


## **Module 11**

# **Setup of Moored Instruments**

## Overview



### Setup of Moored Instruments

- In this module we will discuss:
  - Sample theory (briefly)
  - SeatermV2 and Seaterm, the user interface
  - Status reports, header listings, and data formats
  - Preparing for deployment
  - Conversion of data to a format used by SBE Data Processing

In this module we will discuss setup of moored instruments for deployment, considering sampling theory, the user interface, and instrument status reports.

By the end of this module, you should be able to:

- Understand what sort of time-scale phenomena you can observe with a given sampling rate
- Judge your instrument's readiness for deployment based on its status report
- Prepare your instrument for deployment
- Convert data from your instrument to a format that can be processed by SBE Data Processing

## The Sampling Theorem

**SBE**  
SCIENTIFIC BUSINESS ELECTRONICS

### Measuring Periodic Signals

- Sampling theorem:
  - Given a time or space varying signal,  $x(t)$ , where  $x(t)$  is bandlimited with  $X(\omega)=0$  for  $|\omega| > \omega_m$ .
  - Then  $x(t)$  will be uniquely determined by its samples  $x(nT)$ ,  $n = 0, \pm 1, \pm 2 \dots$
  - if  $\omega_s > 2\omega_m$  Where:  $\omega_s = 2\pi/T$
- Sampling theorem in English:
  - You can observe changes in parameters that occur only half as fast as you are sampling
  - This holds true for distance as well as time

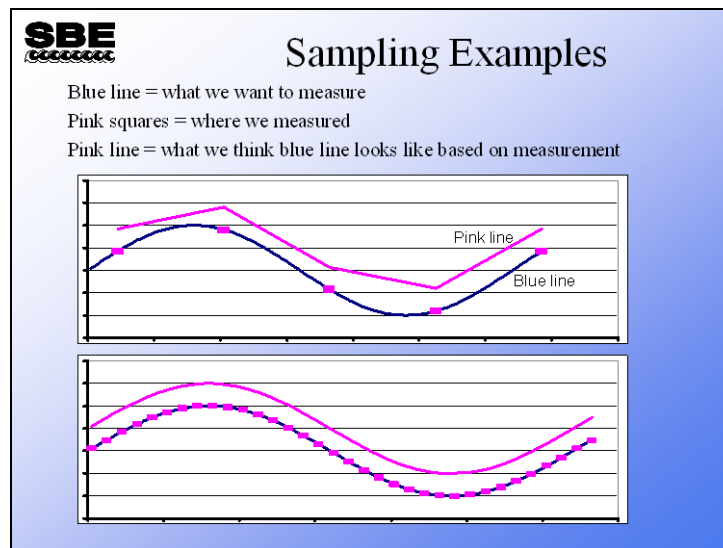
It is rather difficult to understand what this has to do with making measurements in the ocean by looking at the math.

The sampling theorem tells us how much information we can extract from a signal, given the rate at which we sample. In this case, the signal is any of the parameters we measure in the ocean. The sample rate is the rate at which the instrument makes a measurement or takes a scan.

The sampling theorem states *the highest frequency of information we can extract from a signal is half the frequency that we sample*. This means that if you sample at once an hour you can resolve signals that occur with a 2 hour period.

This holds true for distance as well as time. If we place instruments 10 meters apart on our mooring, we can resolve signals that have a length scale of 20 meters.

## Sampling Theorem: Sampling Examples




### ***Top Illustration - Under Sampled:***

Suppose we wish to measure a sine wave that has 16 cycles per hour. However, our sampling equipment is only capable of taking a measurement 5 times per hour. The upper line is our estimate of what the sine wave looks like.

### ***Bottom Illustration - Properly Sampled:***

This signal is properly sampled; the upper line is a complete representation of the lower line.

## Resolution and Sampling Theory




**Resolution and Sampling Theory**

- You can only resolve events that happen at half the frequency you sample
- You can only resolve length scales that are twice the size of your mooring spacing

In the profiling section of the course, you will recall we spent some time considering what sort of events or characteristics in the ocean we could expect to resolve given the sensor characteristics, sampling rates, and the lowering speed of the instrument package. The situation with moored instruments is simpler by virtue of the fact that they stay in one place (hopefully) and their sampling rates are usually slow enough that sensor characteristics are not an issue. Thus, you can resolve events that occur on time scales of 10 seconds to 18 hours given sample rates of 5 seconds to 9 hours.

In the length domain, the characteristics you can resolve depend on the spacing of instruments on the mooring and the space between moorings. As in the time domain, you can resolve characteristics that occur on scales that are twice your spacing on the mooring.

## SeatermV2 and Seaterm: User Interface



**SeatermV2 and Seaterm**

- SeatermV2 and Seaterm provide means to:
  - Set up an instrument for deployment
  - Transfer data from an instrument to user's computer
  - Do data conversion for SBE 37 and SBE 39 to allow further processing with Seasoft's data processing software (SBE Data Processing)

Even in moorings that use inductive telemetry, the data that is stored in the instrument is of interest. There may be occasions where the link between the surface buoy and the ground station malfunctioned, or the telemetered data may be a subset of that collected by the remote instrument. SeatermV2 and Seaterm provide the facility for data transfer and for the conversion of SBE 37 and SBE 39 data to a format that can be used by the processing programs in Seasoft.

## Setting up SeatermV2 for Communications

**SBE**  
SeatermV2

- SeatermV2 is a *launcher*
- Each individual terminal program provides the means to:
  - Set up an instrument for deployment
  - Transfer data from an instrument to the computer
  - Do data conversion for SBE 37 to allow further processing with Seasoft

SeatermV2 launches one of the following programs:

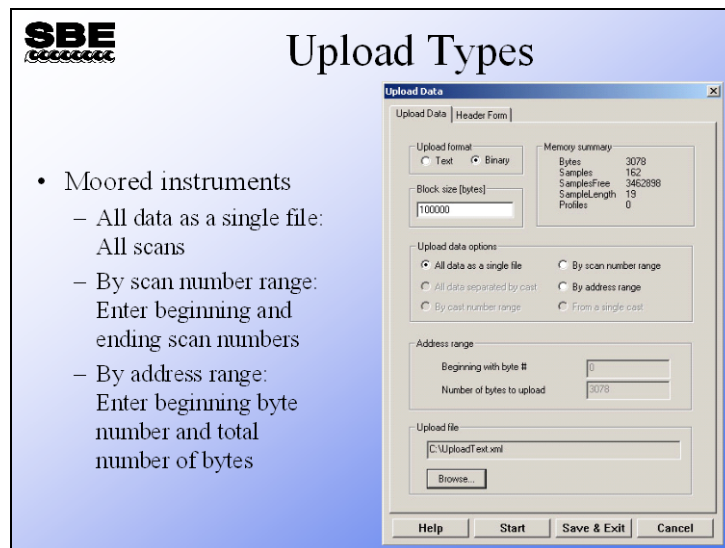
- **Seaterm232** – for instruments that communicate via **RS-232** (standard serial communications)
- **SeatermIM** – for instruments that communicate via **Inductive Modem** telemetry (instruments deployed in parallel, with each instrument assigned a unique ID)
- **Seaterm485** – for instruments that communicate via **RS-485** (4-wire serial communications, instruments placed in parallel on the communications cable, with each instrument assigned a unique ID)

The SBE 37 and 16*plus* V2 are available in models to support each of the three types of communication.

**Notes for Older instruments:** Use Seaterm instead of SeatermV2. Select the instrument in the Configure menu by to set up communication parameters; enter the parameters in the dialog box:

- For the SBE 16, older versions of firmware (EPROM version) do not support a shift to higher baud rate during data transfers. If the firmware selection is not properly set, the data transfer will fail.
- The SBE 16*plus*, 16*plus* V2, 37, and 39 do not support the shift to higher baud rate for data transfer; the user sets the baud rate with the **Baud=** or **BaudRate=** command (see instrument manual for appropriate command). Also, only 8 data bits, 1 stop bit, and no parity is used for these instruments.

## SeatermV2: Upload Types



- Moored instruments
  - All data as a single file:  
All scans
  - By scan number range:  
Enter beginning and ending scan numbers
  - By address range:  
Enter beginning byte number and total number of bytes

**Upload Format:** Seaterm232 and Seaterm485 can upload in text or binary. Binary is approximately twice as fast; the resulting output file is the same, regardless of which upload method you use.

Note: Binary upload is not available with SeatermIM.

**Upload Data Options:** Profiling instruments usually transfer data in casts. Moored instruments typically dump all their stored data at once. Data from an internally recording profiling instrument can be transferred to your computer in one of three ways:

- **All scans:** All scans in the instrument are transferred into 1 file.
- **By scan range:** Enter the range in the dialog box. This is useful when only part of the data is desired.
- **Address range:** Enter the range in bytes in the dialog box. This is useful for diagnostics performed by Sea-Bird.

**Note for newer 37-IM/IMP (acquisition firmware  $\geq$  3.0) and all 37-IMP-IDO and 39-IM:**

Data can be uploaded via RS-232 by connecting to an internal connector, providing much faster upload of large data sets. See the instrument manual for connection and upload details.



## SBE 16plus V2 Status Report

```

SBE
SBE 16plus V2 Status

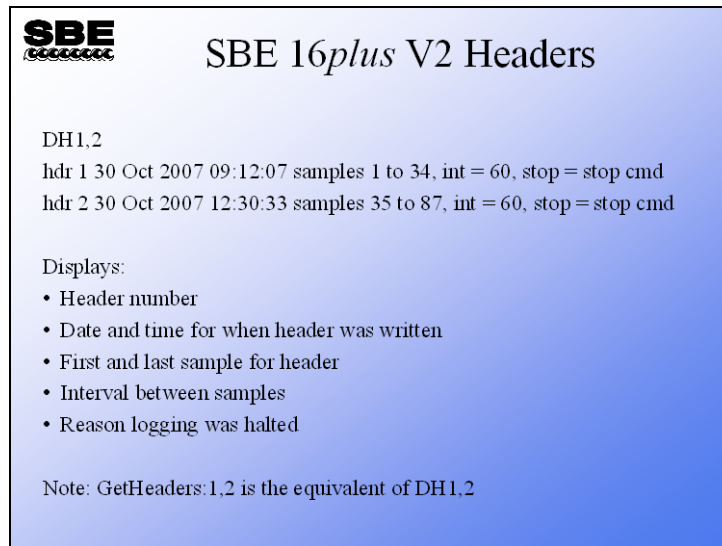
DS
SBE 16plus V 2.0b SERIAL NO. 6001 24 Feb 2008 14:11:48 [DateTime=]
vbatt = 10.3, vliht = 8.5, ioper = 62.5 ma, ipump = 21.6 ma, iext01 = 76.2 ma,
status = not logging
samples = 0, free = 3463060
sample interval=15 seconds, number of measurements per sample=1 [SampleInterval=, NCycles=]
pump=run pump during sample, delay before sampling=2.0 sec [PumpMode=, DelayBeforeSampling=]
transmit real-time = yes [TxRealTime=]
battery cutoff = 7.5 volts
pressure sensor = strain gauge, range = 1000.0 [PType=, PRange=]
SBE 38=no, SBE 50=yes, Gas Tension Device=no [SBE38=, SBE50=, GTD=, DualGTD=]
Ext Volt 0 = yes, Ext Volt 1 = no [Volt0=, Volt1=]
Ext Volt 2 = no, Ext Volt 3 = no [Volt2=, Volt3=]
Ext Volt 4 = no, Ext Volt 5 = no [Volt4=, Volt5=]
echo characters = yes [Echo=]
output format = raw HEX [OutputFormat=]
serial sync mode disabled [SyncMode=]

```

The 16plus V2 status report contains much valuable information:

- Instrument type, firmware revision, serial number, date and time.
- Main battery voltage and back-up battery voltage. Operating current, pump current, and current to external voltage sensors provide an indication of the battery lifetime you can expect, as well as the health of the instrument in general.
- The status entry indicates whether the instrument is logging data or not.
- Number of samples stored and amount of space free.
- Sample interval and number of measurements that are averaged to make a sample.
- Parameters related to the pump and auxiliary sensors –
  - pump mode (no pump, run for 0.5 sec before each sample, or run during each sample)
  - delay before sampling (seconds after switching on power to auxiliary sensors before measurement is made)
- Transmit real-time data or not.
- Battery cut-off voltage, which is the power level that causes the instrument to shut itself off because it does not have sufficient power to sample.
- Pressure sensor type and range are entered at the factory.
- Which auxiliary sensor channels are enabled and logged with the CTD data.
- Whether to show entered commands on screen as you type.
- Output data format
- Serial synchronization mode enabled or disabled.


## SBE 16*plus* V2 Headers



The image is a screenshot of a document titled "SBE 16*plus* V2 Headers". In the top left corner, there is the SBE logo, which consists of the letters "SBE" in a bold, sans-serif font above a stylized wave graphic. The title "SBE 16*plus* V2 Headers" is centered at the top. Below the title, the text "DH1,2" is followed by two lines of header information: "hdr 1 30 Oct 2007 09:12:07 samples 1 to 34, int = 60, stop = stop cmd" and "hdr 2 30 Oct 2007 12:30:33 samples 35 to 87, int = 60, stop = stop cmd". Underneath this, the word "Displays:" is followed by a bulleted list of four items: "Header number", "Date and time for when header was written", "First and last sample for header", and "Reason logging was halted". At the bottom of the screenshot, a "Note:" states that "GetHeaders:1,2 is the equivalent of DH1,2".

Header entries have start date and time, sample range, and sample interval as well as the reason for ending data collection.

## Data Format: SBE 16



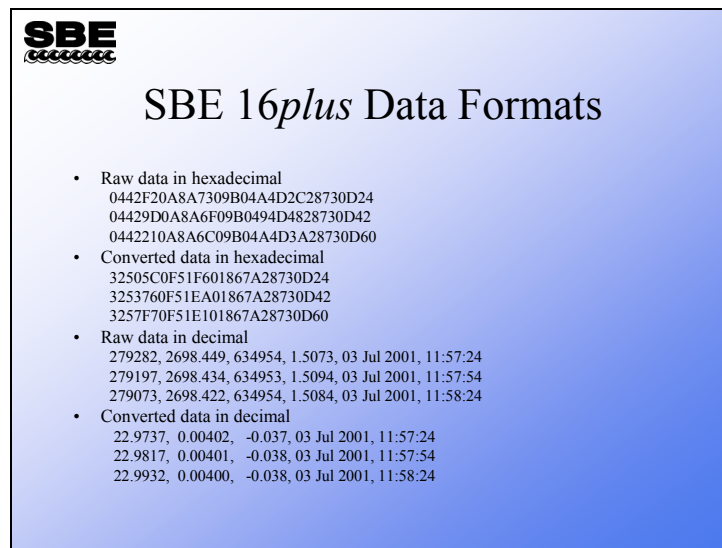
SBE 16 Data

- Data is in hexadecimal format, T and C only
- Adding auxiliary sensors add bytes to the scan

```
S>dd31,10
8A7503DB
8A7D03DB
8A9203DB
8A8D03DB
8ABB03DB
8AD803DB
8AE103DB
8AF603DC
8B0503DC
8B0603DB
S>
```

This data is an ASCII representation of hexadecimal numbers. This representation allows the data file to be readable with any text editor.

## Data Format: SBE 16*plus*



A complete description of these formats is found in the manual.

- **Raw data in hexadecimal example:**

SBE 16*plus* with strain gauge pressure sensor and two external voltages sampled,  
 example scan = tttttccccccppppppvvvvvvvvvvvsssssss  
 = 0A53711BC7220C14C17D820305059425980600

- **Converted data in hexadecimal example:**

SBE 16*plus* with strain gauge pressure sensor and two external voltages sampled,  
 example scan = tttttccccccppppppvvvvvvvvvvvsssssss  
 = 3385C40F42FE0186DE0305059425980600

Note that some of the hexadecimal numbers are scaled. For example, in the raw hexadecimal data, the conductivity frequency is multiplied by 256. In the converted hexadecimal data, temperature is multiplied by 100,000 and conductivity by 1,000,000.


- **Raw data in decimal example:**

SBE 16*plus* with strain gauge pressure sensor and two external voltages sampled,  
 example scan =  
 ttttt, cccc.ccc, pppppp, v.vvvv, v.vvvv, v.vvvv, dd mmm yyyy, hh:mm:ss  
 = 676721, 7111.133, 791745, 2.4514, 0.0590, 0.1089, 12 nov 2000, 12:23:05

- **Converted data in decimal:**

SBE 16*plus* with strain gauge pressure sensor and two external voltages sampled,  
 example scan = ttt.tttt, cc.ccccc, pppp.ppp, v.vvvv, v.vvvv, dd mmm yyyy, hh:mm:ss  
 = 23.7658, 0.00019, 0.062, 0.0590, 0.1089, 12 nov 2000, 12:23:05

## Data Format: SBE 16plus V2



### SBE 16plus V2 Data Formats

- Raw data in hexadecimal  
0442F20A8A7309B04A4D2C28730D24  
04429D0A8A6F09B0494D4828730D42
- Converted data in hexadecimal  
32505C0F51F601867A28730D24  
3253760F51EA01867A28730D42
- Raw data in decimal  
279282, 2698.449, 634954, 1.5073, 03 Jul 2001, 11:57:24  
279197, 2698.434, 634953, 1.5094, 03 Jul 2001, 11:57:54
- Converted data in decimal  
22.9737, 0.00402, -0.037, 03 Jul 2001, 11:57:24  
22.9817, 0.00401, -0.038, 03 Jul 2001, 11:57:54
- Converted data in XML  

```
<?xml?><datapacket><hdr><mfg>Sea-Bird</mfg><model>16plus</model>
<sn>1606001</sn></hdr><data><t1> 22.9737 </t1><c1> 0.00402 </c1>
<p1> -0.037 </p1><dt>2007-11-07T07:34:35</dt></data></datapacket>
```

A complete description of these formats is found in the manual.

- **Raw data in hexadecimal example:**

With strain gauge P and 2 external voltages sampled, example scan =  
 tttttccccppppppvvvvvvvvvvvsssssss = 0A53711BC7220C14C17D82030505940EC4270B

- **Converted data in hexadecimal example:**

With strain gauge P and 2 external voltages sampled, example scan =  
 tttttccccppppppvvvvvvvvvvvsssssss = 3385C40F42FE0186DE030505940EC4270B

Note: some of the hex numbers are scaled. For example, in the raw hex data, conductivity frequency is multiplied by 256. In the converted hex data, T is multiplied by 100,000 and C by 1,000,000.

- **Raw data in decimal example:**

With strain gauge P and 2 external voltages sampled, example scan =  
 tttttt, cccc.ccc, pppppp, v.vvvv, v.vvvv, v.vvvv, dd mmm yyyy, hh:mm:ss  
 = 676721, 7111.133, 791745, 2.4514, 0.0590, 0.1089, 7 nov 2007, 07:34:35


- **Converted data in decimal:**

With strain gauge P and 2 external voltages sampled, example scan =  
 ttt.tttt, cc.ccccc, pppp.ppp, v.vvvv, v.vvvv, dd mmm yyyy, hh:mm:ss  
 = 23.7658, 0.00019, 0.062, 0.0590, 0.1089, 7 nov 2007, 07:34:35

- **Converted data in XML:**

With strain gauge P and 2 external voltages sampled, example scan =  
 <?xml?><datapacket><hdr><mfg>Sea-Bird</mfg><model>16plus</model>  
 <sn>nnnnnnnn</sn></hdr><data><t1> ttt.tttt </t1><c1> cc.ccccc </c1>  
 <p1> pppp.ppp </p1><v0> v.vvvv </v0><v1> v.vvvv </v1>  
 <dt>yyyy-mm-ddThh:mm:ss</dt></data></datapacket>  
 =  
 <?xml?><datapacket><hdr><mfg>Sea-Bird</mfg><model>16plus</model>  
 <sn>1606001</sn></hdr><data><t1>23.7658</t1><c1>0.00019</c1>  
 <p1>0.062</p1><v0>0.0590</v0><v1>0.1089</v1>  
 <dt>2007-11-07T07:34:35</dt></data></datapacket>

## Data Format: SBE 37



### SBE 37 Data Formats

- **Firmware version < 3.0** (SM, SMP, SI, SIP)
  - Format=1: 23.0161, 0.00002, 0.076, 08 Mar 2008, 17:10:23
  - Format=2: 23.0161, 0.00002, 0.076, 03-08-2008, 17:10:23
- **Firmware version  $\geq$  3.0** (SM, SMP, SI, SIP, IM, IMP)
  - OutputFormat=1:  
23.0161, 0.00002, 0.076, 08 Mar 2008, 17:10:23
  - OutputFormat=2 (SM, SMP, SI, SIP) or 4 (IM, IMP):  

```
<?xml version="1.0"?><datapacket><hdr><mfg>Sea-Bird</mfg>
<model>37SM</model><sn>03709999</sn></hdr>
<data><t1> 23.0161</t1><c1> 0.00002</c1><p1> 0.076</p1>
<dt>2008-03-08T 17:10:23</dt></data></datapacket>
```

The basic data format is temperature, conductivity, pressure (if installed), date and time.

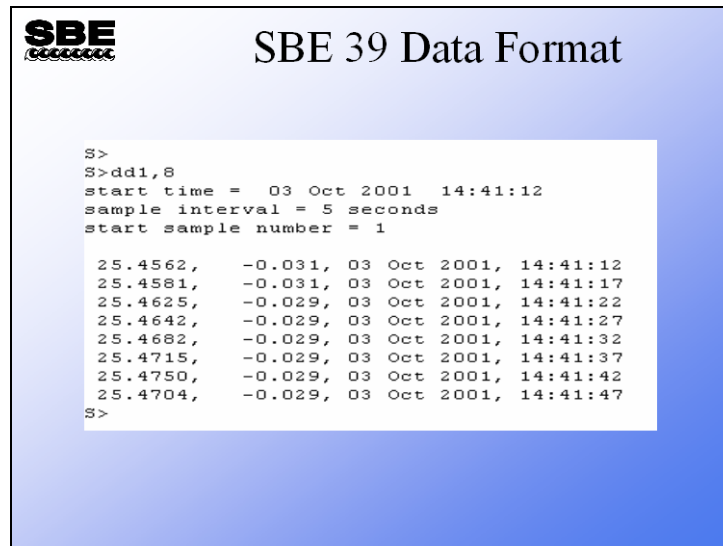
Note: IDO MicroCAT data format is not covered here.

**Older MicroCATs (firmware version < 3.0)** offer two date formats in the data output: dd mmm yyyy or mm-dd-yyyy. This is selected with the **Format=** command. There are some other differences in output format, depending on the specific MicroCAT model:

- 37-IM and 37-IMP: The **Format=** command also affects the output units for conductivity: S/m for **Format=1** and mS/cm for **Format=2**.
- 37-SM and 37-SMP: You can also output salinity and/or sound velocity if desired, with the use of the appropriate setup commands. You can suppress storage and output of date and time with the use of the appropriate setup command.
- 37-SI and 37-SIP: You can also output depth, salinity, sound velocity, and/or density sigma if desired, with the use of the appropriate setup commands. You can suppress output of date and time with the use of the appropriate setup command.

**Newer MicroCATs (firmware version  $\geq$  3.0)** eliminate one of the date format options, but add an XML output option. With the larger memory, date and time are always stored and output.

## Data Format: SBE 39



The screenshot displays the SBE 39 data format output. It shows a series of data points with columns for temperature, pressure, date, and time. The data is presented in a table-like format with commas separating the values.

```
S>
S>dd1,8
start time = 03 Oct 2001 14:41:12
sample interval = 5 seconds
start sample number = 1


25.4562, -0.031, 03 Oct 2001, 14:41:12
25.4581, -0.031, 03 Oct 2001, 14:41:17
25.4625, -0.029, 03 Oct 2001, 14:41:22
25.4642, -0.029, 03 Oct 2001, 14:41:27
25.4682, -0.029, 03 Oct 2001, 14:41:32
25.4715, -0.029, 03 Oct 2001, 14:41:37
25.4750, -0.029, 03 Oct 2001, 14:41:42
25.4704, -0.029, 03 Oct 2001, 14:41:47
S>
```

The SBE 39 data format is temperature, pressure (if installed), date and time.

With the SBE 39-IM, there are options for the date format, similar to the older MicroCATs.

The instrument manuals provide complete details on all the data format options.

## Activity



### Activity: Setup, Logging, and Uploading Data

- Cable your computer to an SBE 37-SM
- Open SeatermV2; select *SBE 37 RS232* to launch Seaterm232
- Send the commands shown in your notes
- Start logging with the **StartNow** command to collect a bench mooring
- Stop logging with the **Stop** command
- Check status with the **DS** command to verify that logging has stopped
- Upload and check out the data in the .XML data file

Type in the commands below (upper case or lower case, it does not matter) or send these commands using the Send Commands window, so everyone's SBE 37 is set up the same.


<i>Send Commands CATEGORY</i>	COMMAND	DESCRIPTION
General Setup	OutputExecutedTag=y	Enable output of executing and executed tags. This makes it easier to use the Send Commands window to transmit commands.
	DateTime=	Set current date and time, month day year hour minute second (mmddyyhhmmss; for example, 10 am on January 30, 2009 is 01302009100000).
	TxRealTime=y	Transmit sample data as it is taken (in <i>real-time</i> ).
Output Format Setup	OutputFormat=1	Output converted decimal data -- temperature, conductivity, pressure (if installed), s
	OutputSal=y	Calculate and output salinity.
	OutputSV=y	Calculate and output sound velocity.
Memory Setup	InitLogging	Reset memory (SBE 37 prompts you to send command twice).
Autonomous Sampling	SampleInterval=10	Set sampling interval to 10 seconds.
Serial Line Sync	SyncMode=n	Disable serial synchronization mode.
Status	DS	Verify setup with status command.

Note about uploaded .xml file: SeatermV2 uploads MicroCAT data into an .xml file.

There is one more step required to get a file with columns of data – converting the .xml file to a .cnv file. We will talk about this in the next few slides.



## Converting Sensor Output to Scientific Units

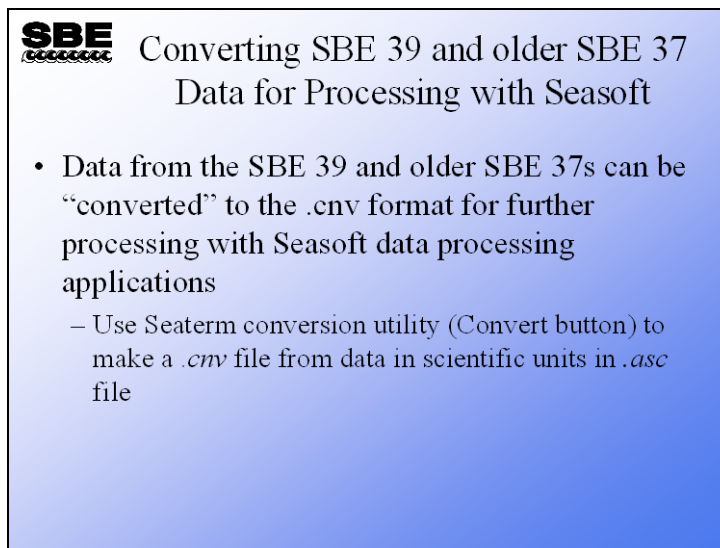


### Converting Sensor Output to Scientific Units

- SBE 16 stores and reports data as frequencies and voltages
  - Use Data Conversion module in SBE Data Processing
- SBE 16*plus* and 16*plus* V2 can report data in scientific units, or as frequencies and voltages
  - Data uploaded as hex, use Data Conversion module
- Newer SBE 37s – SeatermV2 uploads SBE 37 data to an .xml file, converts to a .hex file, and creates a .xmlcon configuration file
  - Use Data Conversion module

- The SBE 16 requires the same sort of data processing as was discussed earlier in the course, beginning with Data Conversion and ending with Bin Averaging.
- The SBE 16*plus* can report data in scientific units. However, when Seaterm’s toolbar is used to retrieve 16*plus* data, Seaterm sets the output format to raw hexadecimal, to provide a file that is compatible with SBE Data Processing. This data requires conversion with SBE Data Processing’s Data Conversion module.
- The SBE 16*plus*V2 can report data in scientific units. However, when SeatermV2’s Upload menu is used to retrieve 16*plus* V2 data, SeatermV2 sets the output format to raw hexadecimal, to provide a file that is compatible with SBE Data Processing. This data requires conversion with SBE Data Processing’s Data Conversion module.
- Newer SBE 37s (SM, SMP, IM, IMP, SI, SIP with firmware  $\geq 3.0$ , and all IDO versions) can report data in scientific units. However, when SeatermV2’s Upload menu is used to retrieve SBE 37 data, SeatermV2 uploads the data to a .xml file, converts the data to a .hex file, and creates a .xmlcon configuration file, to provide a data file and a configuration file that is compatible with SBE Data Processing. This data requires conversion with SBE Data Processing’s Data Conversion module.

## SBE 39 and older SBE 37 Data Conversion Utilities



**SBE** Converting SBE 39 and older SBE 37 Data for Processing with Seasoft

- Data from the SBE 39 and older SBE 37s can be “converted” to the .cnv format for further processing with Seasoft data processing applications
  - Use Seaterm conversion utility (Convert button) to make a .cnv file from data in scientific units in .asc file

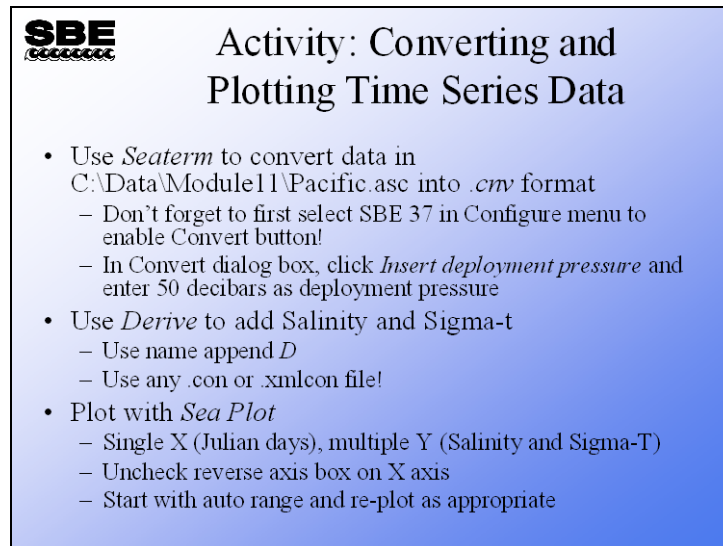
The SBE 37 and 39 can report data in scientific units.

Seaterm uploads data from SBE 39s and older SBE 37s in scientific units, to a .asc data file. This file format is not compatible with Seasoft’s post-processing modules. If you wish to do any processing of this data with SBE Data Processing, you have to convert it to the .cnv file format. There is a utility in Seaterm to perform this conversion; click the Convert button on Seaterm’s toolbar to create a .cnv file.

- To access the Convert button in Seaterm, you must first select the SBE 37 or SBE 39 in the Configure menu.

The .cnv file can then be processed and/or plotted using SBE Data Processing’s modules.

## Activity




**SBE**  
Activity: Converting and Plotting Time Series Data

- Use *Seaterm* to convert data in C:\Data\Module11\Pacific.asc into .cmv format
  - Don't forget to first select SBE 37 in Configure menu to enable Convert button!
  - In Convert dialog box, click *Insert deployment pressure* and enter 50 decibars as deployment pressure
- Use *Derive* to add Salinity and Sigma-t
  - Use name append *D*
  - Use any .con or .xmlcon file!
- Plot with *Sea Plot*
  - Single X (Julian days), multiple Y (Salinity and Sigma-T)
  - Uncheck reverse axis box on X axis
  - Start with auto range and re-plot as appropriate

Note that this exercise is using *Seaterm*, **not** *SeatermV2*, because it is based on real data uploaded from an older SBE 37.

A note about configuration (.con or .xmlcon) files and data from SBE 37 MicroCATs: The SBE 37 is not user-configurable (i.e., it cannot interface with auxiliary sensors), and it outputs converted data in engineering units (using calibration coefficients programmed into the instrument). Therefore, it does not have a configuration file (remember that the configuration file defines what sensor data is in the data stream and also defines the calibration coefficients). But, SBE Data Processing was originally written to process data from instruments that do have a configuration file, and the *Derive* module therefore requires the selection of a .con or .xmlcon file before it will allow you to select the variables to be derived. You can select any .con or .xmlcon file; the selection will have no impact on the results.

## Preparing for Deployment




**Preparing for Deployment**

- Install fresh batteries
- Clear memory of old data
- Check temperature and conductivity and other sensors
- Set internal clock
- Set ID for instruments using inductive modems or RS-485
- Check all cabling, and lubricate and reset underwater connections

We are going to discuss preparation for deployment next. Some of these items need little explanation, while others require more consideration.

Sea-Bird instruments stop recording data when the memory is full. It is very important to clear the memory before you start.

## Preparing for Deployment: Sensor Check




**Pre-Deployment Sensor Check**

- Use the **TS** command to check nominal functionality of all sensors
- Check conductivity zero frequency against calibration sheet

Before you invest a great deal of time and effort deploying a mooring, it is a great idea to send your instrument in for calibration. It is an excellent check on its functionality, not to mention its accuracy. If you are unable to do this, check to see that all the sensors are presenting nominally correct output. Sea-Bird moored instruments, with the exception of the SBE 16, have calibration coefficients stored internally and are able to output measurements in scientific units.

The best spot check for the conductivity cell is to observe its zero conductivity frequency. To do this, rinse the cell in distilled or de-ionized water and shake any remaining drops out. The sensor should read very close to the zero conductivity reading on the calibration sheet. If it does not, the cell may be dirty or damaged. Try cleaning it with a non-ionic detergent such as Triton X.

## Preparing for Deployment: Clocks



### Clock Drift

- All internal clocks use a 32,767 Hz crystal oscillator
  - 32,767 is divided by a binary counter for a 1 Hz clock tick
- The frequency of the crystal is affected by temperature
- Old SBE 37 and old SBE 39 clocks are calibrated and corrected for the temperature effect
- SBE 16*plus* and 16*plus* V2, and newer SBE 37 and SBE 39 offer a temperature-compensated oscillator with extremely low drift

All clocks drift. If you put 10 instruments on a mooring, all set to the same time, and come back 1 year later, they will all report different times. This can lead to some difficulties when you process the data. Clocks drift because of instability of the crystal oscillator that the clocks use to keep time. A small change in oscillator frequency will translate into a clock that runs fast or slow.

For this reason, Sea-Bird calibrates the clock frequency versus temperature and corrects for drift each time an older SBE 37 or older 39 wakes to take a sample. To ensure the time between samples is consistent, each time the instrument wakes and takes a sample, the time to the next sample is set to be the calibrated time interval between samples, not the next clock time. This ensures a data series that is evenly spaced in time. Otherwise, the time between samples would vary depending on temperature's effect on the oscillator.

The SBE 16*plus* and 16*plus* V2, and newer SBE 37s (all IDOs; others with firmware version  $\geq 3.0$ ) and SBE 39s, use a temperature-compensated oscillator that provides a very stable time base. Previously these were not available in a small enough package to make them practical for a moored instrument.


## Preparing for Deployment: Cables



### Check Your Connectors and Cables

- Check all the cabling
- Remove the cables from their bulkhead connectors
  - Inspect the bulkhead connectors and mating pieces
  - Clean and lubricate the bulkhead connectors
  - Burp all air out of the connectors when they are reseated

## Preparing for Deployment: SBE 16*plus* and 16*plus* V2 Pump Operating Modes



SBE 16*plus* and 16*plus* V2  
Pump Modes

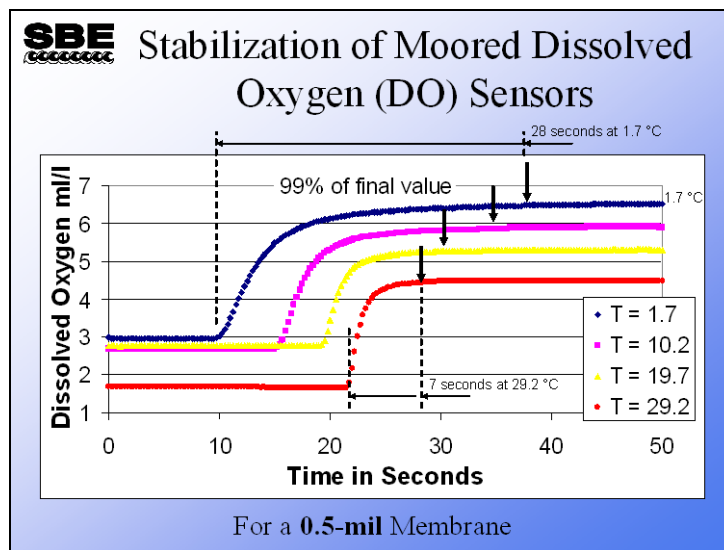
- No pump
  - No mystery here
- Pump for 0.5 seconds before sample
  - T & C only, ventilates the cell, saves power
- Pump on for entire sample period
  - For auxiliary sensors with long time constants
    - Dissolved oxygen
    - Fluorometer

Experience with the SBE 41 Argo CTD has led Sea-Bird to adopt the protocol of ventilating the conductivity cell for a short time before measuring the conductivity, and then turning the pump off. This saves power and has the additional benefit of reducing bio-fouling inside the cell while the pump is off and the instrument is waiting for the next sample. With this protocol the power requirement is small enough to allow long-term deployment at reasonably short sampling intervals (e.g., 15 minutes).

If you are using sensors with a comparatively long time constant, run the pump throughout the sample interval. Recall from the slide that showed the 16*plus* status report, you can set the delay from the start of the sampling period to the moment when the auxiliary sensor is measured. This allows you to give a slow sensor 5 or 6 time constants to equilibrate.

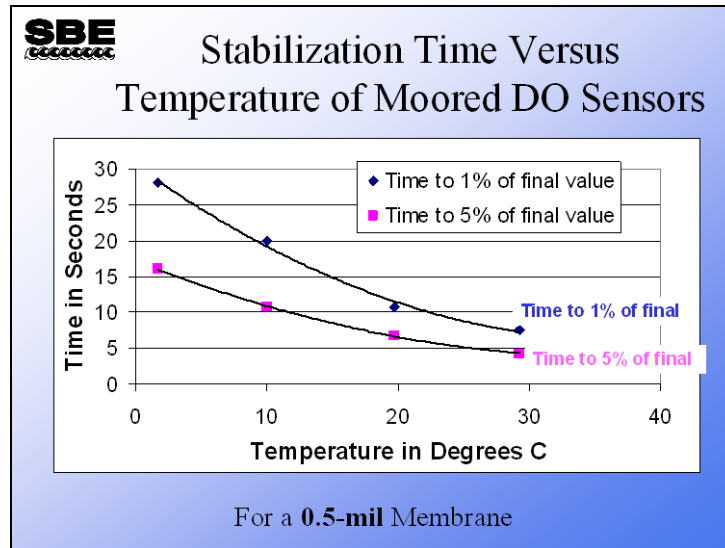


## Preparing for Deployment: Setting Up for Dissolved Oxygen Sensors



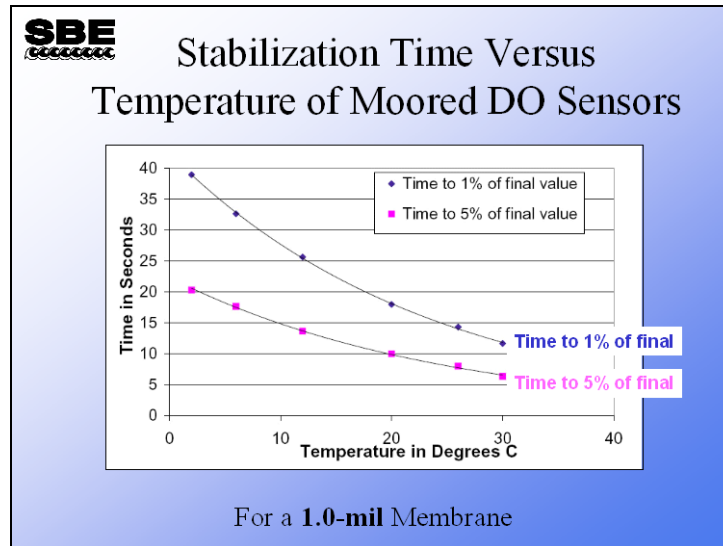
Oxygen sensors installed in moorings are always pumped and are typically plumbed between the pump and conductivity sensor of an SBE 16*plus* or 16*plus* V2. Between samples the pump does not run and water ceases to move past the oxygen electrode. As we have discussed, the electrode of the SBE 43 oxygen sensor is powered by an internal battery. When the water becomes still, the electrode depletes it of oxygen; if the CTD were to continue logging data you would observe oxygen concentration inside the sensor plenum approaching a steady state well below the ambient oxygen levels. When the CTD initiates a normal sampling interval by turning on the pump, you would observe a curve similar to the ones shown above. The water flow establishes a normal boundary layer above the oxygen electrode and the sensor equilibrates to the ambient oxygen level. The time required to reach 99% of the final equilibrium value depends on temperature, warmer water allowing faster equilibration. The arrows on the plot show the point at which the sensor has achieved 99% of the final value at each temperature.

## Preparing for Deployment: Setting Up for Dissolved Oxygen Sensors (*continued*)



This plot is derived from the preceding plot and may be used to determine the time required from power up and pump turn on to the availability of an acceptable dissolved oxygen sample. For example, if you were working in 20° C water and wanted your oxygen data to be better than 1% of actual ambient oxygen concentrations, you would want the sample interval to be longer than 11 seconds. Set the SBE 16*plus* or 16*plus* V2 pump mode to pump during the entire sample time (**MooredPumpMode=2** for 16*plus*; **PumpMode=2** for 16*plus* V2), and set the delay before sampling to 15 seconds (**DelayBeforeSampling=15**). We have allowed 4 extra seconds in our sampling time; this ensures that if the instrument finds itself in colder than expected water, that the sample will still be good. Note that longer pump times reduce battery endurance.


## Preparing for Deployment: Setting Up for Dissolved Oxygen Sensors (*continued*)



**Prior to 2007, all SBE 43s were sold with a 0.5-mil thick membrane.** Beginning in 2007, Sea-Bird began offering two membrane thicknesses – 0.5 mil (faster response, typically for profiling applications and 1.0 mil (slower response but more durable, typically for moored applications).

This plot was derived in a similar fashion to the one on the previous slide, and may be used to determine the time required from power-up and pump turn-on to the availability of an acceptable dissolved oxygen sample with a **1.0-mil membrane**.

## Preparing for Deployment: Starting Logging




### Log Data Now or Later?

- Use the **StartNow** command to begin logging data immediately
- To begin logging later:
  1. Use the **Startmmddyy=** command to enter a date to start logging, and then use the **Starthhmmss=** command to enter a start time on that date  
- Or -  
Use the **StartDateTime=** command to enter a date and time to start logging
  2. Use the **StartLater** command to begin logging on the date and time entered above

You have the option of setting a logging start time and date, allowing you to begin data collection once the mooring is in place. The risk you run is that you might not place the mooring when you planned to. There are always some risks in life. The alternative is to trim off data that is meaningless at the beginning of the file.

**Refer to the instrument manual for the exact commands for your instrument and your firmware version.** For example, the *16plus V2* uses **StartDateTime=** to set the start date and time; the *16plus* uses **Startmmddyy=** and **Starthhmmss=** to set the start date and time.

## Preparing for Deployment: Inductive Modem Systems




### Inductive Modem IDs

- Valid range for ID numbers is 00 – 99
- There must be only one instrument on the loop when setting IDs
- IM receiver is very sensitive; instruments not on the loop, but near the loop, will receive the ID change

Do not change the ID of an inductive modem equipped instrument when another instrument is on the same loop; this will cause both instruments to take the same ID. The telemetry system will not work properly for those instruments, because they will try to use the link at the same time. The inductive modem receiver in the instruments is very sensitive. Two instruments that are side-by-side will take the same ID even if one of them is not connected to the loop. You must separate IM instruments by at least 1 meter when you change IDs.

## Preparing for Deployment: Inductive Modem Systems (continued)




**Getting Data to the Surface and the Shore**

- You are responsible for the communications link between the surface buoy and the shore
  - Satellite
  - Cell phone
  - Radio modem
- You are responsible for a buoy computer that requests data from the moored instruments

The complete inductive mooring system requires a computer at the surface, to send commands to the instruments via the surface inductive modem and to communicate with the shore. Sea-Bird has customers that use ARGOS satellites, some that use cell phones, and some that use radio modems. It is your responsibility to develop the hardware and software to perform these duties.

## Preparing for Deployment: Inductive Modem Systems (continued)




### Requesting Data from Moored Instruments

- **PwrOn** causes the buoy modem to send a wake-up tone down the inductive link
- **GData** tells the moored instruments to prepare a data scan for transmission
- **Dataii** requests the instrument with the ID **ii** to transmit the scan
- **PwrOff** causes all instruments on the mooring to go to sleep

Communication with the moored instruments is simple. Sending **PwrOn** to the surface modem's serial port causes it to wake all the remote instruments with a 4800 Hz tone. Once awakened, the **GData** command causes the modems on the moored instruments to request the last sample or an average of the samples taken from their instruments (depending on the instrument and its setup). Then each instrument in turn is queried for a sample with the **Dataii** command, with **ii** being the instrument ID (for example, **data01** queries the instrument with ID=01). Finally, **PwrOff** puts all the instruments to sleep. If no **PwrOff** is issued, the instruments go to sleep after 2 minutes. Thus, if the sampling protocol requires more than 2 minutes, a **PwrOn** must be issued to make sure that none of the moored instruments has gone to sleep.

## Preparing for Deployment: Inductive Modem Systems (continued)



### Inductive Telemetry Troubleshooting

- Inductive communication is one way at a time (half duplex)
- Instruments with the same ID will compete for the link
- If request for data occurs during data acquisition, the request will be ignored

The inductive link is half duplex. This means that only one device on the link may be actively transmitting data at a time. If the situation occurs where two devices are trying to transmit, the communication will be corrupt.

If two devices have the same ID, the above problem will occur whenever that ID is used.

Usually, moored instruments sample on schedule. If the inductive link becomes active while an instrument is sampling, a request for data may be ignored. This can occur because of clock drift in the buoy. If the buoy is equipped with a GPS, the GPS is a good source of time.

A final note on preparing moored instruments for deployment: **Application Note 83** provides a checklist for moored instruments, intended as a guideline to assist you in developing a checklist specific to your operation and instrument setup.