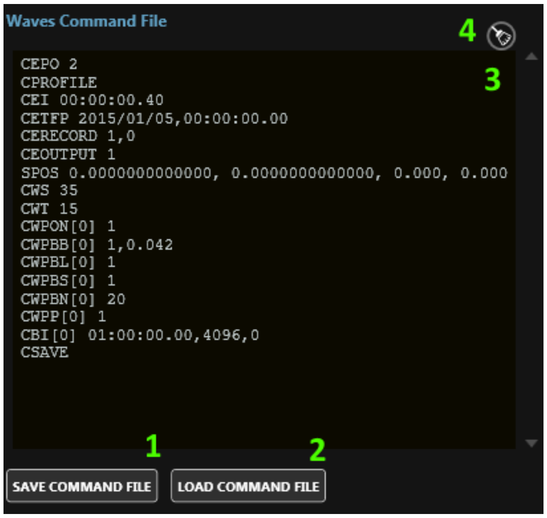
Burst Interval (s): 3600

```
4096 samples in 1 hour .
4 milliseconds  between each sample.
Burst Length: 27 minutes 18 seconds
```

Prediction Model

```
Maximum Velocity:  5.17 m/s
Standard Dev:      0.06 m/s
Range:             20.00 m
```

Customer controlled parameters

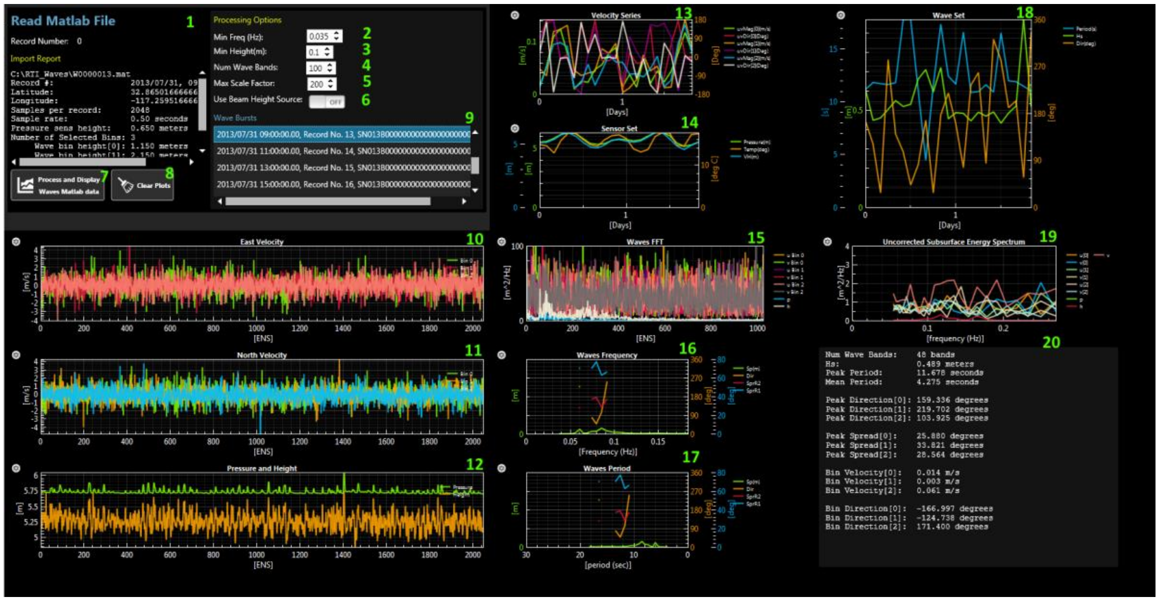


```
CEPO 2
CPROFILE
CEI 00:00:00.40
CETFP 2015/01/05,00:00:00.00
CERECORD 1,0
CEOUTPUT 1
SPOS 0.0000000000000, 0.0000000000000, 0.000, 0.000
CWS 35
CWT 15
CWPON[0] 1
CWPBB[0] 1,0.042
CWPBL[0] 1
CWPBS[0] 1
CWPBN[0] 20
CWPP[0] 1
CBI[0] 01:00:00.00,4096,0
CSAVE
```

Sample Waves deployment file

E7 Data Received From System:

The RTI ADCP/DVL simultaneously measures wave spectra, wave direction, and complete current profiles, for every ping. The RTI ADCP/DVL can also measure current profiles during the wave burst interval, so you can get full coverage during your deployment.



Read Matlab File

Record Number: 0

Import Report

```
C:\RTI_Waves\W0000013.mat
Record #:                 2013/07/31, 09
Latitude:                 32.86501666666
Longitude:                -117.259516666
Samples per record:       2048
Sample rate:              0.50 seconds
Pressure sens height:     0.650 meters
Number of Selected Bins:  3
     Wave bin height[0]:  1.150 meters
     Wave bin height[1]:  2.150 meters
```

Processing Options

- Min Freq (Hz): 0.035
- Min Height(m): 0.1
- Num Wave Bands: 100
- Max Scale Factor: 200
- Use Beam Height Source: OFF

Wave Bursts

- 2013/07/31 09:00:00.00, Record No. 13, SN013B0000000000000000000
- 2013/07/31 11:00:00.00, Record No. 14, SN013B0000000000000000000
- 2013/07/31 13:00:00.00, Record No. 15, SN013B0000000000000000000
- 2013/07/31 15:00:00.00, Record No. 16, SN013B0000000000000000000

```
Num Wave Bands:    48 bands
Hs:                0.489 meters
Peak Period:       11.678 seconds
Mean Period:       4.275 seconds

Peak Direction[0]: 159.336 degrees
Peak Direction[1]: 219.702 degrees
Peak Direction[2]: 103.925 degrees

Peak Spread[0]:    25.880 degrees
Peak Spread[1]:    33.821 degrees
Peak Spread[2]:    28.564 degrees

Bin Velocity[0]:   0.014 m/s
Bin Velocity[1]:   0.003 m/s
Bin Velocity[2]:   0.061 m/s

Bin Direction[0]:  -166.997 degrees
Bin Direction[1]:  -124.738 degrees
Bin Direction[2]:  171.400 degrees
```

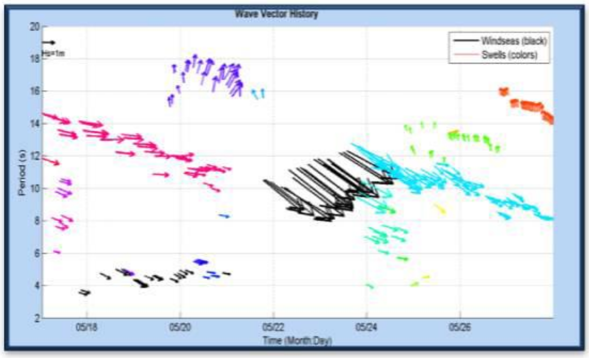
Sample of received data display

E8 Wavector Software

Key Features

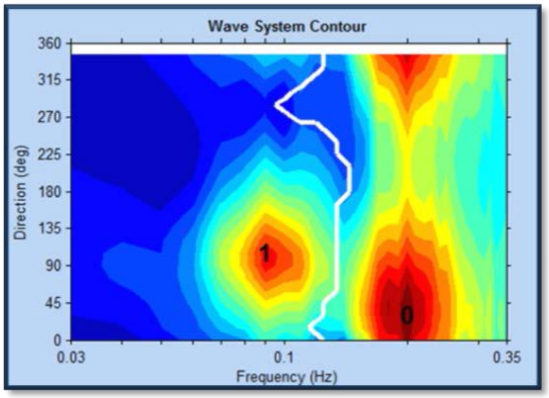
- Directly load wave sensor data (RTI SeaWAVE)
- Import model data (WAVEWATCH III, SWAN)
- Download from national wave archives (NODC, NDBC, and CDIP)
- Convenient data editing and smoothing utilities
- Powerful wind sea and swell system partitioning
- Wave system climatology plots
- Easily export movies and plots
- Save results in EXCEL, MATLAB, or CSV data files

E9 Wavector Plot Outputs

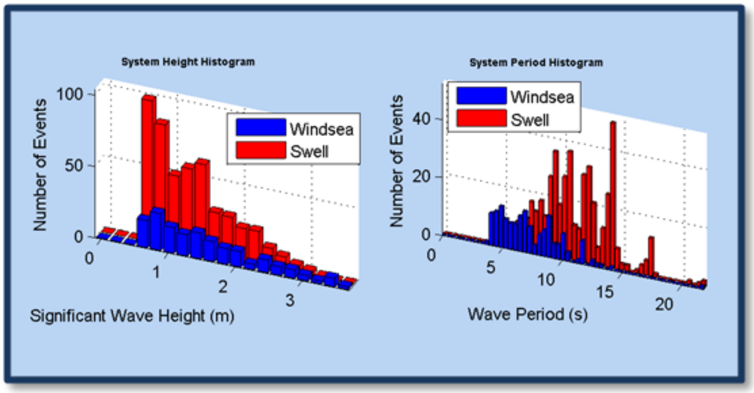


+

Wave System evolution (Vector Plot)



Wind Sea and Swell partitioning



Wave System Climatology