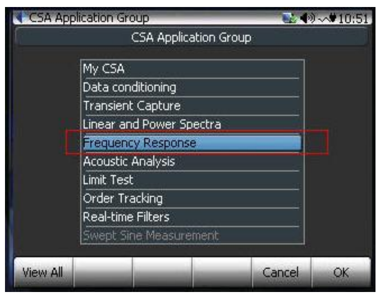
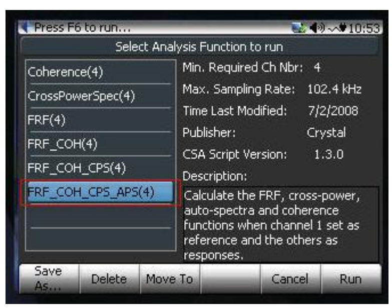
CRYSTAL instruments

4633 Old Ironsides Drive, Suite 304, Santa Clara, CA 95054, USA

Step by Step Tutorial for CoCo-80

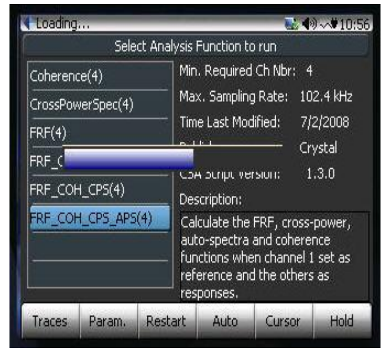
This tutorial demonstrates a step by step approach to set up the signal display, analysis parameters, input and output channels, recording, signal save and other basic operations. It takes approximately 1 hour to complete the entire tutorial. The tutorial is divided into the following sections:

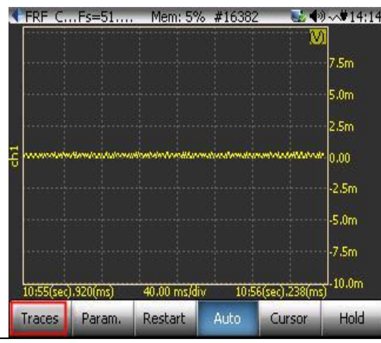
1. Start up and Display Operations
2. Parameters Settings
3. Record the Time Streams and Save the Signals
4. Set up the Trigger for Transient Capture
5. Cursor Operation

<table>
<tr><td>User Operation</td><td>CoCo Screen and comments</td></tr>
<tr><td colspan="2"><h2>Part 1: Start up and Display Operation</h2>Part 1 demonstrates how to start up the CoCo, open a project file, add signals to a window, scale and pan within the window and change the display format.</td></tr>
<tr><td>Press the Power Button and wait for the device to initialize.</td><td></td></tr>
<tr><td>Press the Analysis Button.</td><td></td></tr>
<tr><td>Use the Up/Down Buttons to select Frequency Response Folder and press the Enter Button.</td><td>
</td></tr>
<tr><td>Use the Up/Down Buttons to select FRF_COH_CPS_APS.csa then press the Enter Button.</td><td>

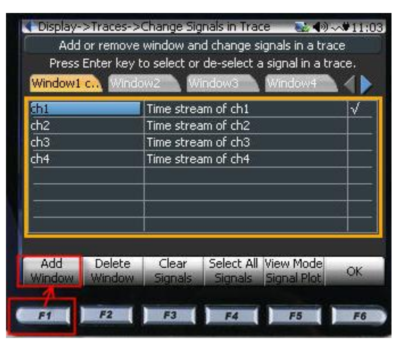
This project computes the auto-power spectra of each channels and the FRF between ch2, ch3 and ch4 to ch1. Ch1 is the excitation channel.</td></tr>
</table>

<table>
<tr>
<td>The CSA will open and the display will show the live signals. Press the F6 Hold Button to stop the display.</td>
<td>

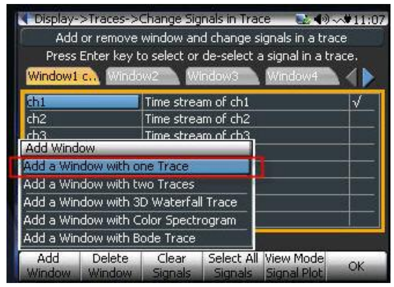
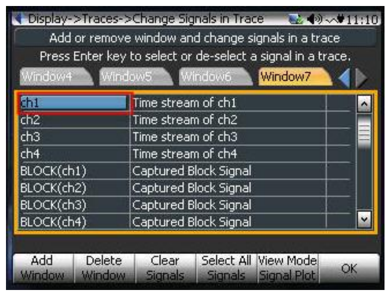
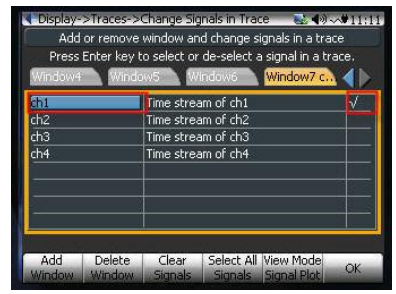
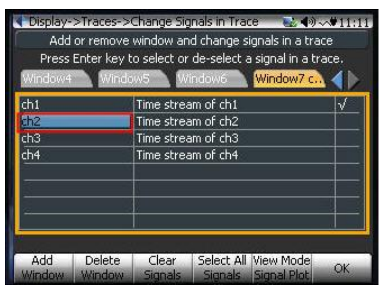
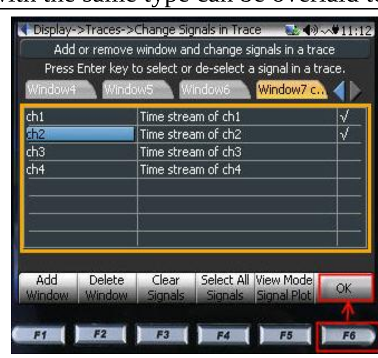



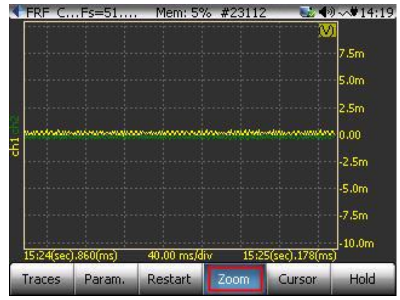
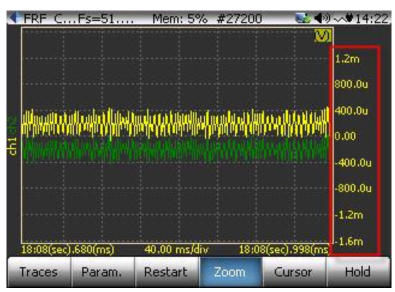
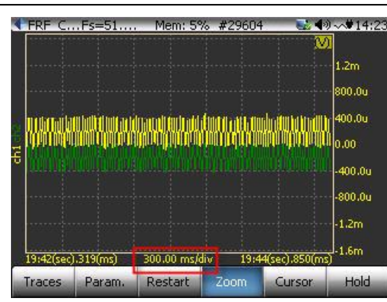
</td>
</tr>
<tr>
<td>Press the F1 Traces Button.</td>
<td>Depending on the project setup during previous use, the screen may show different signals. It can show auto-power spectra, FRF, or time domain signals. A time domain signals is shown in the picture below.


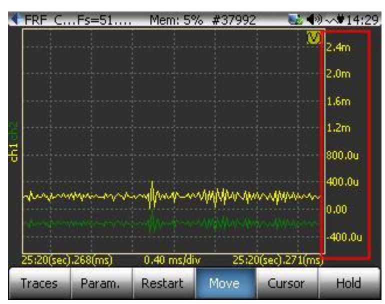
</td>
</tr>
<tr>
<td>Use the Up/Down Buttons to select Trace and Window Settings and press the Enter Button.</td>
<td>

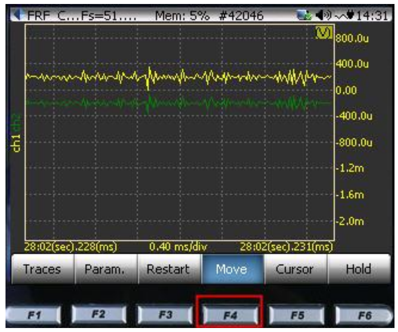

</td>
</tr>
<tr>
<td>Press the F1 Add Window.</td>
<td>


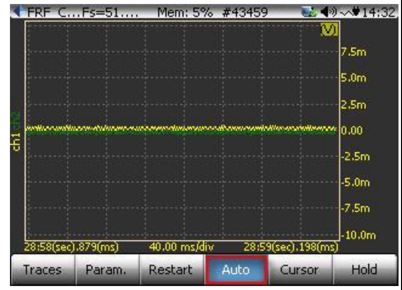
</td>
</tr>
</table>

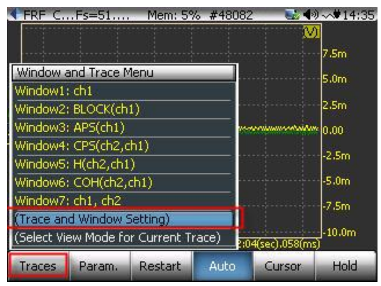
	This step creates a new window and assigns the signals into the window.
Use the **Up/Down Buttons** to select "**Add a Window with one Trace**" And press the **Enter Button.**	 A window may contain one or two traces.
To display continuous time streams, highlight the signal ch1 and press the **Enter Button** to select it.	 
Use the **Up and Down Arrow Buttons** to select ch2 and press the **Enter Button.**	  Only signals with the same type can be overlaid together.
Press the **F6 OK Button.**	 Now a new Window is created.
Press the **F4 Button** to switch from **Auto** to **ZOOM.**	

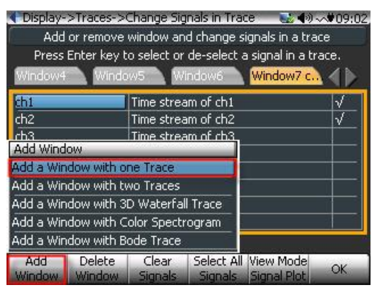
	  When ZOOM is active then the navigation Buttons can be used to manually adjust the display scale. When Auto is active then the display scale is automatically controlled to fit all the data in the window.
Press the **Up Button** a few times to expand the Y axis.	
Press the **Down Button** a few times to reduce the Y axis.	
Press the **Right Arrow Button** a few times to expand the X axis.	

Press the **Left Arrow Button** a few times to reduce the X axis.	
Press the **Shift Button** once to switch from **ZOOM** to **Move** mode.	  When Move is active then the UP and Down Arrow Buttons move the signal plot up or down, left or right to view a specific part of the signal.
Press the **Up Arrow Button** a few times to move the Y axis.	
Press **Down Arrow Button** a few times to move the Y axis.	

<table>
<tr>
<td>Press the F4 Button to change from the Move/Zoom mode to Auto mode.</td>
<td>


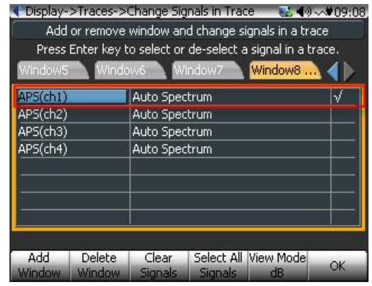
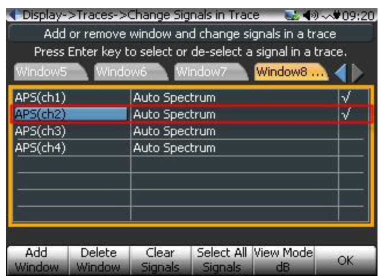
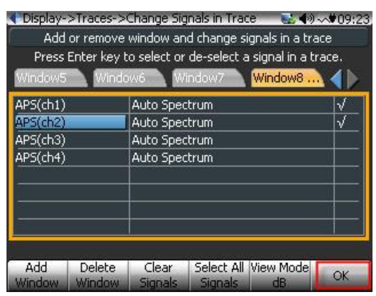


This concludes the tutorial on scaling and moving around the window.
</td>
</tr>
<tr>
<td>Next, display the frequency signals by pressing the F1 Traces Button and select Trace and Window Settings.</td>
<td>


</td>
</tr>
<tr>
<td>Press the F1 Add Window Button and select Add a Window with one Trace.</td>
<td>


</td>
</tr>
<tr>
<td>Next, display the auto power spectra by scrolling down the list, highlighting the signal APS(ch1) and pressing the Enter Button to select it.</td>
<td></td>
</tr>
</table>

<table>
<tr><td></td><td>


</td></tr>
<tr><td>Scroll down one row to the next signal, highlight APS(ch2) and press the Enter Button.</td><td>
</td></tr>
<tr><td>Press the F6 OK Button.</td><td>

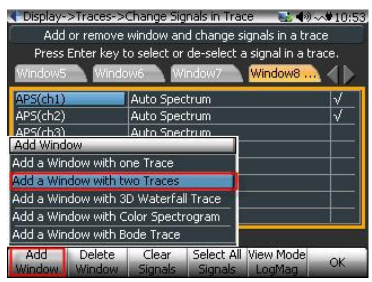
After OK is pressed, you should see the auto-power spectra display.</td></tr>
<tr><td>Next, change the view mode by pressing the F1 Traces Button.</td><td></td></tr>
</table>

	
Use the **Down Arrow Button** and highlight **Select View Mode**.	
Press the **Enter Button.**	
Select **LogMag.**	
Press the **Enter Button.**	

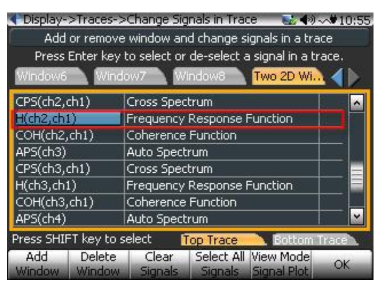
<table>
<tr>
<td></td>
<td>

</td>
</tr>
<tr>
<td>Next create a Window with two traces by pressing the F1 Traces Button.</td>
<td>

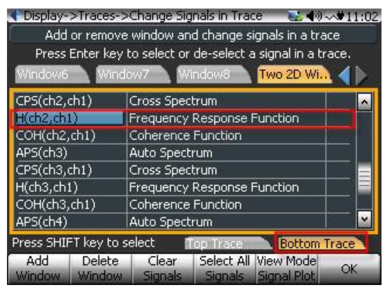
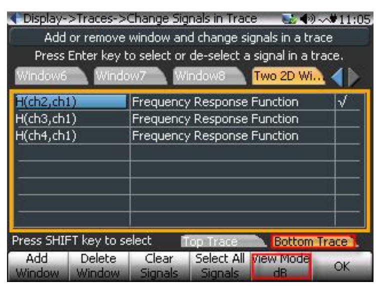
</td>
</tr>
<tr>
<td>Press the F1 Add Window with Two Traces Button.</td>
<td>

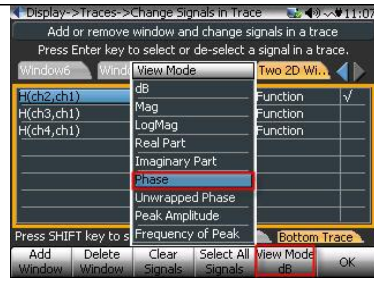
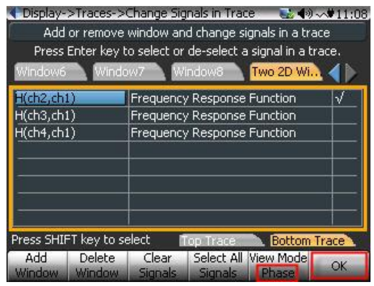


</td>
</tr>
<tr>
<td>Scroll down the list, highlight the signal H(ch2, ch1) and press the Enter Button to select it.</td>
<td>



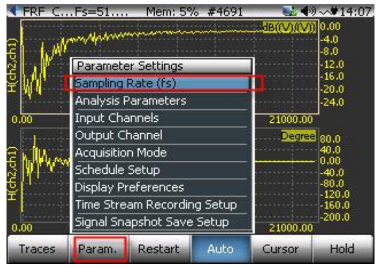
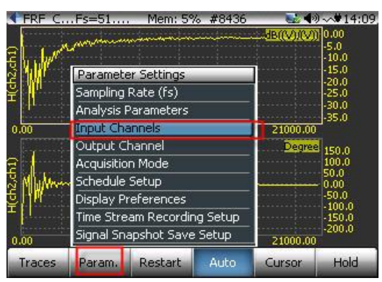
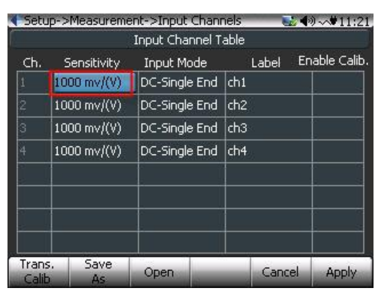
</td>
</tr>
<tr>
<td>Press the Shift Button to select the bottom trace.</td>
<td></td>
</tr>
</table>

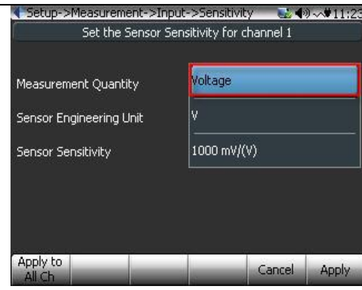
<table>
<tr><td></td><td></td></tr>
<tr><td>Scroll down the list, highlight the signal FRF(ch2, ch1) and press the Enter Button to select it.</td><td></td></tr>
<tr><td>Press the F5 View Mode Button.</td><td></td></tr>
<tr><td>Select Phase and press the Enter Button.</td><td></td></tr>
</table>

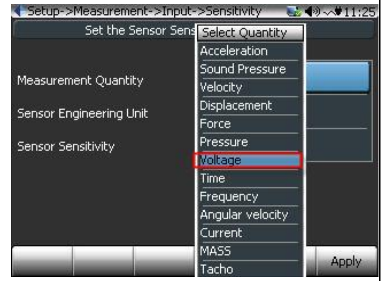
<table>
<tr><td></td><td>
</td></tr>
<tr><td>Press the F6 OK Button.</td><td>
</td></tr>
<tr><td>Then a window with two traces will be displayed.</td><td>
</td></tr>
<tr><td></td><td>At this stage, if you have created too many windows under the Traces menu. You can go back to delete some of them so the windows left are those that you frequently want to see.</td></tr>
</table>

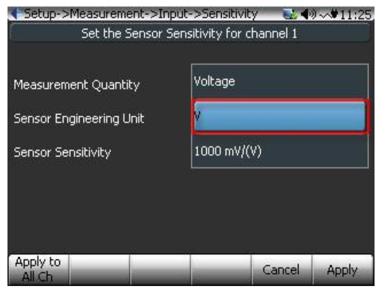
Part 2: Set the Parameters

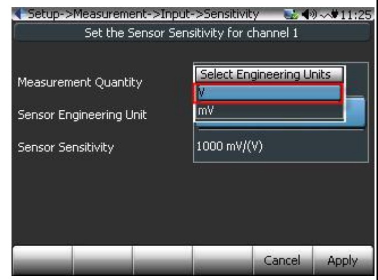
Part 2 demonstrates how to change the sensitivity, units and output channel settings.

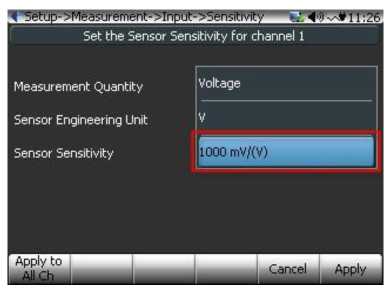
<table>
<tr><td>Press the F2 Param. Button, Select “Sampling Rate” to set appropriate sampling rate.</td><td>
</td></tr>
<tr><td>Press the Enter Button to accept the sampling rate.</td><td>
</td></tr>
<tr><td>Press the F2 Param. Button, use the UP and Down Arrow Buttons to highlight Input Channels and press the Enter Button.</td><td>
</td></tr>
<tr><td>Use the Up and Down Arrow and Enter Buttons to select and set the quantity, sensitivity and engineering unit (details not shown).</td><td>
</td></tr>
</table>

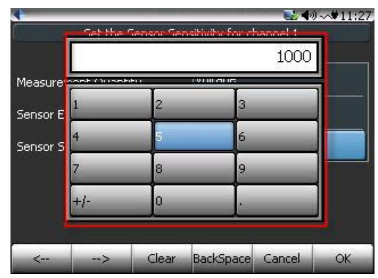
<table>
<tr>
<td></td>
<td>


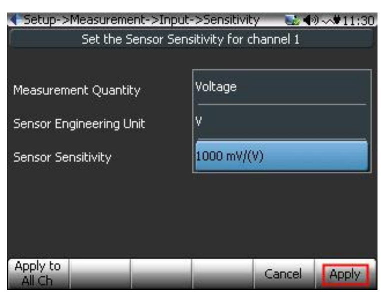


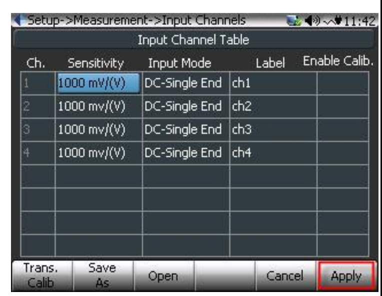




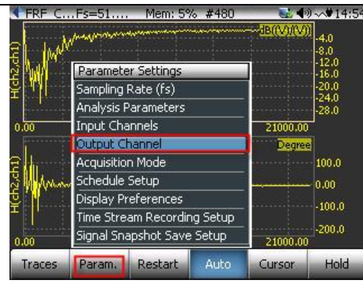
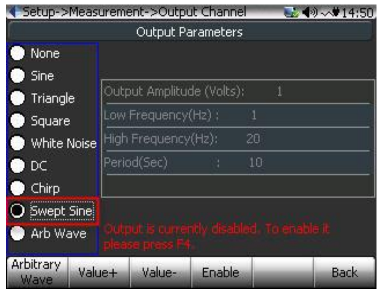
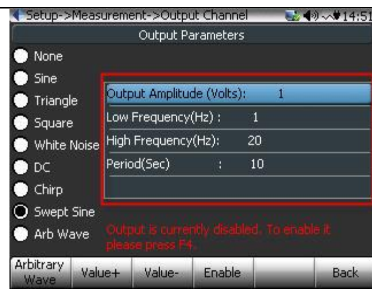
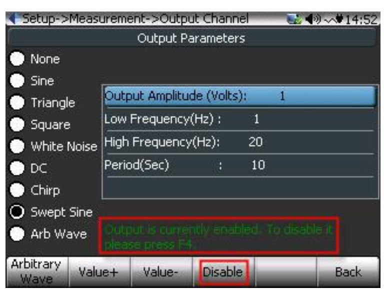


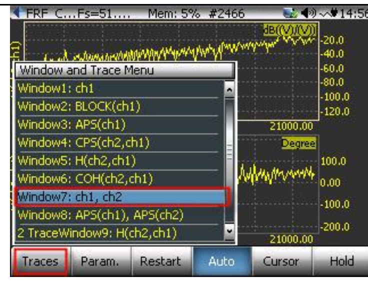
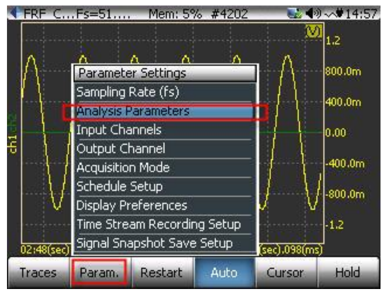


</td>
</tr>
<tr>
<td>Press the F6 Apply Button to accept the changes.</td>
<td>




The F1 Button, Apply to All Ch applies the same setting to all input channels in one step.
</td>
</tr>
<tr>
<td>Press the F2 Param. Button, Use the Up and Down Arrow Buttons to highlight “Output Channel” and press the Enter</td>
<td></td>
</tr>
</table>

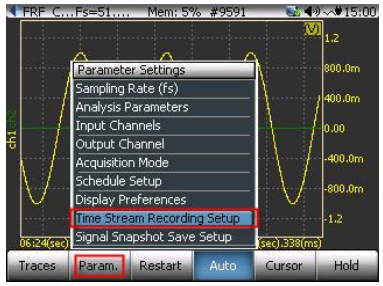
Button.	
Use the **Up and Down Arrow Buttons** to move down to **Swept Sine.**	
Use the **Right Arrow Button** to set the focus to the right pane. You can change the output parameters here (details not shown).	
Press the **F4 Button** to enable the output waveform generator then press the **Display Button**.	
Connect the signal source to Channel 1 using the provided output cable.	After this operation with the signal source is fed into the channel 1, you should see live sine wave in the signal plot.
Press the **F1 Traces Button**, select **Window: ch1,ch2** and press the **Enter Button.**	

<table>
<tr>
<td></td>
<td>
</td>
</tr>
<tr>
<td>Press the F2 Param. Button. Use the Up and Down Arrow Buttons to highlight the Analysis Parameters and press the Enter Button. You can change each of the Analysis Parameters here (details not shown).</td>
<td>
</td>
</tr>
</table>

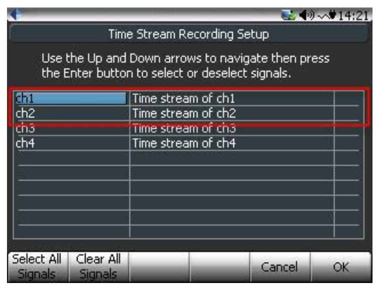
Part 3: Record the Time Streams and Save the Signals

Part 3 demonstrates how to record time streams, save data frames and review saved data on the CoCo.

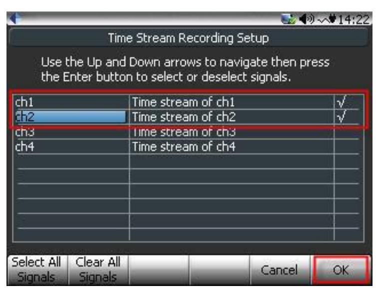
Press the **F2 Params**. **Button**, use **Up and Down Arrow Buttons** to highlight the **Time Streams Setup and** press the **Enter Button**.



Highlight **ch1** and press the **Enter Button** to check it. Highlight **ch2** and press the **Enter Button** to check it.

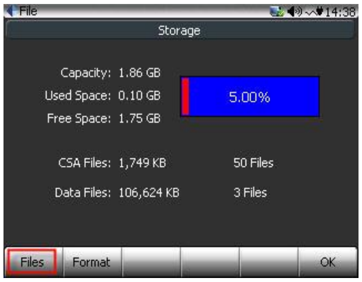
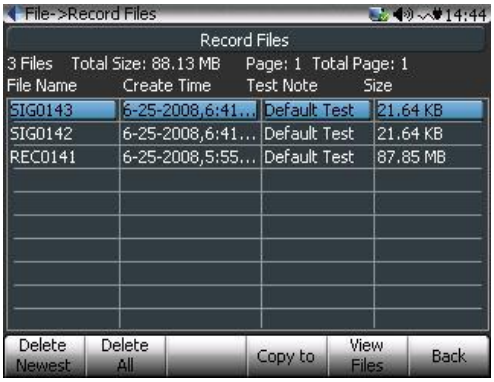


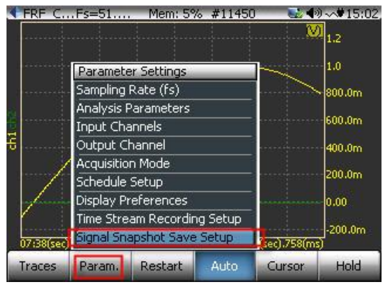
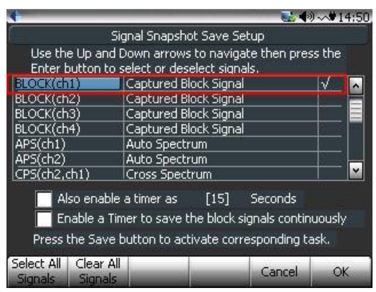
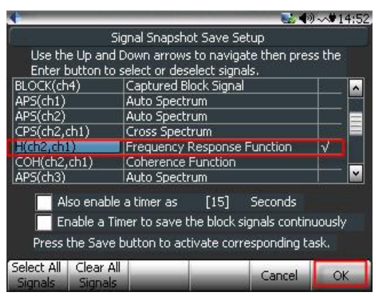
Press the **F6 OK Button.**



This setup is designed for the convenience of time stream recording.

Press the **Rec./Stop Button** once, to start time stream recording.

Press the **Rec./Stop Button** a second time to stop time stream recording.	See the figure above.
Press the **File Button**.	
Press the **F1 Files Button** to see a list of recorded time streams signals or saved frequency signals.	
Use the **Up and Down Arrow Buttons** to highlight a file and press the **F5 View Files Button** to view the highlighted file. Press the **F1 Button** to delete the all the files newer than the file that is highlighted files; Press the **F2 Button** to delete all the files. Press the **F4 Button** to copy the highlighted files to the SD card.	

<table>
<tr><td>Press the Display Button to go back to the signal display.</td><td></td></tr>
<tr><td>Press the F2 Params. Button, use the UP and Down Arrow Buttons to highlight the Signal Snapshot Save Setup and press the Enter Button.</td><td></td></tr>
<tr><td>Highlight Block(ch1) and press the Enter Button. Highlight H(ch2,ch1) and press the Enter Button (or select more signals to save).</td><td></td></tr>
<tr><td>Press the F6 OK Button.</td><td></td></tr>
<tr><td>Press the Save Button twice to save the signals.</td><td>The first time you press the Save Button, a list of commands is shown. Simply press it a second time to save the defined signals.
Now that the signals have been saved you can press the File Button to review or recall the signals.</td></tr>
</table>

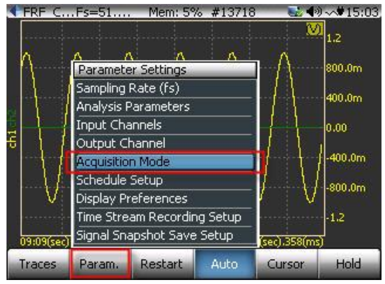
	

Analysis
Display
Setup
File
Enter
Shift
Rec./Stop
Save

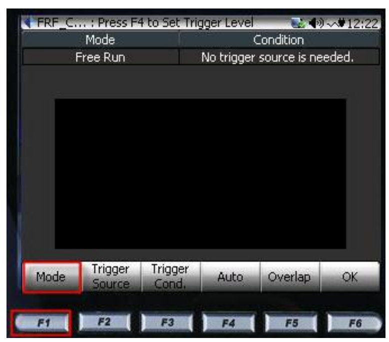
Part 4: Set up the Trigger for Transient Capture

Part 4 demonstrates how to set up a trigger to capture a transient event including setting the trigger mode, level, pre-trigger and accept/reject mode.   Make sure the signal source is still on and ch1 is connected to the source with a cable. If the signal source is not on, please go back to Output Channel setup and enable it

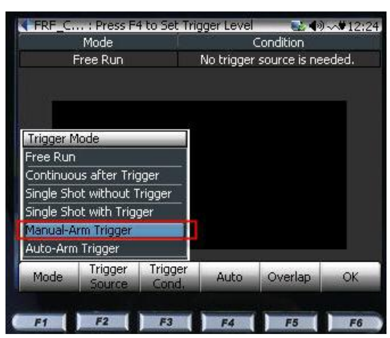
Press the **F2 Params. Button**, use the **Up and Down Arrow Buttons** to highlight **the Acquisition Mode** and press the **Enter Button**.

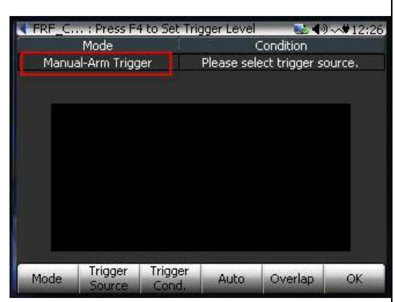


Press the **F1 Mode Button.**



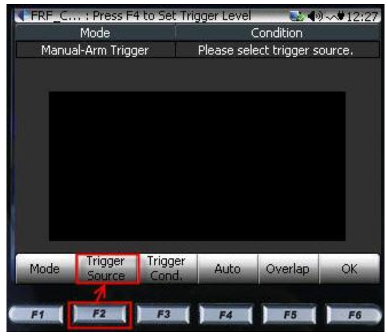
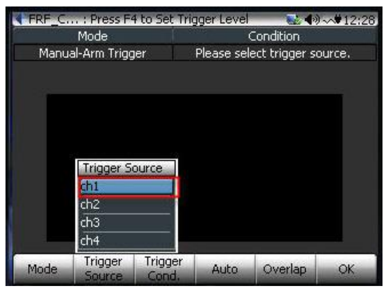
Use the **Up and Down Arrow Buttons** to highlight the **Manual-Arm Trigger** and, press the **Enter Button.**

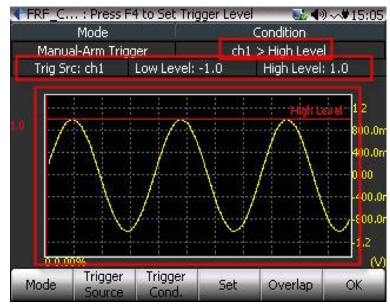
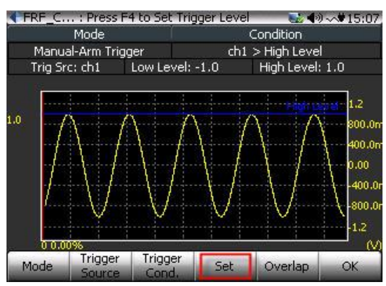


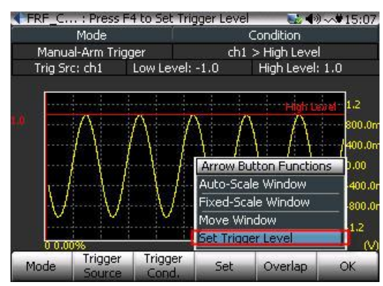
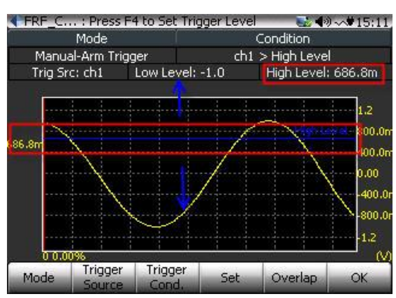


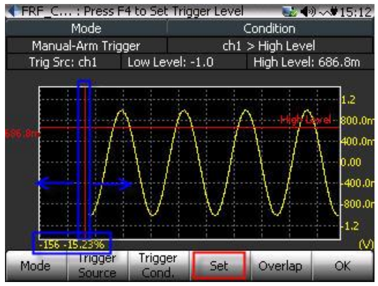
Manual Re-Arm means that in the transient capture, operator you will have a chance to accept or reject the captured time blocks.

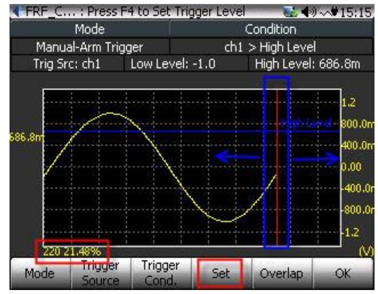
Press the **F2 Trigger Source Button.**

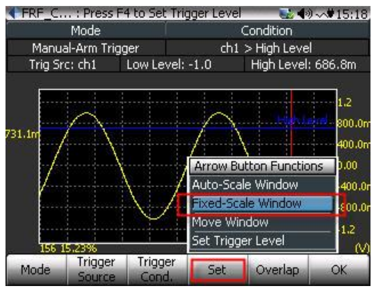
<table>
<tr>
<td></td>
<td>
</td>
</tr>
<tr>
<td>Highlight ch1 and press the Enter Button.</td>
<td>The screen will show a live signal of containing the trigger source.


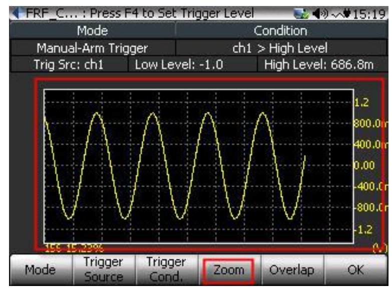

</td>
</tr>
<tr>
<td>Press the F4 Set Trigger Level Button.</td>
<td>This selection puts the navigation buttonButton into the functions of setting trigger level.


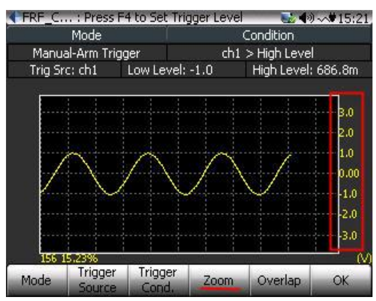

</td>
</tr>
<tr>
<td>Use the Up and Down Arrow Buttons and select Set Trigger Level. Then use the Up and Down Arrow Buttons to change the trigger threshold level. Choose a level that is below the maximum value of the sine wave as shown in the fiture.</td>
<td>
</td>
</tr>
</table>

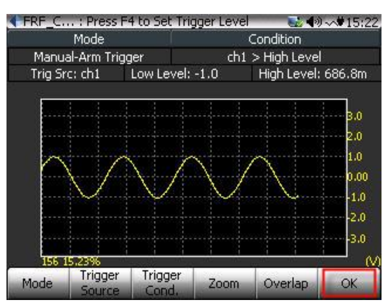
<table>
<tr>
<td>Press the Left and Right Arrow Buttons to set the pre-trigger percentage.</td>
<td>




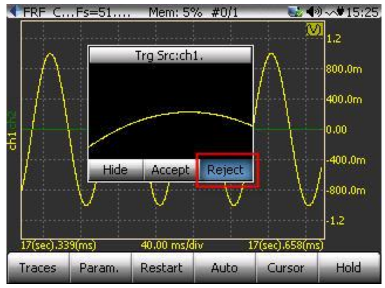
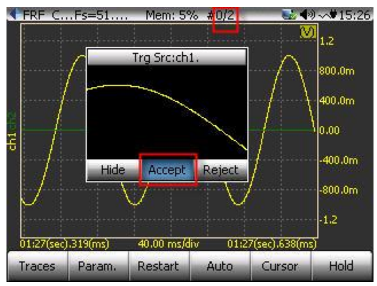
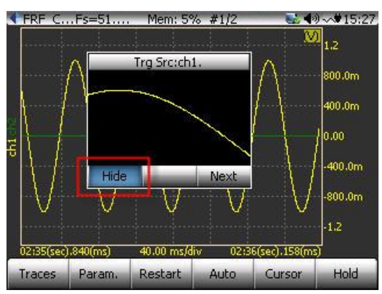
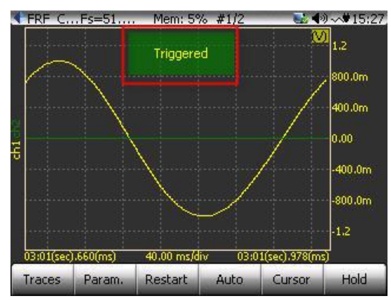
</td>
</tr>
<tr>
<td>Press the F4 Set Button and select Fixed Scale.</td>
<td>


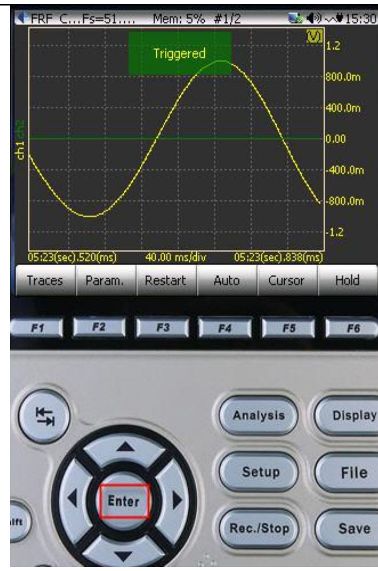


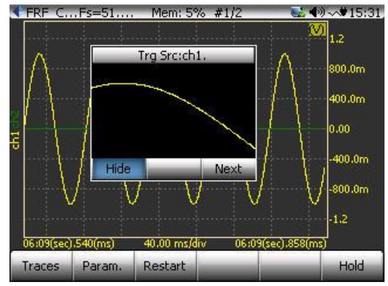
This operation put changes the navigation buttonButtons mode to functions of setting the scaling window instead of setting up the trigger threshold.
</td>
</tr>
<tr>
<td>Use the Up and /Down Buttons to scale the Y axis within the window.</td>
<td>


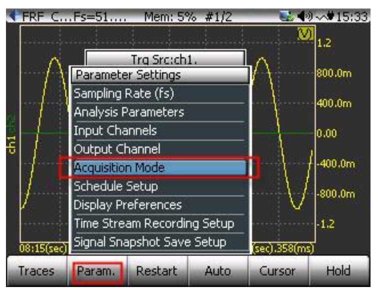
</td>
</tr>
<tr>
<td>Press the F6 Apply OK Button to accept the trigger setup.</td>
<td>


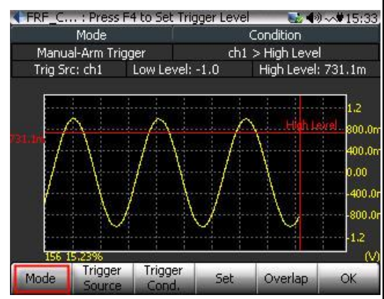
</td>
</tr>
<tr>
<td>Use the Left/Right Arrow</td>
<td></td>
</tr>
</table>

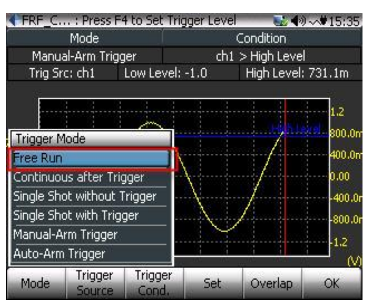
buttonButtons to switch between **Hide**, **Accept** and **Reject**. Select Move to **Reject** and press the **Enter Button.**	
Next select **Accept** and press the **Enter** Button.	This must be the second capture. Accept means that the signals will be accepted for FFT transform and averaged into the spectra and FRF. 
Select **Hide** and press the **Enter Button.**	 
Press the **Enter Button** again.	The trigger operation window will be shown again.

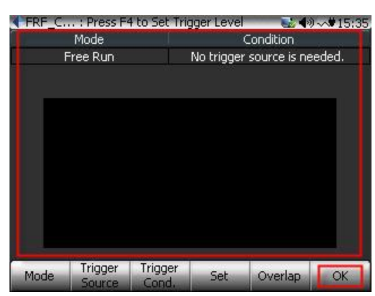
<table>
<tr>
<td></td>
<td>




</td>
</tr>
<tr>
<td>Press the F2 Params. Button, use the Up and Down Arrow Buttons to highlight the Acquisition Mode and, press the Enter Button. ;
Press the F1 Mode Button.</td>
<td>




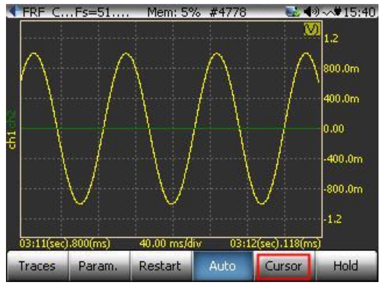
</td>
</tr>
<tr>
<td>Use the Up and Down Arrow Buttons to highlight Free Run, and press the Enter Button. Press the Display Button.</td>
<td>This operation will put change the CoCo back to Free Run mode. In Free-run mode, means the time capture occurs will be done as fast as possible with overlapping.


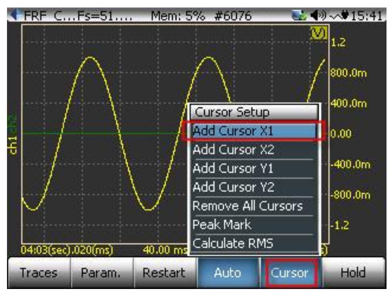
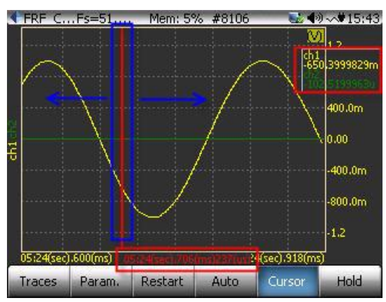


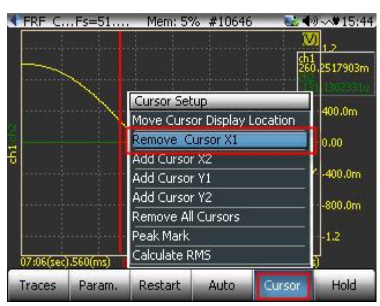
</td>
</tr>
</table>

Part 5: Use Cursors

Part 5 demonstrates how to add cursors to a window and move them to analyze a signal. Make sure the screen contains is in any signal display. If not, press the **Display Button**.

<table>
<tr><td>Press the F5 Cursors Button.</td><td>

This operation puts the navigation buttonButtons into the cursor control mode.</td></tr>
<tr><td>Highlight Add Cursor X1 and press the Enter Button.</td><td>
</td></tr>
<tr><td>Use the Left/Right Arrow Buttons to move the cursor.</td><td>

Note the value of the signals is displayed on the right.</td></tr>
<tr><td>Press the F5 Cursors Button and select Remove Cursor X1 to remove it.</td><td>
</td></tr>
</table>

CoCo-80 Basic DSA User's Manual

Version 1.3

Crystal Instruments Corporation

4699 Old Ironsides Drive, Suite 100

Santa Clara, CA 95054, USA

This page is left blank intentionally

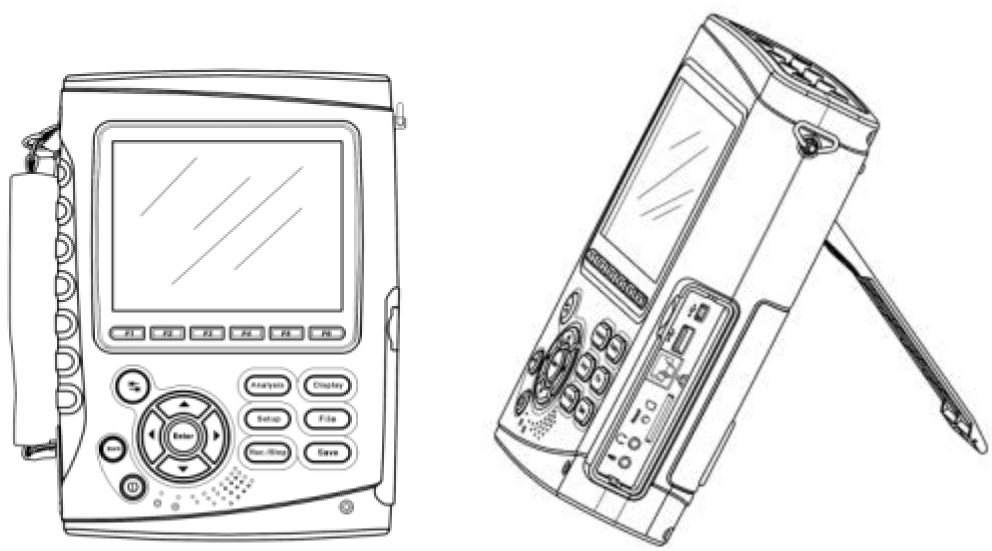
Table of Contents

Table of Figures

1. INTRODUCTION

CoCo-80 is a handheld data recorder, dynamic signal analyzer and vibration data collector that is ideal for a wide range of industries including machine conditioning monitoring, automotive, aviation, aerospace, electronics and military that demand easy, quick and accurate data recording and real-time processing in the field. CoCo-80 is a low-cost, light-weight, battery powered handheld system with unparalleled performance and accuracy. The user interface of CoCo-80 is specifically designed for easy and simple operation while it maintains the capability of providing a wide variety of analysis functions.

The CoCo-80 hardware platform supports two different software working modes: dynamic signal analyzer (**DSA**) and vibration data collector (**VDC**). Each working mode has its own user interface and operation navigation structure. DSA working mode is designed for mechanical structure analysis, testing and optimization or for electrical, geophysics and a wide range of applications. VDC is dedicated to route-based machine vibration data collection and trending. The user will select one of the working modes to execute. This manual describes the DSA working mode of CoCo-80.



- Figure 1. CoCo-80 Hardware.

CoCo-80 is the first battery-powered handheld data acquisition system that matches the performance and functionality of higher-end systems. CoCo-80 is equipped with 4 or 8 input channels and can accurately measure and record both dynamic and static signals. The mass flash memory can record 8 channels of streaming signals simultaneously up to 102.4 kHz. An embedded signal source channel provides various signal output waveforms that are synchronized with the input sampling rate.

CoCo-80 hardware uses dual CPU architecture. An XScale CPU handles the user interface, project configuration, power management, network communication as well as all the peripherals. A high-speed floating point DSP manages the data input/output and real-time processing. CoCo-80 is also configured with large RAM and NAND flash memory for mass data storage. Special thermo and low power design eliminates the need for a cooling fan and increases the battery operating time. Proprietary hardware technology delivers more than 130 dB dynamic range. The extremely high dynamic range eliminates the need for multiple front end gain settings.

The CoCo-80 can also be operated from a DC power source (which will also simultaneously charge the battery). This can be achieved with either the CoCo-80 AC-DC Adapter (P/N 40-115) or an Automotive Cigarette Lighter Adapter.

Revolutionary 24-bit A/D converters, digital technology and unique hardware designed for CoCo-80 offers more than 130 dB dynamic range, 10~100 times higher than competitive products. The high dynamic range and fidelity of the CoCo-80 enables measurement of a wide range of signals, regardless of the input signal magnitude.

CoCo-80 excels in both dynamic and static measurements. When used for dynamic measurements, the input channels offer extremely high-quality dynamic range, signal to noise ratio, cross channel gain match, phase match, and spectrum flatness over an analysis frequency range up to 45 kHz. When it is used to measure the static or quasi-static signals, it offers very high accuracy at DC or near DC frequency.

The CoCo-80 utilizes a new signal processing method, Configurable Signal Analysis (CSA). CSA provides unique flexibility for real time analysis including filtering and spectral analysis.

For DSA applications the CoCo-80 software stores and organizes the data in the popular ASAM-ODS standard. Data may be exchanged with other data formats such as UFF, BUFF, NI-TDM. ASCII, MATLAB or Excel. The ASAM-ODS data standard provides ultimate flexibility and version compatibility. ASAM-ODS data standard is widely supported by the automotive industry and is expanding to aerospace and other areas. For VDC applications the CoCo data is stored and managed by the database.

The handheld system is equipped with two USB ports, 100 BaseT Ethernet, SD-card interface, audio input/output, 5.7 inch color LCD display and a keypad. You can connect the CoCo-80 to a PC, download files and upgrade the software through several means of network connection. The user interface of CoCo-80 is specifically designed for easy and simple operation while it maintains the capability of providing a wide variety of analysis functions.

The CoCo-80 has a weight less than 1.7 kg. Advanced thermal design eliminates the need for a cooling fan reducing operating noise. The fully charged battery life is up to many hours. An AC adapter can be used any time to charge the device and supports unlimited hours of operation.

Compared to handheld data acquisition systems and signal analyzers from the other providers, CoCo-80 delivers a higher measurement dynamic range and accuracy, recording throughput rate and real-time analysis performance. It also provides more powerful communication peripherals.

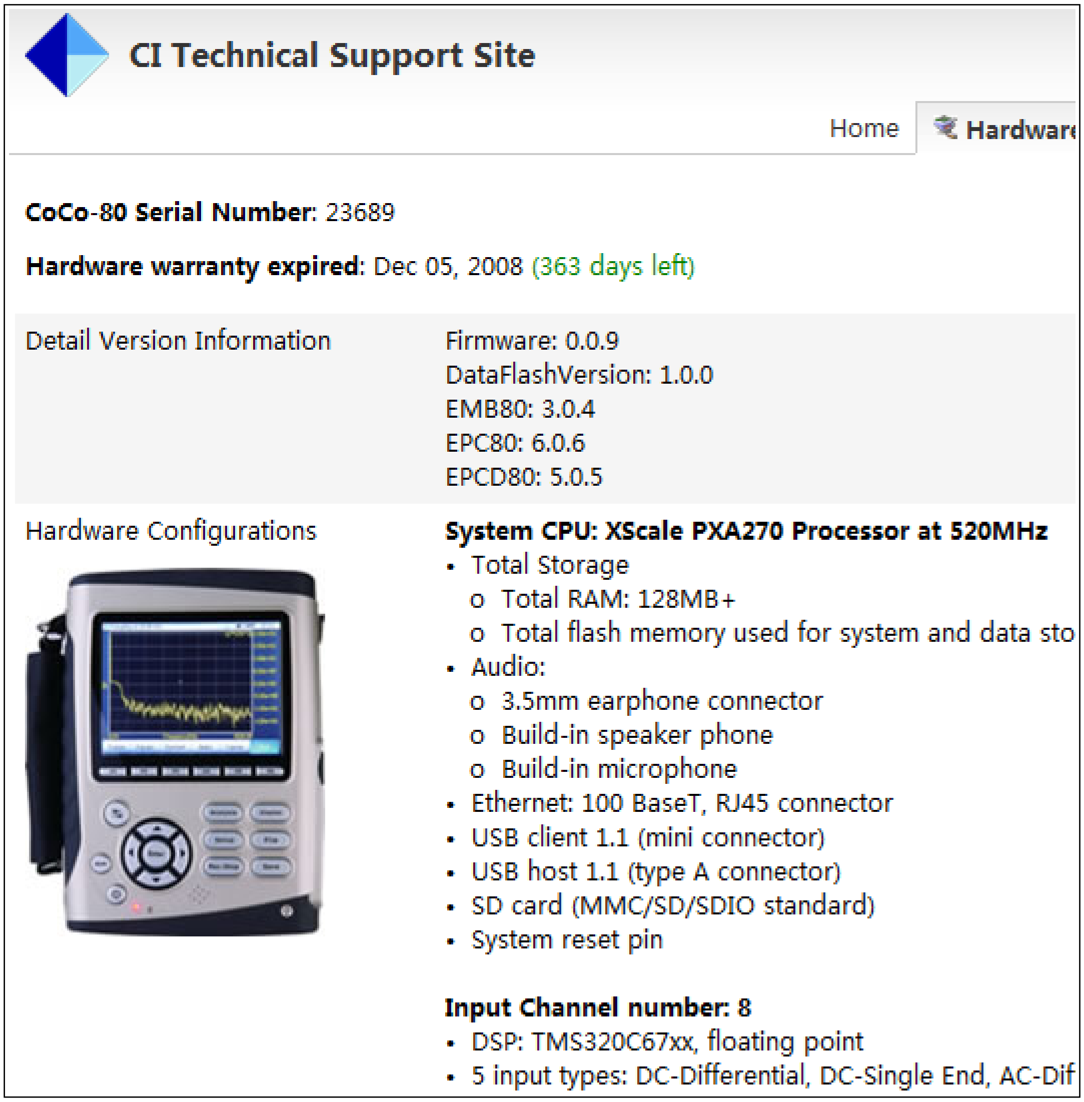
Compared to PC-tethered data acquisition systems and signal analyzers, CoCo-80 does not have the drawbacks such as unreliable data transfer using connection cables. CoCo-80 does not need any additional PC or laptop to operate during field data acquisition. Hence CoCo-80 is much more reliable and easy to operate.

On-Line Support

To access product information about your CoCo-80, please go to the product page of CI website at: http://www.go-ci.com/support.asp, log in with the serial number of the CoCo-80 and the password included in your shipping documents. After you log-in, you will be able to review and download the latest information which is restricted to CoCo80 users, including:

- Product Information
- New CSA projects
- User's Manual
- Shipping and Repair History
- User Forum
- Technical Support
- Software Updates
- Technical Issues

A typical page of CI Technical Support website is shown below:



■ Figure 2. Crystal Instruments CoCo-80 Support Website.

The latest CoCo-80 application software, device drivers or CSA projects can be downloaded while the CoCo-80 subscription is maintained.

Limited Warranty & Limitation of Liability

Each CI product is warranted to be free from defects in material and workmanship under normal use and service. The warranty period is one year for the CoCo-80 hardware and its accessories. The warranty period begins on the date of shipment. Parts, product repairs and services are warranted for 90 days. This warranty extends only to the original buyer or end-user customer of a CI authorized reseller, and does not apply to fuses, disposable batteries or to any product which, in CI's opinion, has been misused, altered, neglected or damaged by accident or abnormal conditions of operation or handling. CI warrants that software will operate substantially in accordance with its functional specifications for one year and that it has been properly recorded on non defective media. CI does not warrant that software will be error free or operate without interruption.

CI authorized resellers shall extend this warranty on new and unused products to end user customers only but have no authority to extend a greater or different warranty on behalf of CI. Warranty support is available if the product is purchased through a CI authorized sales outlet or the Buyer has paid the applicable international price. CI reserves the right to invoice the Buyer for importation costs of repair/replacement parts when product purchased in one country is submitted for repair in another country.

CI's warranty obligation is limited, at CI's option, to refund of the purchase price, free of charge repair, or replacement of a defective product which is returned to a CI authorized service center within the warranty period.

To obtain warranty service, contact your nearest CI authorized service center or send the product, with a description of the difficulty, postage and insurance prepaid (FOB Destination), to the nearest CI authorized service center. CI assumes no risk for damage in transit. Following warranty repair, the product will be returned to Buyer, transportation prepaid (FOB Destination). If CI determines that the failure was caused by misuse, alteration, accident or abnormal condition of operation or handling, CI will provide an estimate of repair costs and obtain authorization before commencing the work. Following repair, the product will be returned to the Buyer transportation prepaid and the Buyer will be billed for the repair and return transportation charges.

THIS WARRANTY IS THE BUYER'S SOLE AND EXCLUSIVE REMEDY AND IS IN LIEU OF ALL OTHER WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. CI SHALL NOT BE LIABLE FOR ANY SPECIAL, INDIRECT, INCIDENTAL OR CONSEQUENTIAL DAMAGES OR LOSSES, INCLUDING LOSS OF DATA, WHETHER ARISING FROM BREACH OF WARRANTY OR BASED ON CONTRACT, TORT, RELIANCE OR ANY OTHERTHEORY.

Since some countries or states do not allow limitation of the term of an implied warranty, or exclusion or limitation of incidental or consequential damages, the limitations and exclusions of this warranty may not apply to every buyer. If any provision of this Warranty is held invalid or unenforceable by a court of competent jurisdiction, such holding will not affect the validity or enforceability of any other provision.

Crystal Instruments Corporation, www.go-ci.com

Safety Information: Read First

The CI CoCo-80 Handheld Data Acquisition System complies with:

EN 61326:1997+A1:1998+A2:2001
EN61000-3-2: 2000 & EN61000-3-3: 1995+A1:2001

Use the CoCo-80 and its accessories only as specified in the User's Manual. Otherwise, the protection provided by the Instrument and its accessories might be impaired.

Condensation may form on the circuit boards when the device is moved from a cold environment to a warm one. In these situations, always wait until the device warms up to room temperature and is completely dry before turning it on. The acclimatization period should take about 2 hours.

For the most accurate measurements a warm-up phase of 20 min is recommended.

The devices have been designed for use in clean and dry environments. It is not to be operated in 1) exceedingly dusty and/ or wet environments, 2) in environments where danger of explosion exists nor 3) in environments containing aggressive chemical agents.

Lay cables in a manner to avoid hazards (tripping) and damage.

A Warning identifies conditions and actions that pose hazard(s) to the user. A Caution identifies conditions and actions that may damage the Instrument.

To avoid electrical shock or fire:

1. Realize that the CoCo-80 is a low voltage measurement instrument.
2. Do not apply input voltages above the rating of the Instrument. You should never apply a voltage that potentially exceeds +/-40V to the Instrument.
3. Review the entire manual before use of the Instrument and its accessories.
4. Do not operate the Instrument around explosive gas or vapor.
5. Before use, inspect the instrument, BNC connectors and accessories for mechanical damage and replace when damaged. Look for cracks or missing plastic. Pay special attention to the insulation surrounding the connectors.
6. Remove the cables and accessories that are not in use.
7. Use the ground input only to ground the Instrument and do not apply any voltage.
8. Do not insert metal objects into connectors.
9. Use only the wall-mount power supply provided by the Crystal Instruments.

AC Adapter Voltage Range

For external power source CoCo-80 uses a wall-mount AC Adapter. The AC Power range is: 100Vac ~ 240Vac.

Maximum Measurement Input Voltage

Maximum Working Input Voltage: 10 V peak. Voltage ratings are given as "working voltage". They should be read as Vpeak for dynamic applications and as V dc for DC applications.
Max. Input Range without damaging the hardware: 40Vpeak.

If Safety Features are Impaired

If the instrument is used in a manner not specified by the manufacturer, the protection provided by the instrument may be impaired. Before use, inspect the test leads for mechanical damage and replace damaged test leads! If the instrument or its accessories appear to be impaired or not functioning properly, do not use it and send it in for repair.

2. QUICK START

This Quick Start section is intended to give a brief introduction to the most basic use of the CoCo-80 system. By following the instructions you will learn how to do the following:

1. Record Time Streams with CoCo-80
2. Install the EDM software to your host PC
3. Download and view the data on host the PC

After completing the Quick Start tutorial you should read the following sections for a more comprehensive description of the system.

Recording Time Streams with CoCo-80

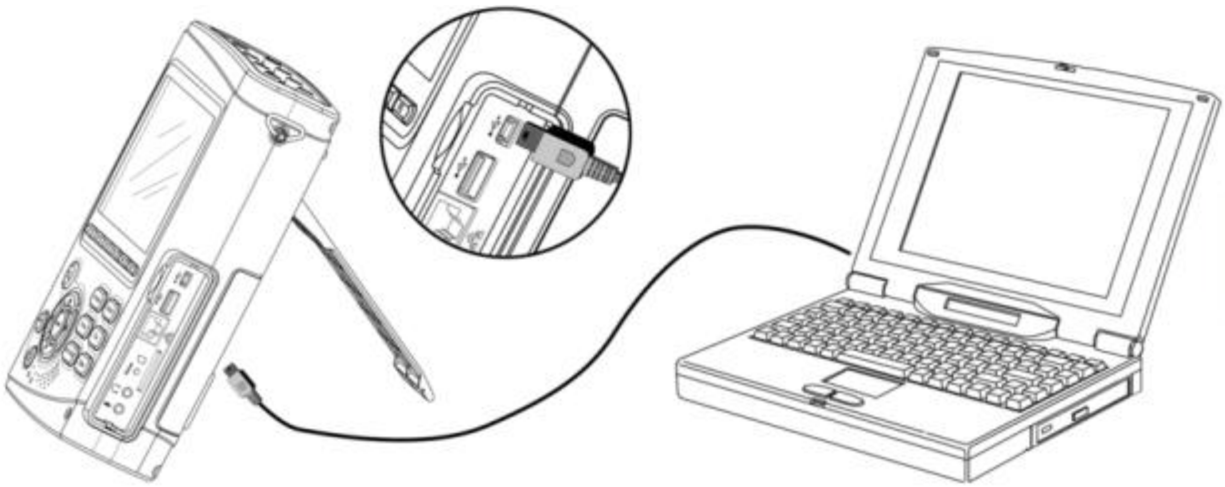
1. Remove the CoCo-80 unit from the shipping packaging.
2. Connect the wall mount power adaptor to the 110V/220V power outlet.
3. Connect the wall-mount power adaptor to the CoCo-80 (if battery is not charged).
4. Push the power button on CoCo-80, wait for about 30 seconds until the Welcome screen is displayed.
5. Use the Up or Down arrow buttons to select one of the CSA project files to run.
6. In the signal display screen, press ***F2 (Param.)*** and select ***Time Stream Recording Setup*** to define data streams for recording. To enable the recording for any signal, first use the up/down arrow navigation buttons to move to the signal, then press the Enter button to select or de-select.
7. Push the ***Rec./Stop*** button to record the signals. After a few seconds push the ***Rec./Stop*** button again to stop the recording.
8. Push the ***File*** button, then the Files soft button to review the recorded signals.

Install Engineering Data management (EDM) software to PC

1. Insert the CoCo-80 Application Software CD into your host computer.
2. Click Install EDM entry to install the EDM software and follow the instructions.
3. Set a local working folder to your computer.

Download Data to the PC

1. Connect the CoCo-80 to your PC using the USB cable provided. Note there are two USB ports on the CoCo-80 device. Connect the USB cable to the USB-client connector shown below.



■ Figure 3. Client USB connector for PC communication.

2. Browse for the device driver on the CD and the Windows operating system will automatically install the CoCo-80 USB driver on the PC.

3. Run the EDM software from the host PC.

4. Click on the Search Button on EDM to search the CoCo-80 devices that are connected to the PC.

5. After the EDM finds the CoCo-80, click Connect.

6. Drag the data file from the CoCo-80 (xxx.atfx), to one of your local folders.

7. Using the mouse, right click on the signal file xxx.atfx on your local folder, and click ***View*** from the pop-up menu.

8. Now the EDM software changes to the Analysis page. Use the mouse to drag the signal ch1 into the center empty area. You will see the waveform that you just recorded with CoCo-80.

After completing this short Quick Start tutorial you should read the following sections and review the complete Users' Manual for a detailed description of the features and operating instructions.

Important Notice about the Concept of CSA

CSA stands for **C**onfigurable **S**ignal **A**nalysis. The first time you use the CoCo-80, you may wonder why some of the signals, or functions, are missing when you run the CSA files. For example, when **DefaultTime(4)** runs, you will find it does not support any acquisition mode, any transient capture or any spectral analysis. You may wonder if this is a software bug. The answer is no. Whether these features are available or not are determined when the CSA project is edited on the host PC.

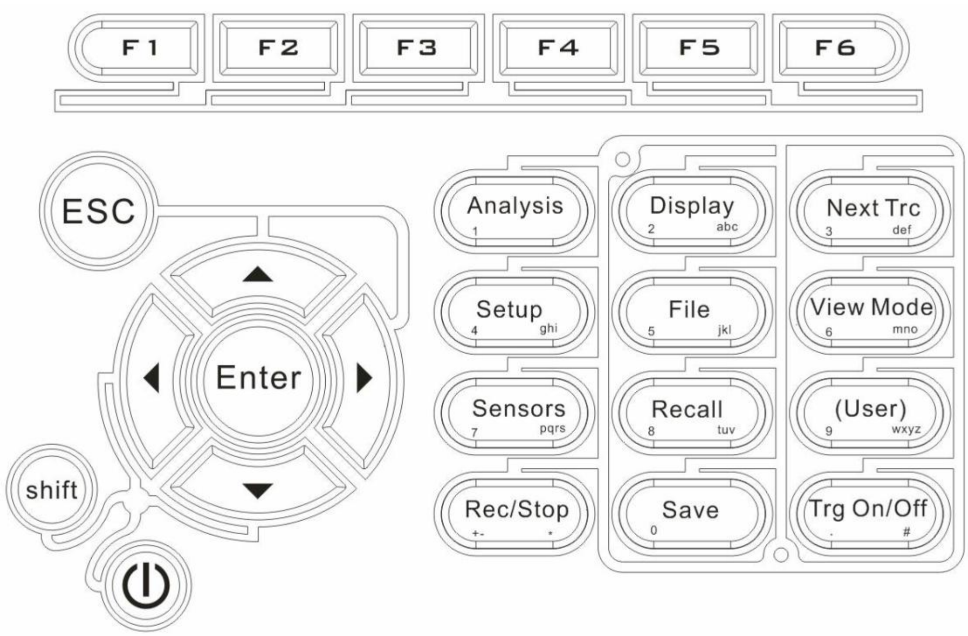
Initially the concept of CSA may appear unconventional to some. However this is the base that provides flexibility and powerful capability to the CoCo-80 and allows each testing case to maintain its simplicity in operation.

3. BASIC COCO-80 OPERATION

This section provides a detailed description of the CoCo-80 device including the user interface, hardware, CSA projects and peripherals.

CoCo-80 User Interface

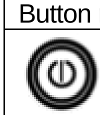
The CoCo-80 menu-driven user interface is easy to use and requires little training. Hard buttons on the front panel are used to initiate function-specific menus. The buttons are divided into three areas. The **navigation button**s include the power, shift, tab, enter and arrow buttons. The **function buttons** include the Analysis, Display, Setup, File, Rec/Stop and Save buttons. The six **soft buttons** located directly below the display change function depending on the current mode selection.



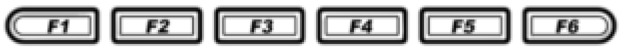
■ Figure 4. Button layout on the CoCo-80 front panel.

Summary of Buttons

The following table gives a brief description of the function of the buttons on the CoCo-80.

Button name	Functions
 Power	Power on the system Power down the system Reset the system (press it 4 seconds or longer)

SHIFT	Shift the functions of the arrow buttons or other buttons
Up arrow	Move the focus up In display window scaling, expand the vertical range In display window scaling, vertically move the display range up (depending on SHIFT position)
Down arrow	Move the focus down In display window scaling, reduce the vertical range In display window scaling, vertically move the display range down (depending on SHIFT position)
Left arrow	Move the focus left In display window scaling, reduce the horizontal range In display window scaling, horizontally move the display range left (depending on SHIFT position)
Right arrow	Move the focus right In display window scaling, increase the horizontal range In display window scaling, horizontally move the display range right (depending on SHIFT position)
Enter	Confirm, accept
ESC	Escape: Cancel, or go back to the previous settings, applies to the scaling of display window, cursor position and other operations
Analysis	Change to analysis screen
Display	Change to the main signal display screen
Setup	Change to the main setup screen
File	Change to the main file view screen
Rec./Stop	Start or stop the time domain data recording designated by the CSA project
Save	Save the signals that are designated by the CSA project
Sensors	Open the input channel setup page to configure the sensors or input channels
Recall	Recall or review the last saved signals
Next Trc	Switch to the next display trace

 View Mode	Open the View mode menu to set up the view mode for the active trace
 Trg On/Off	Turn trigger on or off. When trigger is OFF, it runs in "free run" mode.
 (User)	This button will lead the system to a user previously set CSA and execute it.
	
F1~F6 function buttons	Context dependent function soft buttons

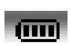
Status Bar

The Status Bar indicates the status of the system.

- Figure 5. CoCo-80 display Status Bar.

Navigation indicates the name of the screen or provides information about the analysis such as sampling rate.

 Volume indicates the volume level for the internal speaker.
Volume for internal speaker (defined in the Setup screen).
 Power indicates battery or line power.

 Battery status indicates the state of charge or if AC power is connected.

System time displays the time (defined in the Setup screen).

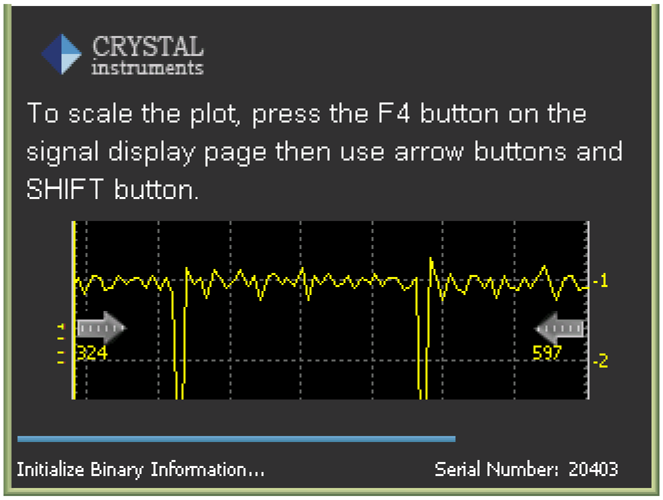
Other status, such as sampling rate, number of averaged frames in spectral processing, number of frames acquired, will be displayed according to installed CSA.

Menu Navigation

The CoCo-80 is operated by moving between screens, entering parameters, and initiating commands with the buttons. This section gives a brief overview of the menu navigation. More detailed information is given in the following sections.

Startup

Press the Power button to power on the unit. The initialization screen shows the startup progress. When the startup sequence is complete the Welcome screen is shown.



- Figure 6. Startup screen is shown during startup sequence.

Power Down

To power down the unit, press the Power button and then press the Turn Off soft button. The Cancel soft button returns to the previous menu without powering down the unit. Lock keypad function can be selected instead of Power Off.

Arrow Buttons

The arrow buttons are used to move the focus from one field to another on the display. By moving the focus you can select deferent fields to enter parameters, select other screens and enter text. They are also used to zoom in and pan around a trace. When cursors are enabled, arrow buttons are used to move the cursor positions. In the trigger setup window, the arrow buttons can be used to move the threshold and trigger delay.

Enter Button

The Enter Button is used to accept an entry or select an item on the display. In general to select an item use the arrow buttons to move the focus to the item and then press the Enter button to select the item.

Shift Button

The Shift Button serves multiple functions depending on the context. In the signal display window, the F4 soft button ZOOM in/out or moves the display. The Shift Button toggles between ZOOM and move. ZOOM changes the size of the plot and move changes the position of the view.

In the Window setup, if you set a two trace window, the Shift button toggles between the top and bottom traces.

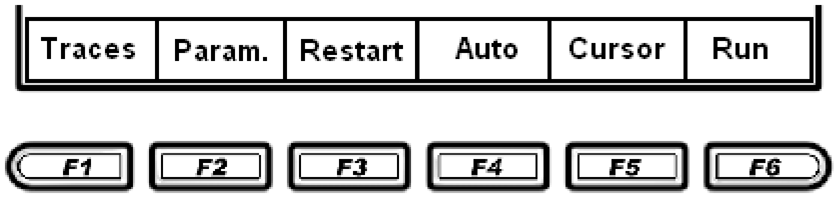
Escape Button

The ESC Button is used to move back to previous screens, or cancel the current action. As the screens are changed using the function or soft buttons, the CoCo-80 remembers the previous screens so that you can easily move back one at time by pressing the Back/Forward button.

Soft Buttons

The F1 – F6 Soft Button functions change depending on which screen is currently shown. Some soft buttons open new screens that include additional soft buttons. To keep the structure clear in the following description the soft button hierarchy will be displayed showing the string of the previous soft buttons or menus in gray and the lowest soft button in black as follows:

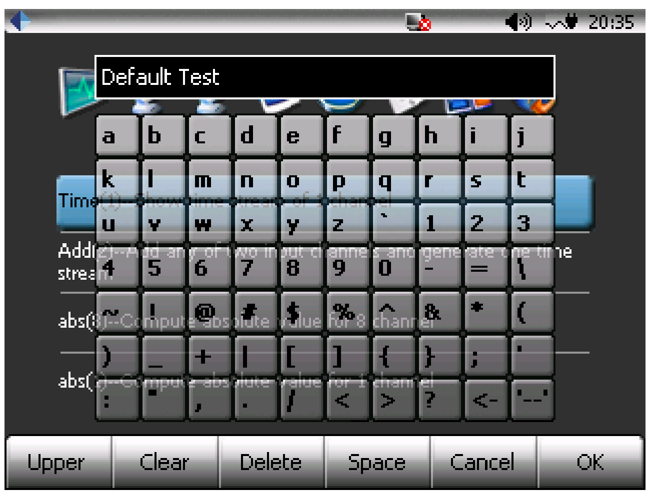
Hard Button Name/Screen Name/Soft Button Name/Screen Name…



■ Figure 7. Soft Buttons change function depending on the current screen.

Text and Number Keypad

Several screens require you to enter text using the keypad. When the text keypad is displayed, use the arrow buttons to move the focus to a letter or number and press the Enter button to select the character. When the text entry is complete press the OK soft button.



■ Figure 8. Text and numbers can be entered in the input screen.

Text Soft Buttons

Upper/Lower toggles the font to upper or lower case font.

Clear deletes all text from the text field.

Delete deletes the character to the left of the cursor.

Space adds a space.

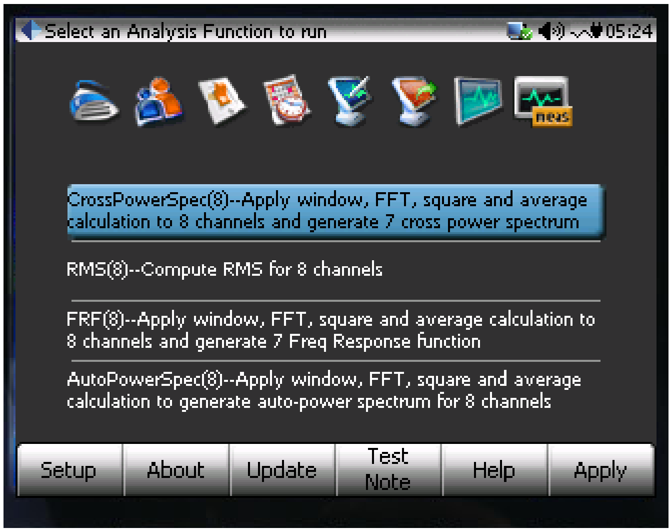
Cancel closes the screen without changing the text.

OK accepts the text and closes the screen.

Note: It is more convenient to use one of 12 hard buttons to enter the number or letters. For example, Button Display can be used to enter number 2 or letter A, B or C. To make the selection, pressing the button quickly will lead the selection to the next one.

Welcome Screen

The Welcome Screen is shown after the system has completed the startup sequence. It shows the available CSA projects at the top and other status at the bottom. It can be used to verify the date and time, network connection settings, add a test note, select a CSA project and run the project.



■ Figure 9. Welcome screen is shown on startup.

Enter Test Note allows you to add text that will be appended to all data files as a file attribute. This is can be used to add notes such as test conditions, locations for later reference and can be used to search through data files by key words. The text typed in the Test Note field will be carried over by the files.

User Information shows the user name and address.

Date and Time indicates the date and time settings of the unit.

Connection indicates the network connection status of the unit.

The CSA project Menu at the top of the screen shows a list of the available CSA projects loaded on the CoCo-80 unit. Use the left and right arrow buttons to select one and press the Run soft button to run the project.

Welcome Soft Buttons

Setup changes to the Setup screen.

About shows the CoCo-80 system information including hardware version, software version, calibration status and software subscription period.

Update connects the CoCo-80 to the CI server via an Internet connection and checks for new files and software updates. You are then prompted to download the updates.

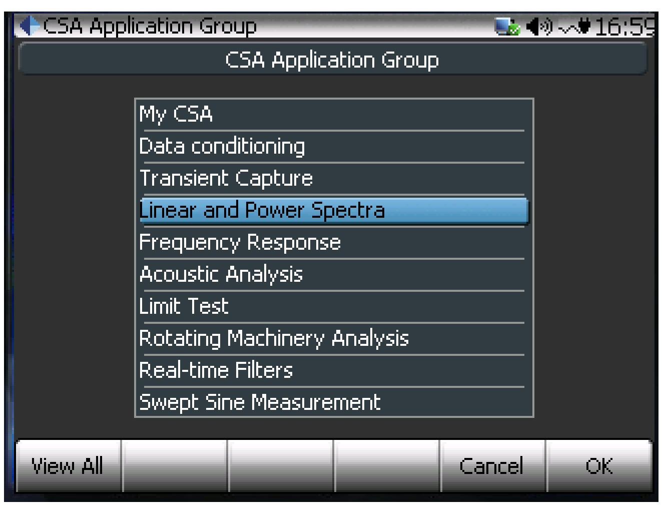
Test Note allows you to enter notes for the test. All data files saved will be attached with Test Note.

Help is the Help document.

Run loads the CSA project and enters into the signal display window.

Analysis Button

The Analysis Button changes the screen to the screen of Configurable Signal Analysis Application Groups.

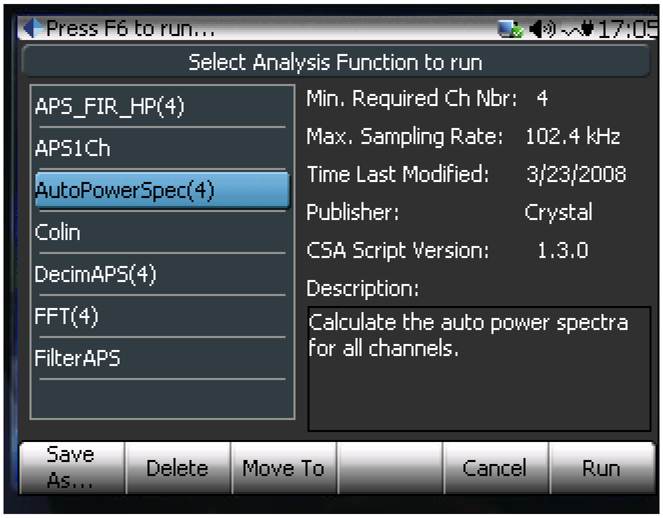


■ Figure 10. CSA Application Groups.

This screen shows several categories of applications. These categories *mostly* match with the template that the CSA uses when it is created with a few exceptions: octave analysis and sound level meter applications are grouped in Acoustic Analysis group. Any CSA with limiting test are grouped in Limit Test group.

After entering one of the application groups, you will see a list of the CSA projects that are loaded on the CoCo-80 in a scroll down menu on the left and information about each project to the right.

Use the up and down arrow buttons to select a project and read the description, maximum sampling rate, time last modified and publisher information on the right. When additional CSA projects are loaded from a PC to the CoCo-80 they will appear on the menu. After selecting one CSA project from the menu press the Run soft Button to load and run the CSA project.



■ Figure 11. Analysis screen is used to select a CSA.

Analysis Soft Buttons

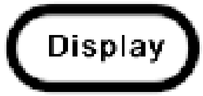
Save As saves the current CSA project with a different name. This can be used to change project parameters and save the new project without overwriting the original project.

Delete removes the CSA project from the CoCo-80 flash memory. The CSA project can be reloaded from the PC if it is accidentally deleted.

Move to lets you move the CSA project file to another group.

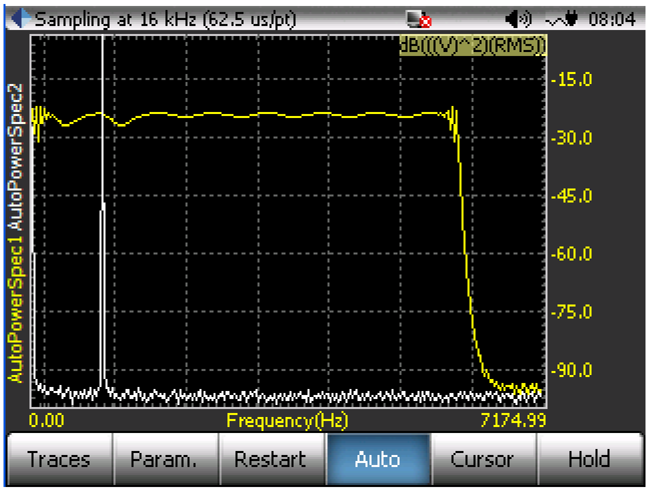
Cancel returns to the previous screen

Run loads the selected project and starts the display. The Enter button also loads and starts the selected project.

Display Button 

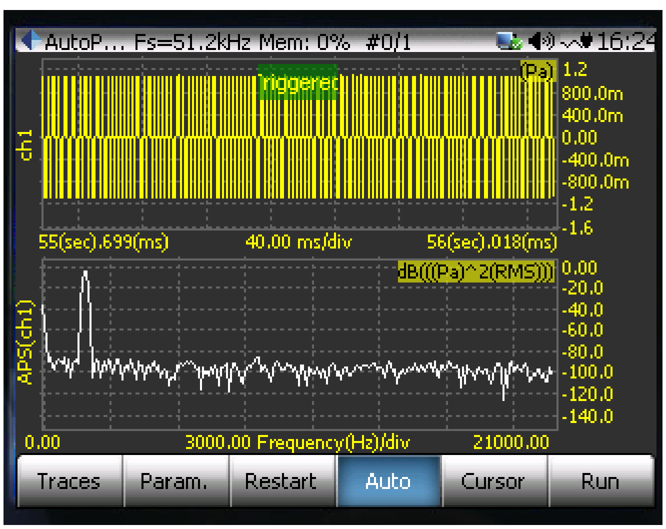
The Display Button changes the screen from the current screen to a center management window for display management. Pressing the Display button and Enter will always lead to displaying the current active window.

The following screen shows the *Signal Display Window*. This is the most frequently used window in this instrument.



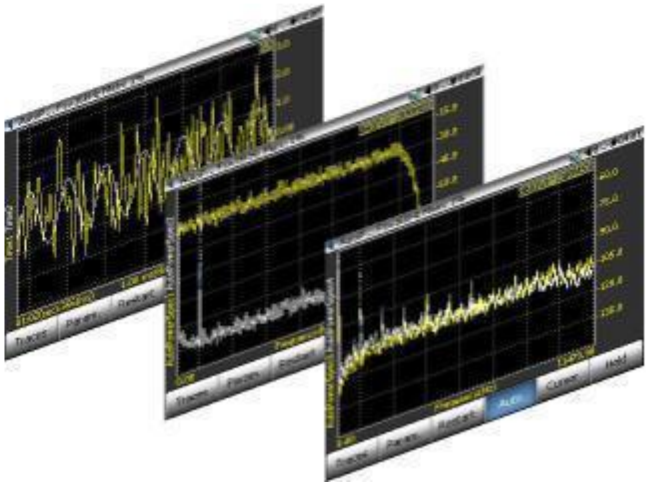
■ Figure 12. Display screen shows a window with one trace.

A signal display window can have either one or two traces. The software allows three types of signal display windows: A window with one trace; a window with top and bottom traces, a window with a 3D waterfall trace. The picture below shows a window with two traces.



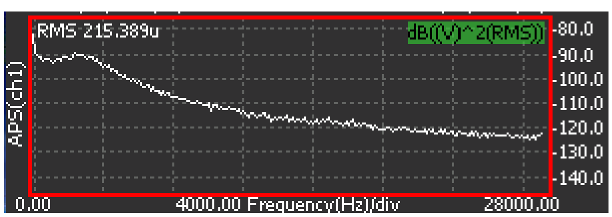
■ Figure 13. Two trace window.

A trace is defined as area display with an axis that can show multiple signals of the same dimension such as time or frequency. Only the signals with the same engineering units in X and Y axis can be overlaid.



■ Figure 14. Multiple traces can be defined and multiple signals can be displayed in each trace.

The traces are periodically updated when the Display is in Run mode. To stop the trace updating press the *Hold* soft button. Note that the trace updating display is independent of the Record operation. This means that while traces are update on the display they are not recorded to memory until the Rec./Stop button is pressed. You should understand the difference between the trace update display and the record feature so that errors are not made in recording.

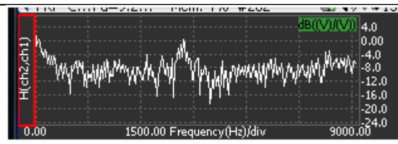
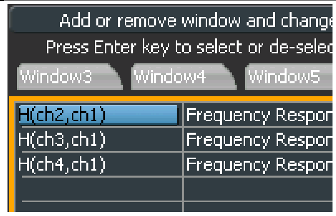
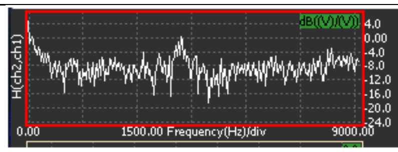
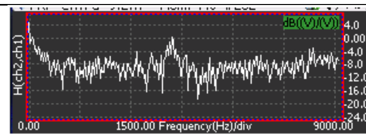
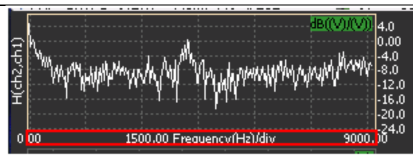
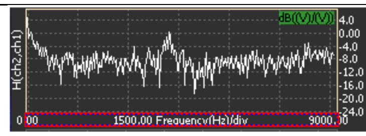
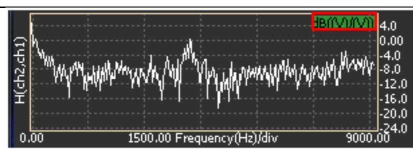
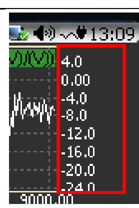
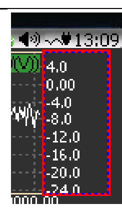


A trace typically consists of five objects:

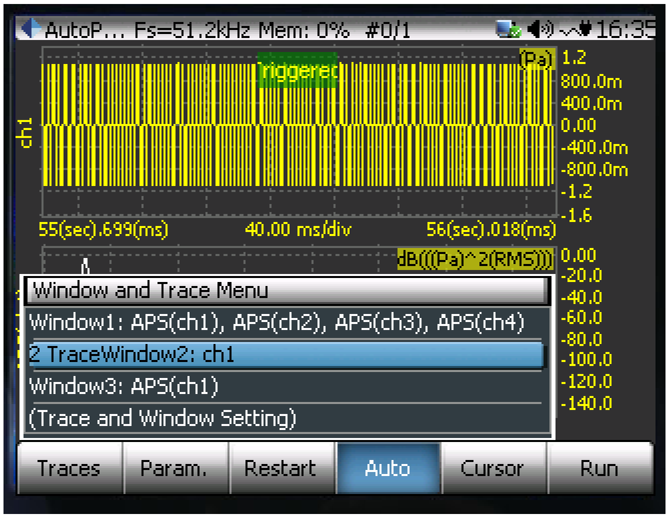
1. The signal label on the left side (displayed as APS(ch1) in this example)
2. The center display area (the area being highlighted)
3. The view mode (displayed as dB((V)^2(RMS)) in this example)
4. The vertical Y-scale range on the right
5. The horizontal X-Scale range on the bottom

User can move around from one object to another. Once one of the objects is highlighted, pressing Enter button will guide to an operation to set the property of that object. Below is a table describing the corresponding property when Enter button is applied to certain area on the screen:

Highlight Area	After pressing Enter button	Description

		Highlight the Y Label area then press the Enter button. CoCo will show ***Change Signals*** in Trace window
		Highlight the center plot display area then press the Enter button. CoCo will stay in ZOOM mode for a few seconds. Use four arrow buttons to scale the window.
		Highlight the X label area then press the Enter button. CoCo will stay in X scaling mode for a few seconds. Use four arrow buttons to scale the X scale.
		Highlight the Y display unit then press the Enter button. CoCo will let you set the view mode.
		Highlight the Y grid area then press the Enter button. CoCo will stay in X scaling mode for a few seconds. Use four arrow buttons to scale the Y scale.

Signal Display Window/**Traces Soft Button** opens the *Trace and Window Setting* Menu. This menu lists the names of the existing windows in the display and is used to change the windows and signals in the display. The menu lists the defined windows at the top of the menu. The display can be changed from one window to another by selecting a different window from the menu and pressing the Enter button. You can define any number of windows to provide a flexible display format.



■ Figure 15. Window and Trace Menu.

Signal Display Window/**Traces/Trace and Window Settings Menu** opens the Trace and Window Setting screen. This screen is used to add new windows and define which signals will be included in each trace.

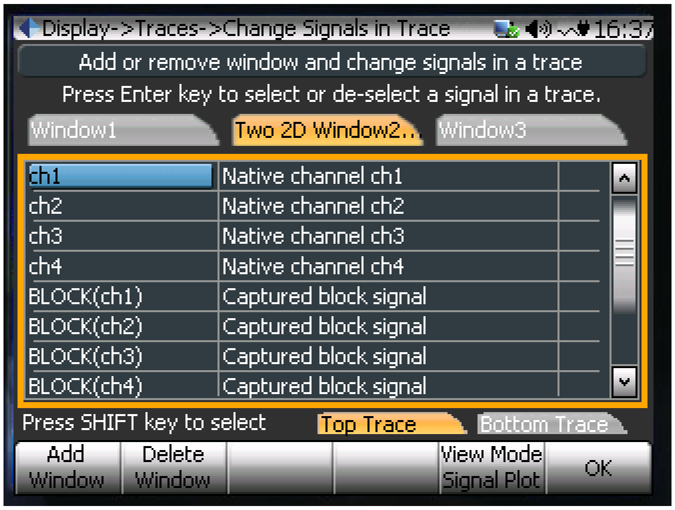
After you select Trace and Window Setting and **Add Window**, you can choose from a single 2D trace, two 2D traces or one waterfall trace.

After you select the window type, you need to assign the signals to be displayed in this window. To select a signal press the Enter button to check it. Only signals that are defined in the CSA as display candidates will be visible and can be added to the trace. This feature is designed to simplify the user interface and optimize the CoCo-80 computation resources. If a signal is not available for display then the CSA must be edited before it can be added.

After the first signal is selected, all the signals with different types to the first one will disappear from the list. You can continue to select the rest of the signals which are with the same type as the first one to overlay them.

Set a Window with Two Traces

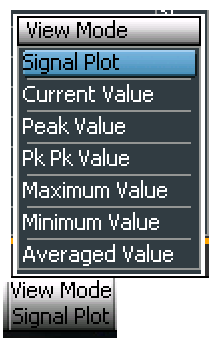
To set up a window with two traces, first select ***Add a Window with two Traces.*** Then the following screen will be displayed:



- Figure 16. Edit window screen is used to add/delete window and add signals to each trace.

In this screen you can set the top and bottom trace by selecting appropriate signals in the same way as you set up the one trace window. **To switch between top and bottom traces within the window, use SHIFT hard key.**

View Mode: The signals can be displayed in either text or graphic plot mode. When it is in the text mode, one of the signatures of the signal must be selected.



- Figure 17 In screen of Trace and Windows Setting different view mode can be selected

Signal Display Window/Traces/Trace and Window Setting Menu/**Signals Displayed in Trace** shows the signals that are attached to the selected trace. Signals can be added to the current trace by selecting them with the arrow buttons and pressing the Enter button to highlight the signal. All highlighted signals are displayed in the current trace. Only signals of the same dimensions can be included in the same trace. For example time and frequency type signals cannot be included in the same trace.

Signal Display Window/Traces/**Trace and Window Setting Menu Soft Buttons**

Add Window creates a new window in the Window List. Windows are named sequentially as Window1, Window2, etc.

Delete Window removed the highlighted window from the Window List.

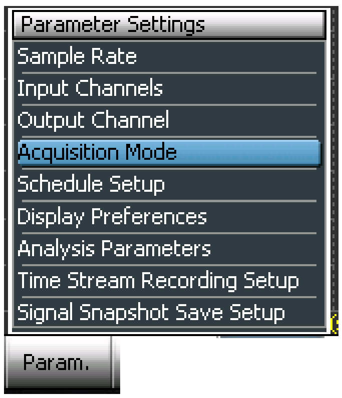
Clear Signals removes all signals from the highlighted trace.

Select All Signals adds all signals to the highlighted trace.

Cancel returns to the previous screen without changing the trace definitions

OK saves the changes to the trace definitions and returns to the previous screen.

Signal Display Window/**Param Soft Button** opens the Parameter Settings Menu. This menu allows you to set parameters for the sample rate, input and output channels, triggering, analysis and time stream recording.



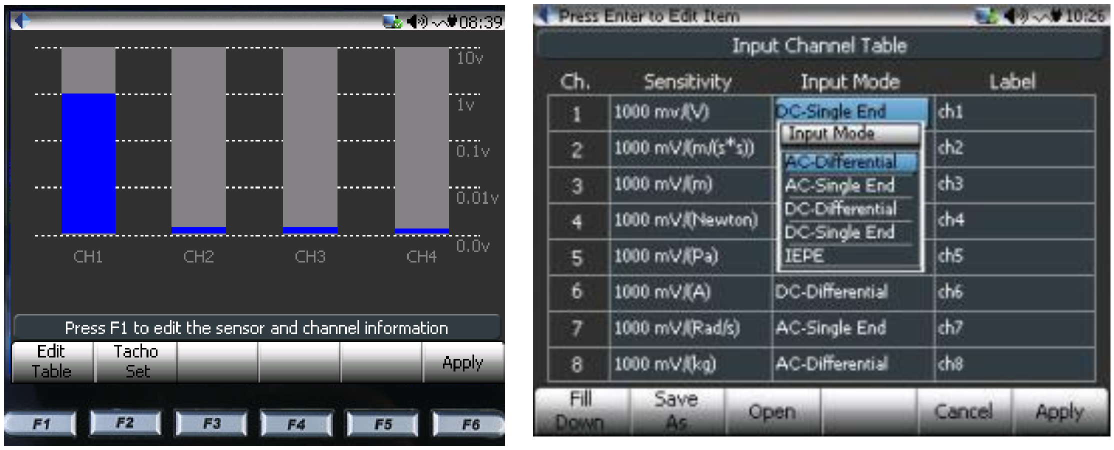
■ Figure 18. Parameter Settings menu.

Signal Display Window/Param Soft Button/**Sample Rate** is used to set the sampling rate for data acquisition. Use the up and down arrow buttons to select from the scroll menu and press the Enter button to accept the setting. Sampling rate is a global setting that applies any loaded CSA.

Signal Display Window/Param Soft Button/**Input Channels** is used to set the sensitivity, input mode and label for the hardware input channels. To edit these parameters use the arrow buttons to select the parameter and press the Enter button. Input Channels is a global setting that applies any loaded CSA.

When the user selects Input Channels menu item, the channel status screen will be shown. It displays the peak magnitude of each channel over a certain period of time. Notice that the vertical scaling of the bars is in logarithmic. The log scaling will help the user see both large and small signals.

Thanks to the high-dynamic technology implemented in the CoCo, as long as the signals are within the full range, the measurement will be reasonably accurate. However if the signals are above the full range, overload will occur and the instrument will flash to warn the user.



■ Figure 19 Left: Input status; Right: Input Channel Table

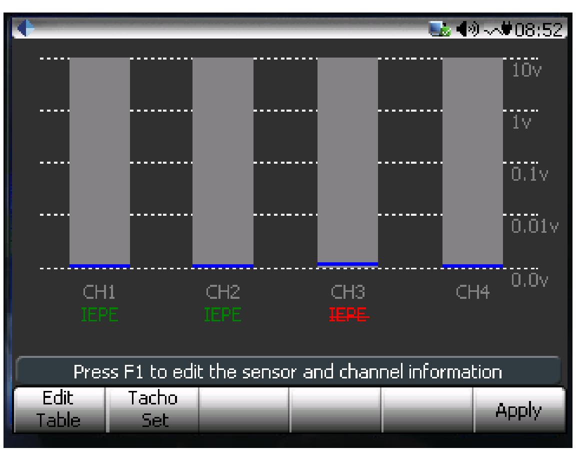
Signal Display Window/Param Soft Button/Input Channel/**Sensitivity** is used to set the physical quantity, units and sensitivity of the input channel. Use the arrow buttons to select the parameter and press the Enter button to select it. The parameters can by applied to all channels using the soft button Apply to all Ch.

Physical Quantity defines the quantity such as acceleration, velocity, displacement, force, voltage, etc.

Units defines the engineering units such as m/s2, cm/s2, gn, etc. for the input channel.

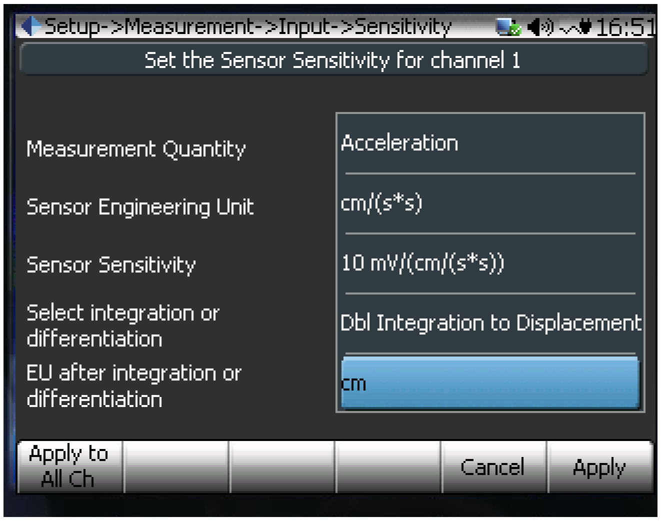
Sensitivity defines the sensitivity in *millivolts/engineering unit* defined in the unit menu. This selection opens a numeric keypad to enter the sensitivity value. Press the OK soft button to accept the value.

When the Physical Quantity is selected as Acceleration, you have the choice to apply a built-in integration or double-integration to generate readings in velocity or displacement. When the Physical Quantity is selected as Velocity, you have the choice to apply integration to displacement. Notice that the algorithms for integration are implemented in the digital domain. They also included a high-pass filter and DC removal routines.



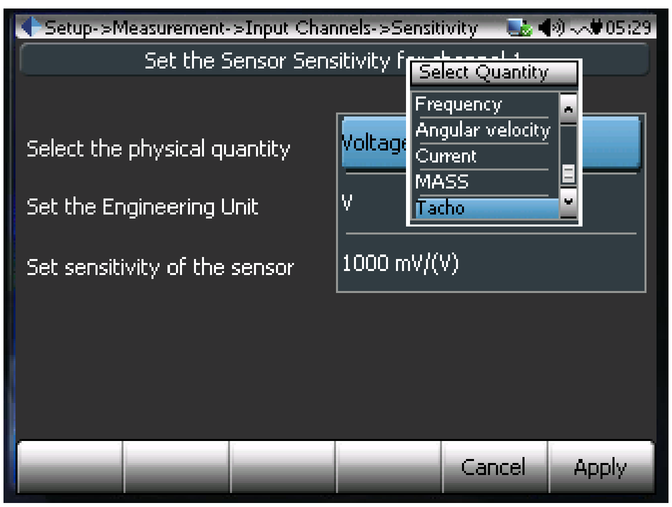
The instrument can automatically detect the status of IEPE sensor connection. If the IEPE type of sensor was not connected correctly, the input channel status will tell.

In the picture above, channel 1, 2 and 3 are enabled with IEPE input mode in software, channel 4 is not. Since channel 1 and 2 are connected with the IEPE sensors, green letters, IEPE, are shown. Channel 3 is not connected to an IEPE sensor therefore IEPE is displayed in red and a crossing line.



■ Figure 20. Apply built-in digital integration or double integration.

Channel 1 is uniquely designed that it can take tacho input. To do so, simply select the Tacho item under the Physical Quantity for channel 1.

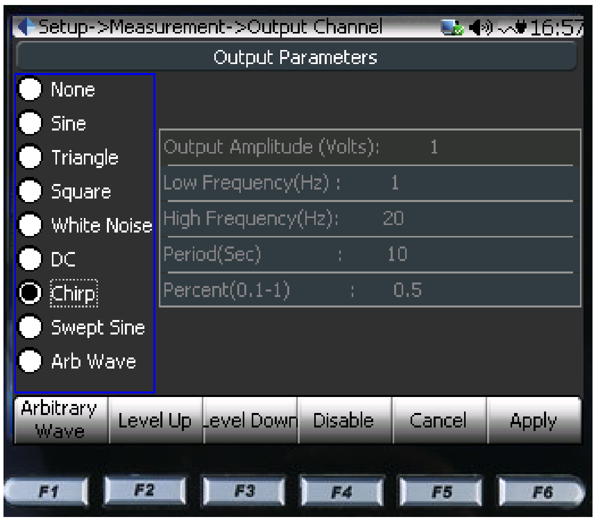


■ Figure 21. Select the Tacho measurement for channel 1.

Signal Display Window/Param Soft Button/Input Channel/**Input Mode** is used to change the input mode. The choices are AC-Differential, AC-Single Ended, DC-Differential, DC-Dingle Ended and IEPE.

Signal Display Window/Param Soft Button/Input Channel/**Label** is used to change the name of the signal. Use the alphanumeric keypad to enter a label name and press the OK soft button to accept it.

Signal Display Window/Param Soft Button/**Output Channel** is used to define the waveform for the output channel. First use the left and right arrow buttons to set the focus. When the focus is set to the left region, you can select one of the signal sources. Use the up/down arrow buttons to select from Sine, Triangle, Square, White Noise, DC, Chirp, Swept Sine or Arbitrary Signal. *None* turns off the output channel. When the focus is set to the right, you can modify the parameters for that particular signal source.

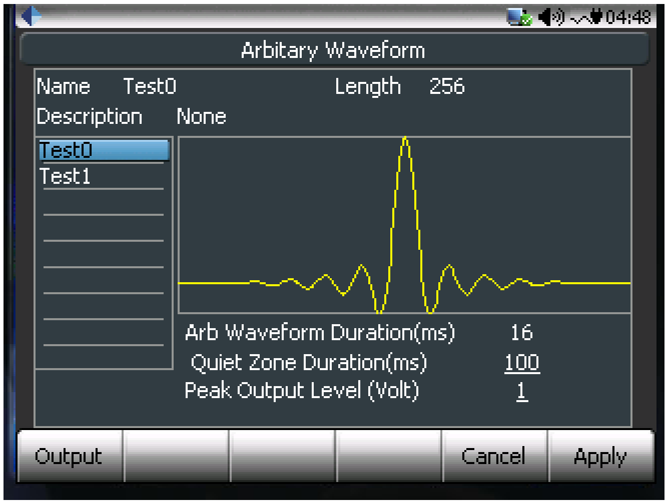


■ Figure 22. Output Channel screen.

For each waveform the parameter settings must also be enters such as range, frequency and amplitude. Output Channel is a global setting that applies any loaded CSA.

Apply saves the settings, activates the output channel and returns to the previous screen.
Cancel discards the settings and returns to the previous screen.

When the *Arb waveform* is selected, you can output an arbitrary waveform file. This file must be uploaded to the CoCo-80 through EDM before it can be used.



■ Figure 23. Arbitrary waveform setup.

In the arbitrary waveform setup, the duration is fixed by the number of points in the arbitrary data file and the sampling rate in use. The Quiet Zone is the time with "zero" output between two arbitrary waveform pulses. The Peak Output Level is the normalized maximum volt for the output waveform. Regardless the value in the arbitrary file, it is always normalized to this peak level volt.

Signal Display Window/Param Soft Button/**Acquisition Mode** is used to configure how the data blocks are captured from the conditioned time streams into the signal analyzer phase. The selections are:

- Free run
- Continuous after trigger
- Single Shot by user
- Single Shot with trigger
- Multiple shots with trigger (auto-rearm)
- Multiple shots with trigger (manual-rearm)

It is important to note that the Acquisition Mode is designed for signal analysis functions only, such as spectrum measurements. The Data conditioning process is not affected by the Acquisition Mode. For example data recording will continue uninterrupted regardless of the Acquisition Mode. Acquisition Mode setting is dependent on the selected CSA.

Signal Display Window/Param Soft Button/**Schedule Setup** is used to configure the automated test.

- Figure 24. Schedule Setup.

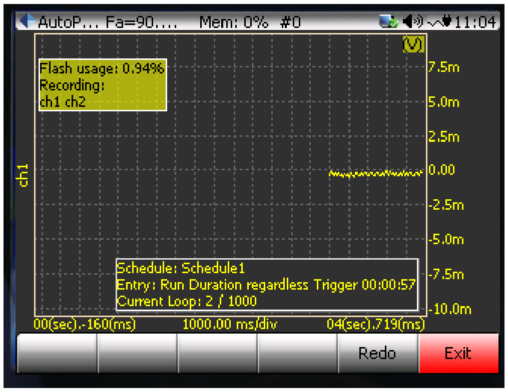
Testing schedule automatically controls the test duration and imitate human operation. Multiple testing schedules can be developed and one is executed at a time. A testing schedule event can include the following events: Loop/End-Loop, Run Duration, Hold, Limit Check on, Limit Check off, Start Recording, Stop Recording, Save Signals, Turn Signal Source On and Turn Signal Source Off.

Activating the testing schedule requires an extra step. This is designed to ensure the test schedule does not start inadvertently. To activate the testing schedule go to the main signal display window and press the Display button for more then 3 seconds then release it. This will activate the test schedule. After the test schedule is activated, all the buttons, except the power button, F5 and F6, will not deactivate because the test is in the automatic mode.

To exit the automated schedule, press F6 the exit button. To redo the schedule, press F5.

Using schedule smartly allows the instrument to divide the total measurement into multiple files. It is easier to retrieve and analyze. The picture below shows a typical schedule with loop/end loop functions. It allows the instrument records 1 hour, stop, run for 1 min, and repeats this recording process 1000 times.

When the schedule is activated, the schedule status will be displayed during the run time.



Use following formula to calculate the total time duration that you can record:

Total Installed Flash Memory in Bytes = (Total Channel Enabled) * (Recording Time in Seconds) * (Sampling Rate) * 8 Bytes * 1.2

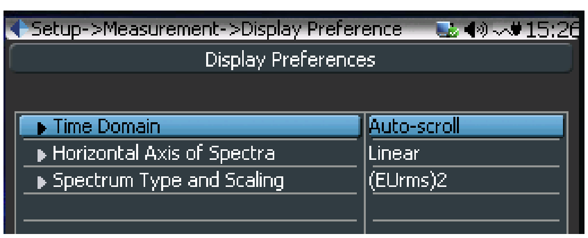
Or

Recording Time in Seconds = Total Installed Memory in Bytes/((Total Channel Enabled) * (Sampling Rate) * 8 Bytes * 1.2)

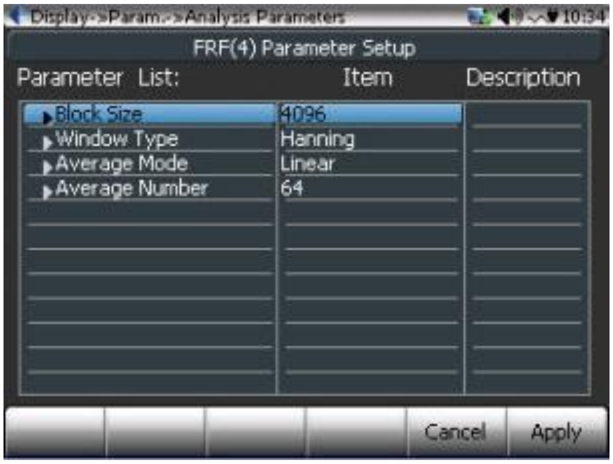
For example if you enabled 6 channels, the sampling rate is 100 Hz with 4GB memory installed:

Recording Time in Seconds = 4GB/(6*100*8*1.2) = 4*1024*1024*1024/(6*100*8*1.2) = 745654 sec (= 207 hours)

Signal Display Window/Param Soft Button/**Display Preference** is used to set up the preferences about signal display. Notice that the spectrum type and horizontal axis, either in linear or log, is set up here.



Signal Display Window/Param Soft Button/**Analysis Parameters** is used to change parameters that are defined in the CSA project. These parameters depend on the definition of the CSA project but may include block size, window type, average mode, average number, excitation and response channel, etc. Refer to the CSA project description in Section 4 for more details. Analysis Parameters setting is dependent on the selected CSA. Different set of CSAs may show completely different Analysis Parameters.

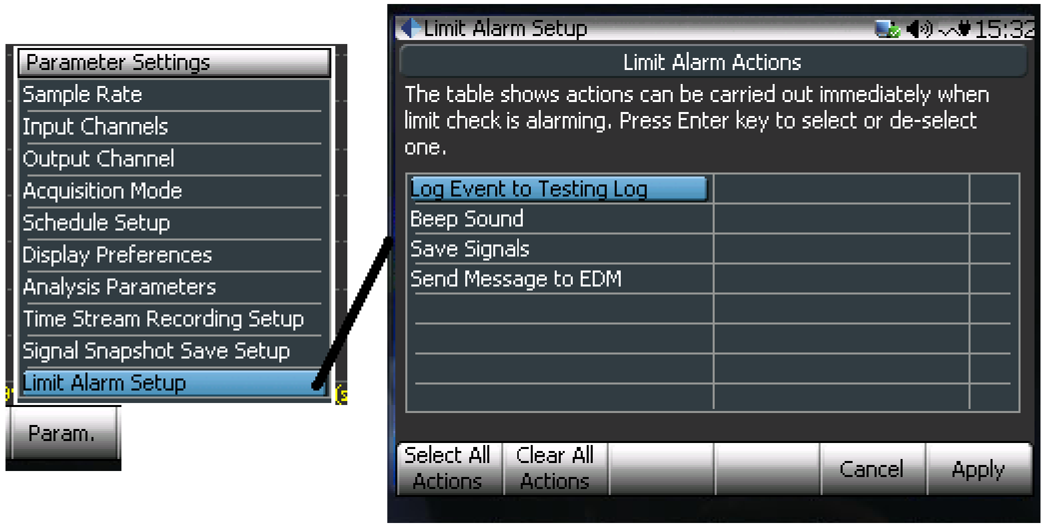


■ Figure 25. Analysis parameters setup screen.

Signal Display Window/Param Soft Button/Time Stream Recording Setup defines which time streams will be recorded to memory when the Rec./Stop button is pressed. To add a stream to the record list select if in the Signal List using the up and down buttons and press the Enter button. Note that the more signal added to the Record List increases the file size of the recording and reduces the record time to flash memory. Only streams that are of interest should be recorded to conserve memory and maximize recording time. Time Stream Recording Setup is dependent on the selected CSA. Note that only signals that are identified as ***Record Candidates*** in the CSA file will be visible and can be recorded. This feature is designed to simplify the CoCo-80 user interface and optimize the device computation resources.

Signal Display Window/Param Soft Button/Signal Snapshot Save Setup defines which signals will be recorded to memory when the Save button is pressed. To add a signal to the record list select if in the Signal List using the up and down buttons and press the Enter button. Signal Snapshot Save Setup is dependent on the selected CSA. Note that only signals that are identified as ***Save Candidates*** in the CSA file will be visible and can be saved. This feature is designed to simplify the CoCo-80 user interface and optimize the device computation resources

Signal Display Window/Param Soft Button/Signal Snapshot Save Setup defines the actions to be taken when testing limit exceeded. Notice that this menu item will only be shown when the CSA project contains limiting check. Usually you won't see this menu entry.



Limit Check Alarm Events include Beep, Screen Flashing, Event Log into Testing Log, Send Message to Host PC, Save Signals.

Signal Display Window/**Restart Soft Button** is used to reset the relative time base of the time streams and also reset the averaging and triggering if these features are used in the current CSA project. The *Restart* will be shown as *Control* button in octave analysis, order tracking, swept sine or other advanced software functions. Restart or Control is used to control the running status without re-initializing the project and test.

Signal Display Window/**Auto-Scale/ZOOM/Move Soft Button** controls the vertical scaling of the trace. Auto applies an automatic vertical scaling so that the scale is continuously adjusted to fill the trace. ZOOM scale turns off the automatic scaling and uses the current scale regardless of the magnitude of the signals. When in the ZOOM scaling mode, the four arrow buttons are used for the purpose of reducing or expanding. Pressing the SHIFT button will switch from ZOOM to Move, or Move to ZOOM. When in the Move mode, the four arrow buttons are used for the purpose of repositioning the window. The following diagram further explains the changes in the three different mode of using the navigation buttons:

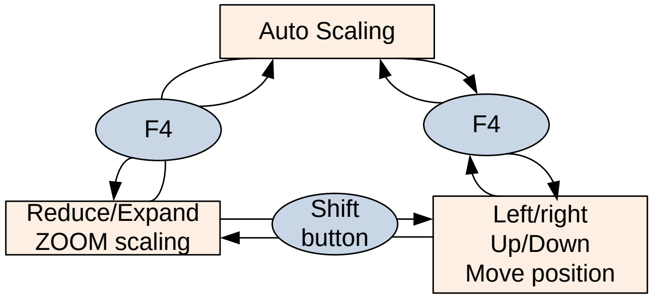


Figure 26. Trace navigation buttons.

Signal Display Window/**Cursor Soft Button** adds a vertical cursor to the trace. Use the right and left arrow keys to move the cursor. The signal values are listed to the right for all signals in a trace. Press the Cursor button again to remove the cursor from the trace.

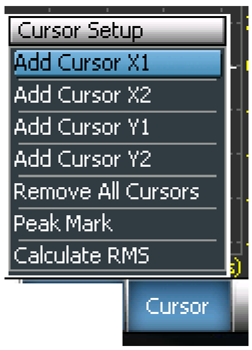
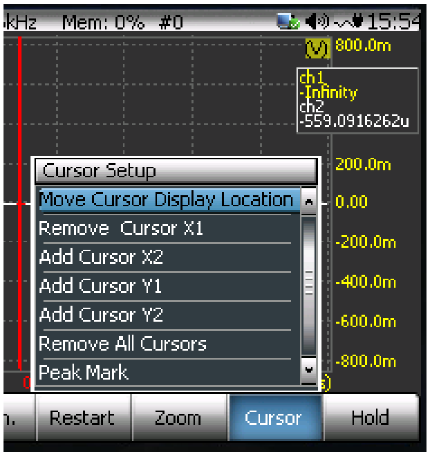


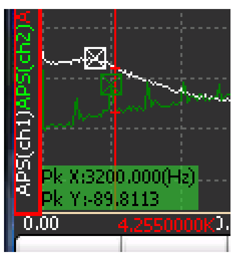
Figure 27. Cursors can be added to a trace.

After the cursor is added, you will see a menu item is added to the Cursor Setup menu, Move Cursor Display Location. If you select it, you will be able to move around the square area of displaying the cursor value by using the four navigation buttons. This function is helpful if you feel the cursor display box is not in the right location. Pressing Enter button will fix the cursor display area.

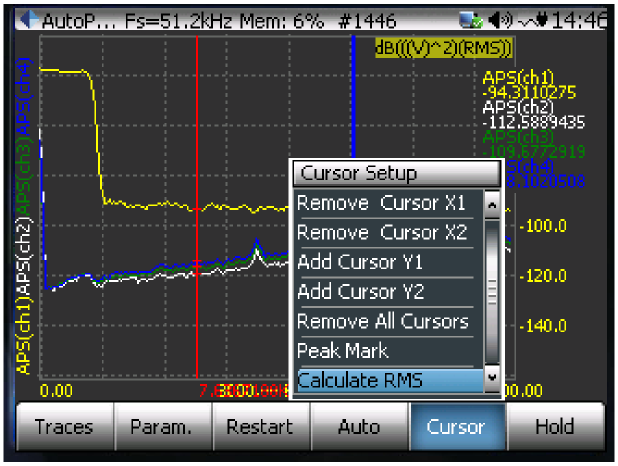


■ Figure 28. Cursor setup.

To search for the peak using cursor, simply press the upper arrow button when a cursor is enabled. The cursor will automatically search a peak of the signal in +/- 10% of the horizontal axis area.

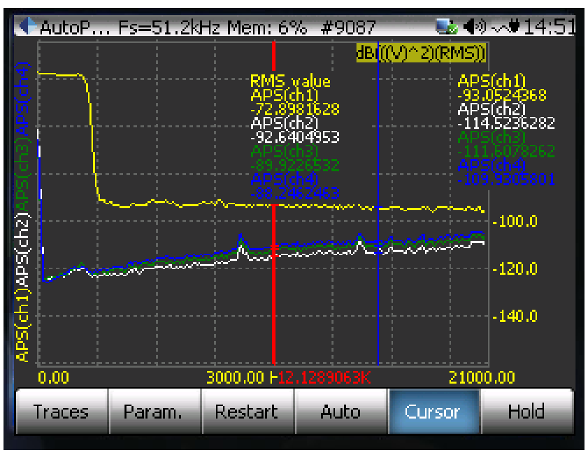


Two vertical cursors, two horizontal cursors and a peak mark can be applied. To calculate RMS within a frequency band for auto spectral signals, select *Calculate RMS* menu item:



■ Figure 29. Cursor setup.

The RMS values will be displayed in the same unit as Y label unit. The RMS is estimated to the energy between two vertical cursors.



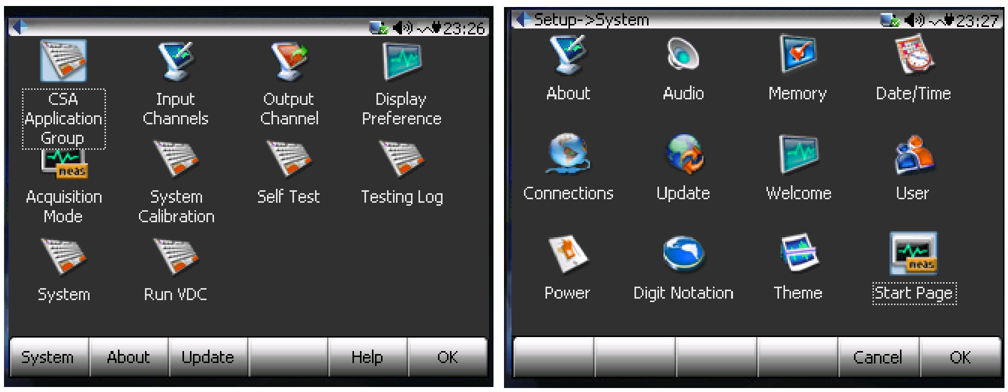
■ Figure 30. Multiple cursor display.

Signal Display Window/**Run-Hold Soft Button** controls the display update and the signal analysis process. When the device is in Run mode the display updates the traces with the signals as fast as possible. When the device is in Hold mode the display stops updating. Note that Run/Hold is independent of Record/Stop. This means that when in Run mode signals are not recorded to memory until the Rec./Stop button is pressed. The record status is indicated by the red record icon blinking at the top of the screen during recording. It is important for you to understand the difference between Run/Hold and Record/Stop so that operator errors are not made in recording signals.

When in Hold mode, the signal analyzer will be held. Processing such as spectral analysis will be frozen.

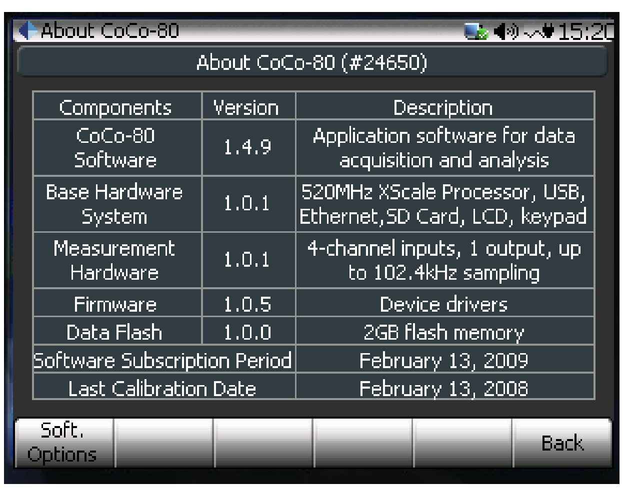
Setup Button

The Setup Button changes the screen to the Main Setup screen. There are two pages of Setup. The first page is all about the measurement settings. Pressing the System icon (F1) will prompt the second page. the second page allows you to change the system parameters including audio, memory, date/time, connection, update, welcome, owner, power, digit notation, theme and measurement settings. Use the arrow buttons to select one of the setting icons and press the Enter button to select it. These settings are described below.



- Figure 31. Setup Screen

About displays hardware and software version information, software subscription period and calibration status.


About CoCo-80

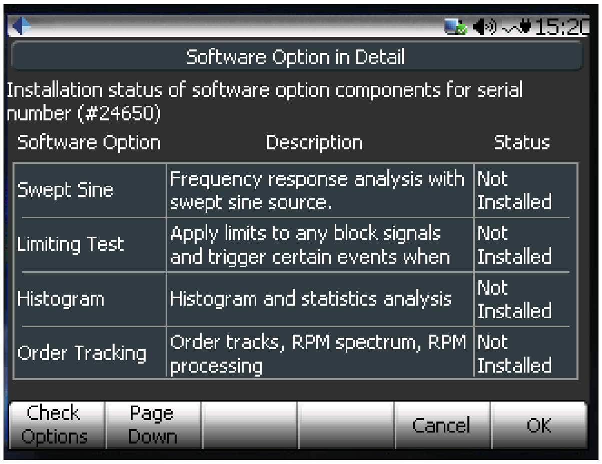
About CoCo-80 (#24650)

Components	Version	Description
CoCo-80 Software	1.4.9	Application software for data acquisition and analysis
Base Hardware System	1.0.1	520MHz XScale Processor, USB, Ethernet,SD Card, LCD, keypad
Measurement Hardware	1.0.1	4-channel inputs, 1 output, up to 102.4kHz sampling
Firmware	1.0.5	Device drivers
Data Flash	1.0.0	2GB flash memory
Software Subscription Period		February 13, 2009
Last Calibration Date		February 13, 2008

Soft. Options
Back

- Figure 32. About Box

Press the F1 Software Option button, the screen will show all installed or uninstalled options.



■ Figure 33. Software Options.

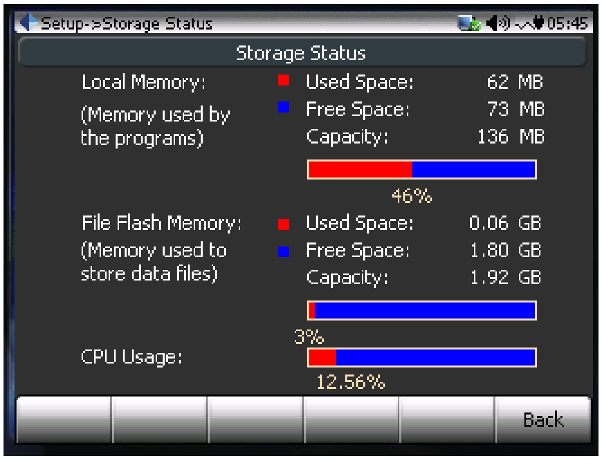
If you press the F1 ***Check Options*** button, the CoCo will check the available software options that can be installed on the remote CI server.

Audio allows you to change the audio feedback settings including keypad, power button and alarm sounds. The speaker volume and microphone level can also be changed. Voice annotation is controlled through Audio setup as well.

■ Figure 34. Audio Settings including Voice Annotation.

Among these settings, the ***Use microphone to record the voice annotation*** and ***Use headphone to listen to any input channel*** are advanced audio functions. User needs to purchase this software option to enable such setting.

Memory displays the status of the CoCo-80 memory. This includes local memory used by the CoCo-80 software and the flash memory used to store recorded data. This display can be used to monitor the remaining flash memory remaining during field operations. When flash memory is full then the data must be downloaded to the PC and removed from the CoCo-80 before more data can be recorded.



■ Figure 35. Memory and DSP CPU usage.

Date/Time allows you to enter the current date and time so that this information can be included as a file attribute with the data files.

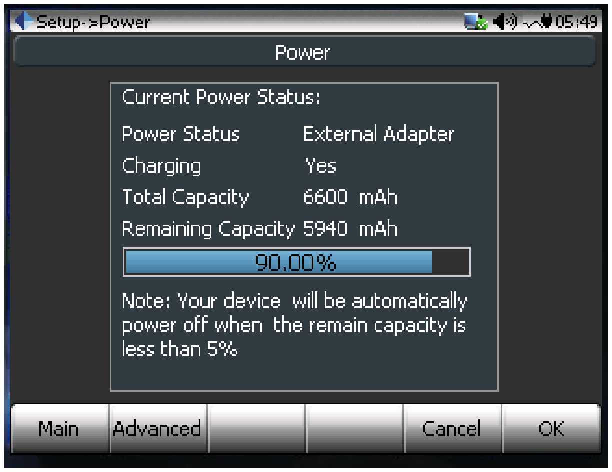
Connections displays the status of the Ethernet, USB Client or Wireless connections. The IP Setup soft button allows you to specify a fixed IP address or to use DHCP.

Update allows the CoCo-80 to check for new software components on the Crystal Instruments server and conduct online software updates at the user's request. CoCo-80 must be connected to the Internet using Ethernet or wireless when on-line update is performed

Welcome shows the welcome screen that lists the available CSA projects loaded on the CoCo-80 and other status.

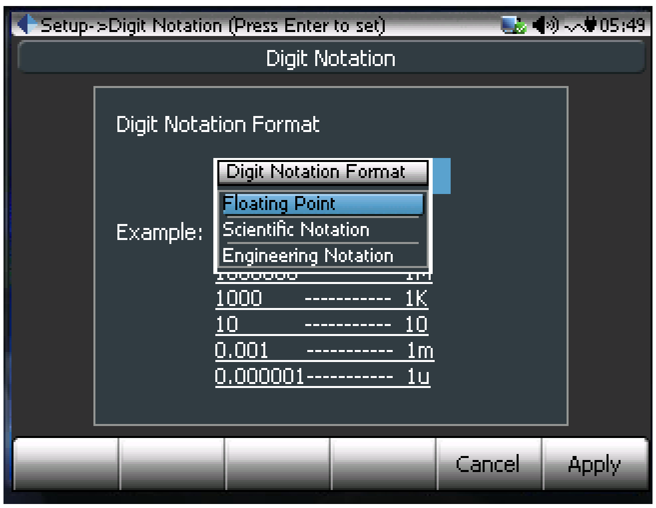
User shows the information recorded for the user of the hardware including Name, company, address, telephone and Email and appends the information as an attribute to all data files. This information can be edited by selecting it with the arrow buttons and pressing the Enter button.

Power indicates the status of the power including the Remaining capacity of the battery. The Advanced soft button allows you to customize the power settings to optimize the battery life for specific conditions including Automatic mode which maximized the battery life by automatically turning off the LCD and the backlight and Ethernet. Maximum Active Mode keeps all components on but uses the maximum power consumption.



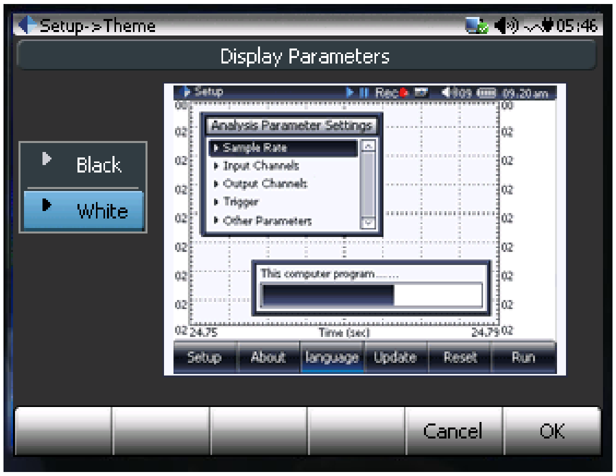
■ Figure 36. Power Status Screen.

Digit Notation is used to change the format that numbers are displayed on the CoCo-80. The choices include: Floating Point, Scientific or Engineering notation.



■ Figure 37. Digit Notation Settings.

Theme changes the display from black to white background.



■ Figure 38. Theme Settings: Black or White Style.

Measurement Page shows the Measurement Settings including Input, Output, Display Preference, Acquisition Mode, System Calibration, Self Test and Testing Log.

■ Figure 39. Setup/Measurement Screen shows the Input, Output, Display and Acquisition Mode settings.

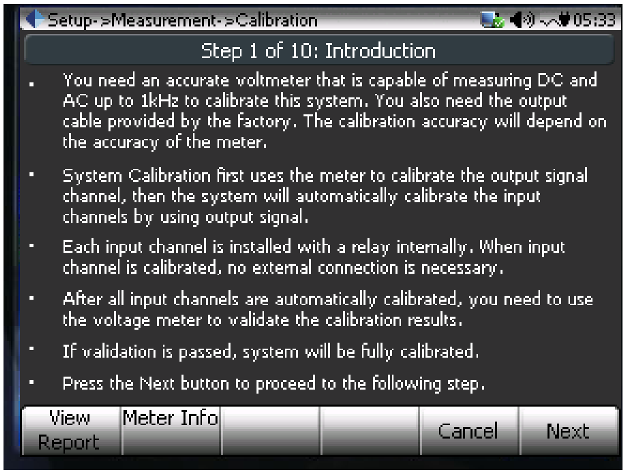
Setup/**CSA Application Group** is the same as pressing Analysis button.

Setup/**Input Channels** changes the display to the Input Channel Table. This is described above under the Analysis/Parameters section.

Setup/**Output Channel** changes the display to the Output Parameter screen. This is described above under the Analysis/Parameters section.

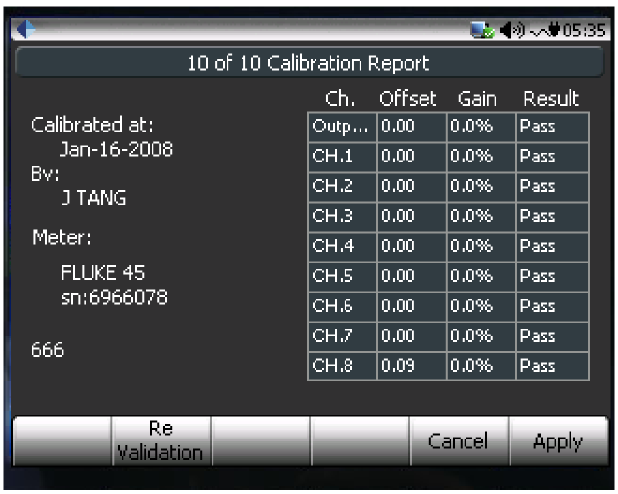
Setup/**System Calibration** is used to calibrate the input and output channels. This procedure requires a high precision calibrated AC/DC volt meter. CI recommends that this task is performed by the factory or designated companies that are familiar with the calibration process.

The picture below shows one of the ten steps to conduct the calibration. It will take about 10 minutes to complete a calibration.



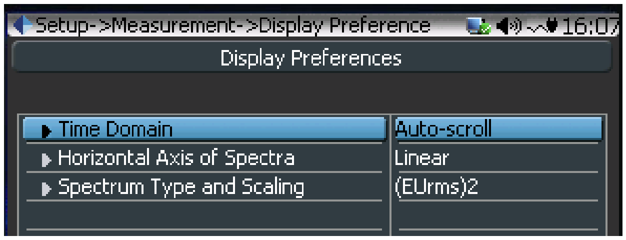
■ Figure 40. System Calibration Screen.

Calibration results can be viewed on the CoCo-80 by pressing View Report in the above screen.



■ Figure 41. Calibration Report.

Setup/**Display Preference** changes to the Signal Display Preferences that are applied to all traces.



- Figure 42. Display Preference screen.

Setup/**Time Domain** Block, Scroll or Auto-scroll. The block display option displays one block of data when the buffer is filled. After a buffer is displayed, the system will grab another buffer a fast as possible for the next block display. Some portion of time signals between buffers will be ignored for display. The scrolling display shows data scrolling continuously on the screen like a strip chart with no gaps in the scrolling display. The auto-scrolling option enables the system to choose the best display method for the user defined horizontal time interval. Usually if the total time interval is less than 0.5sec, it is displayed one block at a time as in block mode while a longer time interval will let the system display the signals with the scrolling method.

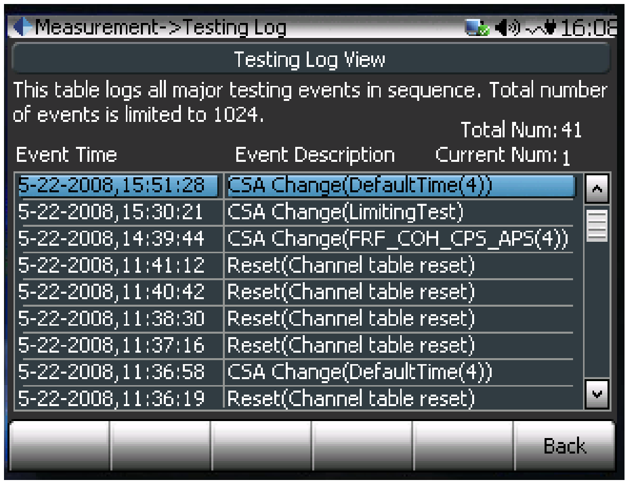
Horizontal Axis of Spectra specifies the format of the horizontal axis of spectra as either linear or log scale.

Spectrum Type and Scaling specifies the vertical scaling for spectrum traces. The choices include: $(EUrms)^2$, EUpeak, EUrms, and $(EU)^2$/Hz. This selection only affects the auto-power spectra or linear FFT spectra.

Setup/**Acquisition Mode** Soft Button specifies the trigger settings including free run, continuous after trigger, single shot without trigger, single shot with trigger (manual re-arm trigger), or automatic re-arm.

Setup/**Self Test** Soft Button allows you to run a self test for the hardware without using external meter. A built-in precision signal source is used to check whether the input channels are in the reasonable range. If the circuitry of any channel is damaged or goes out of range, the Self Test will tell. Self Test does not change or replace the last time calibration results.

Setup/**Testing Log** record the most recent activities happens to the measurement. A sample of testing log is show below.



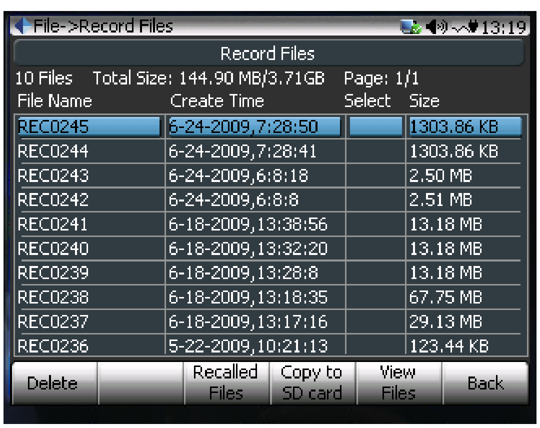
- Figure 43. Test Log View.

Setup/**Run VDC** icon will allow you to switch to the vibration data collector mode. If your CoCo is not installed with VDC mode, this icon will not be effective.

File Button 

The File hard button displays the Storage Capacity status including capacity, used space and free space. Also the number of files is listed including CSA files and measurement data files.

File/**Files** displays a list of all the record files including the name, created time, test note and size.



- Figure 44. Record Files screen shows the names and other attributes of files stored on the CoCo-80.

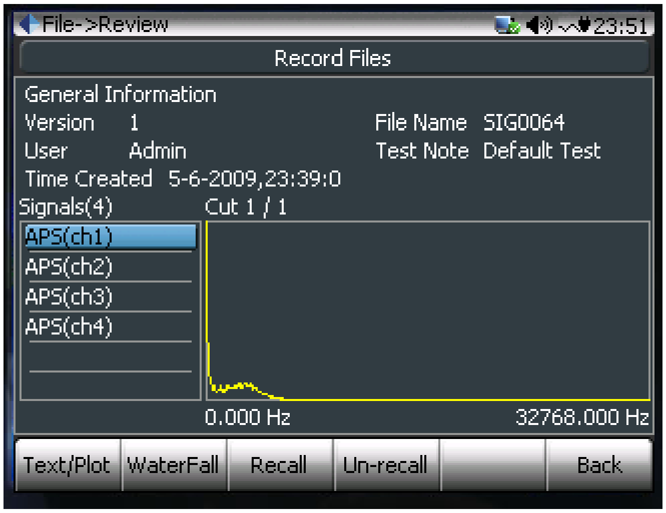
Delete->Delete Newest allows you to delete the newest a few files including and after the file that is being highlighted from the flash memory. For example if there are 4 files in order in the list, File4, File3, File2 and File1 where File4 the latest and File2 is being highlighted, when Delete Newest is pressed, File4, File3 and File2 will be removed.

Delete->Delete All allows you to delete all files for the flash memory at once.

Copy to SD Cards allows the user copying the signals or recording files into the SD memory card.

Recalled Files allows the user viewing all the recalled signal files. Recall is a useful operation to review the data files in a signal display trace.

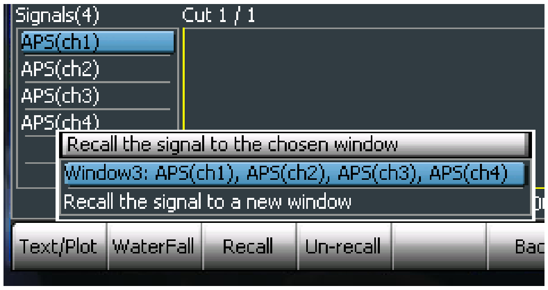
View Files shows a summary of the data file. The Text/Plot soft button allows you to display a text view including type, points, sample rate and units or change to a low resolution of the time record.



■ Figure 45. View the file content per signal.

Once a signal file is opened, the user can show it in text mode or plot mode. this is for the purpose of simple preview. To view the signals in the display plot trace, please use the recall function.

To recall a signal, press the F3 recall button:



The CoCo will allow the user to recall a signal into an existing trace, or a new trace. Recalling into existing trace will overplot this signal with the all existing ones with the same type.

Rec./Stop Button

The CoCo-80 combines two traditional instruments, a data recorder and a signal analyzer into one hardware platform. Due to the requirement of different use cases, the user interface design is a bit unconventional. Two hard buttons, Rec./Stop and Save are designed for storage; two soft buttons, Restart and Run/Hold are designed for controlling the data processing flow.

For details please refer to later Chapter about saving data.

Save Button

The Save Button is used to save block signals to memory. These include transient time signals and spectra signals, which unlike time stream data are a snapshot of a single block of data at the time the Save button is pressed. This can be used to capture averaged spectra or transient events at any time. You can choose which signals are saved when you press the Save Button by selecting **Signal Snapshot Save Setup** under the Param. Button.

When the CoCo-80 is connected to a PC the saved block signals can be downloaded using the EDM software.

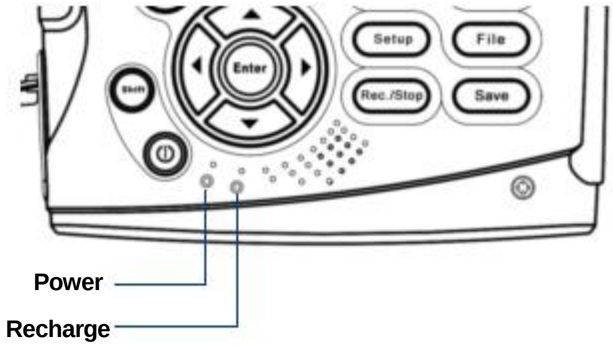
CoCo-80 Startup and Shutdown

This section describes power on and off, lock the keypad and how to reset the CoCo-80.

Power on and off the CoCo-80

The power button is located at the lower-left corner on the keypad. The very first time the CoCo-80 is used, it is necessary to set the clock time. All the data acquired and stored will include the clock time as a file attribute with a clock time accuracy of seconds.

There are two LEDs on the front panel. The one on the left close to the power button is an indicator for the system *on* or *off*. When the system is turned on, it will be lit red. The LED on the right is the indicator for external power charging. When the CoCo-80 is being charged, it will be lit in red. When the system is fully charged and still connected to the external DC power, it will be lit in green.



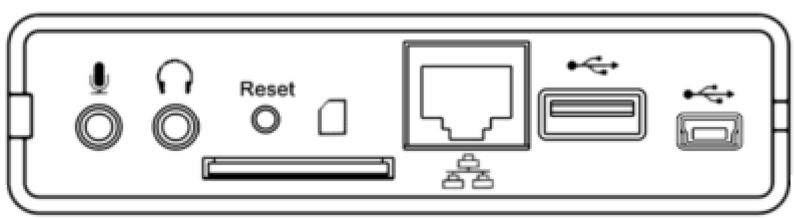
■ Figure 46. Two LEDs showing power and recharge status.

System Reset

In the rare event of a system lock up the power on/off button may not respond. To restore the unit you can reset the system in one of two ways.

Reset the system by Pushing the Reset Pin

You can reset the system by inserting a pin or paper clip through the reset hole. The Reset pin hole is shown below.



■ Figure 47. Reset pin hole can be used to shutdown the CoCo-80.

Reset the system using the Power Button

You can reset the system by pressing the power button for more than 4 seconds which will force the system to shut down. After the system is shut down, it can be rebooted by pressing the power button again.

CoCo-80 Software Disaster Recovery through EDM

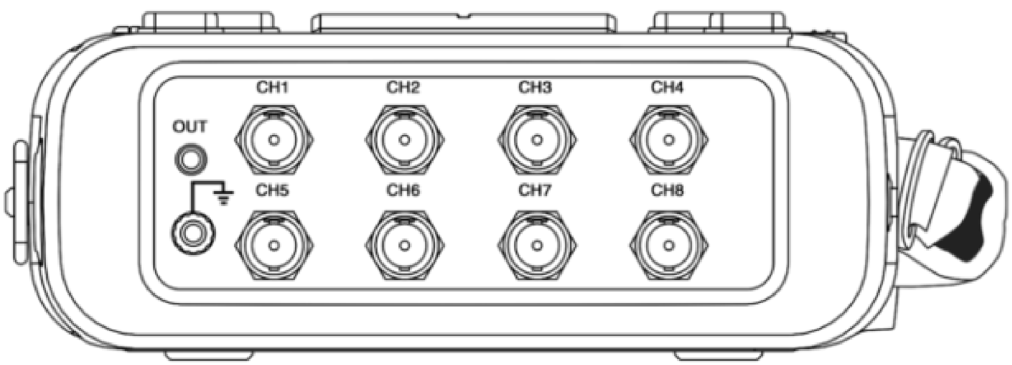
In the case that the CoCo-80 application software programs are completely corrupted due to an unknown reason, you can also use EDM, the host software, to restore the CoCo-80 back to its original state when CoCo-80 is connected to the host via USB.

Keypad Lock

To avoid accidental mistaken operation to the keypad, it can be locked by pressing the power button and then pressing the Lock the Keypad Soft Button. To do this, simply press the Power button and make the second selection.

CoCo-80 Input Connections

This section describes the CoCo-80 input connections and the related circuit design including a description of single ended versus double ended AC versus DC coupling and IEPE.



■ Figure 48. BNC input connectors, output and ground connector.

System Calibration

The CoCo-80 loads factory calibration data during start-up, eliminating the need for daily calibration checks. Although the CoCo-80 does not require daily field calibration, CI recommends an annual calibration and performance verification by local CI service centers.

To execute the System Calibration, first press the Setup hard button, then select System Calibration icon and press the Enter button.

DC-Differential

DC-Differential allows measurement of signals with a non-zero mean, DC component and uses differential input mode. Non-zero mean signals are typically low frequency signals or signals that are measured relative to ground. Differential mode is recommended when measuring signals with a common mode voltage (CMV). CMV is an in-phase signal that appears simultaneously on both input terminals of an input channel. Provided the sum of the signal and the CMV do not saturate the input and cause clipping, the measurement will be accurate. If the signal and CMV exceed the input range then the signal will be clipped and produce erroneous results. If the signal and CMV are very high and exceeds the maximum over-voltage rating of the instrument front end then the data will be erroneous and the hardware can be damaged. This must be avoided to protect the hardware from permanent damage.

DC-Single End

DC-Single End allows measurement of signals with a non-zero mean, DC component and uses single ended input mode. Single ended mode is recommended for most cases and when no CMV exists. This is the case when measuring the output of sensor amplifiers. A CMV will produce noise in single ended mode.

AC-Differential

AC-Differential applies a low frequency high-pass filter to the input filtering the DC component of the signal. The result is a zero mean signal. This is most commonly used for dynamic signals with CMV.

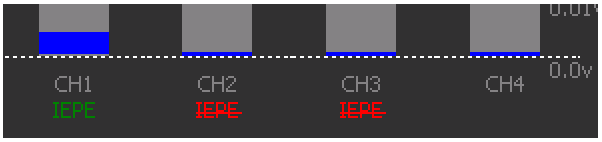
AC-Single End

AC-Single End mode combines the AC filter with single ended mode. This is most commonly used for dynamic signals with no CMV such as measuring the output of an amplifier.

IEPE (ICP)

The CoCo-80 supports IEPE constant current output type for its input channels. The built-in circuit is powered by a 4mA constant current source at roughly 21 Volts. IEPE refers to a type of transducer that is packaged with a built-in current source. IEPE is an acronym for ***Integral Electronic Piezoelectric***. IEPE requires an AC filter so DC measurements are not possible when IEPE is enabled. CoCo has a cut-off frequency of 0.3Hz@-3dB for the IEPE input mode.

CoCo can automatically detect the IEPE sensor connection when the IEPE input mode is enabled. The sensor indication is shown in following three modes:



A green IEPE sign indicates that the IEPE is set in the channel table and the IEPE sensor is detected; a red IEPE sign indicates that the IEPE mode is set in the channel table but the sensor has not been detected. This is a faulty mode; the empty space means that this channel is not set to IEPE.

CoCo-80 Output Connections

The CoCo-80 includes one output channel that can act as a function generator and provides a variety of waveforms synchronized with the input channel sampling rate. The output channel is a 0.3 mm stereo jack. A stereo jack to BNC adaptor is provided with the unit. For each waveform the parameters such as amplitude and frequency can be specified with the Output Parameters screen from the Display screen and the Param. soft button. The output waveforms include: None, Sine, Triangle, Square, White Noise, DC, Chirp, and Swept Sine.

CoCo-80 Peripherals and Accessories

This section describes the peripherals and accessories available on the CoCo-80 including SD Card, audio devices, Ethernet, USB, audio and battery. The CoCo-80 includes interfaces to many peripheral devices. These can be connected to the hardware via the connectors shown below.

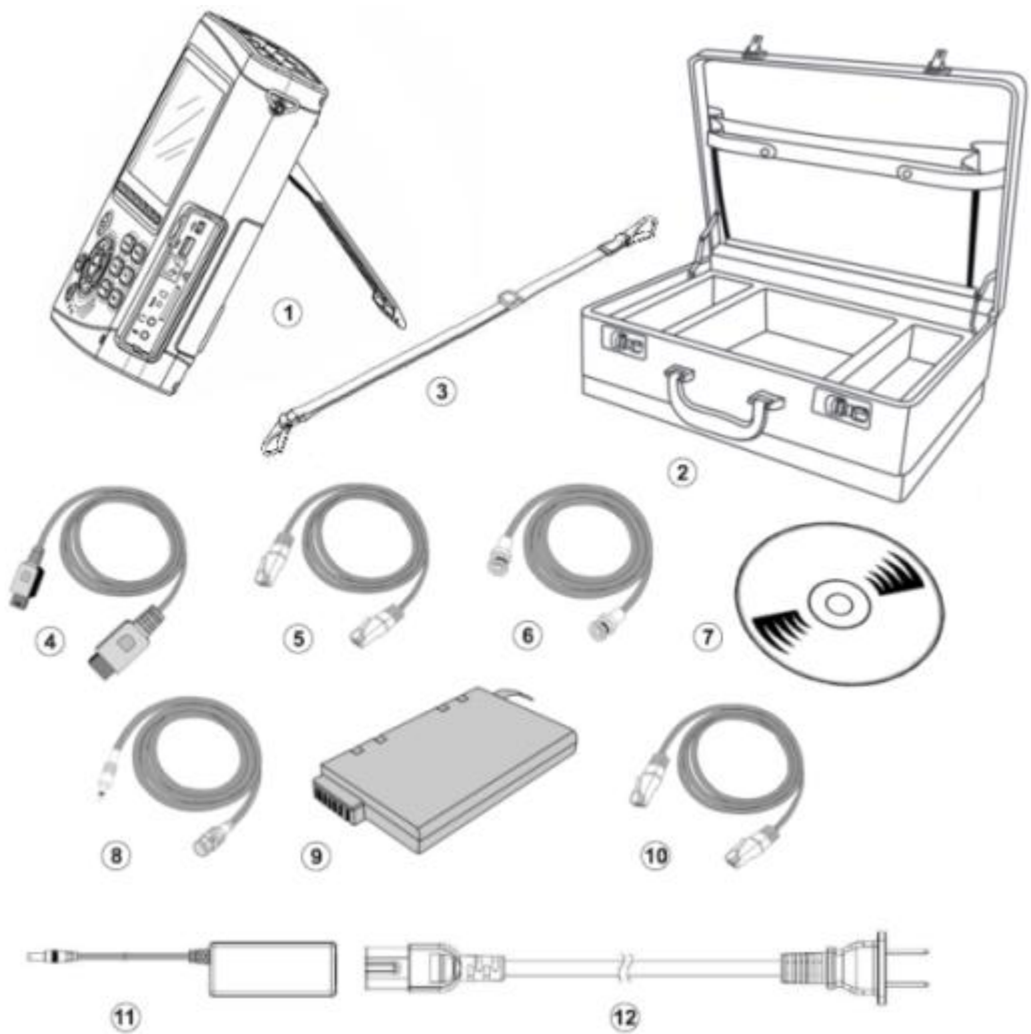


Figure 49. CoCo-80 Peripherals and Accessories.

Item Description:

#	Description
1	CoCo-80 Handheld Data Acquisition System
2	Suitcase with foam inside
3	Hang Strap
4	USB cable
5	Regular Ethernet cable
6	BNC cable
7	CD for EDM, the host software, User's Manual in PDF
8	Cable for Output (Signal Source)
9	Main Battery (installed)
10	Cross-Over Ethernet Cable
11	AC/DC Power Adapter
12	Power Cable to AC Outlet

Figure 50. CoCo-80 peripheral connections.

Ethernet

CoCo is equipped with an RJ 45 - 100 BaseT Ethernet jack to connect to a local area network or directly to a PC. A cross-over Ethernet cable must be used to connect the CoCo-80 to a PC **directly**. If CoCo is connected to a network hub, router or a switch, then a regular Ethernet cable (not a crossover cable) should be used.

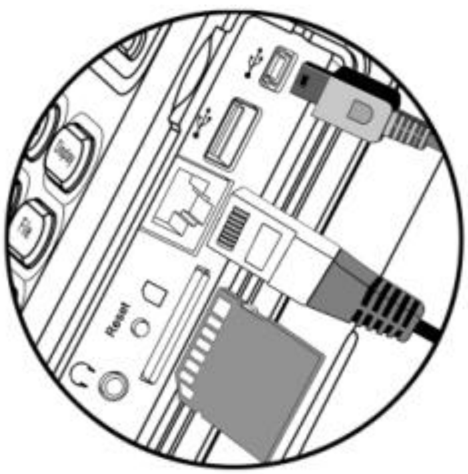
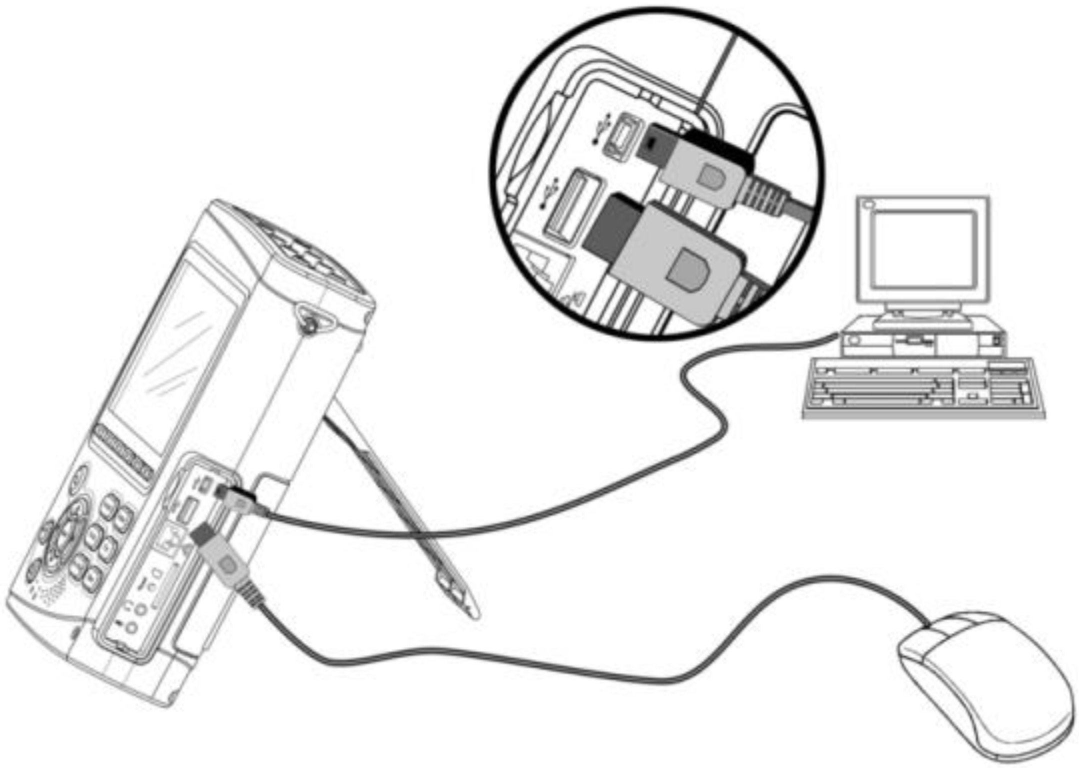


Figure 51. Ethernet connection.

USB Ports

The CoCo-80 has two USB ports, one USB-client (mini-USB) and one USB-host (type A). They are fully compliant with USB 2.0 full speed specification and backward compatible with USB 1.1. The shapes of two ports are different, as shown below:



■ Figure 52. CoCo-80 has two USB ports: client for PC connection and host for peripheral connection.

The USB-client port is used to establish communication between the CoCo-80 and a PC. When the USB-client port is used, CoCo-80 device acts as a slave unit.

The USB-host port is used to establish communication between the CoCo-80 and other USB-based peripherals, such as a USB-mouse, or a USB memory stick. In this case, the CoCo-80 acts as a USB master device.

Mouse Support

USB Mouse is supported with following operations: F1~F6 function buttons, two virtual keypads, scrolling and make selections in any combo box, ZOOM-in scaling, ZOOM-out scaling the graph.

To ZOOM-in the graph, hold the left button of the mouse and drag to the area that you intent, then release the left button.

To ZOOM-out the graph to the previous scaling stage, double-click on the graph.

SD Card Interface

The MMC/SD-Card interface is designed to be used for multiple purposes, mainly the high density memory card. The official information about the MMC/SD-card can be found on the official site: http://www.sdcard.org/

The user can copy the recorded signal files from the internal flash memory to SD memory card or directly record the time stream data to SD memory card.

Audio Devices

CoCo-80 has the following built-in audio devices:

- 3.5mm stereo jack connector for an earphone
- Built-in speaker
- Built-in microphone

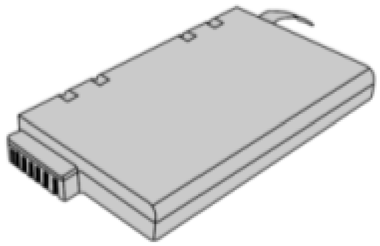
The earphone and speaker are used to generate status sounds that provide audio feedback to the use such as:

- AC adapter is connected
- AC adapter power was disconnected
- System boot-up successful
- System boot-up failure

Battery

There are two batteries inside the CoCo-80 device, the clock battery and the main battery. The clock battery is only used maintain power to the internal clock. It is located inside the hardware and should be replaced when necessary by an authorized CI service center and should not be replaced by the user. The main battery is used to power the instrument. The main battery is a Lithium-Ion type cell with a capacity of up to 6600 milliamp-hours. The main battery is located inside the enclosure and can be replaced by opening the lid on the back of the CoCo-80.

To recharge the main battery, simply connect the AC adaptor between the CoCo-80 and the AC power source. The power source must be in the range of 100 - 250 VAC. When the CoCo-80 is turned on, a battery capacity symbol is shown on the status bar that indicates the state of charge of the battery.



- Figure 53. CoCo Battery.

Battery Charger

This is an optional accessory. This charger can charge the main battery without using CoCo. It is convenient to use this charger to charge an extra main battery while one is in use. This charger is designed and made by Crystal Instruments.

DC/DC Converter for Car Cigarette

This is a DC-DC adapter using automobile cigarette lighter, voltage isolated. Input: 9~30VDC. Output 15V/3A (±10%).

With this converter, the user can use the power from the car cigarette adapter to support the CoCo.

CoCo-80 On-Line Updates

The CoCo-80 application software has the capability to check for software updates from the CI web server when you connect the CoCo-80 device to the Internet. You first connect the CoCo-80 to a local network using regular Ethernet. After you connect it, press Setup button and click the Update icon. The CoCo-80 will first check the connection status, and then a connection will be established.

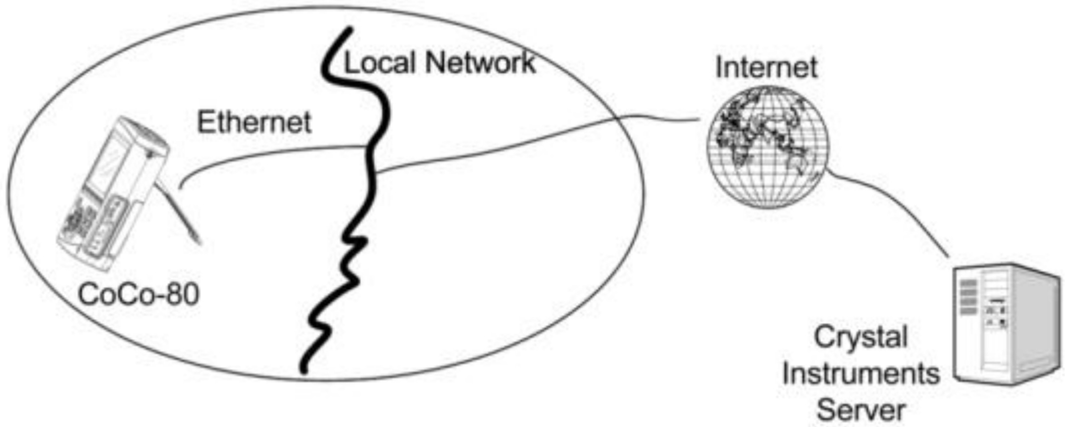


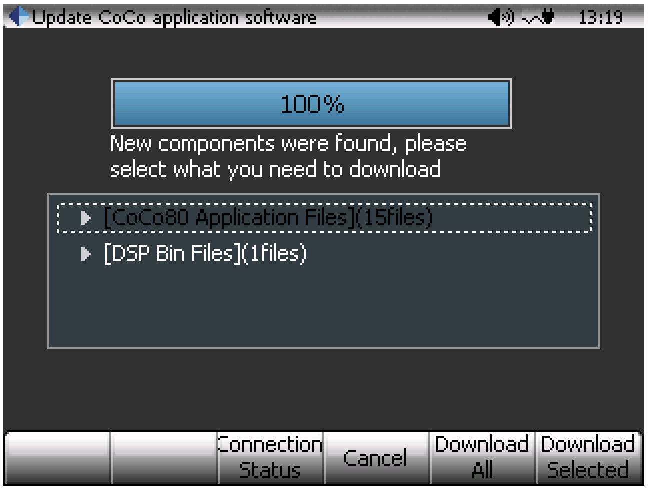
Figure 54. Network connection for CoCo-80 update.

After communication is established, the CoCo-80 will check with the server to verify if the software subscription is valid. If the CoCo-80 is in the valid software subscription period, it will then check the latest software components available on the server and download them to the CoCo-80 after the user's approval.

Two types of software components can be updated:

- CoCo-80 application software
- CSA projects

The CoCo-80 user interface will always ask the user's confirmation before the software is downloaded. When the new CoCo-80 application software is downloaded, you will be asked to confirm to overwrite the old version with the new version. Then the older version will be overwritten.



■ Figure 55. On-line update detection status screen.

When new CSA projects are downloaded, if the new CSA files take the same file name as the old ones, the old CSA files will be renamed to the CSA files with sequence number added. This approach will prevent overwriting the old CSA files that may have been changed by the user.

If the connection to the Internet could not be established, please press the Setup button and click on the Connections icon. This will lead you to the Ethernet network setup. The most common problem is caused by inappropriate IP address setting. Most often, your LAN requires you set up the IP as "Dynamically obtain an IP via DHCP". Please refer to section "Configuring the CoCo-80 Network Settings" in this manual for more details.

Advanced Audio Functions

CoCo-2, CoCo-80 and CoCo-90 are all equipped with advanced audio functions. These audio functions allow you to listen to the vibration or any measurement quantity or record voice annotations during signal recording. This document describes how to use the audio functions.

The advanced audio features can be summarized as following:

- You can listen to any measurement input using headphones without interrupting the measurement or recording process.
- The audio monitoring is automatically scaled to the listening range and the headphone audio can be manually adjusted.
- You can record voice annotations at any time and length during time stream recording.
- A customized microphone is available with a push button to control voice annotation recording.
- Voice annotations can be replayed on the CoCo hardware through headphones.
- Voice annotations are attached to each recorded file, and can be played back on the PC using the EDM software
- CoCo can play back any recorded time streams using its output port. The output port can drive another audio device such as headphones or external speakers.

These advanced audio functions require the following minimum hardware and software versions: CoCo Software Version ≥ 1.7.8; Base Hardware System Version ≥2.0.9; Measurement Hardware Version ≥ 10.1.0; Firmware Version ≥ 1.5.0.

Hardware Audio Peripherals

Three hardware audio peripherals are used for the advanced audio functions:

1. Internal Speaker
2. External Headphone
3. External Microphone

The internal speaker is used to generate system-related signals, such as the sound simulating the key press, power-on/off or alarm. Voice annotations and measurement input audio can only be played back through headphones and not through the internal speaker.

- Figure 56 Built-in Speaker

The external headphone jack uses the 3.5mm stereo jack connector. You can connect any headphone to this connector.

- Figure 57 An example of headphone

The headphone jack is located at the second to the left with a headphone symbol. Voice annotations and measured input audio can be played back through the headphones.

■ Figure 58 Connectors

The external microphone must be ordered from CI. It is designed so that when the microphone button is pushed, the voice annotation recording is activated. The microphone jack connector is on the left side of the peripheral panel. Do not use any microphone other than the specified CI microphone because without the microphone button hardware, you will not be able to start a voice annotation recording.

■ Figure 59 Microphone with push button (part # CoCo-A12)

Audio Functions

The audio functions are controlled through the CoCo, Setup->Audio Setup screen.

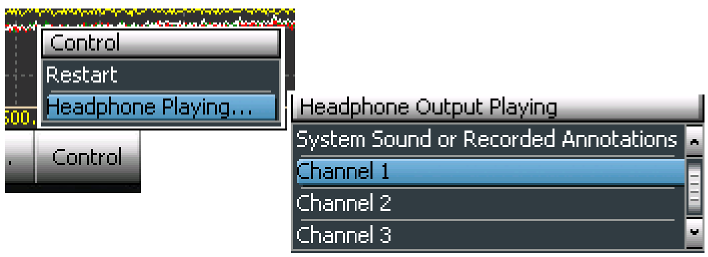
■ Figure 60 Audio Setting page

Keypad Sound: Enable and select the internal speaker sound output when any of the buttons are pressed.

<u>Power Button Sound</u>: Enable and select the internal speaker sound output when the power button is pressed.
<u>Alarm Sound</u>: Enable and select the internal speaker sound output for system alarms.
<u>Use microphone to record the voice annotation</u>: Enable the external microphone recording function. When this item is checked and the user presses the connected microphone button, the voice annotation is recorded until the button is released. Multiple annotations can be recorded during a measurement. If this item is not checked, the microphone button will not activate any voice recording.
<u>Use headphone to listen to any input channel</u>: Enable the external headphone listening function.

Headphone Listening

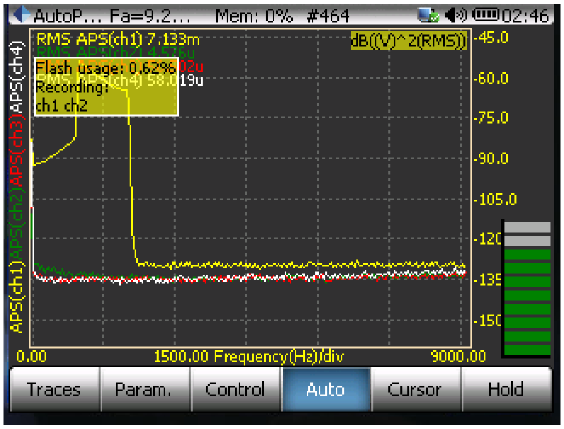
When "<u>Use headphone to listen to any input channel</u>" is enabled, under the F3 Control Button of the signal display screen, you will see the Headphone Playing menu item. Select one of the input channels. If you do not want to listen to the input channels, then set the selection to "System Sound or Recorded Annotations."



- Figure 61 Select the Channel for Headphone Listening

Record Voice Annotations

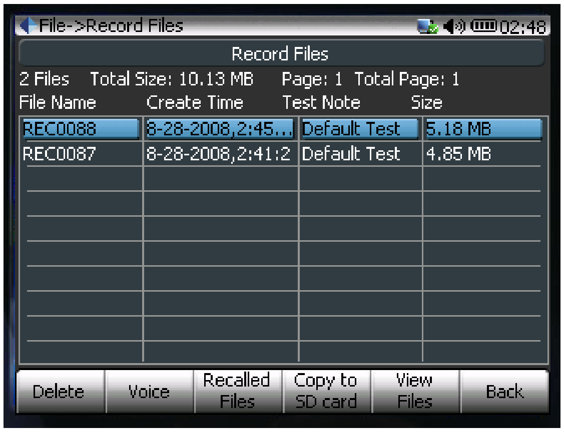
After "<u>Use microphone to record the voice annotation</u>" is checked in the audio setup, connect the external microphone (Part #CoCo-A12) to the microphone jack. Press the Rec/Stop button to record the time signals. While the time signals are being recorded, you can press the microphone button to record your voice annotation. The voice annotations will be attached to the recorded time streams. The green bar on the right bottom corner on the screen indicates the volume of the signal received by the microphone.



■ Figure 62 Monitor the volume of the microphone input

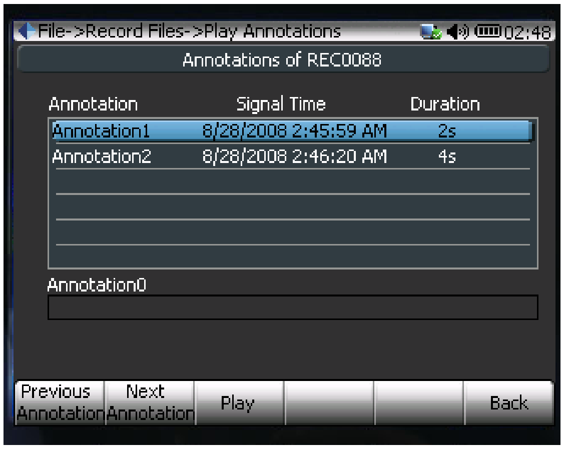
Playback the Voice Annotations on CoCo

To play back the voice annotation, first press the File button, then the F1 Files, then the F2 Voice button.



■ Figure 63. Play back voice annotations from the File View.

The F3 Play button allows you to hear the previously recorded voice annotation. Then you can use the F1 Previous Annotation or F2 Next Annotation Buttons to play all the annotations. If the Voice button is not shown, it means the signal file saved has no voice annotation attached.



■ Figure 64. Play all annotations using Next and Previous buttons.

Voice annotations will be listed under each recorded or saved signal files and can be played back with EDM PC software.

Playback the recorded signals from output channel

The CoCo can playback any of the recorded signals from its output channel. To do so, open a recorded file and press F5 the Playback button.

If the user made voice annotations while the time stream is recorded, the voice annotations will be played back as well.

After the signal playback is finished, the screen will show a message of ***Waveform playback finished.***

Notice that the recorded signal will be played back at the sampling speed of when it was acquired. Inside CoCo, the A/D converters and D/A converter share the same sampling clock. Due to this design, when the signal is played back, input signal cannot be analyzed.

CSA — Configurable Signal Analysis

This section describes the Configurable Signal Analysis concept that is the basis for the CoCo-80 functionality and allows advanced users to customize the analysis features to suit individual needs. This section gives a brief description that is intended for the basic user. It does not describe writing projects for advanced users. For more on writing CoCo-80 analysis functions refer to the manual about CSA Editor that comes with EDM host software.

When the CoCo-80 powers up the Welcome screen is shown. From this screen the use must select one of the CSA projects loaded on the CoCo-80. When a CSA project is selected, the project defines the settings and analysis functions that are computed by the CoCo-80. These settings include the following:

- Parameters used by the data conditioning functions such as Add, Subtract, Multiply, Divide, Square, Square Root, RMS, Scale, Offset, Decimate
- Parameters used by the signal analyzer functions such as FFT, Auto Power Spec, Coherence, FRF
- Time Stream Data Recording Settings
- Block Data Save Settings
- Trace Settings

The CSA is designed to control how the data is processed, not how the data is acquired. When the CSA is changed, the processing functions are changed according to the new CSA, but the data acquisition parameters do not change. For this reason, the following settings have global effect and are not part of CSA project:

- Sampling Rate
- Input Channels: sensitivity, coupling, channel labels
- Output Channel: output waveform settings

All pre-programmed CSA projects have predefined parameters that are loaded when the project is selected. You can modify the parameters on the CoCo-80 from the ***Param*** Soft Button in the Display screen. Modified CSA projects can be saved with a different name using the Save As soft Button in the Analysis screen so that the original projects are not overwritten.

Most pre-programmed CSA projects carry a variable called *Maximum Sampling Rate*. This is the sampling rate that this CSA can safely execute without exceeding its computational resource limit. Maximum Sampling Rate is used to limit the selection of the sampling rates.

Preprogrammed CSA projects

The CoCo-80 is preprogrammed with a set of default CSA projects which provide a wide range of options that meet most users' needs. Additional CSA projects may be downloaded from the Crystal Instruments web site. In addition, by using CSA Editor the advanced users may edit or develop their own customized CSA projects to meet their specialized needs. Typical default CSA projects are given below.

- Table 1. Preprogrammed CSA project Descriptions. Yellow-highlighted are most often used

CSA Group	CSAs used for 4-channel CoCo-80	CSAs used for 8-channel CoCo-80	Description
Data Conditioning	RMS(4).csa	RMS(8).csa	Calculate the RMS of each input channel. Overlap ratio and average time are changeable.
	Time(4).csa	Time(8).csa	Apply no data conditioning for any channel. Only show the native input channels.
	PkPk(4).csa	PkPk(8).csa	Calculate the peak-to-peak value for each channel.
	Subtract(4).csa	Subtract(8).csa	Channel 1 is subtracted from each other channel.
	Multiply(4).csa	Multiply(8).csa	Multiply each channel by channel 1.
	Add(4).csa	Add(8).csa	Add channel 1 to other channels.
	Integration(4).csa	Integration(8).csa	Digitally integrate each channel. Suitable for signals with higher frequency content.
	IntegrationLow(4).csa	IntegrationLow(8).csa	Digitally integrate each channel. Suitable for signals with low frequency content.
	OffsetScale(4).csa	OffsetScale(8).csa	Apply an offset and a multiplier to each channel. The offset and multiplier can be modified in Analysis Parameters.
Transient Capture	Transient(4).csa	Transient(8).csa	Time streams from each channel are captured into block signals by enabling Acquisition Mode.
	Capture1Ch(4).csa	Capture1Ch(8).csa	Block-capture the channel 1 with up to 64k buffer size
Linear and Power Spectra	AutoPowerSpec(4).csa	AutoPowerSpec(8).csa	Transform the time streams into block signals then apply data window and FFT to calculate auto power spectra. FRF/Coh will not be calculated.
	APS1Ch(4).csa	APS1Ch(8).csa	Only calculate the auto spectrum for 1 channel with up to 64K buffer size
	FFT(4).csa	FFT(8).csa	Transform the time streams into block signals then apply data window and FFT.

Frequency Response	FRF(4).csa	FRF(8).csa	Calculate the frequency responses when channel 1 set as reference and the others as responses.
	CrossPowerSpec(4).csa	CrossPowerSpec(8).csa	Calculate the cross power spectra when channel 1 set as reference and the others as responses.
	FRF_COH(4).csa	FRF_COH(8).csa	Calculate the FRF and coherence when channel 1 set as reference and the others as responses.
	FRF_COH_CPS_APS(4).csa	FRF_COH_CPS_APS(5).csa	Calculate the FRF, cross-power, auto-spectra and coherence functions when channel 1 set as reference and the others as responses.
Real-time Digital Filter	DecimFltr(4).csa	DecimFltr(6).csa	Apply n stage of 2:1 decimation to each input channel to obtain the signals with lower sample rate.
	FIR(4).csa	FIR(8).csa	Apply FIR Low Pass, High Pass, Band Pass, and Band Stop filters with filter length 67.
	RemezFltr(4).csa	RemezFltr(8).csa	Apply Remez FIR Low Pass, High Pass, Band Pass, and Band Stop filters with filter length 67.
	IIRFltr(4).csa	IIRFltr(8).csa	Apply IIR Butterworth Low Pass, Butterworth High Pass, Chebyshev Band Pass, and Elliptic Band Stop filters with filter order 7.
Acoustic Analysis	OCT(4).csa	OCT(8).csa	Apply 1/1, 1/3, 1/6 or 1/12 octave filters to time streams and generate octave spectra and filter r.m.s. time traces. It conforms to ANSI std. S1.11:2004 and IEC 61260-1995.
	OCT RPM(4).csa	OCT RPM(8).csa	Apply 1/1, 1/3, 1/6 or 1/12 octave filters to time streams and generate octave spectra and filter RPM traces. It conforms to ANSI std. S1.11:2004 and IEC 61260-1995.
	SLM(4).csa	SLM(8).csa	Sound Level Meter (SLM) template provides various overall sound level readings, time and frequency weighting according to IEC 61672-2002.
	SLM RPM(4).csa	SLM RPM(8).csa	Sound Level Meter (SLM) template provides various overall sound level readings, time and frequency weighting according to IEC 61672-2002.
Order Tracking	ORDTRK(4).csa	ORDTRK(8).csa	Normalized Order Spectra and Order Tracks
	ORDTRK(4)_CFS.csa	ORDTRK(8)_CFS.csa	Constant Frequency Bands
	ORDTRK(4)_Phase.csa	ORDTRK(8)_Phase.csa	Order Tracks with Phase

LimitTest	TimeLimiting(4).csa	TimeLimiting(8).csa	Apply limit test to the time stream signal of each channel. Limits are edited in CSA Editor.
	APSLimiting(4).csa	APSLimiting(8).csa	Apply limit test to the auto power spectrum of each channel. Limits are edited in CSA Editor.
	Transient_Limit(4).csa	Transient_Limit(8).csa	Apply the limit test to transient captured signals. Limits are edited in CSA Editor.
My CSA	Histogram(4).csa	Histogram(8).csa	View Histogram and statistical analysis of each channel.

When a CSA project is running you can choose to display, record or save data streams or signals.

Change CSA projects from the CoCo-80

A CSA project specifies the analysis settings and functions including: analysis parameters and functions, time stream recording, block data save and trace settings. After a CSA is selected it can be modified from the CoCo-80 to change these parameters using the ***Traces*** and ***Param*** soft buttons in the Display screen. The modified CSA can then be saved using the Save As soft button in the Analysis screen. This allows modified CSAs to be saved and used again later.

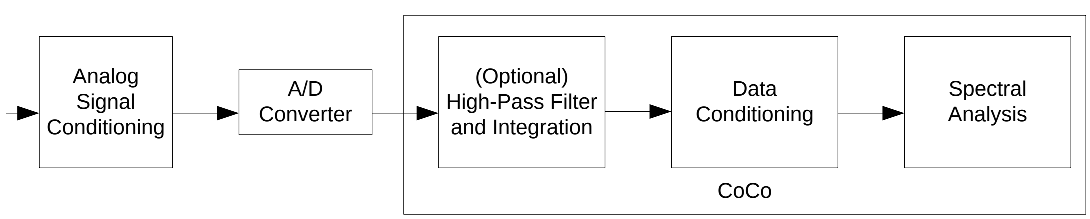
Editing CSA from the EDM Software

CSA files can also be edited or new CSA files can be created from scratch from the EDM software on a PC. This feature allows the advanced end user to create custom analysis functions to suit their special needs. This advanced topic is not covered in this manual. Refer to the CI Support web page for documentation about editing CSA from the EDM software.

To initiate the CSA Editor, click on the CSA Editor icon on the upper-left corner of EDM.

The Analog Signal Conditioning and Data Pre-Conditioning

In a global picture, the signals will go through the following stages in an analyzer:



First the analog signals will be processed by the analog signal conditioning circuitry. It usually includes an input mode selection logic, the high-pass filter for AC coupling and constant current source for IEPE. Then the analog signals will be digitized simultaneously by multiple A/D converters. The digital signals coming out of A/D converters, after a unique calibration process, will be fed into the DSP chip inside of the CoCo box.

The CoCo may first turn on a pre-data conditioning algorithm. This pre-data conditioning algorithm may apply the high-pass filters so to reduce the DC drift, or convert the acceleration signals into velocity or displacement. It will also apply the appropriate engineering unit setting to the input signals.

Then the data streams will be fed into a user controlled Data Conditioning module, and then spectral analysis. This part will be explained in detail in the next section.

The Data Processing Flow of CoCo

CoCo-80 combines two instruments, a data recorder, and a signal analyzer into one system. It is important to understand the differences between these two functions. The following sections provide details of each.

The data conditioning and recording phase includes processing the data from native acquisition channels and data conditioning. Data conditioning operations include +-*/, filtering, integration, differentiation, calibration and other math operations that can be applied to the continuous time streams. All the signals in the data conditioning and recording stage are continuous time streams with a fixed sampling rate and do not include any gaps. Time streams can be displayed or recorded.

The signal analyzer phase includes Acquisition Mode, and CSA based block-by-block processing. The acquisition mode controls how the continuous time streams are captured in fixed block size records. The processing phase applies algorithms such as spectral analysis to the block-by-block signals. Figure 65 shows how the input data is processed in the data conditioning and signal analyzer phases.

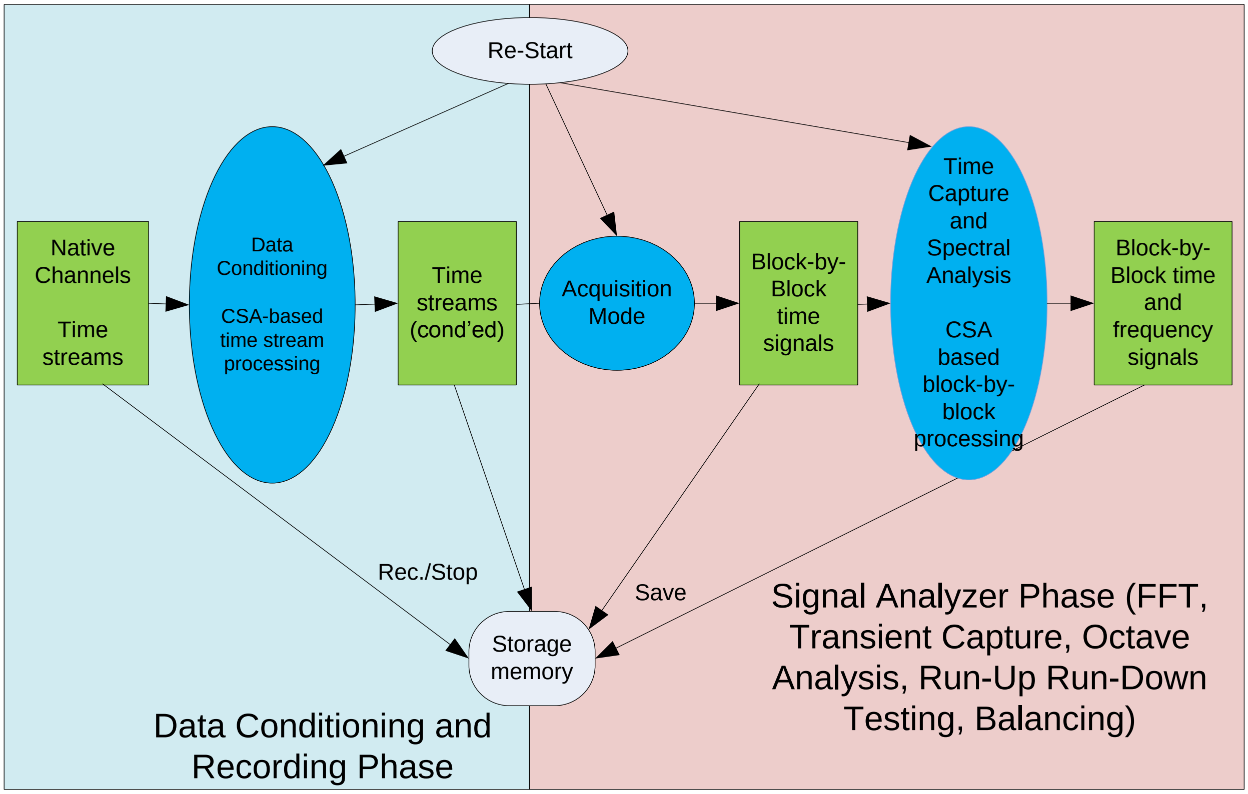


Figure 65. Data Process Flow Diagram.

The system has two dedicated buttons that control the data acquisition.

Rec./Stop button only applies to the continuous time streams in the data conditioning phase while the ***Save*** button only applies to the captured time buffer and spectral signals in the signal analyzer phase. Recording and Save cannot be performed at the same time. In the other words, when the time streams is being recorded, the spectral analysis cannot be saved. To save a spectral analysis signal the time recording must first be stopped.

Re-Start soft button re-initializes the data conditioning, acquisition mode control and spectral analysis. It resets the timer of time streams, re-arm the trigger and reset the average number of spectral analysis.

Block Size governs the size of transient capture or FFT processing in the signal analyzer phase. Block Size has no influence on the length of the time streams in the data conditioning phase.

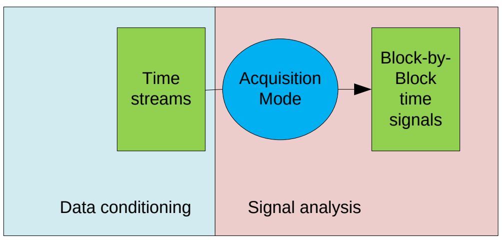
Both time streams and block-by-block signals can be displayed with continuous update (Run mode) or frozen on the screen (Hold mode). In the signal display window the F6 software button is assigned to the ***RUN/Hold*** function. *Hold* means the display and spectral analysis are frozen on the screen. **Hold** does not stop the data conditioning and recording process. If data is being recorded and the Hold button is pressed, the data will continue to be recorded until the Rec/Stop button is pressed again.

The actual Data Conditioning and Signal Analysis function processing are defined by a special technology, CSA, as described in the following section.

Acquisition Mode

This section describes Acquisition Mode which acts as a trigger to capture block by block data from continuous time streams.

The instrument separates the data processing into three stages: data conditioning, acquisition mode and signal analysis. Acquisition Mode controls how the continuous time stream data is captured for block-by-block processing. Acquisition Mode control is applied after data conditioning and before the signal analysis stage. If a CSA does not include a block capture function then Acquisition Mode will not be used.



■ Figure 66. Data processing is separated into three stages.

Note: in the description below, sometimes when we say "capture a block of data", it really means that multiple blocks of data are captured from their own time streams. These blocks are all accurately time-synchronized.

Acquisition Mode

Acquisition Mode defines how the device responds when a trigger event is detected and includes the following options.

Free Run displays block data acquired from the time stream as fast as possible or at the overlap rate by the user. Free Run is commonly used to analyze the random or irregular signals.

Continuous after Trigger waits until a trigger event is detected. After the first trigger event, averaging is reset and the system runs in Free Run mode.

Single Shot with Trigger waits until a trigger event is detected. After the trigger event, the spectral analysis, if there is any, will reset its average back to 1, and waiting for the next trigger. This mode is the best if you want to observe the time signal block by block at a certain trigger event.

Single Shot without Trigger waits until you press the Run button, then acquires one block of data and returns to the wait condition. This mode is the best if you simply want to observe a time signal block by block at an arbitrary time.

Auto-Arm Trigger waits until a trigger event is detected, then a block is acquired and the system returns to the wait condition. If another trigger event is then detected the new block of data will be acquired. This processes will continue indefinitely with no user interaction. Averaging is reset after each trigger event. This mode is the best for repetitive data capture and spectral analysis if you have confidence in the signal quality. Caution should be used because a desired captured block of data can be replaced by a new undesired block of data if an addition trigger event occurs. Use Manual-Arm Trigger to ensure that the last block is not automatically replaced by a new triggered signal.

Manual-Arm Trigger waits until a trigger event is detected, then a block is acquired and the system changes to a hold condition. After the data capture, you will be prompted with an Accept/Reject dialog box. If you accept the new data capture then the block will be included into the average and the system will return to the wait mode. If you reject the new data capture, the new block will be discarded and the system will return to the wait for trigger mode. This mode is the best for applications such as impact hammer testing where you may not have confidence in the signal quality of some of the data blocks.

Trigger Source

Trigger Source defines what signal is used to determine a trigger event. Any time stream that is set as trigger source candidate in the CSA can be selected as trigger source on the CoCo-80. If a signal is not identified as a trigger source candidate in the CSA file then the signal will not appear on the list. This feature is designed to simplify the user interface and optimize the CoCo-80 computation resources.

The candidates of Acquisition Mode selection and Trigger Source selection will be defined by the CSA editor. The CSA editor will assign some the data streams after the data conditioning as candidates of trigger sources. For example, in a CSA there are 8 channels, if you only select ch1 and ch2 time streams as candidates of trigger source and then this CSA will only show ch1 and ch2 on the trigger source selection menu.

You may also define time streams other than native channels as trigger source candidate. For example, if in the CSA an RMS measurement is derived from ch1, this RMS time stream can be used as a trigger source.

Trigger Condition

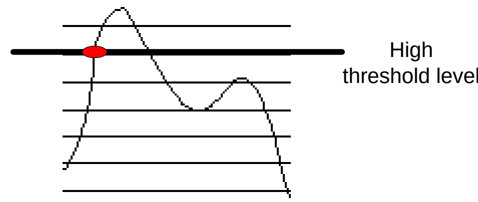
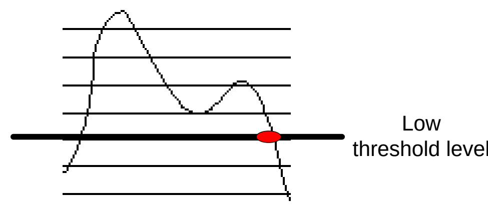
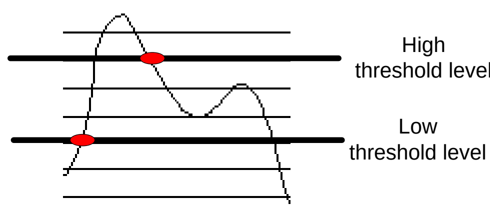
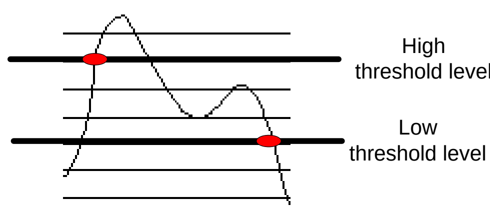
Trigger Condition defines when a trigger is detected based on the signal level and the slope. The four choices are:

1. ***Trigger Source > High Level (rising edge)***
2. ***Trigger Source < Low Level (falling edge)***
3. ***Low Level < Trigger Source < High Level (level trigger)***
4. ***(Trigger Source > High Level) OR (Trigger Source < Low Level) (edge trigger)***

There are two types of trigger detection, one is called edge detection; the second level detection. In the trigger conditions above, 1, 2 and 4 are edge detection and 3 the level detection. Edge detection compares at least two sample points against the threshold level. Level detection only detects one sample point.

When ***Free Run*** is selected, trigger source and level are not needed.

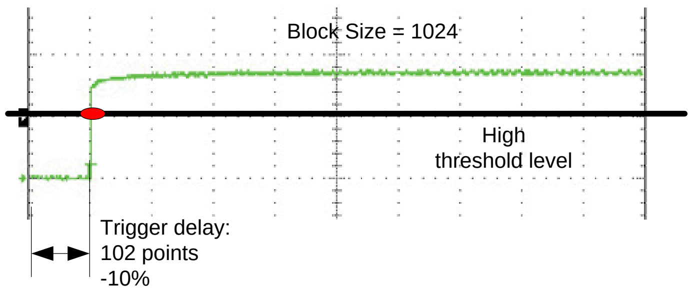
The table below visually explains when the trigger event will happen in these four conditions. The red mark shows the instant in time that the trigger event is detected:

Trigger Condition	Visual Explanation
Trigger Source > High Level (rising edge)	
Trigger Source < Low Level (falling edge)	
Low Level < Trigger Source < High Level (level trigger)	
(Trigger Source > High Level) OR (Trigger Source < Low Level) (edge trigger)	

Trigger Delay

Trigger delay allows a captured signal to include some data before or after the trigger event. This is done by defining some number of points, or the percentage of the total Block Size, that the capture occurs after the trigger event. For example, if the Block Size is set to 1024 and the trigger delay is 10%, the data capture will happen 102 points after the trigger event.

A negative trigger delay is more common for transient data capture. Negative trigger delay means that the data capture will include data points before the trigger event. For example, a -10% trigger delay means that the data capture will include 102 data points before the trigger event with Block Size 1024. Some instruments call a negative trigger delay a ***Pre-Trigger***. The following picture shows the concept of a negative trigger delay:



■ Figure 67. Pre-Trigger (negative delay) example.

Overlap

When overlap is enabled, then the data is averaged from data frames that are overlapping. This reduces the averaging time. Overlap is only used when the Acquisition Mode is set to ***Free Run*** or ***Continuous after Trigger.*** Otherwise it is not used. Continuous capture without further trigger can also use overlapping.

No Overlap – Overlap is not applied.
Automatic – System determines the best overlap rate
25% Overlap – Frames are overlapped by 25%.
50% Overlap – Frames are overlapped by 50%.
75**% Overlap** – Frames are overlapped by 75%.

Acquisition Mode Setup

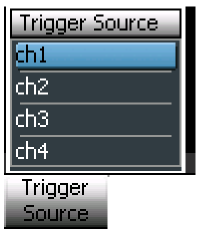
This section explains how to set up the acquisition mode and the trigger related parameters. First select the Acquisition Mode under the Param. Setting then the acquisition mode screen will be shown.

The soft buttons are assigned with different functions:

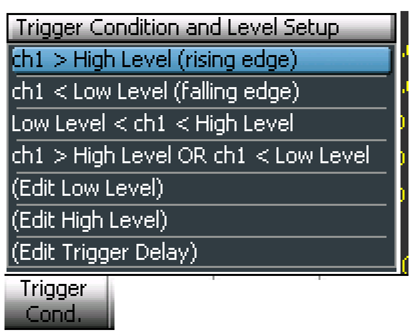
Press F1 to select one of the acquisition modes:

When the Acquisition Mode is not **Free Run**, the Trigger Source, Trigger Condition must be defined.

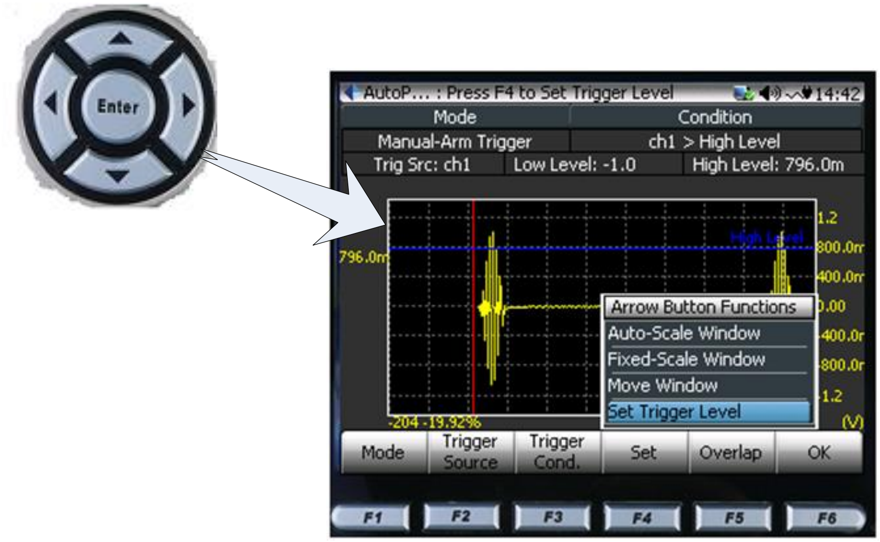
Press F2, Trigger Source, to select one of the time streams as the trigger source. These time streams are set as Trigger Source Candidates by the CSA Editor when this CSA project is created on the host PC.



Press F3, *Trigger Cond.*, to select one of the trigger conditions. You can also key in the trigger level(s) using Editing Level entry.



The arrow buttons can also be used to set the trigger level and delay settings.



Use the arrow buttons to change the trigger level and delay while the data stream from the trigger source is displayed.

Press F4 to activate one of four functions for the navigation arrow buttons

1. Auto Scaling Window

2. Fixed Scale Window: Arrow buttons used for expanding or reducing the scales

3. Moving window: Arrow buttons used to shift the positions of the window

4. Arrow buttons are used to set the high threshold level, low threshold level, and trigger delay

The method 3 is a more convenient way to set the trigger threshold level instead of using the editing tool under F3. The editing tool allows you to set the trigger level to a precise value.

Press F5 to set the overlap rate. This factor will only have effect when the acquisition mode is set as **Free Run** or **Continuous after Trigger**.

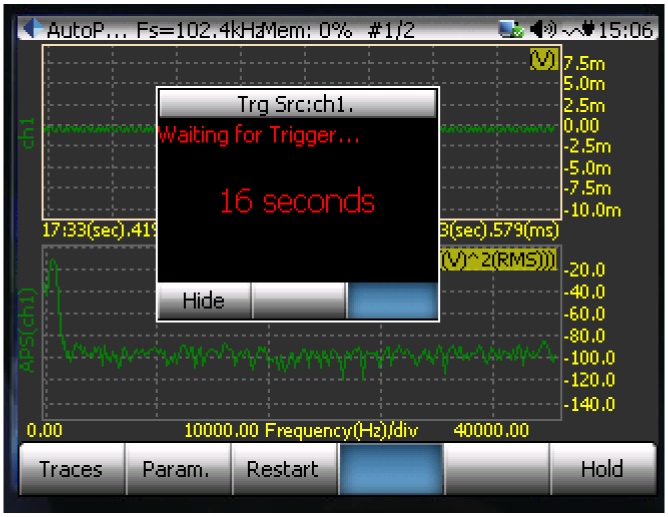
After all trigger parameters are set, press F6 (OK) button then the system will exit to the main measurement display window.

Using a Trigger during Measurement

This section explains the trigger operation while making measurements. Manual-Arm Triggering is the most common mode and will be described first and in the most detail. The other types will be explained briefly afterwards.

Manual Arm Trigger

When the Acquisition mode is setup then a small popup window is displayed as shown in Figure 68 indicating that the system is waiting for a trigger event. No signals are displayed until a trigger event is detected.

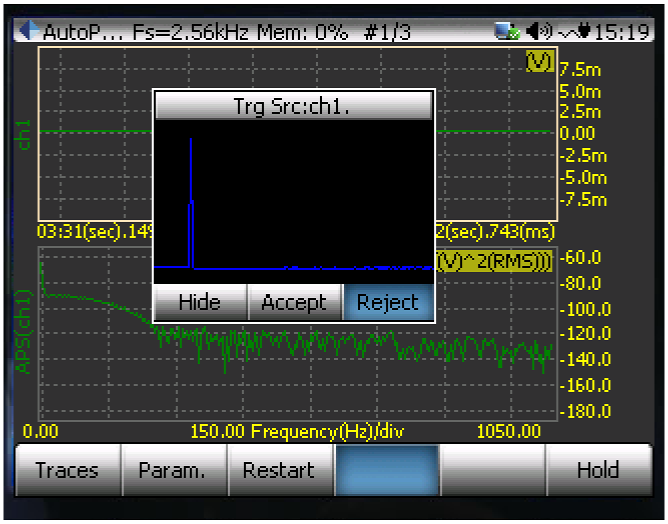


■ Figure 68. Waiting for trigger message.

You can change from the waiting for trigger mode to Hold mode by pressing the F6 (Hold) button. The window will close and the system will change to Hold mode.

Press Restart (F3) or Run (F6) to reopen the window and return to the waiting for trigger mode.

When a trigger event occurs due to the Trigger Source signal meeting the trigger condition, the popup window will show a block of capture of data from the trigger source signal.



■ Figure 69. Trigger window with Accept or Reject options.

The display will depend on the type of signals in the pane.

1. If a time stream is displayed then the display will update continuously. You will not notice the difference before or after trigger event for the time stream.

2. If a block signal is displayed, the block signal(s) in the background window will be updated with the new content.

3. If a block signal in the frequency domain is displayed, it will not be updated because you have not "accept" the time signal yet.

Now you may do one of two things: **Accept** or **Reject**.

If you press the **Accept** button, the acquired block signals will be passed to the signal analysis stage, usually windowing, FFT and spectral analysis. Then you can continue to the **Next** frame of capture.

After you press the **Next** key, the system will go back to waiting in trigger mode.

If you press **Reject** then the captured time signals will be discarded and will not be sent to signal analysis stage. After the **Reject** action, the system goes back to waiting in trigger mode.

The number showed on the top status bar, #N, indicates the number of the frames of the time captures that have been *accepted* and averaged into the spectra.

After you press the **Hide** key, the small from window will disappear. Press **Enter** to show this window again.

During trigger operation you can switch the main display window to any trace. This can be helpful to view the time stream selected as the trigger source to tune the level and slope settings.

Save Data to flash Memory or SD memory card

Introduction

Ranging from different applications, several ways are devised to save the signals that are being measured. The media of storage can be either internal flash memory or SD memory card.

Save Long Time Waveform Signal: the time streams can be saved either automatically by a preset schedule or manually.
Save Block Signals: The transient capture time signals, frequency signals or other block signals can be saved automatically or manually.
Save Points: The current value of the time streams, or RMS of a spectrum, or multiple statistics of signals, can be saved automatically or manually in to one file over long period of time. This is particularly useful in the monitoring applications.

The data can be saved either manually or automatically.

Save Long Time Waveform Signals

The Rec./Stop Button is used to control the streaming of time stream data to memory. After a CSA project is selected, pressing the Rec./Stop button will start the display and also start recording the time stream to memory. The red flashing Rec icon at the top of the screen indicates that the data is recording.

To stop the recording press the Rec./Stop button again. The red flashing Rec icon will not be displayed indicating that the recording has stopped.

Before a time stream can be recorded it must be defined in the Parameter Settings/Time Steam Recording Setup. If no time streams are defined in this setting when the Rec./Stop button in pressed then a message will indicate that no signal are selected.

The Rec./Stop button can also be pressed after the Run button is pressed. The Run button starts the display of live signals but does not start recording. After a recording is stopped the display will continue to display live signals until the Hold button is pressed.

A special data compression algorithm is developed in order to save the storage space. It only applies to time stream recording.

For uncompressed data, use following formula to calculate the total time duration that you can record:

Total Installed Flash Memory in Bytes = (Total Channel Enabled) * (Recording Time in Seconds) * (Sampling Rate) * 8 Bytes * 1.2

Or

Recording Time in Seconds = Total Installed Memory in Bytes/((Total Channel Enabled) * (Sampling Rate) * 8 Bytes * 1.2)

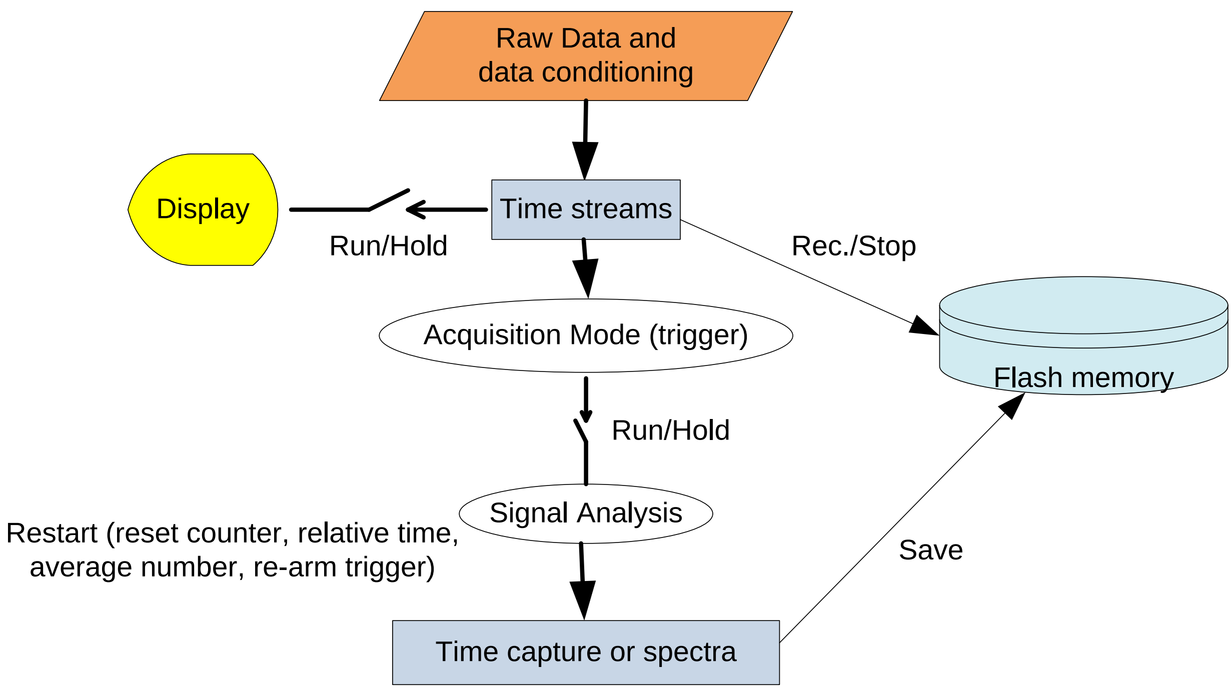
For example if you enabled 6 channels, the sampling rate is 100 Hz with 4GB memory installed:

Recording Time in Seconds = 4GB/(6*100*8*1.2) = 4*1024*1024*1024/(6*100*8*1.2) = 745654 sec (= 207 hours)

When data compression is used, the storage space will be doubled. The spectrum dynamic range of compressed data will be reduced to about 100dB.

If the storage space and downloading time is not an issue for your application, then data compression should not be used.

The figure below illustrates the difference between the concepts of **Display - Run/Hold**, **Time Stream - Record/Stop** and **Signal - Save**. The Display mode is independent of the Record or Save functions. When you change the Display mode between Run and Hold it has no effect on the Save or Record functions. That means that time streams can continue to be recorded when the display is in Hold mode.



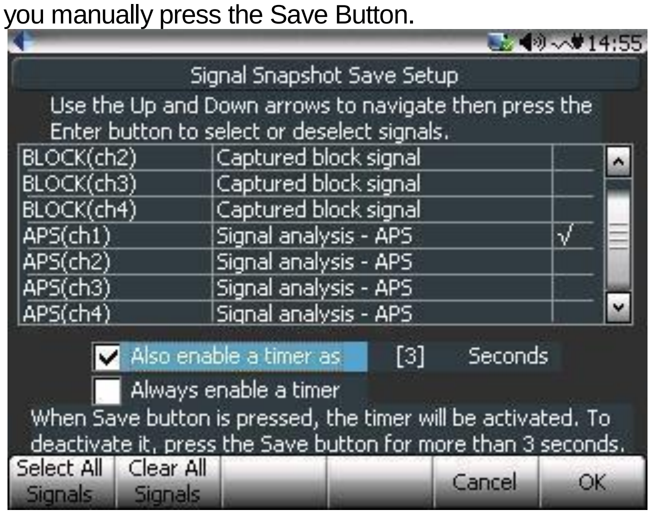
■ Figure 70. Illustration of the difference between Display Run/Hold, Time Stream Record/Stop and Signal Save.

Save Block Signals

Data can be saved by defining which signals to save and under what conditions. Select Signal Snapshot Save Setup under the Param. Button.

■ Figure 71. Signal snapshot save setup.

Choose which signals should be saved by using the Up and Down Arrows and the Enter Button to add a check next to the desired signals. These signals will be saved during a measurement when you manually press the Save Button.



■ Figure 72. Signal snapshot save setup.

In addition, these signals can be automatically saved by placing a check next to Also enable a timer and specifying the number of seconds between automatic saves. The signals can be saved with no delay between blocks by selecting Save Signals Continuously. This option can be used to view all data blocks on a waterfall plot.

Save Points

Save Points function saves a data point per signal at one time. This function is particularly useful in the very long period monitoring applications. For example people can save and monitor the vibration or acoustic level over a few months by looking at the data points saved every hour.

All data points in one test will be saved into one data file. The user can easily open, view and analyze the data files using EDM PC software.

To set up the Save Points, first go to Param.->Signal Save Predefined List screen, then press F4 the Save Point Setup button, a tab display will be shown. The user can select one of following items for any time signal to save:

Current Value, Max, Min, Peak, Average, RMS.

Or the user can select one of following items for spectrum signals to save:

Peak, RMS, Frequency of Peak

Or the user can select one of following items for Sound Level Meter measurement to save:

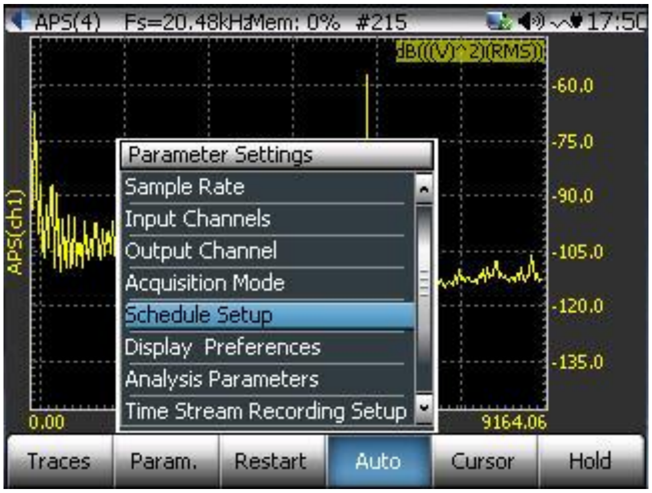
Leq, Lmax

Using Schedule to Save Data

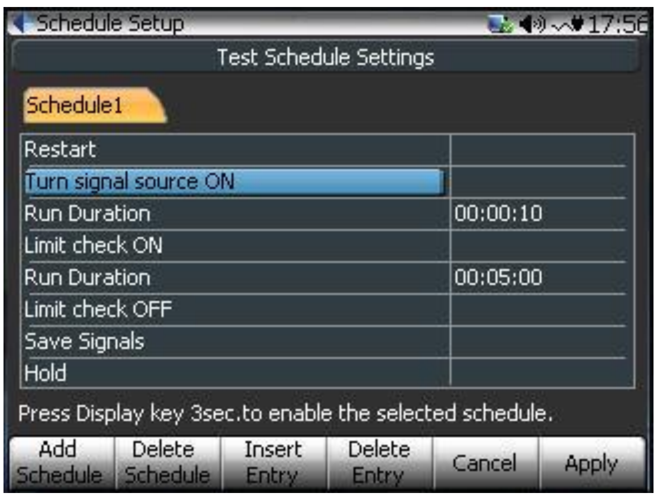
An automated schedule can be developed for recording the time streams, saving the block signals or data points.

- Restart
- Hold
- Run Duration regardless Trigger
- Run Duration after Trigger
- Waiting for one time
- LOOP
- END LOOP
- Limit check ON
- Limit check OFF
- Start Recording
- Stop Recording
- Save Signals
- Turn signal source ON
- Turn signal source OFF
- Activate Timer to Save Signals
- Deactivate Timer to Stop Saving Signals
- Set all input mode

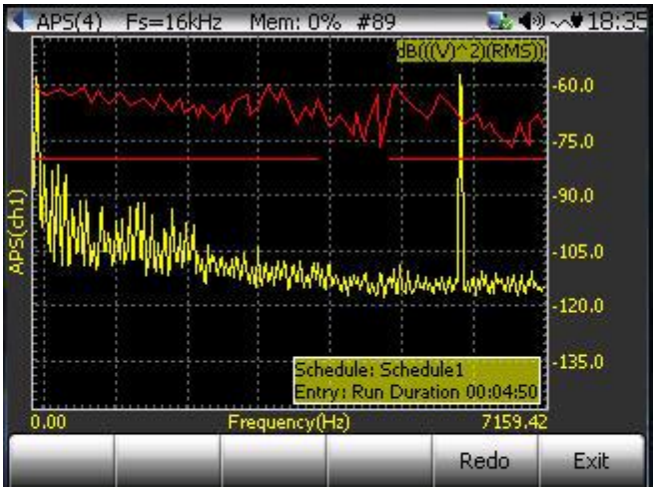
To make a schedule, first go to the Schedule Setup screen:



Insert appropriate entries into the schedule:



To activate the schedule, press the Display button for more than 3 seconds, then release the button.



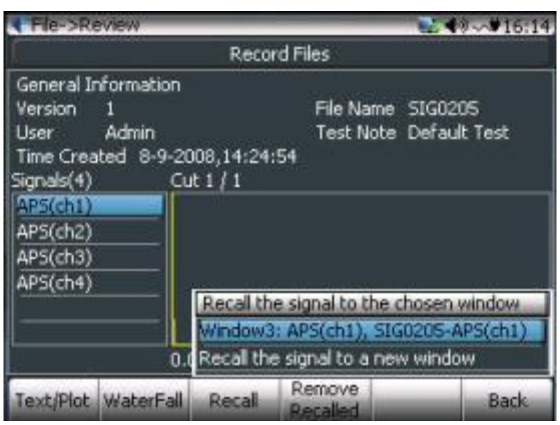
To deactivate the schedule, press F6 the Exit button in the signal display screen.

Recall Signals

Signals that are the result of a current measurement are named "live" signals. Occasionally it is helpful to compare live signals with previously saved signals and stored on the CoCo device. This can be done with the recall feature. Recalled signals can be overlaid with live signals for comparison or displayed in a separate window. Signals can also be un-recalled which removed them from all displays but does not affect the data saved on the CoCo.

To recall the signals that you just saved, simply press the Recall hard button.

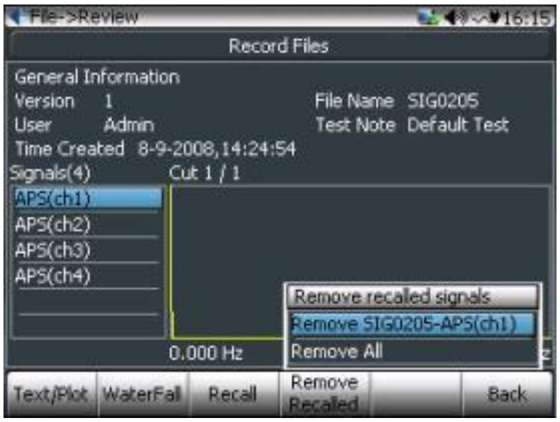
To recall a signal that you previously saved, press the File Button, then press the F1 Files Button then press the F5 Review Button, Next use the up and down arrow buttons to highlight a signal to be recalled and press the. F3 Recall Button. Next highlight a signal from the Record Files list and then press the F3 Recall Button.



■ Figure 73. File review screen for recalling signals.

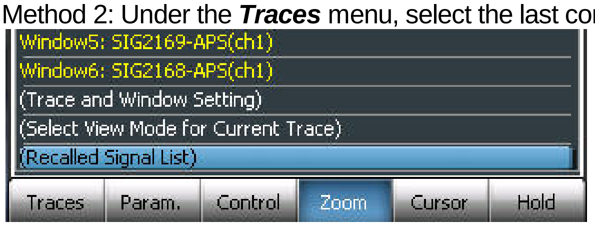
The CoCo will show a menu listing all available windows that this signal can be recalled into. The last item is always "Recall the signal to a new window". This item will create a new display window in the current active project and display the recalled signal into this window. Other selections will allow the recalled signal to be overlaid with the others.

After a signal is recalled, it can then be un-recalled. This removes the signal from the all displays however the original data file remains stored on the CoCo. To un-recall a signal, take one of the following two actions.

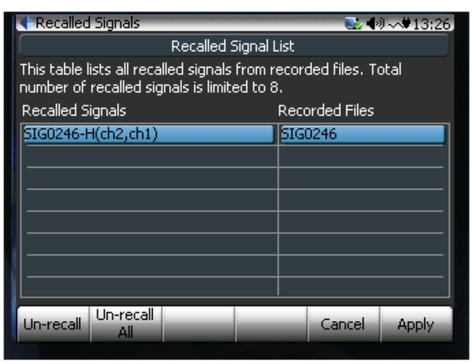


■ Figure 74. Recall signal pop up menu.

Method 1: In the above file menu, press F4 to see all signals that can be un-recalled. You can either unrecall one signal or un-recall all of them.
Method 2: Under the ***Traces*** menu, select the last command *(Recalled Signal List)*:

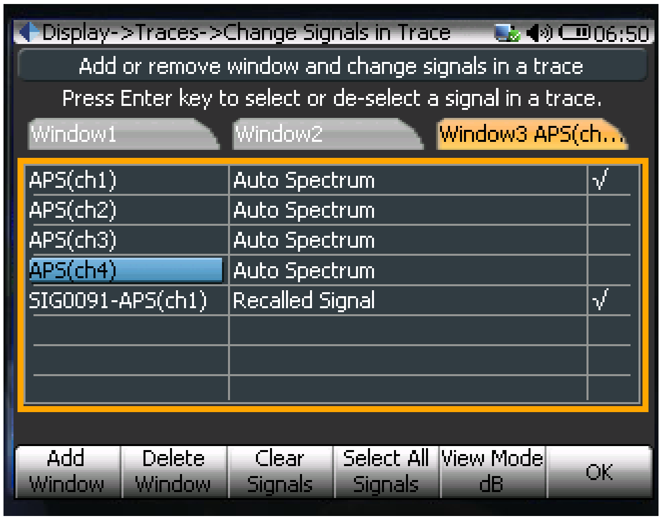


■ Figure 75. Recall signals from the trace menu.



■ Figure 76. Un-recall signal.

The recalled signals will have a signal name with a prefix of its file name. The recalled signals can be displayed in the same way as live signals.

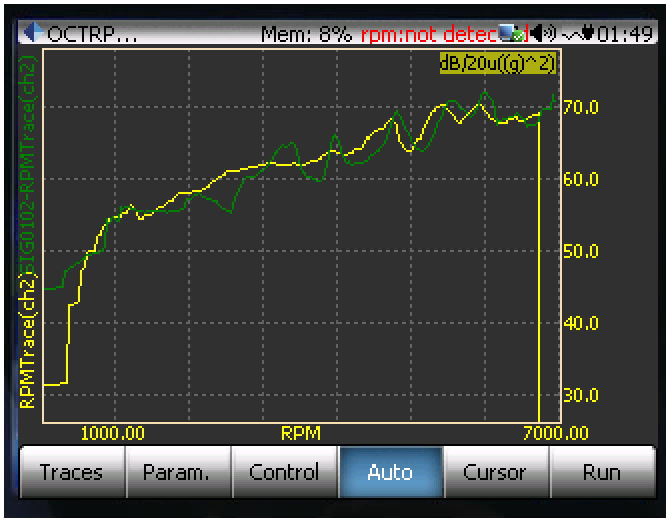


■ Figure 77. Recalled signals appear with file name prefix.

The picture below shows a recalled signal in green color overlaid with a live signal in yellow.

■ Figure 78. Live signal overlaid with recalled signal.

The picture below shows the saved RPM trace signal in octave analysis that is overlaid with the current live measurement.



■ Figure 79. RPM live signal overlaid with recalled signal.

4. BASICS OF DYNAMIC SIGNAL ANALYSIS

DSA, often referred to Dynamic Signal Analysis or Dynamic Signal Analyzer depending on the context, is an application area of digital signal processing technology. Compared to general data acquisition and time domain analysis, DSA instruments and math tools focus more on the dynamic aspect of the signals such as frequency response, dynamic range, total harmonic distortion, phase match, amplitude flatness etc.. In recent years, time domain data acquisition devices and DSA instruments have gradually converged together. More and more time domain instruments, such as oscilloscopes, can do frequency analysis while more and more dynamic signal analyzers can do long time data recording.

DSA uses various different technology of digital signal processing. Among them, the most fundamental and popular technology is based on the so called the Fast Fourier Transform (FFT). The FFT transforms the time domain signals into the frequency domain. To perform FFT-based measurements, however, you need to understand the fundamental issues and computations involved. This Chapter describes some of the basic signal analysis computations, discusses antialiasing and acquisition front end for FFT-based signal analysis, explains how to use windowing functions correctly, explains some spectrum computations, and shows you how to use FFT-based functions for some typical measurements.

General Theory of Spectral Analysis

In this Chapter we will use standard notation for different signals. Each type of signal will be represented by one specific letter. For example, "G" stands for a one-side power spectrum, while "H" stands for a transfer function.

The following table defines the symbols used in this Chapter:

Cyx	Coherence function between input signal x and output signal y
Gxx	Auto-spectral function (one-sided) of signal x
Gyx	Cross-spectral function (one-sided) between input signal x and output signal y
Hyx	Transfer function (Frequency Response) between input signal x and output signal y
k	Index of a discrete sample
Rxx	Auto-correlation function of signal x
Ryx	Cross-correlation function between input signal x and output signal y
Sx	Linear spectral function of signal x
Sxx	Instantaneous auto-spectral function (one-sided) of signal x
Syx	Instantaneous cross-spectral function (one-sided) between input signal x and output signal y
t	Time variable
x(t)	Time history record
X(f)	Fourier Transform of time history record

Fourier Transform

Digital signal processing technology includes FFT based frequency analysis, digital filters and many other topics. This chapter introduces the FFT based frequency analysis methods that are widely used in all dynamic signal analyzers. CoCo has fully utilized the FFT frequency analysis methods and various real time digital filters to analyze the measurement signals.

The Fourier Transform is a transform used to convert quantities from the time domain to the frequency domain and vice versa, usually derived from the Fourier integral of a periodic function when the period grows without limit, often expressed as a Fourier transform pair. In the classical sense, a Fourier transform takes the form of

$$X(f) = \int_{-\infty}^{\infty} x(t)e^{-j2\pi ft}\, dt$$

where

x(t)	continuous time waveform
f	frequency variable
j	complex number
X(f)	Fourier transform of *x(t)*

Mathematically the Fourier Transform is defined for all frequencies from negative to positive infinity. However, the spectrum is usually symmetric and it is common to only consider the single-sided spectrum which is the spectrum from zero to positive infinity." For discrete sampled signals, this can be expressed as

$$X(k) = \sum_{n=0}^{N-1} x(k)e^{-j2\pi kn/N}$$

where

x(k)	samples of time waveform
n	running sample index
N	total number of samples or "frame size"
k	finite analysis frequency, corresponding to "FFT bin centers"
X(k)	discrete Fourier transform of *x(k)*

In most DSA products, a Radix-2 DIF FFT algorithm is used, which requires that the total number of samples must be a power of 2 (total number of samples in FFT = 2^m , where m is an integer).

Data Windowing

The Fourier Transform assumes that the time signal is periodic and infinite in duration. When only a portion of a record is analyzed the record must be truncated by a data window to preserve the frequency characteristics. A window can be expressed in either the time domain or in the frequency domain, although the former is more common. To reduce the edge effects, which cause leakage, a window is often given a shape or weighting function. For example, a window can be defined as

$w(t) = g(t)$	$-T/2 < t < T/2$
$= 0$	elsewhere

where $g(t)$ is the window weighting function and T is the window duration.

The data analyzed, $x(t)$ are then given by

$$x(t) = w(t)\, x(t)'$$

where $x(t)'$ is the original data and $x(t)$ is the data used for spectral analysis.

A window in the time domain is represented by a multiplication and hence, is a convolution in the frequency domain. A convolution can be thought of as a smoothing function. This smoothing can be represented by an effective filter shape of the window; i.e., energy at a frequency in the original data will appear at other frequencies as given by the filter shape. Since time domain windows can be represented as a filter in the frequency domain, the time domain windowing can be accomplished directly in the frequency domain.

In most DSA products, rectangular, Hann, Flattop and several other data windows are used;

Rectangular Window

$$w(k) = 1 \qquad 0 \le k \le N\text{-}1$$

Hann Window

$$w(k) = 0.5 * (1 - \cos(2\pi k/(N-1))) \qquad 0 \le k \le N\text{-}1$$

Because creating data window attenuates a portion of the original data, a certain amount of correction has to be made in order to get an un-biased estimation of the spectra. In linear spectral analysis, an *Amplitude Correction* is applied; in power spectral measurements, an *Energy Correction* is applied. See the sections below for details.

Linear Spectrum

A linear spectrum is the Fourier transform of windowed time domain data. The linear spectrum is useful for analyzing periodic signals. You can extract the harmonic amplitude by reading the amplitude values at those harmonic frequencies.

An averaging technique is often used in the time domain when synchronized triggering is applied. Or equivalently, the averaging can be applied to the complex FFT spectra.

Because the averaging is taking place in the linear spectrum domain, or equivalently, in the time domain, based on the principles of linear transform, averaging make no sense unless a synchronized trigger is used.

Most DSA products use the following steps to compute a linear spectrum:

<u>Step 1</u>
First a window is applied:

$$x(t) = w(t)\, x(t)'$$

where $x(t)'$ is the original data and $x(t)$ is the data used for the Fourier transform.

<u>Step 2</u>

The FFT is applied to *x(t)* to compute *X(k)*, as described above.

Step 3
Averaging is applied to *X(k)*. Here Averaging can be either an Exponential Average or Stable Average. Result is *Sx'*.

Sx' = Average (X(k))

Step 4
To get a single-sided spectrum, double the value for symmetry about DC.
An Amplitude Correction factor is applied to *Sx'* so that the final result has an un-biased reading at the harmonic frequencies.

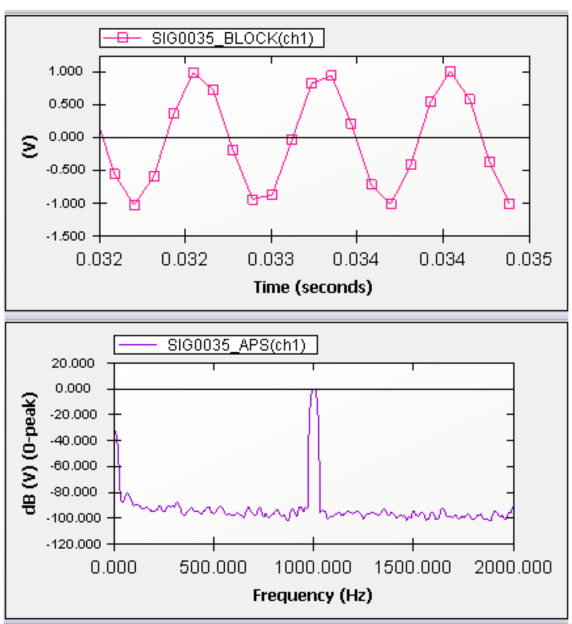
$$Sx = 2 \bullet Sx' / AmpCorr$$

where AmpCorr is the amplitude correction factor, defined as:

$$AmpCorr = \sum_{k=0}^{N-1} w(k)$$

where *w(k)* is the window weighting function.

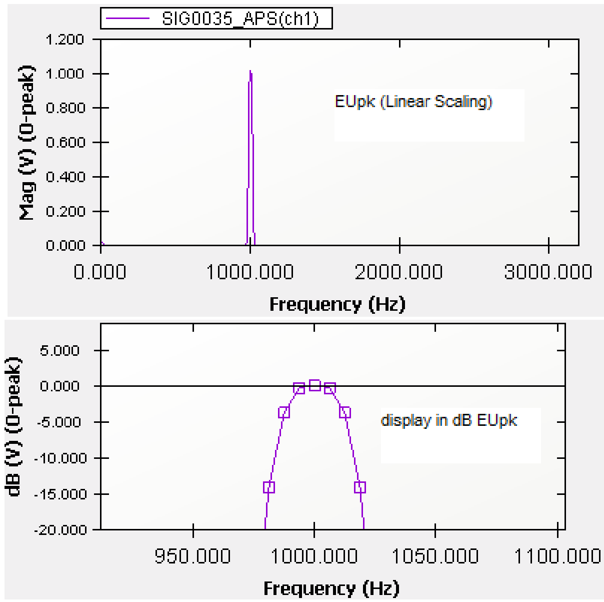
This correction will make the peak or RMS reading of a sine wave at specific frequency correct regardless of which data window is applied. For example, if a 1.0 volt amplitude 1kHz sine wave sampled at 6.4kHz is analyzed with a Linear Spectrum with Hann window, you will get following the spectral shape:



■ Figure 80. Sine wave with Hanning window applied to the spectrum.

The top picture is the digitized time waveform. The sine-wave is not smooth because of the low sampling rate relative to the frequency of the signal. However the well known Nyquist principle indicates that the frequency estimate from the FFT will be accurate as long as the sampling rate is more than twice of the signal frequency. The frequency spectrum of the period signal will show the accurate frequency and level. Note for a more accurate sample of the time waveform a higher sampling rate is required.

Figure 81 illustrates a windowing function applied to a pure sine tone.



■ Figure 81. Hanning windowing function applied to a pure sine tone.

The top picture is displayed in EUpk, i.e., the peak of the spectrum is scaled to the actual 0 peak level, which is 1.0 in this case. The bottom picture shows the same signal with the dB scale applied. Since we use 0dB as reference, the 1.0 Vpk is now scaled to 0.0 dB. With the dB display, we can see frequency points around the peak causing by the Hanning window.

In many DSA products, amplitude correction is automatically applied when you select different Spectrum Types. For details about Spectrum Types, see the section entitled Spectrum Types in this Chapter.

The linear spectrum is saved internally in the complex data format with real and imaginary parts. Therefore, you should be able to view the real and imaginary parts, or amplitude and phase of the spectrum.

Power Spectrum

Spectral analysis is popular in characterizing the operation of mechanical and electrical systems. A type of spectral analysis, the power spectrum (and power spectral density (PSD)), is especially popular because a "power" measurement in the frequency domain is one that engineers readily accept and apply in their solutions to problems. Single channel measurements (auto-power spectra) and two channel measurements (cross-power spectra) both play important roles.

In power spectrum measurements, window ***amplitude correction*** is used to get un-biased final spectrum amplitude reading at specific frequency. In PSD or energy spectral density (ESD) measurements, window ***energy correction*** is always used to get an un-biased spectral density or energy reading.

To compute the spectra listed above, the instrument will follow these steps:

Step 1
A window is applied:

$x(k) = w(k)\, x(k)'$

where $x(k)'$ is the original data and $x(k)$ is the data used for a Fourier transform.

Step 2
The FFT is applied to $x(t)$ to compute Sx

$$Sx = \sum_{n=0}^{N-1} x(k) e^{-j2\pi kn/N}$$

Next the so called periodogram method is used to compute the spectra with area correction. Using Sx.

Step 3

Calculate the Power Spectrum $Sxx = Sx\, Sx^{*} / (AmpCorr)^{2}$

Or calculate the Power Spectral Density $= Sx\, Sx^{*}\, T / EnergyCorr$

Or calculate the Energy Spectral Density $= Sx\, Sx^{*}\, T^{2} / EnergyCorr$

where T is the time duration of the capture. The symbol * is for complex conjugation. *EnergyCorr* is a factor for energy correction, which is defined as:

$$EnergyCorr = \frac{1}{N}\sum_{k=0}^{N-1} w(k)^{2}$$

N is the total number of the samples and $w(k)$ is window function.

For any power spectral measurement of the three types listed above, the EU is automatically chosen as EU_{rms} because only EU_{rms} has a physical meaning related to signal power.

After the power spectra are calculated, the averaging operation will be applied. More details will be discussed in the next sections for averaging operation.

Spectrum Types

Several Spectrum Types are given for both Linear Spectrum and Power Spectrum measurements in CoCo and EDM. The concept of spectrum type is explained below in detail.

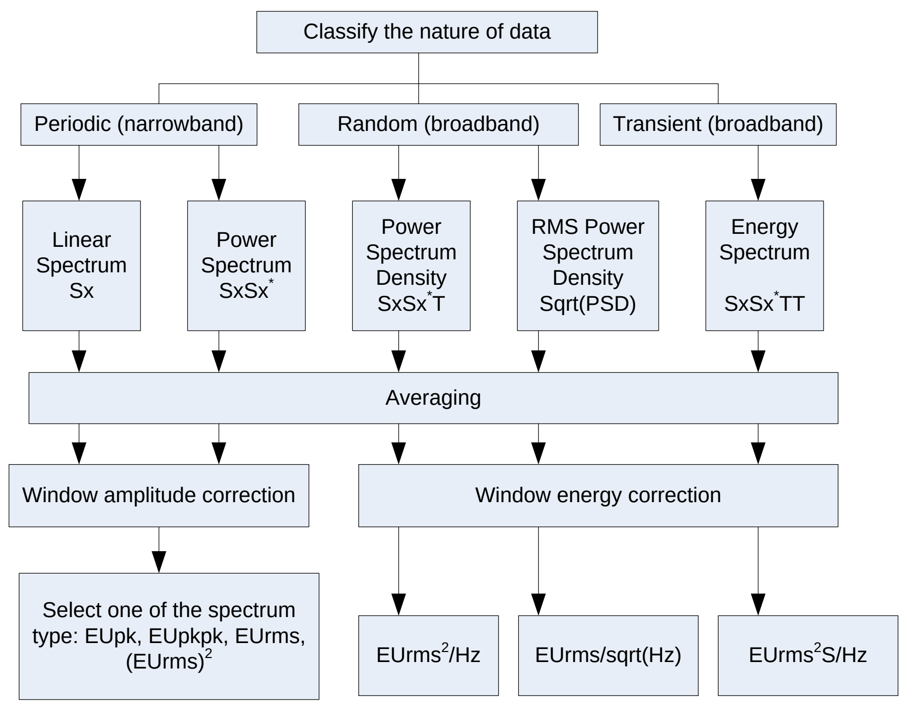
First let's consider the signals with periodic nature. These can be the signals measured from a rotating machine, bearing, gearing, or anything that repeats. In this case we would be interested in amplitude changes at fundamental frequencies, harmonics or sub-harmonics. In this case, you can choose a spectrum type of EU_{pk}, EU_{pkpk} or EU_{rms}.

A second scenario might consist of a signal with a random nature that is not necessarily periodic. It does not have obvious periodicity therefore the frequency analysis could not determine the "amplitude" at certain frequencies. However, it is possible to measure the r.m.s. level, or power level, or power density level over certain frequency bands for such random signals. In this case, you must select one of the spectrum types of EU_{rms}^2/Hz, or EU_{rms}/sqrt(Hz), which is called power spectral density, or root-mean squared density.

A third scenario might consist of a transient signal. It is neither periodic, nor stably random. In this case, must select a spectrum type as EU^2S/Hz, which is called energy spectrum.

In many applications, the nature of the data cannot be easily classified. Care must be taken to interpret the data when different spectrum types are used. For example, in the environmental vibration simulation, a typical test uses multiple sine tones on top of random profile, which is called Sine-on-Random. In this type application, you have to observe the random portion of the data in the spectrum with EU_{rms}^2/Hz and the sine portion of the data with EU_{pk}.

Figure 82 shows a general flow-chart to choose one of the measurement techniques and spectrum types for linear or auto spectrum:

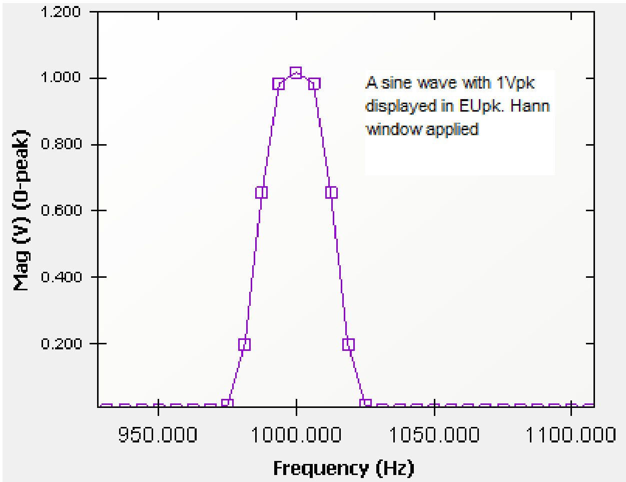


■ Figure 82. Flow chart to determine measurement technique for various signal types.

The following figures illustrate the results of different measurement techniques on a 1 volt pure sine tone. The figures include RMS, Peak or Peak-Peak value for the amplitude, or power value corresponding to its amplitude. Notice these readings can only be applied to a periodic signal. If you applied these measurement techniques to a signal with random nature, the spectrum would not be a meaningful representation of the signal.

EU_{pk} or EU_{pkpk}

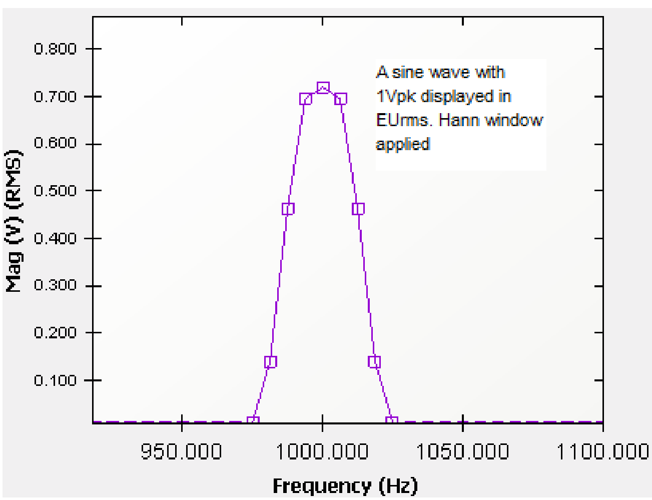
The EU_{pk} and EU_{pkpk} displays the peak value or peak-peak value of a periodic frequency component at a discrete frequency. These two spectrum types are suitable for narrowband signals.



■ Figure 83. A sine wave is measured with EUpk spectrum unit. The sine waveform has a 1V amplitude.

EU_{rms}

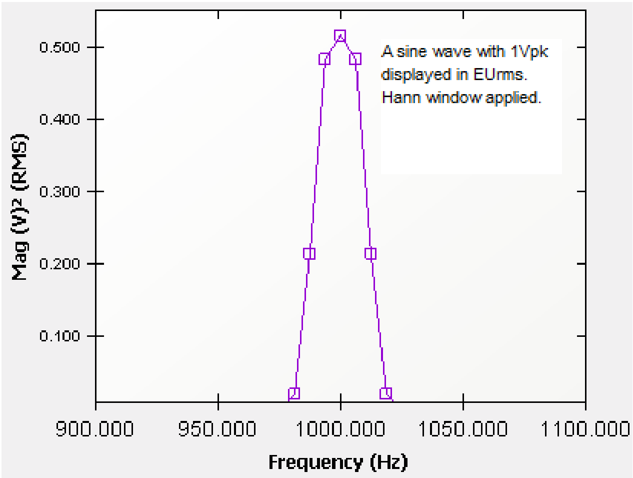
The EU_{rms} displays the RMS value of a periodic frequency component at a discrete frequency. This spectrum type is suitable for narrowband signals.



■ Figure 84. A sine wave is measured with EUrms spectrum unit. The peak reading is 0.707V. The sine waveform has a 1V amplitude.

$(EU_{rms})^2$ Power spectrum

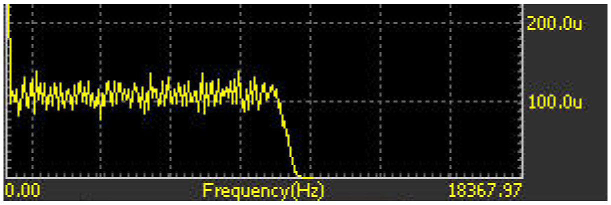
The $(EU_{rms})^2$ displays the power reading of a periodic frequency component at a discrete frequency. This spectrum type is suitable for narrowband signals.



■ Figure 85. A sine wave is measured with (EUrms)2 spectrum unit. The peak reading is $0.5V^2$. The sine waveform has a 1V amplitude.

EU^2/Hz, Power Spectrum Density

The EU^2/Hz is the spectrum unit used in power spectrum density (PSD) calculations. The unit is in engineering units squared divided by the equivalent filter bandwidth. This provides power normalized to a 1Hz bandwidth. This is useful for wideband, continuous signals. EU^2/Hz really should be written as $(EU_{rms})^2$/Hz. But probably due to the limitation of space, people put it as EU^2/Hz.



■ Figure 86. White noise with 1 volt RMS amplitude and $volts^2$/Hz PDS units.

Figure 86 shows a white noise signal with $1V_{rms}$ amplitude or $1V^2$ in power level. The bandwidth of the signal is approximately 10000 Hz and the V^2/Hz reading of the signal is around 0.0001 V^2/Hz. The 1 V RMS can be calculated as follows:

$1\ V_{rms} = \text{sqrt}(10000Hz * 0.0001\ V^2/Hz)$

EU^2S/Hz, Energy Spectrum Density

The EU^2S/Hz displays the signal in engineering units squared divided by the equivalent filter bandwidth, multiplied by the time duration of signal. This spectrum type provides energy normalized to a 1Hz bandwidth, or energy spectral density (ESD). It is useful for any signals when the purpose is to measure the total energy in the data frame. Figure 87 shows a random signal with a 1 volt RMS level in the ESD format.

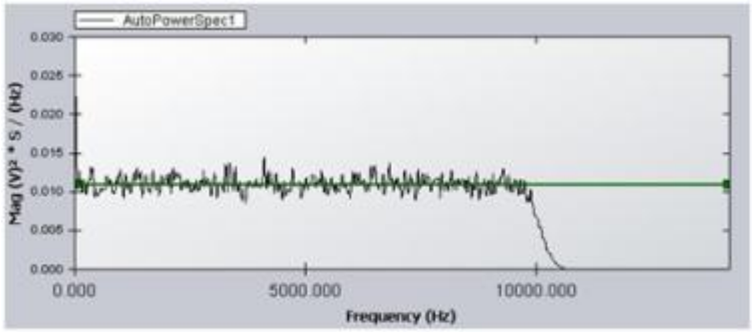


Figure 87. Random signal with 1 volt RMS amplitude and Energy Spectrum Density format.

The ESD is calculated as follows:

Values for ESD = values of PSD * Time Factor

were the Time Factor = (Block size)/∆f and ∆f is the sampling rate / block size.

Notice that in **EU^2/Hz**, or **EU^2S/Hz,** EU really means the RMS unit of the EU, i.e., EU_{rms}.

It should also be noted that since a window is applied in time domain, which corresponds a convolution in the linear spectrum, we cannot have both a valid amplitude and correct energy correction at the same time. Use Figure 82 to select appropriate spectrum types.

In a Linear Spectrum measurement, a signal is saved in its complex data format which includes both real and imaginary data. Then is averaging operation applied to the linear spectrum. In a Power Spectrum measurement, the averaging operation is applied to the squared spectrum, which has only real part. Because of different averaging techniques, the final results of Linear Spectrum and Power Spectrum will be different even though the same spectrum type is used.

Spectrum Types selection only applies to Power Spectrum and Linear Spectrum signals. Spectrum Types do not apply to transfer functions, phase functions or coherence functions.

Cross Spectrum

Cross spectrum or cross power spectrum density is a frequency spectrum quantity computed using two signals, usually the excitation and response of a dynamic system. Cross spectrum is not commonly used by its own. Most often it is used to compute the frequency response function (FRF), transmissibility or cross correlation function.

To compute the cross-power spectral density *Gyx* between channel x and channel y:

Step 1, compute the Fourier transform of input signal x(k) and response signal y(k):

$$Sx = \sum_{n=0}^{N-1} x(k)\, w(k) e^{-j2\pi kn/N}$$

$$Sy = \sum_{n=0}^{N-1} y(k)\, w(k) e^{-j2\pi kn/N}$$

Step 2, compute the instantaneous cross power spectral density

Syx = Sx Sy T*

Step 2, average the *M* frames of *Sxx* to get averaged PSD *Gxx*

Gyx' = Average (Syx)

Step 3, Compute the energy correction and double the value for the single-sided spectra

Gyx = 2 Gyx' / EnergyCorr

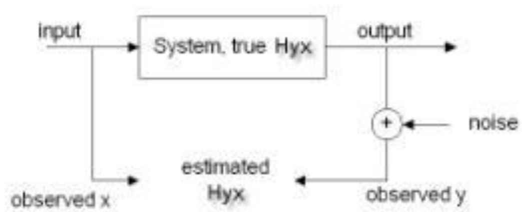
Frequency Response and Coherence Function

The cross power spectrum method is often used for estimating the frequency response function (FRF) between channel x and channel y. The equation is:

$$H_{yx} = G_{yx} / G_{xx}$$

where *Gyx* is the averaged cross-spectrum between the input channel x and output channel y. *Gxx* is the averaged auto-spectrum of the input. Either power spectrum, power spectral density or energy spectral density can be used to compute the FRF because of the linear relationship between input and output.

Using the cross-power spectrum method instead of simply dividing the linear spectra between input and output to calculate the FRF will reduce the effect of the noise at the output measurement end, as shown below.



■ Figure 88. Frequency response function computation.

The frequency response function has a complex data format. You can view it in real and imaginary or magnitude and phase display format.

The coherence function is defined as:

$$C_{yx}^{\ 2} = \frac{|G_{yx}|^2}{G_{xx}G_{yy}}$$

where *Gyx* is the averaged cross-spectrum between the input channel x and output channel y. *Gxx* and *Gyy* are the averaged auto-spectrum of the input and output. Either power spectrum, power spectral density or energy spectral density can be used here because of the linear relationship between input and output so that any multiplier factors will be cancelled out.

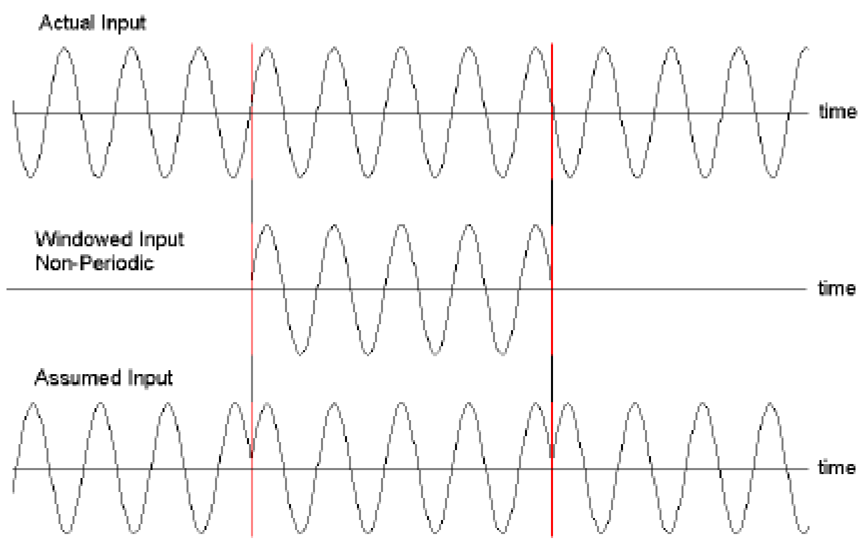
Coherence is a statistical measure of the how much of the output is caused by the input. The maximum coherence is 1.0 when the output is perfectly correlated with the input and zero when there is no correlation between input and output. Coherence is calculated by an average of multiple frames. When it is computed for only one frame, then the coherence function has a meaningless result of 1.0 due to the estimation error of the coherence function.

The coherence function is a non-dimensional real function in the frequency domain. You can only view it in the real format.

Data Window Selection

Leakage Effect

Windowing of a simple signal, like a sine wave may cause its Fourier transform to have non-zero values (commonly called leakage) at frequencies other than the frequency of this sine. This leakage effect tends to be worst (highest) near sine frequency and least at frequencies farthest from sine frequency. The effect of leakage can easily be depicted in the time domain when a signal is truncated. As shown in the picture, after data windowing, truncation distorted the time signal significantly, hence causing a distortion in its frequency domain.



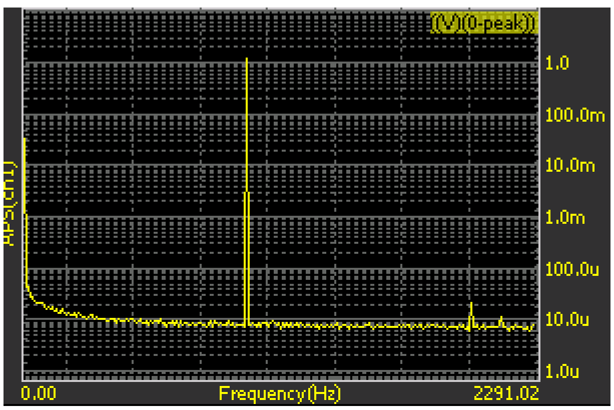
■ Figure 89. Illustration of a non-periodic signal resulting from sampling.

If there are two sinusoids, with different frequencies, leakage can interfere with the ability to distinguish them spectrally. If their frequencies are dissimilar, then the leakage interferes when one sinusoid is much smaller in amplitude than the other. That is, its spectral component can be hidden or masked by the leakage from the larger component. But when the frequencies are near each other, the leakage can be sufficient to interfere even when the sinusoids are equal strength; that is, they become undetectable.

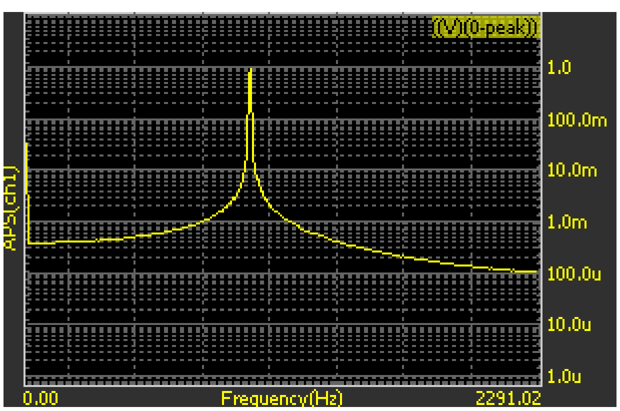
There are two possible scenarios that leakage does not occur. The first is that when the whole time capture is long enough to cover the complete duration of the signals. This can occur with short transient signals. For example in a hammer test, if the time capture is long enough it may extend to the point where the signal decays to zero. In this case, data window is not needed.

The second case is when a periodic signal is sampled at such a sampling rate that is perfectly synchronized with the signal period, so that with a block of capture, an integer number of cycles of the signal are always acquired. For example, if a sine wave has a frequency of 1000Hz and the sampling rate is set to 8000Hz. Each sine cycle would have 8 integer points. If 1024 data points are acquired then 128 complete cycles of the signal are captured. In this case, with no window applied you still can get a leakage-free spectrum.

Figure 90 shows a sine signal at 1000 Hz with no leakage resulting in a sharp spike. Figure 91 shows the spectrum of a 1010 Hz signal with significant leakage resulting in a wide peak. The spectrum has significant energy outside the narrow 1010 Hz frequency. It is said that the energy leaks out into the surrounding frequencies.

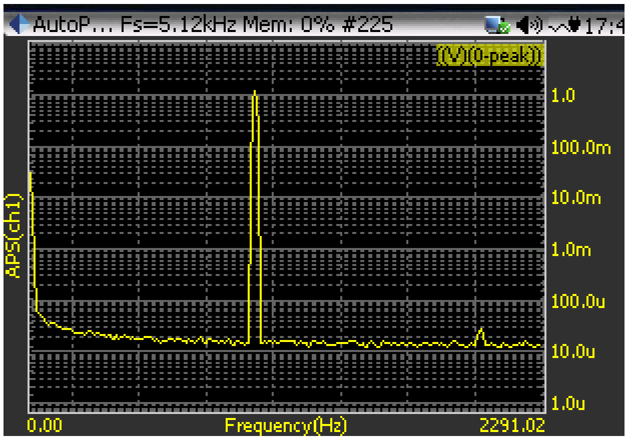


■ Figure 90. Sine spectrum with no leakage.



■ Figure 91. Sine spectrum with significant leakage.

Several windowing functions have been developed to reduce the leakage effect. The picture below shows a Flattop window applied to the same sine signal with frequency 1010Hz:



■ Figure 92. Sine spectrum with Flattop windowing function.

When Flattop window is used, the leakage effect is reduced. Both the sine peak and noise floor can be seen now. However, such data windowing operation also makes the spectrum peak "fatter" and less accurate. In the rest of the sections we will discuss how to choose different data windows.

Data Window Formula

In this section, we will describe the math formula that we used for each data window.

Uniform window (rectangular)

$$w(k) = 1.0$$

Uniform is the same as no window function.

Hamming window

$$w(k) = 0.53836 - 0.46164 \cos\left(\frac{2\pi k}{N-1}\right)$$

Hann window

$$w(k) = 0.5 - 0.5 \cos\left(\frac{2\pi k}{N-1}\right)$$

The Hann and Hamming windows are in the family known as "raised cosine" windows, are respectively named after Julius von Hann and Richard Hamming. The term "Hanning window" is sometimes used to refer to the Hann window, but is ambiguous as it is easily confused with Hamming window.

Blackman window

$$w(k) = 0.84 - 0.5\cos\frac{2\pi k}{N-1} + 0.08\cos\frac{4\pi k}{N-1} \qquad for\ k = 0 \sim N-1$$

Flattop window

$$w(k) = 1 - 1.93\cos\frac{2\pi k}{N-1} + 1.29\cos\frac{4\pi k}{N-1} - 0.388\cos\frac{6\pi k}{N-1} + 0.032\cos\frac{8\pi k}{N-1} \qquad for\ k = 0 \sim N-1$$

Kaiser Bessel window

$$w(k) = 1.0 - 1.24\cos\frac{2\pi k}{N-1} + 0.244\cos\frac{4\pi k}{N-1} + 0.00305\cos\frac{6\pi k}{N-1} \qquad for\ k = 0 \sim N-1$$

Exponential Window

The shape of the exponential window is that of a decaying exponential. The following equation defines the exponential window.

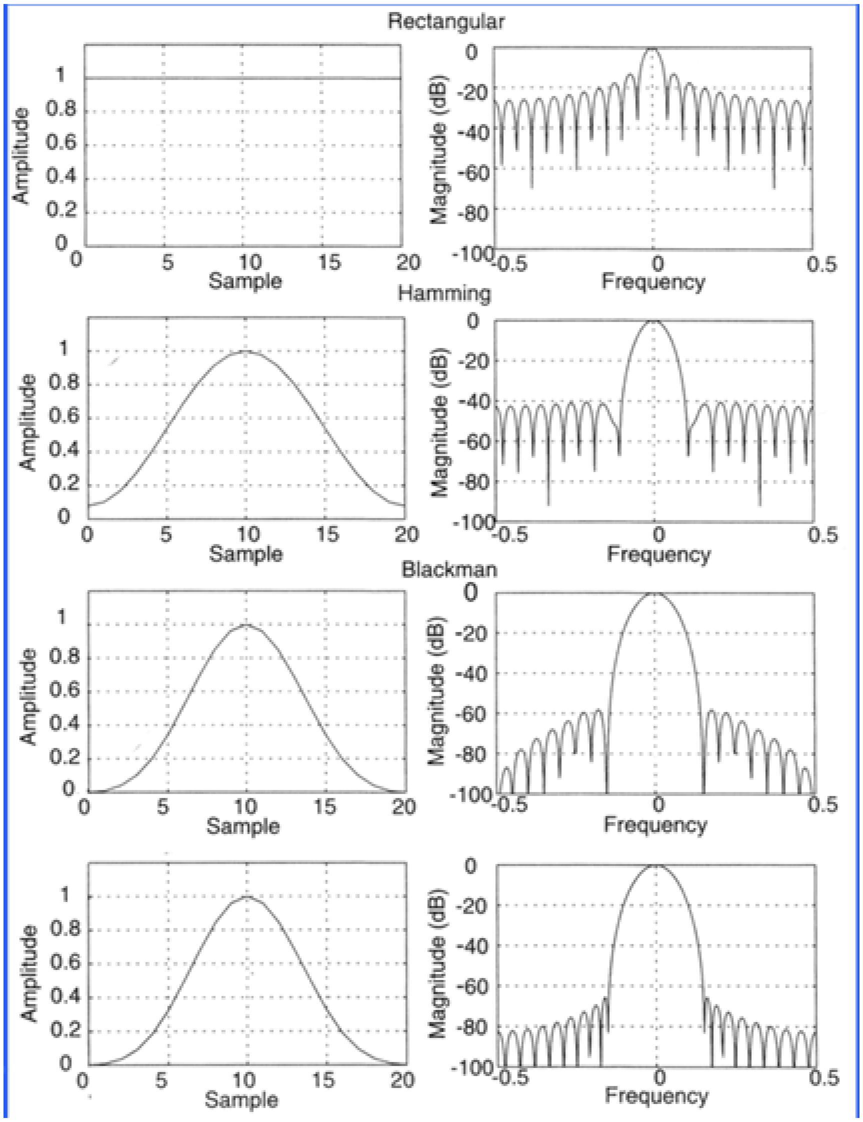
$$w(k) = e^{\left(\frac{k \ln(final)}{N-1}\right)}$$

$$for\ k = 0 \sim N-1$$

where N is the length of the window, w(k)is the window value, and ***final*** is the final value of the whole sequence. The initial value of the window is one and gradually decays toward zero.

How to Choose the Right Data Window

In this section we will discuss how to choose the data window. Figure 93 shows the spectral shape of four typical windows corresponding to their time waveform.



■ Figure 93. Spectral shape of common windowing functions.

It can be seen that the spectral shape of the data window is always symmetric. The spectral shape can be described as a main lobe and several side lobes.

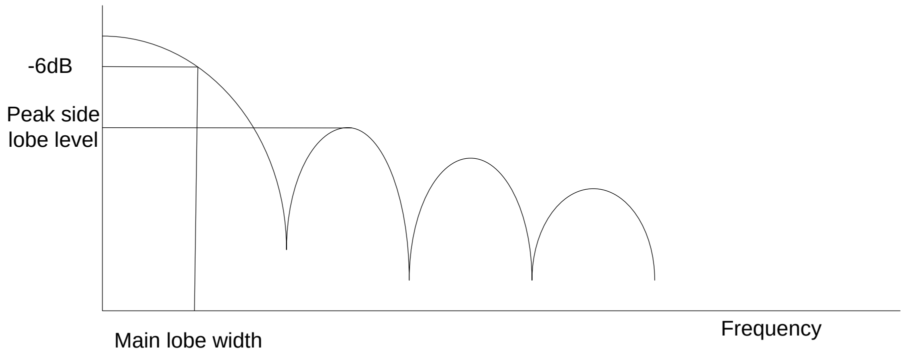


Figure 94. Window frequency response showing main lobe and side lobes.

The following table lists the characteristics of several data windows.

Frequency Characteristics of Data Windows

Window	–3 dB Main Lobe Width (bins)	–6 dB Main Lobe Width (bins)	Maximum Side Lobe Level (dB)
Uniform (none)	0.9	1.2	–13
Hanning	1.4	2.0	–32
Hamming	1.3	1.8	–43
Blackman	1.6	2.3	–58
Flattop	2.9	3.6	–44

Main Lobe

The center of the main lobe of a window occurs at each frequency component of the time-domain signal. By convention, to characterize the shape of the main lobe, the widths of the main lobe at –3 dB and –6 dB below the main lobe peak describe the width of the main lobe. The unit of measure for the main lobe width is FFT bins or frequency lines.

The width of the main lobe of the window spectrum limits the frequency resolution of the windowed signal. Therefore, the ability to distinguish two closely spaced frequency components increases as the main lobe of the smoothing window narrows. As the main lobe narrows and spectral resolution improves, the window energy spreads into its side lobes, increasing spectral leakage and decreasing amplitude accuracy. A trade-off occurs between amplitude accuracy and spectral resolution.

Side Lobes

Side lobes occur on each side of the main lobe and approach zero at multiples of f_s/N from the main lobe. The side lobe characteristics of the smoothing window directly affect the extent to which adjacent frequency components leak into adjacent frequency bins. The side lobe response of a strong sinusoidal signal can overpower the main lobe response of a nearby weak sinusoidal signal. Maximum side lobe level and side lobe roll-off rate characterize the side lobes of a smoothing window. The maximum side lobe level is the largest side lobe level in decibels relative to the main lobe peak gain.

Guidelines of Choosing Data Windows

If a measurement can be made so that no leakage effect will occur, then do not apply any window (in the software, select Uniform.). As discussed before, this only occurs when the time capture is long enough to cover the whole transient range, or when the signal is exactly periodic in the time frame.

If the goal of the analysis is to discriminate two or multiple sine waves in the frequency domain, spectral resolution is very critical. For such application, choose a data window with very narrow main slope. Hanning is a good choice.

If the goal of the analysis is to determine the amplitude reading of a periodic signal, i.e., to read EU_{pk}, EU_{pkpk}, EU_{rms} or EU_{rms}^2, the amplitude accuracy of a single frequency component is more important than the exact location of the component in a given frequency bin, choose a window with a wide main lobe. Flattop window is often used.

If you are analyzing transient signals such as impact and response signals, it is better not to use the spectral windows because these windows attenuate important information at the beginning of the sample block. Instead, use the Force and Exponential windows. A Force window is useful in analyzing shock stimuli because it removes stray signals at the end of the signal. The Exponential window is useful for analyzing transient response signals because it damps the end of the signal, ensuring that the signal fully decays by the end of the sample block.

If the nature of the data is has a random nature or unknown, choose Hanning window.

Averaging Techniques

Averaging is widely used in spectral measurements. It improves the measurement and analysis of signals that are purely random or mixed random and periodic. Averaged measurements can yield either higher signal-to-noise ratios or improved statistical accuracy.

Typically, three types of averaging methods are available in DSA products. They are:

Linear Averaging, **Exponential Averaging**, and **Peak-Hold**

Linear Averaging

In linear averaging, each set of data (a record) contributes equally to the average. The value at any point in the linear average in given by the equation:

$$Averaged = \frac{Sum\ of\ Records}{N}$$

N is the total number of the records. The advantage of this averaging method is that it is faster to compute and the result is un-biased. However, this method is suitable only for analyzing short signal records or stationary signals, since the average tends to stabilize. The contribution of new records eventually will cease to change the value of the average.

Usually, a target average number is defined. The algorithm is made so that before the target average number reaches, the process can be stopped and the averaged result can still be used.

When the specified target averaging number is reached, the instrument usually will stop the acquisition and wait for the instruction for another collection of data acquisition.

Moving Linear Averaging

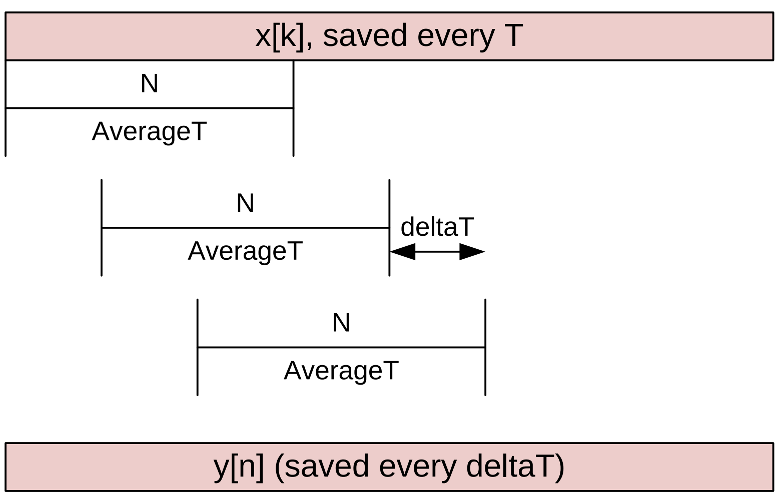
In a regular Linear Average, the data rate of the output of the averaging operator is only 1/N of that of the original signal. Therefore more averages takes longer to compute. Thus averaging will increase the time of the measurement. To reduce the time a Moving Linear Averaging can be used. Moving Linear Averaging uses overlapped input data points to generate more than 1/N results within a period of time. Moving linear average has the advantage that the resulted trace update time can be much shorter than the linear averaging period.

Moving Linear Average is computed by

$$y[n] = \frac{1}{N}\sum_{j=0}^{N-1} x[n-j]$$

Where x[k] is the input data, with sampling rate of T, y[n] is the output data, with Trace Update rate deltaT, AverageT is the period of Linear Average and ,N is the total samples used for Linear Average. N = AverageT/T

The Moving Linear Averaging is illustrated in Figure 95. Assume the averaging period is *AverageT* but the progressive time for each averaging operation is *deltaT*, the output buffer will have a data rage of *deltaT* instead of *AverageT*.



■ Figure 95. Illustration of moving linear average.

The Moving Linear Average is useful in many situations. For example, in Sound Level Meter, Leq is defined as a linear averaged value over a long period of time, say 1 second to 24 hours. Assume the AverageT is 1 hour, without moving linear average, in a 24 hours period, you can only get 24 readings. This is not very useful. With moving averaging, you can get the readings in every 1 second, for the linear averaging of the past 1 hour.

Exponential Averaging

In exponential averaging, records do not contribute equally to the average. A new record is weighted more heavily than old ones. The value at any point in the exponential average is given by:

$$y[n] = y[n-1] * (1-\alpha) + x[n] * \alpha$$

where $y[n]$ is the *nth* average and $x[n]$ is the nth new record. α is the weighting coefficient. Usually α is defined as 1/(Number of Averaging). For example in the instrument, if the Number of Averaging is set to 3 and the averaging type is selected as exponential averaging, then $\alpha = 1/3$

The advantage of this averaging method is that it can be used indefinitely. That is, the average will not converge to some value and stay there, as is the case with linear averaging. The average will dynamically respond to the influence of new records and gradually ignore the effects of old records.

Exponential averaging simulates the analog filter smoothing process. It will not reset when a specified averaging number is reached.

The drawback of the exponential averaging is that a large value may embed too much memory into the average result. If there is a transient large value as input, it may take a long time for y[n] to decay. On the contrary, the contribution of small input value of x[n] will have little impact to the averaged output. Therefore, exponential average fits a stable signal better than a signal with large fluctuations.

Peak-Hold

This method, technically speaking, does not involve averaging in the strict sense of the word. Instead, the "average" produced by the peak hold method produces a record that at any point represents the maximum envelope among all the component records. The equation for a peak-hold is

$$y[n] = \mathrm{MAX}_{\mathrm{j}=0}^{\mathrm{N}-1}(x[n-j])$$

Peak-hold is useful for maintaining a record of the highest value attained at each point throughout the sequence of ensembles. Peak-Hold is not a linear math operation therefore it should be used carefully. It is acceptable to use Peak-Hold in auto-power spectrum measurement but you would not get meaningful results for FRF or Coherence measurement using Peak-Hold.

Peak-hold averaging will reset after a specified averaging number is reached.

Time versus Power Spectrum Averaging

Averaging can be applied to either time domain or power spectrum. If you want to reduce the spectral estimation variance, use power spectral averaging. If you want to extract repetitive or periodic small signals from a noisy signal, you can use triggered capture and average them in time domain. Time averaging must be performed with on a triggered event so that the time signal of one average is correlated with other similar measurements. Without time synchronizing mechanism, averaging makes no sense.

Power Spectrum Averaging is also called RMS Averaging. RMS averaging computes the weighted mean of the sum of the squared magnitudes (FFT times its complex conjugate). The weighting is either linear or exponential. RMS averaging reduces fluctuations in the data but does not reduce the actual noise floor. With a sufficient number of averages, a very good approximation of the actual random noise floor can be displayed. Since RMS averaging involves magnitudes only, displaying the real or imaginary part, or phase, of an RMS average has no meaning and the power spectrum average has no phase information.

Table 2 gives a summary of the averaging methods described above.

- Table 2. Summary of Averaging Methods.

Time Averaging	**Power Spectrum Averaging**
No statistical spectral estimate, for deterministic signals only.	Statistical spectral estimate, for signals with random characteristics.
Signal must have periodic components.	Applicable to both pure random and mixed random/periodic signals.
Improve SNR.	Does not improve SNR.
Requires a synchronized trigger in fixed relation to the signals.	Does not require a synchronized trigger.

In CoCo-80, the user can select ***Linear, Exponential*** and ***Peak Hold*** for power spectral averaging and select ***Time Linear*** or ***Time Exponential*** for time domain averaging.

Spectrum Estimation Error

You may wonder how much confidence we should have when we take the spectral measurement. This is an academic topic that can go very deep. First you must classify your signal types. If you are measuring a deterministic signal, with very few averaging, the spectrum estimation can be very accurate. If the signal has a random nature, with partially random, or significant measurement noise, more averaging must be used.

Assume the time data is captured from a stationary random process and we calculate various spectra using window, FFT and averaging techniques, how much we can trust the measured spectra can be measured by a statistical quantity, *standard deviation*. Here are a few useful equations to compute the standard deviation of the spectra when linear averaging is used:

Functions being estimated	**Standard Deviation**
Auto-spectrum Gxx	$\frac{1}{\sqrt{n}}$
Cross-spectrum |Gyx|	$\frac{1}{\lvert Cyx \rvert \sqrt{n}}$
Coherence Function Cyx^2	$\frac{(1-Cyx^2)\sqrt{2}}{\lvert Cyx \rvert \sqrt{n}}$
Frequency Response Function Hyx	$\frac{\sqrt{(1-Cyx^2)}}{\lvert Cyx \rvert \sqrt{2n}}$

where *n* is the average number in linear averaging. The transfer function is computed in the cross-power spectrum method as presented earlier.

Assume a signal is random and has an ***expected*** power spectral density at 0.1 V^2/Hz. The goal of a measurement is to average a few power spectra and to estimate such an expected value. If the average number is 1, meaning, with no average, the standard deviation of the error of such a measurement will be 100%. When we average two frames of auto power spectra, the standard deviation of the error will become $\frac{1}{\sqrt{2}} = 70.7\%$ When the average number is increased to 100, the standard deviation of the error of the reading is 10%. This means that the reading is likely in the neighborhood of (0.1±0.01) V^2/Hz

Now if this signal has a deterministic nature, say a sine wave, the spectral estimation error will only be applied to the random portion, i.e., the noisy portion, of this signal.

Coherence Value Affects Accuracy of Spectra Estimate

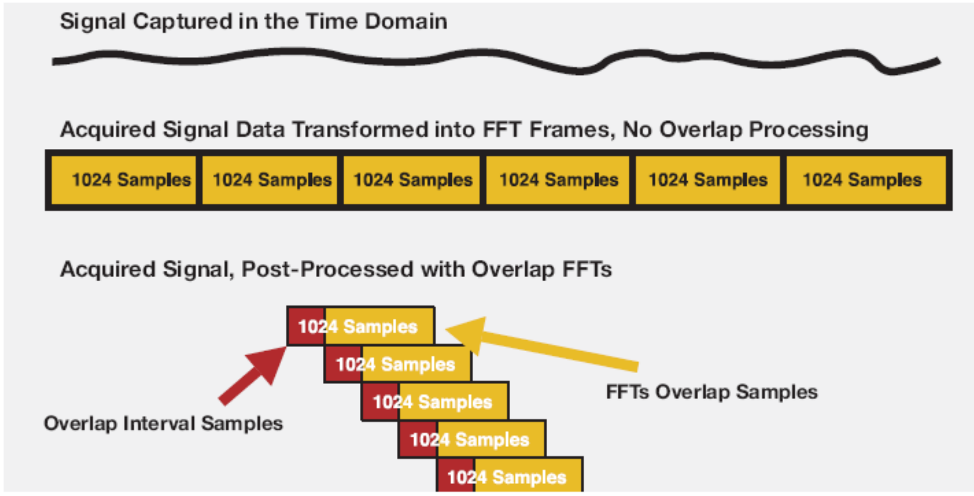
As shown above, the spectral estimation error is related to the coherence value between the input and output. When signal to noise ratio is low, the coherence value will be much less than 1.0 which results in very inaccurate estimation to the FRF and other spectra signals.

To improve the coherence value, the key is to increase the signal intensity of the excitation signal at the frequency range of interest.

One of the unique features of CoCo is that the random signal of its signal source is well-controlled within the analysis frequency band. This design allows the excitation focus its energy between DC and analysis frequency. The FRF estimation with such excitation will be more accurate.

Overlap Processing

To increase the speed of spectral calculation, overlap processing can be used to reduce the measurement time. The diagram below shows how the overlap is realized.



- Figure 96. Illustration of overlap processing.

As shown in this picture, when a frame of new data is acquired after passing the Acquisition Mode control, only a portion of the new data will be used. Overlap calculation will speed up the calculation with the same target average number. The percentage of overlap is called overlap ratio. 25% overlap means 25% of the old data will be used for each spectral processing. 0% overlap means that no old data will be reused.

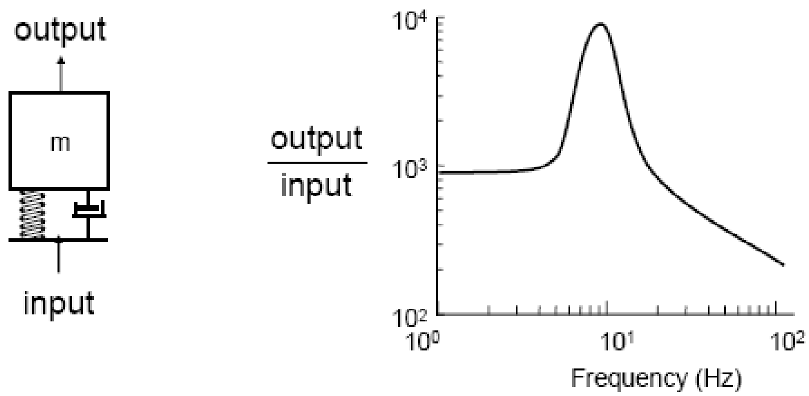
Overlap processing can improve the accuracy of spectral estimation. This is because when a data window is applied, some useful information is attenuated by the data window on two ends of each block. However, it is not true that the higher the overlap ratio the higher the spectral estimation accuracy. For Hanning window, when the overlap ratio is more than 50%, the estimation accuracy of the spectra will not be improved.

Another advantage to apply overlap processing is that it helps to update the display more quickly.

Single Degree of Freedom System

This section briefly discusses the single degree of freedom (SDOF) system as background for the frequency response function and damping estimation methods.

The vibration nature of a mechanical structure can be decomposed into multiple, relatively independent Single-Degree-Of-Freedom systems. Each SDOF system can be modeled as a mass fixed to the ground by a spring and a damper in parallel as shown in Figure 97. The frequency response function (FRF) of this mechanical system is also shown.



■ Figure 97. SDOF system and their frequency response.

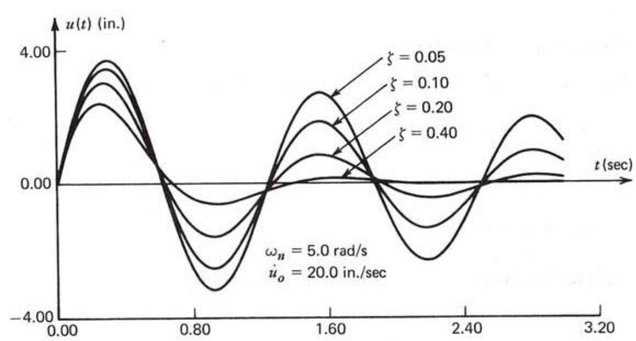
The differential equation of motion for this system is given by

$$m\ddot{x} + c\dot{x} + kx = f(t)$$

The natural frequency ω_n and damping ratio ζ can be calculated from the system parameters as

$$\omega_n^2 = \frac{k}{m}, \ \ and \quad 2\zeta\omega_n = \frac{c}{m}$$

where m is the mass, k is the spring stiffness and c is the damping coefficient.
The natural frequency, ω_n , is in units of radians per second (rad/s). The typical units displayed on a digital signal analyzer are in Hertz (Hz). The damping ratio, ζ, can also be represented as a percent of critical damping – the damping level at which the system experiences no oscillation. This is the more common understanding of modal damping.. Figure 97 illustrates the response of a SDOF system to a transient excitation showing the effect of different damping ratios.

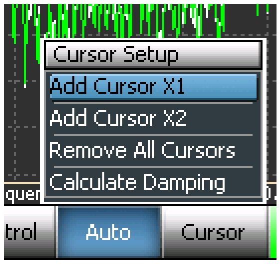


■ Figure 98. Step response of a SDOF system with different damping ratios.

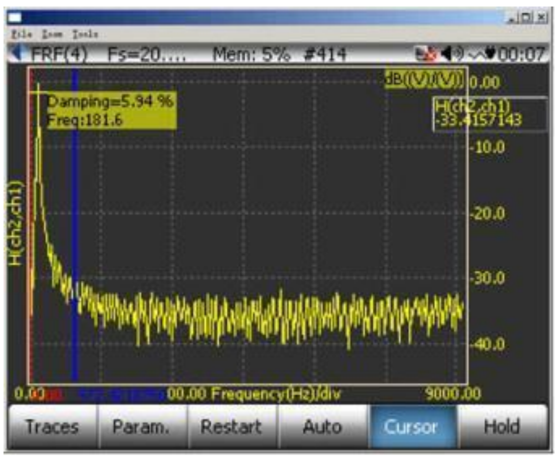
A SDOF system with light damping factor will have longer oscillation in a transient process. This is why the exponential window may be chosen to reduce the leakage effect in its spectral analysis. On the other hand, the exponential window may artificially change the spectral shape so the estimated damping is biased. Since most modern instruments allow can capture very long buffer of data, it is preferred to increase the buffer length instead of applying the exponential window.

Damping factor and the resonant frequency can be calculated by using curve fit method that applies to the FRF signal. Without curve fit, the estimation of damping value can go off significantly especially in a light damping system.

In CoCo or EDM software, the user can use dual cursors to calculate these values, as show in the picture below.



Enable two cursors and cover the peak area, then select ***Calculate Damping*** command.



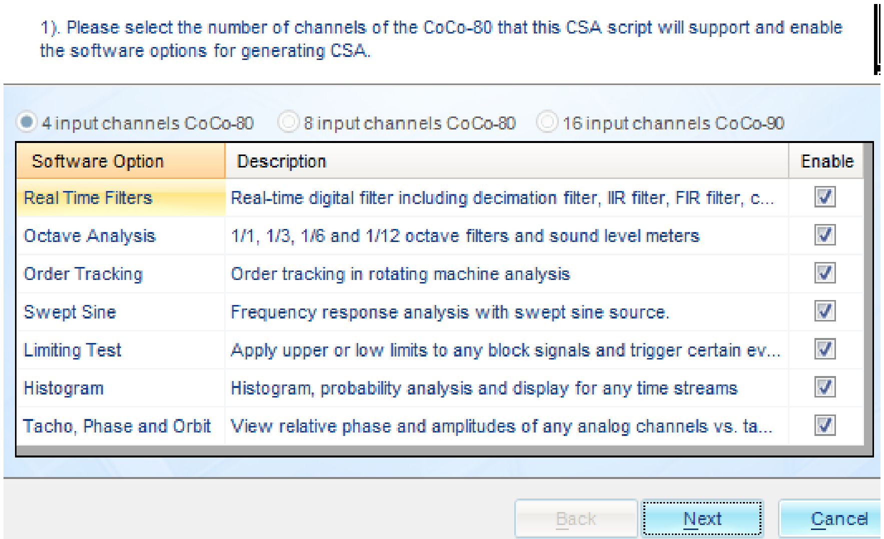
CSA Editor Operation for Spectral Analysis

This section describes the operation of CSA Editor related to FFT based spectral analysis. For general operation of CSA Editor, refer to the CSA User's Manual.

CSA Editor Wizard

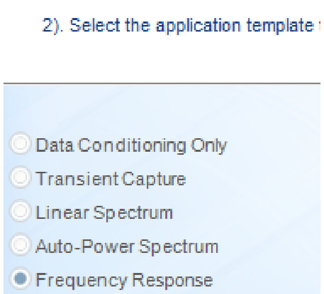
This section summarizes how to create a CSA project for general spectral analysis in the CSA Editor. We strongly recommend that you read the CSA Editor User's Manual to gain more detail information before proceeding with this chapter.

To start, click on the CSA Editor icon in the upper-right corner in EDM and start the CSA Editor. The CSA Editor Wizard dialog box will be displayed.



■ Figure 99. CSA Editor Wizard Software Option selection.

Although it does not show up in the software option list, the FFT based spectral analysis is a the heart of the software and is always installed and enabled in the CSA Editor by default. This is true regardless of whether any other software options are enabled in the display.



■ Figure 100. CSA Editor Wizard template selection.

When you are asked to select one of the templates to use, you must selection one of the following: ***Linear Spectrum***, ***Auto Spectrum*** or ***Frequency Response*** as the template in order to calculate the appropriate spectra. The software will open the CSA Application Group associated with the template that you choose.

Linear Spectrum

This application is the best choice for applying data conditioning to native time streams, transforming time streams into block signals and applying data windowing and FFT calculations.

Auto Power Spectrum

This application is the best choice for applying pply data conditioning to native time streams, transforming time streams into block signals and applying data windowing and FFT to calculate auto spectra. FRF/Coh is also calculated in this application.

Frequency Response

This application is best for applying data conditioning to native time streams, transforming time streams into block signals then applying data windowing and FFT to calculate auto spectra, cross spectra, frequency response and coherence functions.

The software determines many factors based on the template selection. Table 3 shows the availability of measurement quantities for the different templates of basic spectral analysis:

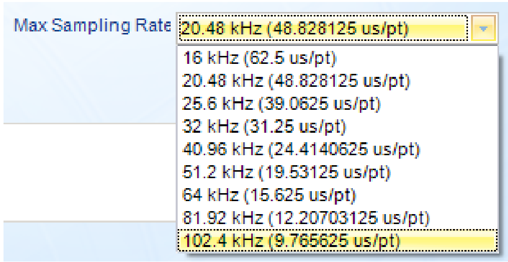
- Table 3. Comparison of measurement quantities for different spectral analysis templates.

CSA Template	Time streams of native channels	Time streams for each data conditioning output	Acquisition Mode, Blocked Time Capture	FFT and Auto-Power Spectra	Cross Spectra	FRF/Coh
Data Conditioning Only	Yes	Yes	No	No	No	No
Transient Capture	Yes	Yes	Yes	No	No	No
Linear Spectrum	Yes	Yes	Yes	Yes	No	No
Auto Power Spectrum	Yes	Yes	Yes	Yes	No	No
Frequency Response	Yes	Yes	Yes	Yes	Yes	Yes

The table illustrates that the *Frequency Response* template contains the most complete measurement quantity set. If you want to measure time streams, block time captures, auto and cross spectra, FRF and coherence all together, select the *Frequency Response* template.

Does this mean that you should always select the *Frequency Response* template because it has all the measurements? The answer is NO. The more measurement quantities that are selected, the more DSP resources that are required. DSP resources refers to both real time computational capability and memory resources. Sometime you have to make a tradeoff between speed, size of the data buffer and the number of measurements taken.

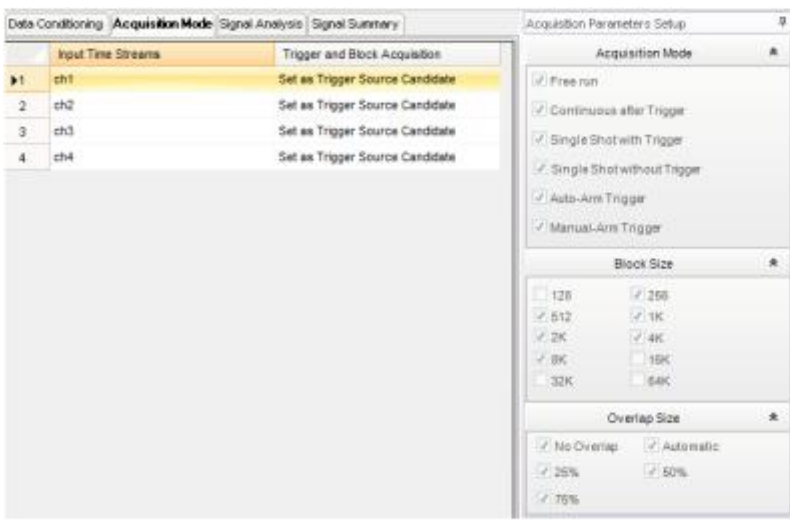
As a rule of thumb, always select the minimum function sets that you need to use. It will save computational resource on DSP.



■ Figure 101. CSA Wizard Template sampling rate selection.

Next, you must choose the ***Maximum Sampling Rate,*** which is a number that controls the highest sampling rate that this CSA project uses. If you allocate too many computational tasks in the following steps, then the sampling rate may have to be reduced. The Editor will give you chances to make the changes later, if necessary. It is best to choose the lowest sampling rate that you know that you will need for your application to conserve DSP resources.

After the CSA Editor Wizard is finished, you may apply various data conditioning modules to the time streams. For details about how to apply the data conditioning modules refer to the User's Manual of CSA Editor. Then go to Acquisition Mode, follow the instructions in the Manuals to set the *trigger source candidate*, *Acquisition Mode*, *Block Size* and *Overlap Size*.



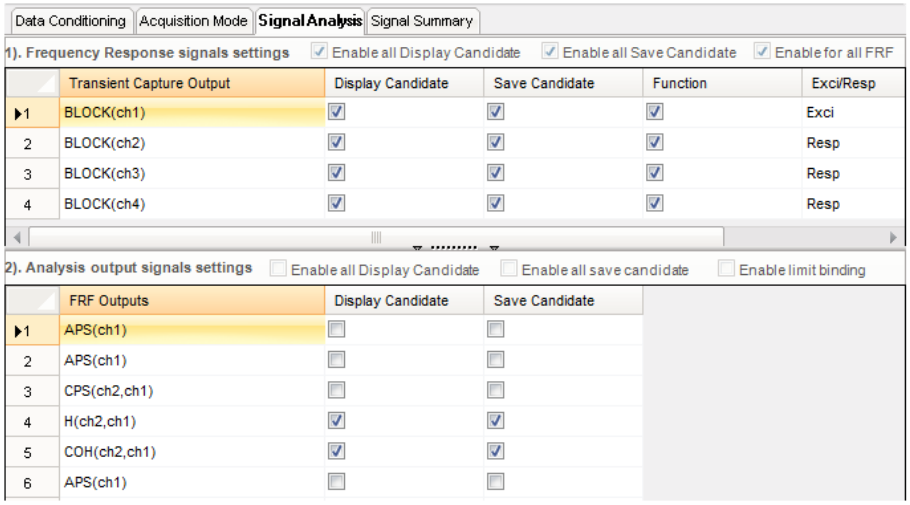
■ Figure 102. CSA Editor Wizard Acquisition Mode selection.

Select the Signals to Compute

The Signal Analysis tab in the CSA Editor is the area to enable which signals will be computed. You can change all the names of the signals. By default, the following signal name conventions are used:

CHn: the native raw data, in the form of a time stream, of nth input channel
BLOCK(chn): the block captured signal of the nth input channel
APS(x): the auto power spectrum of time stream x
CPS(y,x): the cross power spectrum of excitation x and response y
COH(y,x): the coherence function of the excitation x and response y

FRF(y,x): the FRF of excitation x and response y
APS(x): the auto power spectrum of time stream x



■ Figure 103. CSA Editor Wizard Signal Analysis signal selection.

In this setting, the signals set as *Display Candidates* will be selectable for display on CoCo; only the signals that are set as *Save Candidates* will be selectable for Save on CoCo. You should only enable the minimum necessary signals that are required in order to conserve DSP resources and simplify the user interface. Enabling too many signals will make the user interface more complicated to navigate than is necessary.

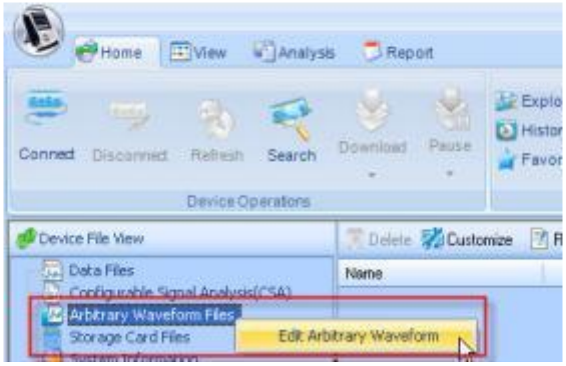
For FRF or cross power computation, you must designate one channel as excitation channel. Once this channel is designated, the other channels will be set as response channels automatically.

For cross channel signals, the default signal names always follow the convention (y,x) where y measures the system response and x the excitation. All signals can be edited in the CSA Editor.

Editing an Arbitrary Waveform

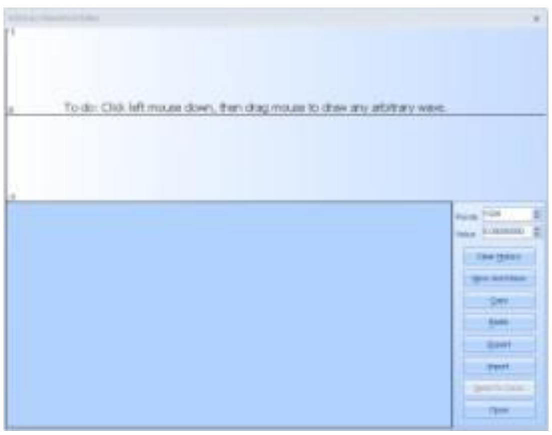
The CoCo output channel can generate an arbitrary waveform which can be programmed on the EDM software. To create an arbitrary waveform you must edit it in the EDM software and then upload it to the CoCo hardware. The waveforms can either be imported from a text file, or drawn by hand.

In the EDM software, right click on "Arbitrary Waveform Files" to open the Arbitrary Waveform Editor



- Figure 104. Edit arbitrary waveform in EDM software.

To draw a waveform by hand click and hold the mouse button then draw the desired Arbitrary waveform on the top half of the window.



- Figure 105. Arbitrary waveform editor in EDM software.

After the waveform is completed, the value of each point can be changed by entering the value in the box.

New Arb Wave - creates a blank pane for another arbitrary waveform.

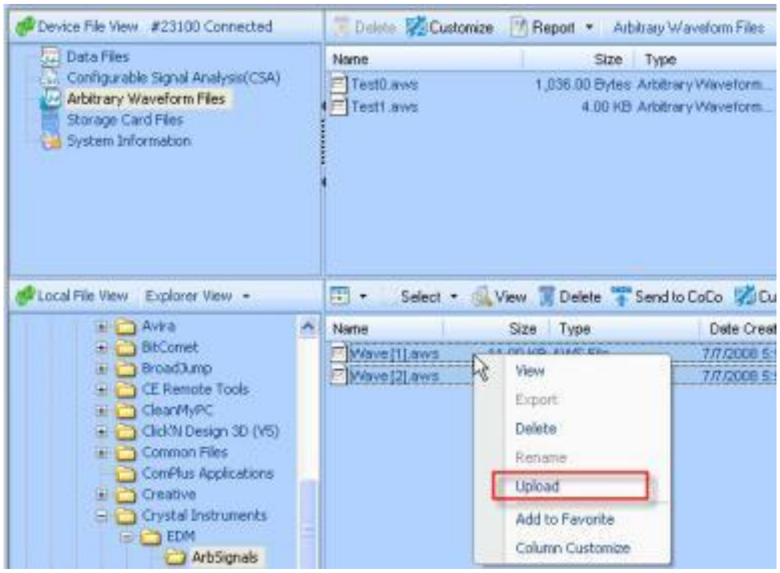
Copy and Paste let you copy the current waveform into a new waveform.

Export lets you save the arbitrary waveform as a text file which can be opened in a text editor or spreadsheet.

Import lets you import an arbitrary waveform that is saved as a text file.

You can also right click on the name of waveform. This opens a pop up menu with copy, paste delete, export and import.

Finally Right click on the arbitrary waveforms and upload them. Then the arbitrary waveforms are ready to be output in CoCo.



■ Figure 106. Upload the arbitrary waveform to the CoCo hardware.

Finally click on the Send to CoCo button to save the arbitrary waveform file on the CoCo hardare.

Validation

After the CSA Wizard is complete and the CSA file is created, connect the host PC to the CoCo device and press the Validate icon to validate the CSA. It may take a few minutes to finish the validation.

The validation process analyzes the CSA file for internal consistency and estimates the required DSA resources required to run the CSA file on the CoCo device.

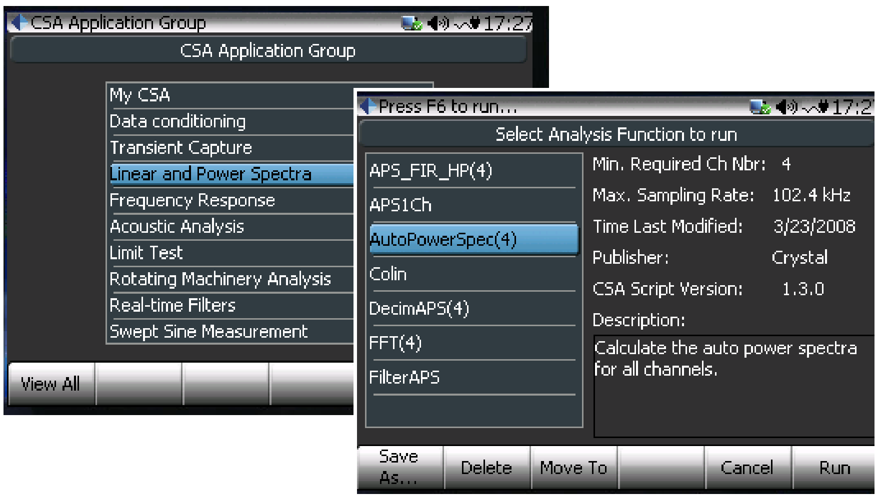
If the Validation passes, then press *Send to CoCo* command in the Validation dialog box to send the CSA project file to CoCo. Alternatively you can manually upload it to CoCo. The CSA uploaded will be classified into different CSA Application Groups based on the template that was used.

CoCo-80 Operation for Spectral Analysis

This section describes the operations of CoCo that are specifically related to the FFT spectral analysis. For general operations of CoCo, refer to previous Chapters of this manual.

Select a CSA Project

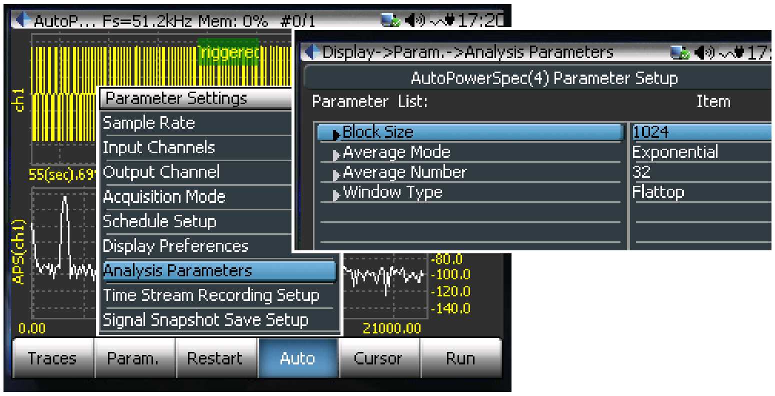
To run a spectral analysis CSA press the Analysis button and select either the Linear and Power Spectra or the Frequency Response Application group, then select one of the CSA spectral analysis projects.



■ Figure 107. CoCo Select Analysis Function selection display.

Set Analysis Parameters for Spectral Analysis

To set the parameters for spectral analysis, press the Param. Button in the signal display window, select Analysis Parameters, then set the parameters.



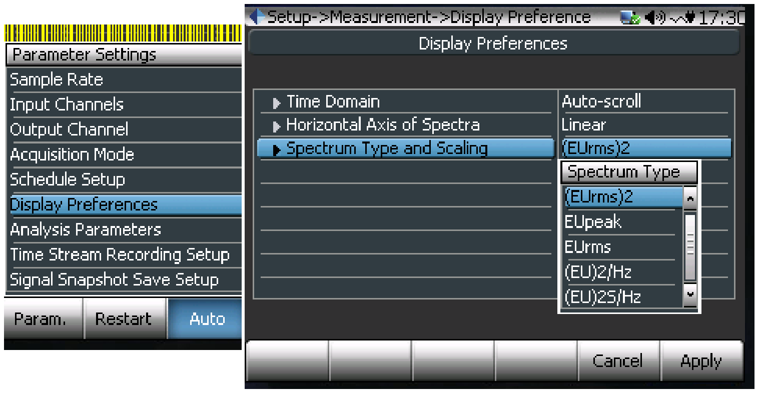
■ Figure 108. CoCo Analysis Parameters selection display.

Block Size: the block size of the time block signals.
Average Mode: Exponential, Linear or Peak Hold applied to frequency domain, power spectra averaging. Time Linear and Time Exponential applies to time domain averaging.
Average Number: the Average number of linear averages. When exponential average is selected as the average mode, 1/(Average Number) is used as the exponential factor.
Window Type: type of data window.

The selection candidates of these fields are defined in the CSA Editor. That is, the maximum sampling rate, maximum number of input channels, etc are defined when you create the CSA. For example if you select a Maximum Sampling Rate of 1 kHz in the CSA Editor then higher sampling rates will not be available on the CoCo device. If a higher sampling rate is required then you must modify the CSA and download it to the CoCo. Although this behavior may seem limiting, it should be noted that it allows the user to choose exactly the analysis functions and optimize the performance of the CoCo device to suit your specific needs and is one of the unique features of the CoCo system.

Set the Spectrum Type

To set the Spectrum type, select the *Display Preferences* under the *Param.* Button and choose from the list.

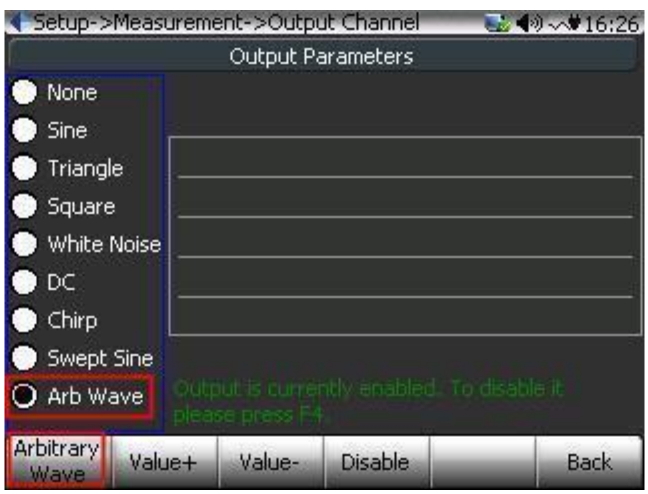


■ Figure 109. CoCo. Spectrum Type selection display.

Set the Output Channel Parameters

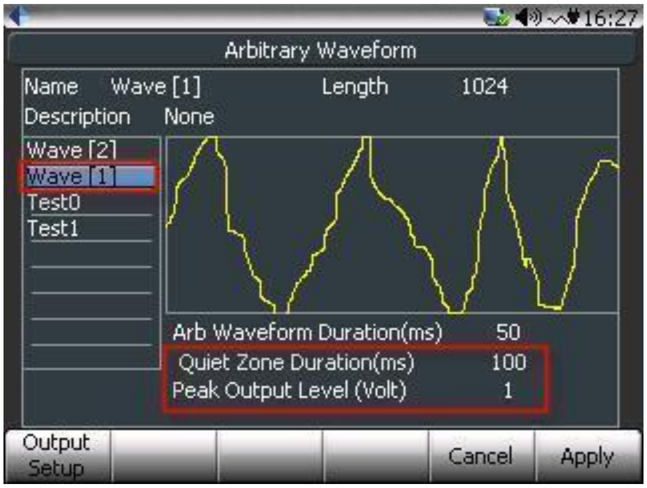
To enable the output channel as a function generator select Output Channel from the Param. Button. Next select the waveform. To set the amplitude and frequency and other parameters move the cursor to the parameter and press the Enter Button to edit the parameter.

To select an arbitrary waveform first select Arb Wave as the output type, then press the Arbitrary Wave Button to choose from all the wave files that are loaded on the CoCo.



■ Figure 110. Output parameters.

Select the waveform from the list on the left and use the Arrow Buttons to move the cursor to the quiet zone, duration and peak output level settings on the right. Press the Enter Button to edit any of these parameters.

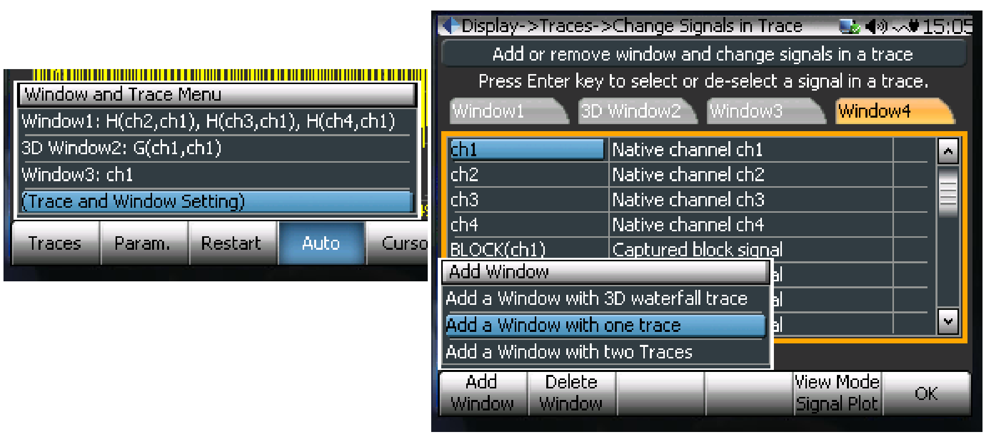


■ Figure 111. Arbitrary waveform settings.

To start the output channel press the Enable Button.

Create Display Window and Set up the Trace

To create a display window select Trace and Window Setting under the Param. Button. Use the soft buttons to add or delete a window, clear the signals from the current window, select all signals or change the view mode to show a numerical value. Change the signals in each window by selecting a specific window tab with the left or right buttons then editing the window settings.



■ Figure 112. CoCo Add Window display.

dB and Linear Magnitude

Most often, amplitude or power spectra are shown in the logarithmic unit decibels (dB). Using this unit of measure, it is easy to view wide dynamic ranges; that is, it is easy to see small signal components in the presence of large ones. The decibel is a unit of ratio and is computed as follows.

$$dB = 10\log_{10}(Power/Pref)$$

where Power is the measured power and P_{ref} is the reference power.

Use the following equation to compute the ratio in decibels from amplitude values.

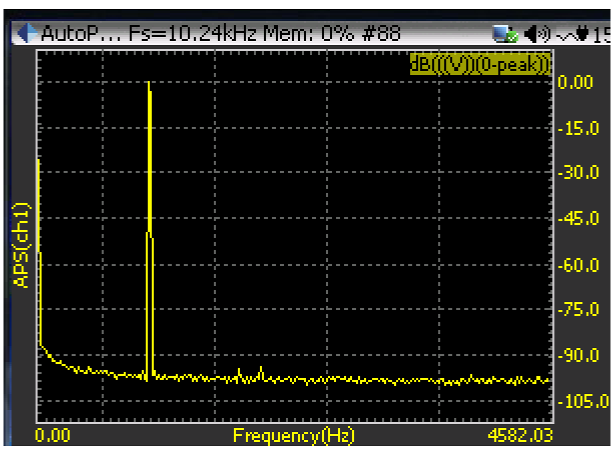
$$dB = 20\log_{10}(Ampl/A_{ref})$$

where Ampl is the measured amplitude and A_{ref} is the reference amplitude.

When using amplitude or power as the amplitude-squared of the same signal, the resulting decibel level is exactly the same. Multiplying the decibel ratio by two is equivalent to having a squared ratio. Therefore, you obtain the same decibel level and display regardless of whether you use the amplitude or power spectrum.

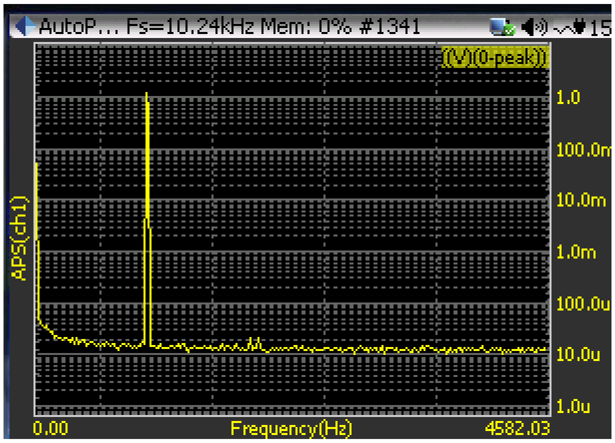
As shown in the preceding equations for power and amplitude, you must supply a reference for a measure in decibels. This reference then corresponds to the 0 dB level. Different conventions are used for different types of signals. A common convention is to use the reference 1 Vrms for amplitude or 1 Vrms squared for power, yielding a unit in dBV or dBVrms. In this case, 1 Vrms corresponds to 0 dB. Another common form of dB is dBm, which corresponds to a reference of 1 mW into a load of 50 Ω for radio frequencies where 0 dB is 0.22 Vrms, or 600 Ω for audio frequencies where 0 dB is 0.78 Vrms.

The picture below shows a sine wave with 1V amplitude displayed in dB. Because the reference is 1Vpk , it shows the peak value of this sine wave as 0dB.



■ Figure 113. Show a 1Vpk sine signal in frequency domain with dB scaling.

Another display format is called Log, or LogMag. The Log display shows the signal scaled logarithmically with the grid values and cursor readings in actual engineering value. The picture below shows the same signal in LogMag.



■ Figure 114. A 1Vpk sine signal in frequency domain with LogMag scaling.

When dB reference is not specified, the dB reference is 1.0 engineering unit. In acoustics application, the dB reference for the sound pressure value is set to 20uPa. The same input signal will result in different dB readings when dB reference is changed.

Set Acquisition Mode

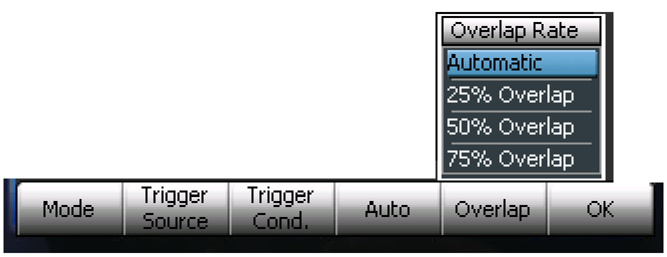
The appropriate acquisition mode should be set to transform the time streams into blocks. The details of acquisition mode for transient capture are described in the next chapter, *Transient Capture and Hammer Test.*

For frequency analysis that use stable and continuous excitation signals, use either *Free-Run* or *Continuous after Trigger mode* in the Acquisition Mode selection.

For details about setting the acquisition mode, refer to the Basic Operation of CoCo-80.

Set Overlap Ratio

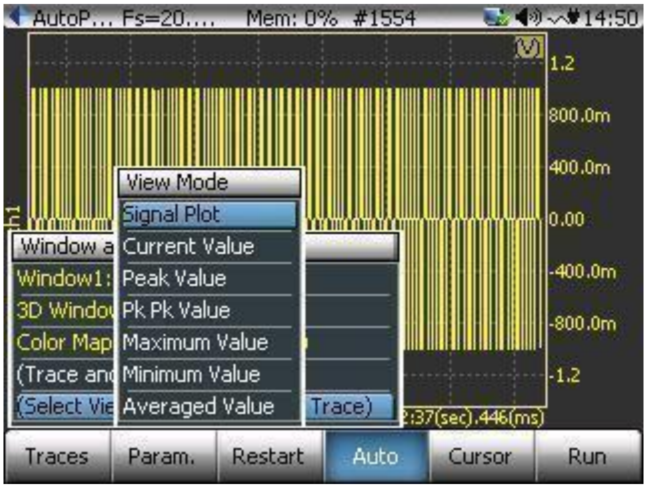
The overlap ratio is set in the Acquisition Mode. The overlap ratio setting will only be effective when the Mode is selected as Free-run or Continuous after Trigger. For triggered transient capture, there will be gaps between frames and an Overlap Ratio can not be applied.



■ Figure 115. Overlap Rate selection.

Select the View Mode

The View Mode defines how the data will be displayed on the screen. To change the view mode select View Mode for Current Trace under the Param. Button.



■ Figure 116. View mode.

Signal Plot displays a graph of the plot vs. time or frequencies.

Current Value shows a numerical display of the current value of the signal.

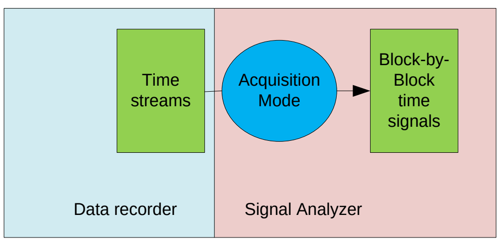
Peak Value, Pk Pk Value, Maximum Value, Minimum Value and Averaged Value show a numerical display of the results of the current data block.

5. TRANSIENT CAPTURE AND HAMMER TESTING

This chapter will demonstrate how to use CoCo to conduct hammer testing. Hammer testing refers to impact or bump testing that is conducted using an impact hammer to apply an impulsive force excitation to a test article while measuring the response excitation from an accelerometer or other sensor. This type of measurement is a transient event that usually requires triggering, averaging and windowing. First, let's briefly review the Transient Capture function on CoCo.

Transient Capture

Transient Capture is one of the most common used functions for dynamic data acquisition. In CoCo the Transient Capture is implemented by setting up the Acquisition Mode. Acquisition Mode defines how to transform the time streams into block by block time signals. It sets the trigger and the overlapping processing. Before the Acquisition Mode stage, the instrument acts as a data recorder while after the Acquisition Mode, it is acts as a signal analyzer.



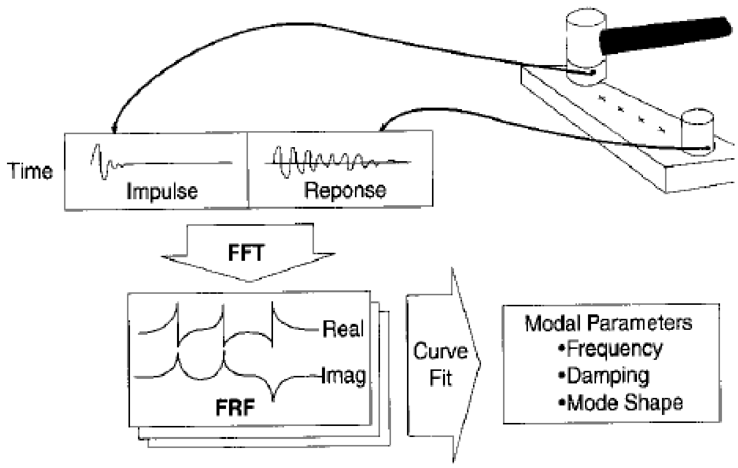
■ Figure 117. Transient capture operation on CoCo.

Besides Acquisition Mode, you must first enable at least one time stream as a trigger candidate in the CSA Editor. Trigger candidates are those time streams that can be selected as a trigger source. The names of these trigger candidates will be passed to the CoCo. During runtime, one of the trigger source candidates must be selected as the trigger source.

Impact Hammer Testing

Typically impact hammer testing is conducted with a signal analyzer to measure FRFs of the device under test. The FRFs can be used to determine the modal properties of the device such as the natural frequencies and damping ratios. In addition the data can be exported to third party modal analysis software to compute mode shapes.

An impact hammer test is the most common method of measuring FRFs. The hammer imparts a transient impulsive force excitation to the device. The impact is intended to excite a wide range of frequencies so that the DSA can measure the vibration of the device across this range of frequencies. The bandwidth or frequency content of the excitation input depends on the size and type of impact hammer that is used. The dynamic force signal is recorded by the DSA. After the impact, the device vibrations are measured with one or more accelerometers or other sensor and recorded by the DSA. The DSA then computes the FRF by comparing the force excitation and the response acceleration signals. Impact testing is depicted in Figure 118.

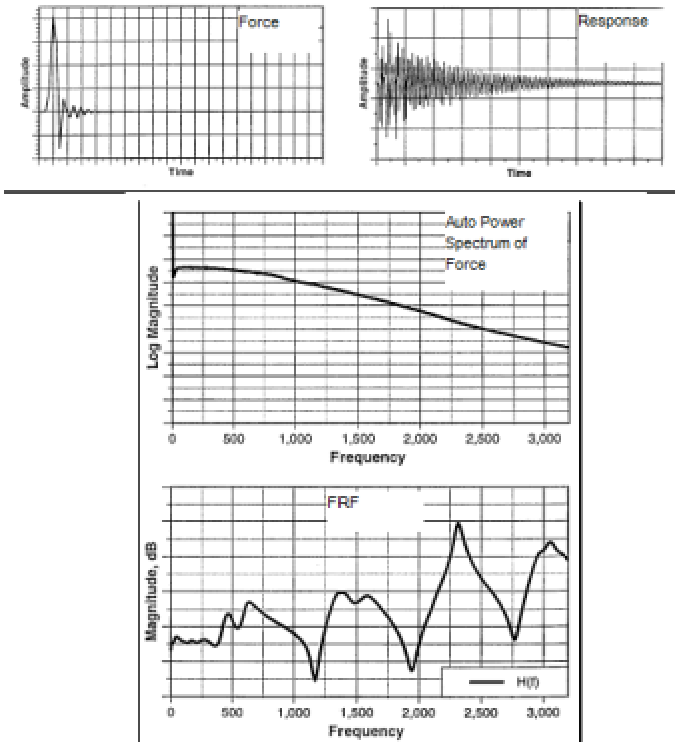


■ Figure 118. Illustration of a typical impact test and signal processing.

The following equipment is required to perform an impact test:

1. An impact hammer to excite the structure. With CoCo we recommend using an impact hammer with IEPE output, which allows the hammer to be connected directly to the analyzer without extra signal conditioning.
2. One or multiple accelerometers that are fixed on the structure. Again, IEPE accelerometers can be used directly with CoCo without additional signal conditioning.
3. Coco Signal Analyzer
4. The CoCo can be used to extract the resonance frequencies and damping factors of the structure. In addition third party software can be used to extract modal shapes and animate the vibration modes.

A wide variety of structures and machines can be impact tested. Of course, different sized hammers are required to provide the appropriate impact force, depending on the size of the structure; small hammers for small structures, large hammers for large structures. Realistic signals from a typical impact test are shown in Figure 10.



- Figure 119. Typical impact test data. Top left shows excitation force impulse time signal, top right shows response acceleration time signal and bottom shows FRF spectrum.

Impact Test Analyzer Settings

The following settings are used for impact testing.

1. **Trigger Setup** including trigger level and pre-trigger delay are used to capture the transient signal for FRF processing. It is important to capture the entire short transient signal it in the sampling window of the FFT analyzer. To insure that the entire signal is captured, the analyzer must be able to capture the impulse and impulse response signals *prior to the occurrence of the impulse* with the pre-trigger.

2. **Force & Exponential Windows**. Two common time domain windows that are used in impact testing are the force and exponential windows. These windows are applied to the signals after they are sampled, but before the FFT is computed in the analyzer.

   The *force window is used to remove noise* from the impulse (force) signal. Ideally, an impulse signal is non-zero for a small portion of the sampling window, and zero for the remainder of the window time period. Any non-zero data following the impulse signal in the sampling window is assumed to be measurement noise. CoCo has a unique way to implement the force window. This was discussed in the data windowing section in the previous chapter.

   The exponential window is applied to the impulse response signal. The *exponential window is used to reduce leakage* in the spectrum of the response.

3. **Accept/Reject**: Because accurate impact testing results depend on the skill of the operator, FRF measurements should be made with **averaging**, a standard capability in all modern FFT

analyzers. FRFs should be measured using at least 4 impacts per measurement. Since one or two of the impacts during the measurement process may be bad hits (too hard causing saturation, too soft causing poor coherence or a double hit causing distortion in the spectrum), an FFT analyzer designed for impact testing should have the ability to accept or reject the result of each impact after inspecting the impact signals. An *accept/reject* capability saves a lot of time during impact testing since you don't have to redo all measurements in the averaging process after one bad hit.

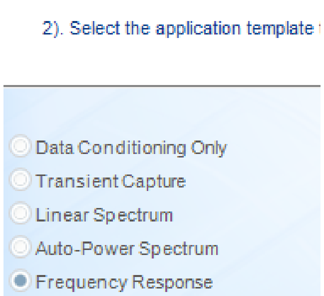
4. **Modal Damping Estimation**. The *width of the resonance peak* is a measure of modal damping. The resonance peak width should also be the same for all FRF measurements, meaning that *modal damping is the same in every FRF measurement*. A good analyzer should provide an accurate damping factor estimate. CoCo uses a curve fitting algorithm to estimate the damping factor. The algorithm reduces the inaccuracy caused by the poor spectrum resolution or noise.

5. **Modal Frequency estimation**. The analyzer must provide capability of estimating the resonance frequencies.

CSA Editor Operation for Transient Capture

This section describes the operation of the CSA Editor that is related to transient data capture and impact testing. For general operation of the CSA Editor, refer to the CSA User's Manual.

In the CSA Editor Wizard, select one of the templates that contain the data transient capture capability. Transient Capture is available in the following templates: *Transient Capture*, *Linear Spectrum*, *Auto-Power Spectrum* and *Frequency Response*.

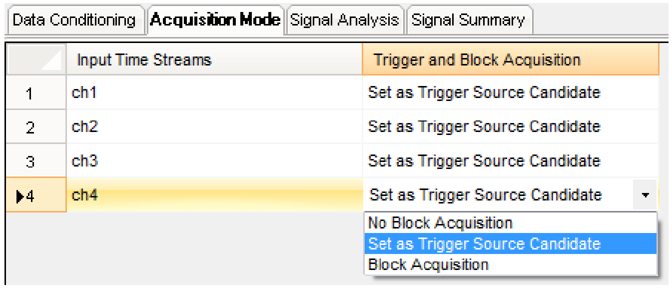


■ Figure 120. CSA Editor Wizard template selection.

The *Data Conditioning Only*, *Octave Analysis*, *Order Tracking* and *Swept Sine* templates do not use regular transient capture to acquire the data.

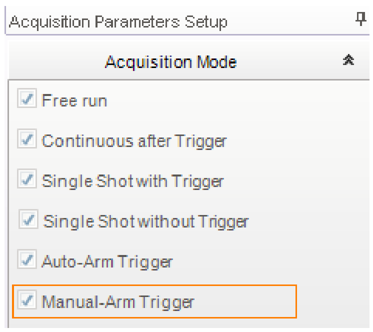
After the Wizard is finished and data conditioning is applied as necessary, go to Acquisition Mode to define the Trigger and Block Acquisition settings and the Acquisition Mode.

At least one input time stream must be selected as the *Set as Trigger Source Candidate*. The items enabled as trigger source CAN BE SELECTED as trigger source when this CSA project runs on CoCo. Notice that the time streams after the data conditioning can also be selected as trigger candidates. Also note that items not specified as trigger candidates can not be used as triggers on the CoCo without modifying the EDM script and downloading it to the CoCo.



- Figure 121. CSA Editor Wizard Acquisition Trigger and Block Acquisition settings.

To acquire the data block by block, enable the appropriate acquisition modes. For a hammer test, you must enable *Manual-Arm Trigger* which will activate "*Accept/Reject*" logic when a block of signals are captured on CoCo.



- Figure 122. Acquisition Mode selection.

You must validate and upload the CSA to CoCo after editing.

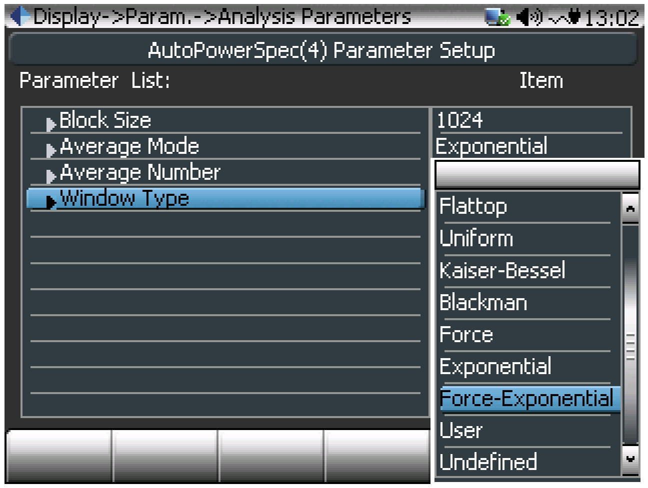
CoCo Hardware Operation for Transient Capture

Select a CSA Project

This section discusses the CoCo settings that are specifically related to transient capture applications. For a complete explanation of these settings refer to the Basic CoCo Operation section. To run a Transient Capture CSA press the Analysis button and select a CSA Application Group that includes transient capture option. These include: Transient Capture, Line and Power Spectrum, and Frequency Response. Then choose an Analysis Function from the CSA files on the CoCo to run.

Analysis Parameters - Window Type

First you must specify the Analysis Parameters under the Param. Button. Select the averaging mode, averaging number and data window type. Transient Capture commonly uses the Force, Exponential or combination Force-Exponential data window function. Press the Apply button to accept the settings.

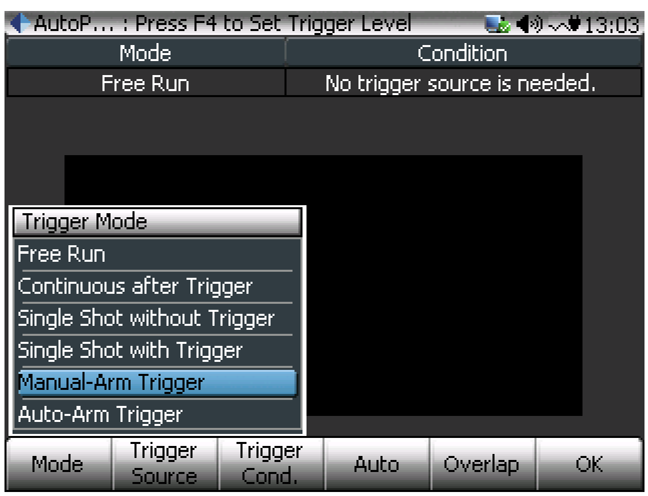


■ Figure 123. Select the data window type for transient capture.

Acquisition Mode

Next select Acquisition Mode under the Param. Acquisition Mode controls how the data is acquired and under what conditions. It includes setting the trigger mode, trigger source, level, conditions and overlap.

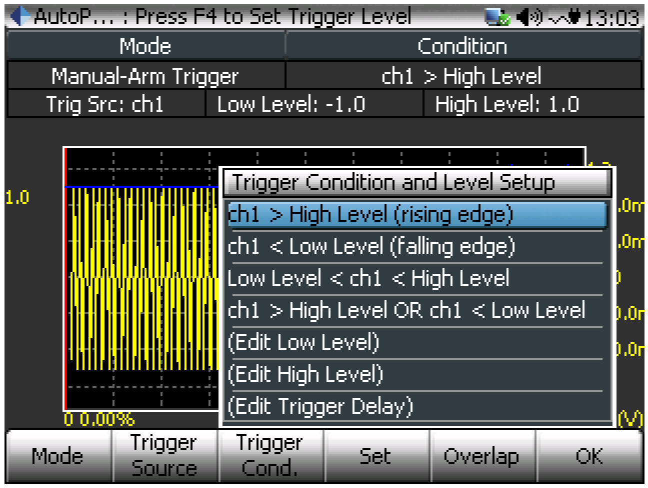
Select Trigger Mode using the Mode Button. Transient capture projects such as impact hammer tests typically use Manual-Arm Trigger or Auto-Arm Trigger. Auto-Arm automatically accepts the data frame into the average and prepares the trigger for the next signal. Manual-Arm provides a graphical display of the data and allows you to accept or reject the frame into the average.



■ Figure 124. Trigger Modes for Transient Capture.

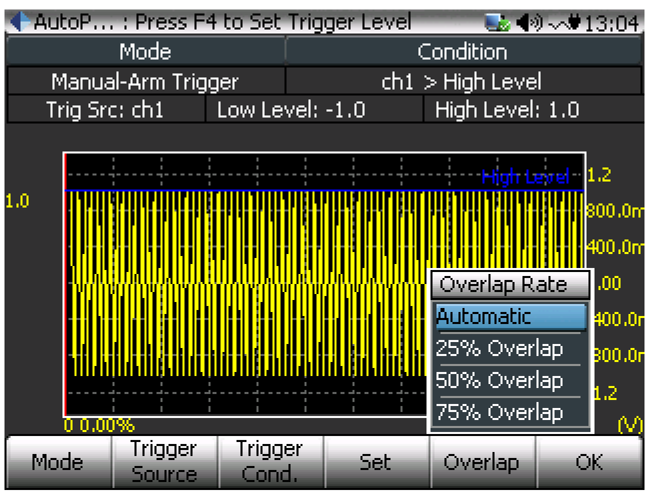
Trigger Source defines which signal to use as the trigger source. Only signals specified in the CSA script are available as trigger sources. If a signal is not available then it can be added as a trigger source by editing the CSA file and downloading it to the CoCo hardware.

The Trigger Condition and Level Setup define the conditions that will trigger the acquisition. You can also edit the high and low level and the trigger delay. Alternatively you can change the level settings with the up and down arrow buttons.



■ Figure 125. Trigger conditions and level setup for transient capture.

Overlap defines the amount of overlap between frames for averaging to reduce the time required to acquire a large number of averages.

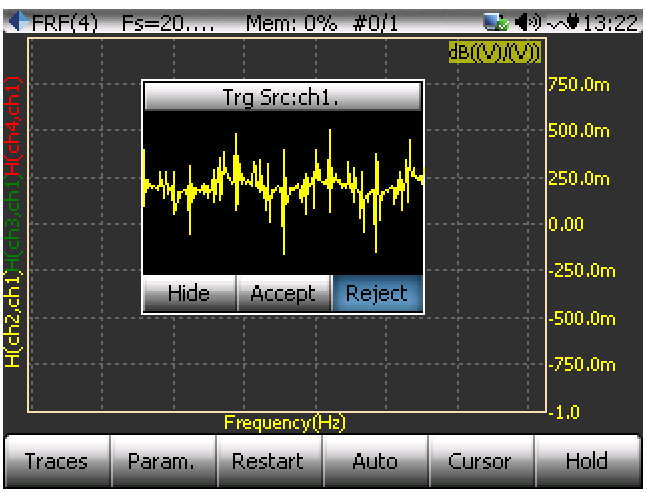


■ Figure 126. Overlap rate for transient capture.

After the Parameter Settings are specified the CoCo begins to wait for a trigger event. A window displays the time elapsed before a trigger event is detected.

■ Figure 127. Waiting for trigger message.

When a trigger event is detected and Manual-Arm Trigger is selected, then a small window will show the data frame and give you the option to accept or reject the data. Accept will include the frame into the average and then ask you to proceed to the next trigger by pressing the Next button. Reject will discard the frame, not include it in the average and return to the waiting for trigger mode. If Auto-Arm Trigger mode is selected then the system will automatically return to the wait mode after each trigger event with no user intervention.



■ Figure 128. Accept/Reject display for transient capture.

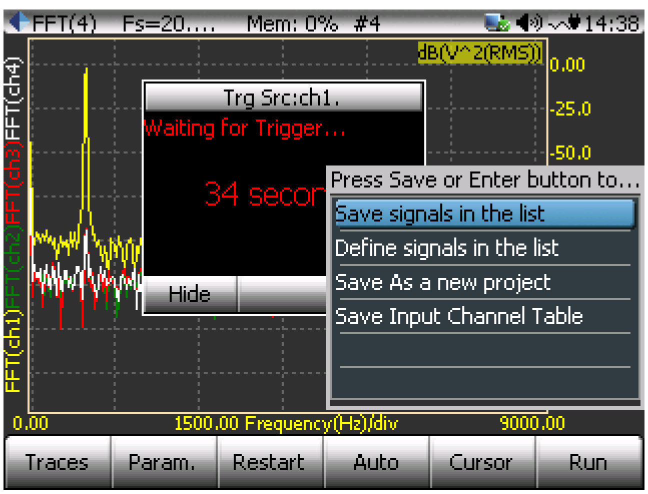
The frame average number is displayed in the status bar to help you monitor how many averages have been recorded. When the averaging mode is set to linear and you reached the averaging number you are prompted to restart a new test by pressing the Run Button.

When the averaging mode is set to exponential then new frames will be acquired and included in the average until you press the Hold Button. The system does not stop when the average number reaches the averaging number. The averaging number only defines the behavior of the averaging function. Exponential averaging is intended for continuous averaging to help observe how a signal changes over time or converges to a mean.

■ Figure 129. When averaging is complete you can restart a new test with the Run Button.

Save Averaged Data

Data can be saved at any time by pressing the Save Hardware Button. This opens a menu with several options. Press the Save Button again to save the signals in the save list. This can be done in the middle of an average or at the end.



■ Figure 130. Save data by pressing the Save hardware button.

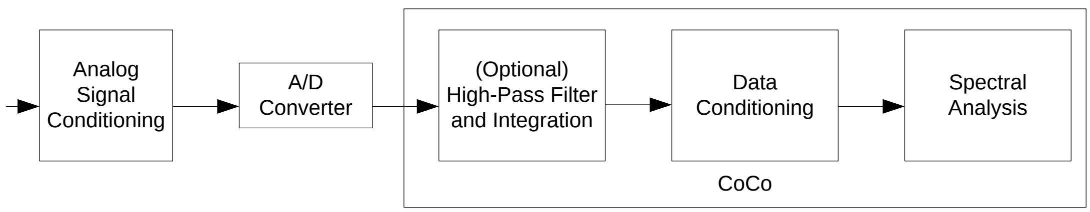
Select Define signals in the list to add or remove signals from the save list or setup automatic data save feature.

6. BUILT-IN DIGITAL INTEGRATION AND FILTERING

Introduction to Digital Integration

Ideally a measurement is made using a sensor that directly measures the desired quantity. For example an accelerometer should be used to measure acceleration, a laser velocimeter pr velocity pickup should be used to measure velocity and an LVDT should be used to measure position. However since position, velocity and acceleration are related by the time derivatives it should be possible to measure an acceleration signal and then compute the velocity and position by mathematical integration. Alternatively you can measure position and compute velocity and acceleration by differentiating. The integration can be performed at the analog hardware level or at the digital level.

The CoCo provides a means to digitally integrate or double integrate the incoming signals. The integration module fits into the very first stage after data is digitized, as shown below:



There are several issues to address in such implementation:

1. The integration and double integration algorithm has to be accurate enough and it must find a way to reduce the effects of a DC offset. A tiny initial value, offset in the measurement or temperature drift before the integration, may result in a huge value after single or double integration. This DC effect can be removed using a high-pass filter.

2. The initial digital signal must have a high signal to noise ration and high dynamic range. The integration process in essence will reduce the high frequency energy and elevate the low frequency components. If the original signals do not have good signal noise ratio and dynamic range, the signals after integration and double integration will have too much noise to use. The noise will corrupt the integrated signal.

3. The instrument must be able to set two different engineering units: one engineering unit for the input transducer and a second engineering unit after the integration. For example, first the instrument must provide a means to set the sensitivity of the sensor, say 100*mV/g*. After the double integration the instrument must have the means to set the engineering unit to a unit that is compatible with the integration such as *mm*.

The CoCo instrument handles these three issues effectively so you can get reliable velocity or displacement signals from the acceleration measurement, or displacement signals from the velocity measurement. The CoCo hardware has a unique design to provide 130dB dynamic range in its front-end measurement. The signals with high dynamic range will create better results after digital integration.

Since such build-in integration is conducted in the time domain before any other data conditioning or spectral analysis, the time streams generated after the digital integration can be treated in the same way as other time streams. They can be analyzed or recorded.

CoCo also provides differentiation and double differentiation to calculate the acceleration or velocity from velocity or displacement transducers. Differentiation is not as common as integration.

It must be noticed that the displacement after double integration to the acceleration is not the same as that measured by a proximity probe. A proximity probe measures the relative displacement between an moving object to the fixed coordinates seated by the probe. The accelerometer and its integration value can only measure the movement of the moving object against the gravity field.

Sensor Consideration

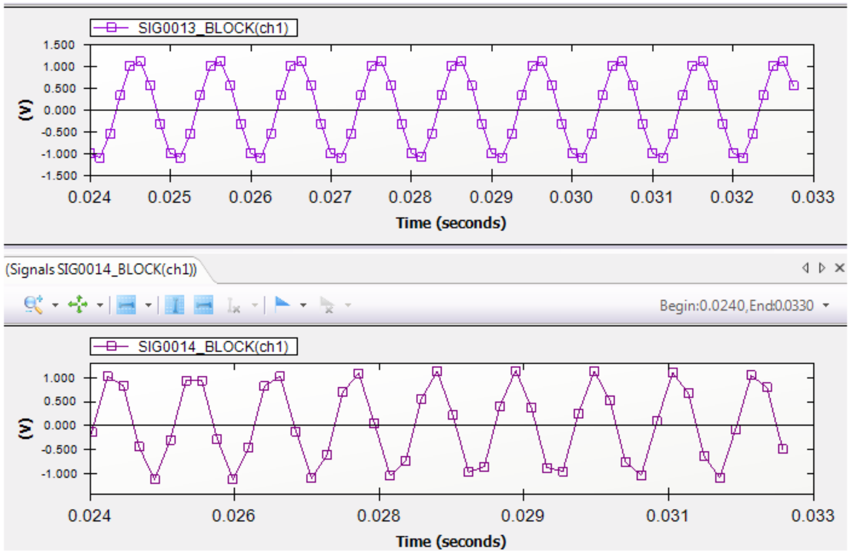
Accelerometer signals that are non-dynamic, non-vibratory, static or quasi-static in nature (low acceleration of an automobile or flight path of a rocket) are typically integrated in the digital domain, downstream of the signal conditioner. Piezoelectric and IEPE accelerometers are commonly used to measure dynamic acceleration and, therefore, dynamic velocity and displacement. They should not be used to measure static or quasi-static accelerations, velocities, or displacements because the IEPE includes analog high pass filtering in the sensor conditioning that cuts out any low frequency signal. At frequencies approaching 0 Hz, piezoelectric and IEPE accelerometers cannot, with the accuracy required for integration, represent the low frequency accelerations of a test article.

When this slight inaccuracy is integrated in order to determine velocity and displacement, it becomes quite large. As a result, the velocity and displacement data are grossly inaccurate. A piezoresistive or variable-capacitance accelerometer is a better choice for low frequency signals and for integration. These types of sensors measure accelerations accurately at frequencies approaching 0 Hz. Therefore the integration calculation of velocity and position can be used to produce accurate results.

Calculation Errors in Digital Integration

Two types of calculation errors can be caused by digital integration: low sampling rate and DC offset.

The sampling rate of a signal must be high enough so that the digital signal can accurately depict the analog signal shape. Some people may think that according to the Nyquist sampling theorem as long as the sampling speed is more than twice of the frequency content of the signals before the integration, the integration results should be acceptable. This is not true. Satisfying the Nyquist frequency only ensures an accurate estimate of the frequency of a measurement. Integration error can still occur of a signal is not sampled at more than twice the signal frequency. Figure 131 shows a 1kHz sine wave sampled at 8kHz and 5.12kHz.

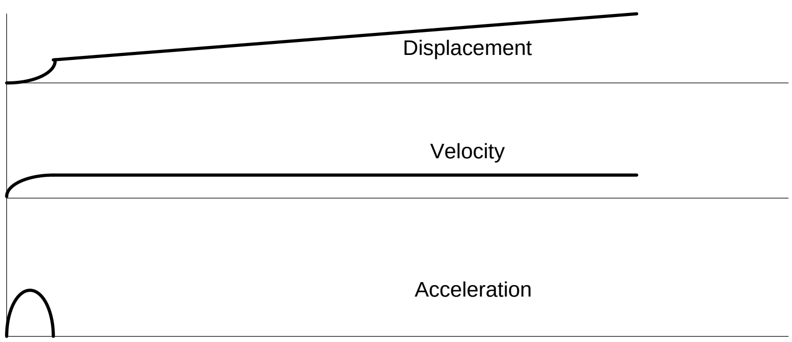


■ Figure 131. A 1 kHz sine wave sampled at 8 kHz (top) and also sampled at 5.12 kHz (bottom).

It is clear that the higher the sampling frequency, the closer this digitized signal is to the true analog waveform. When the sampling rate is low, the digital integration will have significant calculation error. For example the 5.12 kHz sampled signal is not symmetric about 0 volts so the integration will drift and a double integration may grow with accumulated error very fast.

In general, you should use a sampling rate 10 times higher than the frequency content that is of interest in the signal when you apply numerical integration.

DC offset is the second type of digital integration error and can be more severe. It is caused by any measurement error before integration and may result in huge amplitude errors after the integration. Figure 132 shows how a small measurement error in acceleration will create a constant DC offset in the acceleration integrated to compute velocity and result in a drift and eventually an infinite large magnitude of displacement after double integration.



■ Figure 132. A small error in acceleration results in a DC offset in velocity and a huge drift in displacement.

Of course, the computed velocity and displacement signals are unrealistic. They are artifacts of the integration errors. In order to remove such a problem caused by inaccurate measurement and digital integration, a high pass filter can be applied before or after the integration. It should be noted that the high-pass filter will distort the waveform shape to some extent because it alters the low frequency content of the signal. However this effect must be tolerated if numerical integration is used.

Digital High-Pass Filter

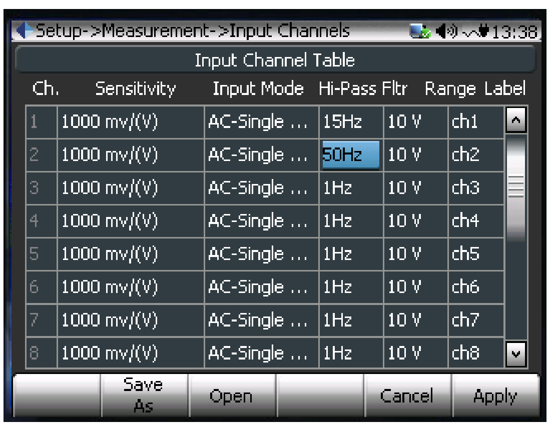
The most effective way to remove the DC drift effect as described above is to apply a high pass digital filter to the continuous time streams. In CoCo, a unique algorithm is realized so that even the data is sampled at high rate, the high pass filter can still achieve very low cutoff frequency.

The filter cutoff frequency is specified at -3dB attenuation.

To remove unwanted signals at or near DC, please set up the cutoff frequency of the digital high-pass filter as high as possible as long as it won't chop off useful frequency content of your interest.

To give an example, if you are not interested in any frequency less than 20Hz, then you can set the cutoff frequency to approximately 10Hz. With this setting, the amplitude attenuation at 20Hz will be less than 1dB.

The following picture shows that ch1 sets the high-pass filter at 15Hz and ch2 at 50Hz. Others are at 1Hz.


Setup->Measurement->Input Channels 13:38
Input Channel Table

Ch.	Sensitivity	Input Mode	Hi-Pass Fltr	Range	Label
1	1000 mv/(V)	AC-Single ...	15Hz	10 V	ch1
2	1000 mv/(V)	AC-Single ...	50Hz	10 V	ch2
3	1000 mv/(V)	AC-Single ...	1Hz	10 V	ch3
4	1000 mv/(V)	AC-Single ...	1Hz	10 V	ch4
5	1000 mv/(V)	AC-Single ...	1Hz	10 V	ch5
6	1000 mv/(V)	AC-Single ...	1Hz	10 V	ch6
7	1000 mv/(V)	AC-Single ...	1Hz	10 V	ch7
8	1000 mv/(V)	AC-Single ...	1Hz	10 V	ch8

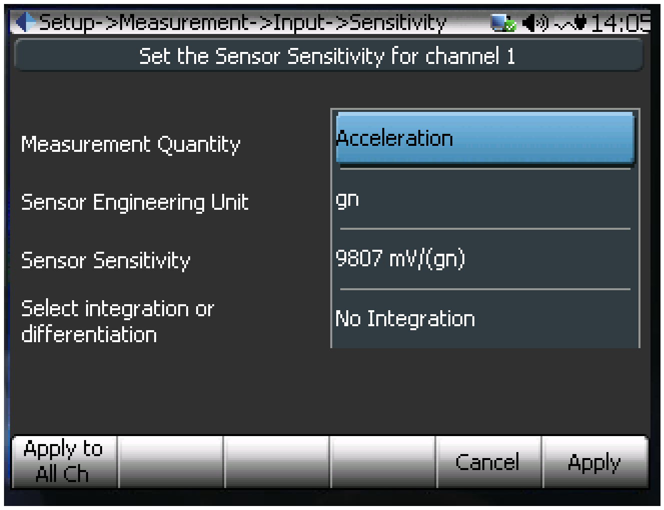
Save As | Open | Cancel | Apply

CoCo-80 Operation

Integration can be enabled in the Input Channel table. In CoCo, to set up the built-in integration or double integration, you must set two engineering units. The first one is for sensor sensitivity; the second for the engineering unit after the integration or differentiation. For example, you can choose either *g* or m/s^2 as the engineering unit used for the accelerometers. After the double integration, you can choose one from the list of *meter*, *cm*, *mm* or other displacement units for displacement presentation.

Example:

First select the Acceleration or Velocity in the input channel table:

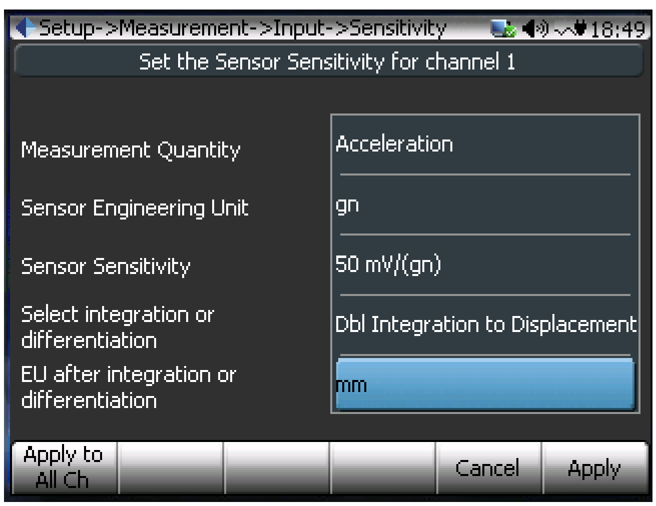


■ Figure 133. Sensor sensitivity without integration.

Then select the engineering unit of the sensor to be used and set its sensitivity.

Under the item of Selection integration or differentiation, select appropriate item. The example shows *Double Integration to Displacement* is selected

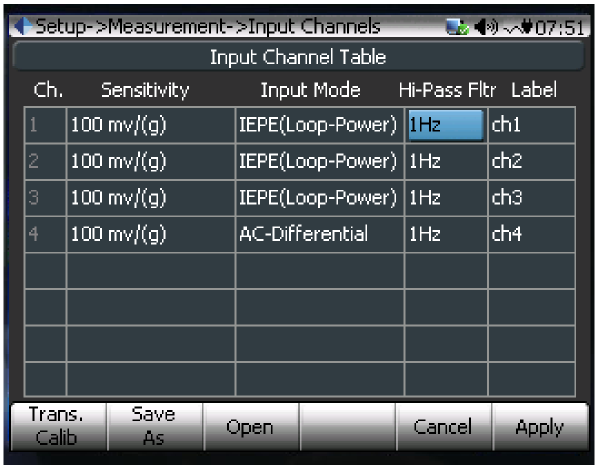
Select the appropriate engineering units for displacement. The example shows displacement units of millimeters.



■ Figure 134. Sensor sensitivity window with double integration and units.

With this setup, the accelerometer is set to measure *g* while the displacement is using *millimeter* as the output unit. The sensor sensitivity 50mV/g transforms the input voltage into *g* appropriately.

To enable or set the high pass filter, simply go to the channel table and high-light the column of ***Hi-Pass Fltr***, and press Enter button. Each channel can have its independent cutoff filter values.

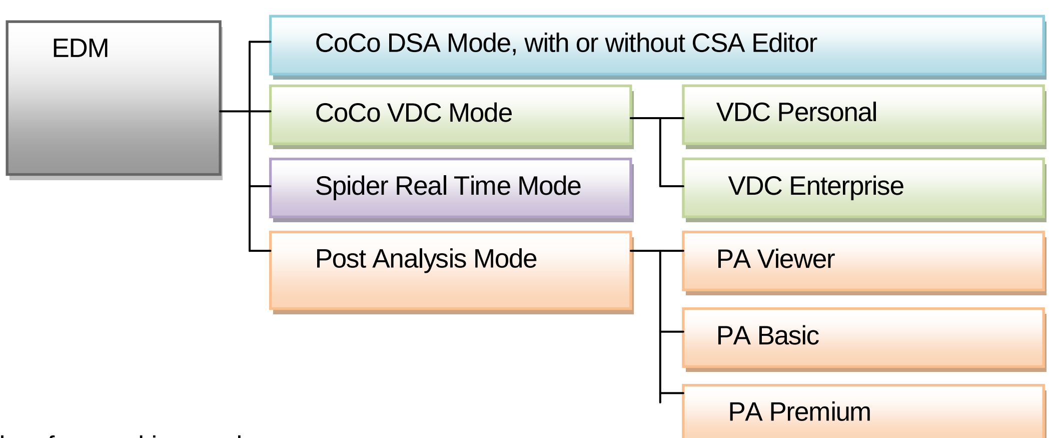


7. EDM PC SOFTWARE

This section briefly describes the Engineering Data Management (EDM) software that is used to download data from the CoCo-80 to a PC, view, analyze and export the data. For details about EDM software installation and operation please refer to EDM User's Manual.

The Engineering Data Management (EDM) is a PC software used for data management, post signal processing, viewing, report and the connection between the Crystal Instruments hardware, the PC and the data storage system. EDM provides connectivity to one or more CoCo or Spider devices. It provides data management tools that allow you to search through many tests, records and view file properties or waveform characteristics. The analysis tools allow you to display data in a wide variety of formats and configurations and let you identify important signal characteristics using cursors. The report tool allows you to document the hardware configuration or data analysis results in a user formatted document.

The basic structure of EDM software is:



EDM has four working modes:

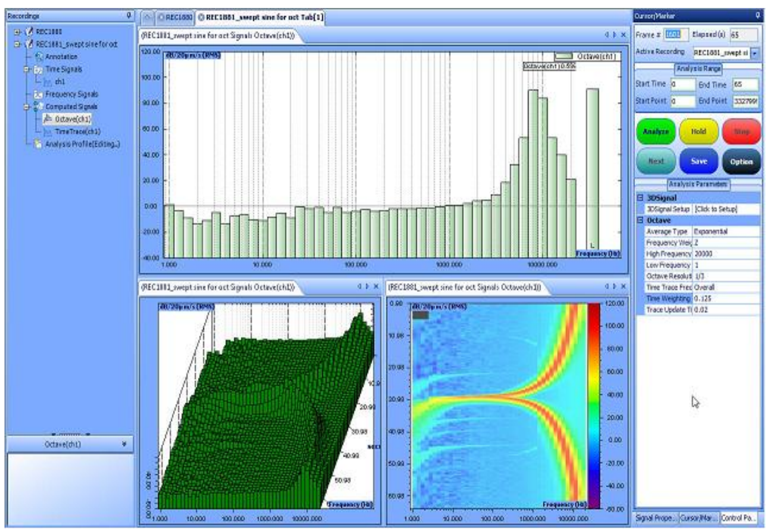
- **<u>CoCo-80 DSA mode</u>**: accesses CoCo-80 in its DSA mode, download files and view data files. CSA Editor, a tool of editing CoCo testing projects, will be included in this mode.

- **<u>CoCo-80 VDC mode</u>**: creates route data collection database, upload settings to CoCo, download data to PC, trending and alarm analysis. There are two versions of VDC modes: personal version allows the user access the database on his local PC. Enterprise version allows multiple user access the database on the LAN.

■ Figure 135: EDM in VDC Mode, Signal Analysis Display

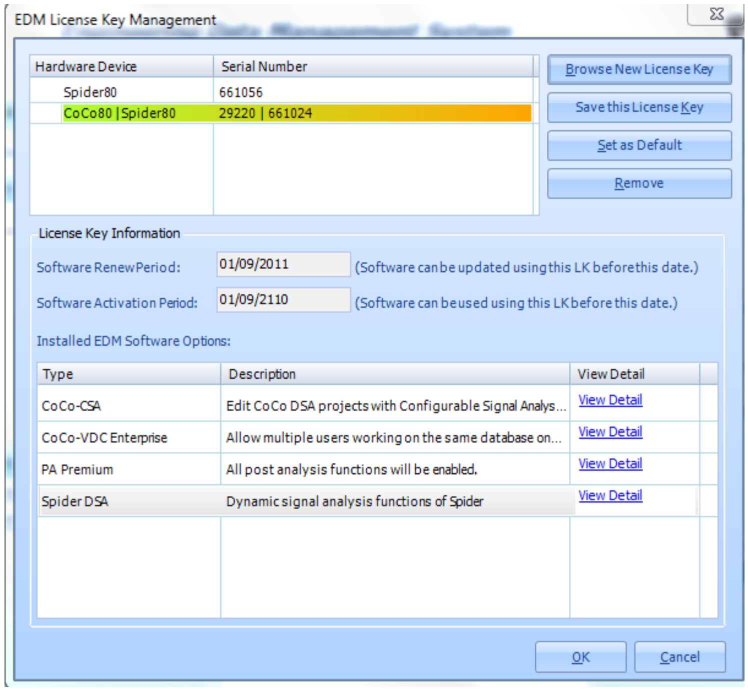
- **<u>Spider Real-Time Mode</u>**: operates on Spider hardware in real-time.

- **Post Analysis Mode**: analyzes the data files on PC using various algorithms. PA has three versions: PA Viewer allows the user to view the data and create report; PA Basic has FFT spectral analysis and 3D signal display functions; PA Premium has all post analysis functions.



EDM software is registered to a CoCo or a Spider device. To activate the EDM software, the user must have a License Key. EDM software uses a License Key file to enable or disable certain functions. License Key is also used to control the Activation Period and Software Subscription Renew period. Multiple License Keys can be installed in one EDM installation. This allows an instance of EDM runs multiple hardware devices.

A typical management page for license keys is shown below:



Software Renew Period: this is the time period that this EDM installation can be upgraded using the current installed License Key. When the time expires, the EDM software will still be functional but cannot be updated.
Software Activation Period: this is the time period that this EDM installation can be used using this License Key.

The CoCo DSA Mode of EDM

The EDM Software functions as both the means of transferring date from the CoCo-80 to the PC and also as a data management and analysis tool. The main features of the software include: transferring data between the CoCo-80 and a PC, viewing, searching and exporting data to other formats and using the analysis tools to measure signal characteristics.

Data Transfer

After a connection is established between the CoCo-80 and a PC, the EDM software manages the transfer of data between the two devices. The data includes recorded time streams, saved signals, and CSA projects. When EDM detects the connection, the software displays a list of the files available for transfer and allows the use to initiate the download to the PC. After files are downloaded they can be deleted from the CoCo-80 flash memory to create free space for new data files.

Data Management

The nature of signal measurement generates a large number of records. The EDM software provides tools to manage this data to simplify searching, review and exporting the data. Data can be searched by key words, date or time, size or other file attributes. Data can be previewed via thumbnail representations of the data or by text file attributes. Data can be replayed within the search tool including the ability to scroll through a long time stream to verify that the record contains the required properties. EDM simplifies the process of exporting data from the native ASAM ODS format to other popular universal formats including UFF, BUFF and ASCII.

Data Analysis

The EDM software includes basic analysis tools that help measure signal characteristics such as zoom and pan and cursors. Multiple signals can be overlaid on one trace for comparison. Long time streams can be played back and time or frequency data can be displayed.

CoCo-80 – PC Communication

The first step in downloading data from the CoCo-80 to a PC is to establish communication between the two devices. CoCo-80 is equipped with a number of hardware connectivity functions for easy communication with a host PC. These include:

- USB port
- 100MbaseT Ethernet
- Wireless 802.11b/g using SD card

You can choose one of following four typical connections:

- Connect CoCo-80 to a PC directly using a USB cable
- Connect CoCo-80 to a PC directly using Ethernet via cross-over cable
- Connect CoCo-80 to a local network using Ethernet where a host PC resides on the local network
- Connect CoCo-80 to a local network using a wireless SD card

The table below summarized the configuration for these connections.

■ Table 4. PC to CoCo-80 Configuration Summary.

Connection method	CoCo-80 Configuration	Host PC Configuration
Connect CoCo-80 to a PC directly using USB	No special configuration required	Install the EDM host PC software Install the CoCo-80 USB RNDIS Driver
Connect CoCo-80 to a PC directly using Ethernet via cross-over cable	CoCo-80 must be configured with a fixed static IP	Host PC IP must be configured with fixed static IP at the same subnet mask as that of CoCo-80
Connect CoCo-80 to a local network	If DHCP server is installed on the local network, CoCo-80 can obtain	If DHCP server is installed on the local network, host PC can obtain

using Ethernet where a host PC resides on the local network	an IP address automatically. If DHCP server is not installed on the local network, fixed static IP address must be configured on CoCo-80.	an IP address automatically. If DHCP server is not installed on the local network, fixed static IP address must be configured on the host PC. Same subnet mask must be used.
Connect CoCo-80 to a local network using wireless SD card	If DHCP server is installed on the local network, CoCo-80 can obtain an IP address automatically If DHCP server is not installed on the local network, a fixed static IP address must be configured on CoCo-80	If DHCP server is installed on the local network, host PC can obtain an IP address automatically" If DHCP server is not installed on the local network, fixed static IP address must be configured on the host PC. The same subnet mask must be used.

In this table, *DHCP (dynamic host configuration protocol) server* refers to a piece of software installed on the local area network, either wired or wireless, that supports the "Obtain an IP address automatically" function on any networked device. DHCP is commonly used in most office networks.

Transfer Data Files to the Host PC

To transfer the recorded data files to a PC, you must:

1. Establish a physical network connection between the CoCo-80 and a PC. This can be done by using either the Ethernet, USB-client port, or SD wireless card.

2. Execute the EDM software on the PC.

3. Download the data files from the CoCo-80 to PC using EDM software.

The data files will be automatically stored in the ASAM-ODS format. They can be converted into other formats with the EDM software.

Configuring the CoCo-80 Network Settings

CoCo-80 Network Settings must be configured when an Ethernet or SD Wireless card are used for communicating with the host. When USB is used for the connection, this section can be ignored.

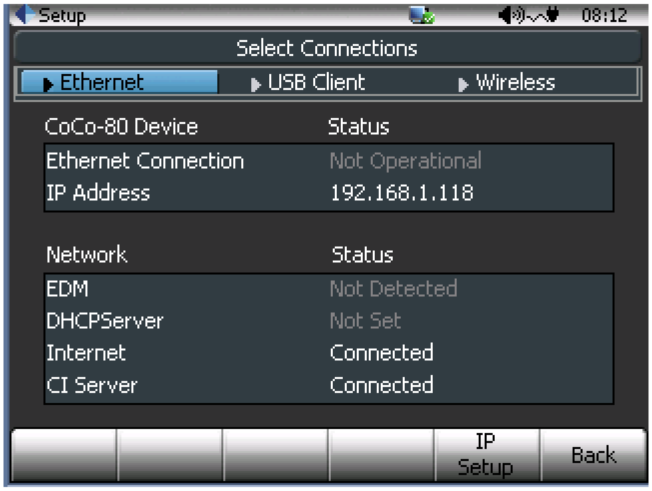
To configure the network settings for the CoCo-80, complete the following steps:

1. Power on the CoCo-80. In the Welcome page, move the focus to IP address and press ENTER.

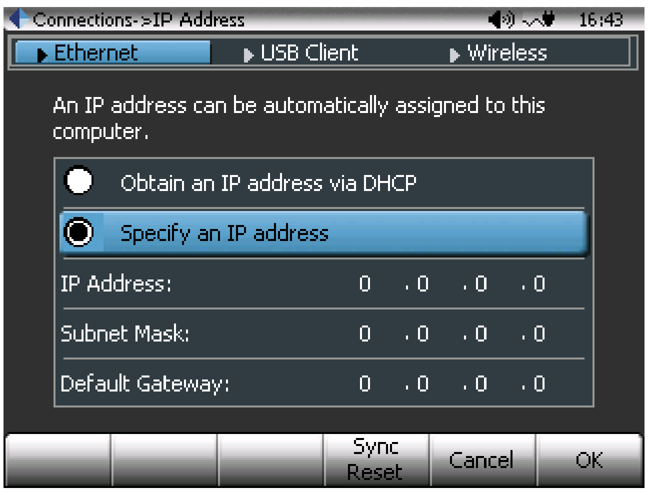
2. In the **IP Settings** window click **Edit the IP settings** to specify a static IP address and subnet mask. Type in the **IP address** and **Subnet mask**. You must specify a static IP address and use a crossover cable to directly connect the CoCo-80 to the host computer. In this case, both the Gateway and DNS server fields must be blank or set to zeroes. If the

network uses a DHCP server and you are not directly connecting the CoCo-80 to the host computer with a crossover cable, click the **Obtain IP address via DHCP server** option button.

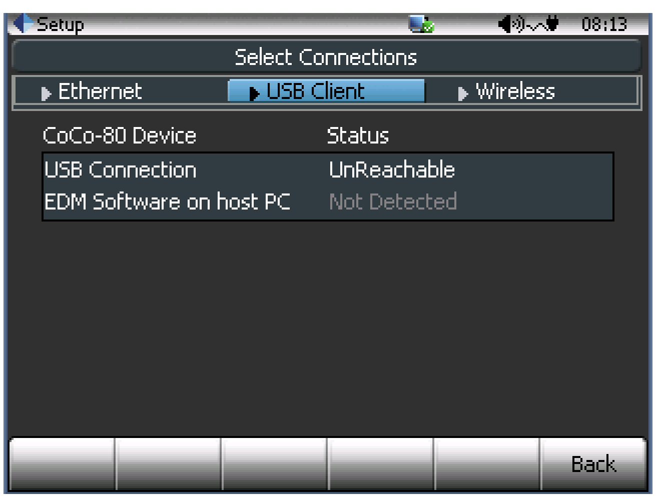
3. Click **OK** to apply the changes.



■ Figure 136. Ethernet connection status.



■ Figure 137. Specify a static IP address for the Ethernet connection.



■ Figure 138. USB Connection Status Display.

Access Code: This is a special setup to prevent an unauthorized user from accessing the CoCo-80 on the LAN. It also provides a means for the EDM software to access a particular CoCo-80 on the network. The EDM requires the CoCo-80 Access Code to access it over the LAN.

Configuring the Host PC Network Settings

If the host system is a PC which is only connected directly to the CoCo-80 using a cross over Ethernet cable, you can manually configure the TCP/IP settings. You can also use the "alternate configuration" functionality to maintain seamless operations on both office and private networks without having to manually reconfigure the TCP/IP settings. Choose whichever method best applies to your system configuration and connectivity needs. Refer to the Microsoft support website for more information on "alternate configuration" http://support.microsoft.com/kb/283676.

Note: You must be logged onto the host system as an Administrator in order to change network settings. Contact the system administrator to get access to the necessary privileges.

Connect CoCo-80 to a PC directly using USB client

A USB connection is the easiest method to connect the CoCo-80 to a PC. This requires the following steps:

1. Install the EDM software on the PC. Install the RNDIS USB driver on the PC.
2. Connect CoCo-80 to the PC through the provided USB cable. This cable has a mini-client port connecting to the CoCo-80 and a flat USB port connecting to the PC.

Connect CoCo-80 to a PC directly using Ethernet via cross-over cable

Another way to connect the CoCo-80 to a PC directly is to use the Ethernet port and a CAT-5 cross-over cable. The advantage of using Ethernet compared to USB is that the data transfer speed is faster with Ethernet. The disadvantage is that you must configure the IP settings on the host PC so it can communicate with the CoCo-80.

In this case, both PC and CoCo-80 must be configured with a fixed IP address with the same subnet mask. The host PC can also use the Alternative Configuration feature for convenient communication with its office local area network. Alternate Configuration is a networking option within Windows to maintain seamless operations on both office and home networks without having to manually reconfigure TCP/IP settings. Refer to the Microsoft support website for more information of this feature.

http://support.microsoft.com/kb/283676

Connect CoCo-80 to a local network using Ethernet

In this connection case, if DHCP server is not installed on the local area network, both PC and CoCo-80 must be configured with a fixed IP address with the same subnet mask. If DHCP server is installed, then both the PC and the CoCo-80 can use the **Obtain an IP address automatically** function.

Connect CoCo-80 to a local network using wireless SD card

In this connection case, if DHCP server is not installed in the local area network, both PC and CoCo-80 must be configured with a fixed IP address with the same subnet mask. If DHCP server is installed, both the PC and the CoCo-80 can use the **Obtain an IP address automatically** function.

AmbiCom WL11 or WL54-SDIO Wireless LAN Card is a compact size wireless card for the SDIO capable PDAs and other SDIO compatible mobile computing device using Microsoft Windows Mobile 2003 and Windows Mobile 5.0 operating system. In addition to the slim and ultra lightweight SD design, the Wireless SD Card also features secure data transfer and full privacy, exceptional range and data rate, and meets Wi-Fi certification standards for total interoperability with other 802.11b/g equipment. More information is available at

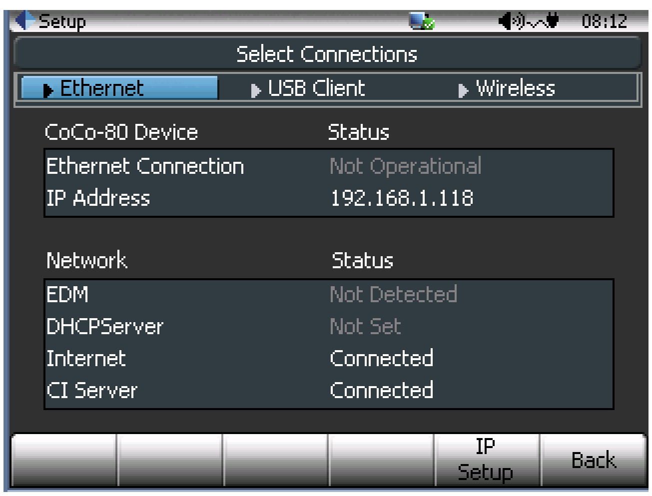
http://www.ambicom.com/products/wave2net/wl54-sdio10.html

Network Connection Diagnosis

The following section describes methods for diagnosing network connectivity from the CoCo-80 or the PC which may be helpful when setting up the network connection.

Diagnosis from the CoCo-80 side

A tool is provided to detect the existing network settings from the CoCo-80 side. Push the Setup button, and select the Connections icon and press the Enter button, the connection status is shown below:



■ Figure 139. Ethernet connection status screen.

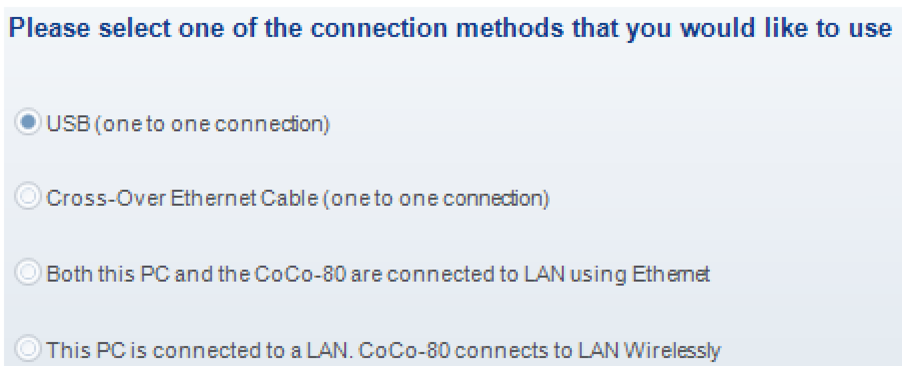
The network setting detection shows the following status:

- Hardware: indicates whether the Ethernet, USB port or Wireless card inside the CoCo-80 device are functional.
- IP Address: indicates the IP address of the CoCo-80.

- DHCP server: indicate whether the CoCo-80 has detected a DHCP server on the local area network.
- EDM: indicates whether CoCo-80 is connected to the EDM, the host software on a PC.
- Internet: indicates whether the CoCo-80 is connected to the Internet.
- CI server: indicates whether the CoCo-80 is detecting the Crystal Instruments server. The CI server is used to host new software to keep the CoCo-80 up to date.

Diagnosis from the PC

The connection between the CoCo-80 and a PC is managed within the EDM software on the PC. The EDM software provides connection diagnosis capability. The Connection Wizard dialog box will show one of following four connection pictures:



■ Figure 140. EDM network connection status screen.

Select appropriate connection type, and follow the online instructions. The EDM will provide diagnose information.

Data Format

The data format within the CoCo-80 and the EDM software is the ASAM ODS File format. ASAM ODS files have the suffix ATFX. EDM also interfaces to other file formats including NI-TDM, MatLab, UFF, BUFF and user-defined ASCII files.

ASAM ODS (Open Data Service)

The rapid progress in hardware and software leads to storage of data in many different data base systems as well as under different hardware and/or server generations. During development and production of complex products, a huge mass of data is produced. Today, data are stored within the automotive industry in a standardized format specified by the ASAM ODS workgroup. ASAM stands for ***Association for Standardization of Automation and Measuring Systems***, and ODS stands for “Open Data Services”. The CoCo-80 uses the ASAM ODS data format as the internal data format and data is saved by default in this format when it is downloaded from the device to a PC.

The ASAM ODS standard has the fundamental quality of storing data with an architecture-independent method. This leads to great advantages when exchanging data between different sources and possible prospective customers.

Many systems in test, evaluation, and simulation environments have their own proprietary formats to store data. These formats usually are very different from each other regarding the description of the configuration (unit under test, test sequence, test equipment, etc.) as well as the way results are stored (database, binary files, etc.).

The main objectives for a standardization of data access interfaces are to reduce costs and risks within projects, and to provide a reliable basis for implementations in the area of data storage and data usage. Using standardized interfaces and common structures minimizes the efforts for the system integration within the heterogeneous environments discussed above and makes it much easier to exchange data.

Because of these benefits the ASAM ODS data format was chosen as the internal format for the CoCo-80 and the EDM software.

UFF Files

The CoCo-80 and EDM Software also support the Universal File format (UFF). This format was originally developed by the Structural Dynamics Research Corporation (SDRC) in the late 1960s and early 1970s to facilitate data transfer between computer aided design (CAD) and computer aided test (CAT) in order to facilitate computer aided engineering (CAE). SDRC, as part of EDS, continues to support and utilize the UF formats as part of their CAE software. Currently, MTS, Noise and Vibration Division supports and continues to develop IDEAS software in the test area that utilizes UF formats.

The formats were originally developed as 80 character (card image), ASCII records that occur in a specific order according to each UF format. As computer files became routinely available, single UF formats were concatenated into computer file structures. Recently, a hybrid UF file structure (UF Dataset 58 Binary) was developed for experimental data that allows data to be stored in a more efficient binary format.

Before the introduction of ASAM ODS, the use of the Universal File Format as a de-facto "standard" has been of great value to the experimental dynamics (vibration and acoustic) community, particularly in the area of modal analysis. Both users and vendors have benefited from this de-facto standard.

The EDM software will be able to export the data into UFF (Dataset 58) and BUFF (Dataset Binary 58). For more information on UFF refer to http://www.sdrl.uc.edu/uff/uff.html.

The Binary 58 Universal File Format (BUFF)

The CoCo-80 and EDM software also support the BUFF format. The basic (ASCII) universal file format for data is universal file format 58. This format is completely documented by SDRC and a copy of that documentation is on the UC-SDRL web site (www.sdrl.uc.edu/UFF2/58.asc). The universal file format always begins with two records that are prior to the information defined by each universal file format and ends with a record that is placed after the information defined by the format. First of all, all records are 80 character ASCII records for the basic universal file format. The first and last record are start/stop records and are always -1 in the first six columns, right justified (Fortran I6 field with -1 in the field). The second record (Identifier Record) always contains the universal file format number in the first 6 columns, right justified.

This gives a file structure as follows (where b represent a blank character):

```
bbbb-1
bbbb58
...
...
...
bbbb-1
```

The Binary 58 universal file format was originally developed by the UC-SDRL in order to eliminate the need to compress the UFF 58 records and to reduce the time required to load the UFF 58 data records. The Binary 58 universal file format yields files that are comparable to compressed files (approximately 3 to 4 times smaller than the equivalent UFF 58 file). The Binary 58 universal file format loads approximately 30 to 40 times faster than the equivalent UFF 58 file, depending upon the computing environment. This new format was submitted to SDRC and subsequently adopted as a supported format.

The Binary 58 universal file format uses the same ASCII records at the start of each data file as the ASCII dataset 58 but, beginning with record 12, the data is stored in binary form rather than the specified ASCII format. The identifier record has the same 58 identifier in the first six columns, right justified, but has additional information in the rest of the 80 character record that identifies the binary format (the size of the binary record, the format of the binary structure, etc.).

```
 -1
 58b   x   y     11     zzzz   0   0       0       0
...
... (11 ASCII header lines)
...
...
... (zzzz BINARY bytes of data, in format specified by x and y, above)
... (interleaved as specified by the ASCII dataset 58)
...
    -1
```

When reading or writing a dataset 58b, care must be taken that the binary data immediately follows the ASCII header lines and the closing '   -1' immediately follows the binary data. The binary data content is written in the same sequence as the ASCII dataset 58 (i.e. field order sequence). The field size is NOT used, however the data type (int/float/double) content is. Note: there are no CR/LF characters embedded in or following the binary data

ASCII UFF

The CoCo-80 and EDM software also support the ASCII UFF format. The ASCII UFF file format is a form using the ASCII type to represent all the data sets. For details, see: http://www.sdrl.uc.edu/uff2/58.asc

MATLAB file

This is the standard file that can be imported into Matlab.

NI-TDM file

This is a structured data format that is defined and widely used by the LabView from National Instruments.

User Defined ASCII file

This is the ASCII files where you have the freedom to define its attributes and header format.

.CSV (Microsoft Excel) File

This is the ASCII file that the Microsoft Excel can directly read.

.WAV File

This is the sound wave files that can be played by most of the media players. Due to limited information a wave file can carry, the wave files exported only contain very basic waveform shape and it does not hold any attribute information of ODS. You are expected to use the .WAV file to listen to its sound effect, instead of for data processing.

APPENDIX

Version

Version	Release Date	Comments
0.2	8/01/07	First Draft
0.90	11/12/07	Revised
0.94	2/15/08	Many pictures added
0.99	7/16/08	Basic manual includes basic, spectral analysis and impact testing.
1.01	2/3/09	Move built-in integration from Advanced to here
1.2	7/16/09	Add descriptions for added 6 buttons. Minor changes
1.3	1/27/2010	Add playback function. Change CI address

Users' Manual Typeface

Headings Arial Black 12 and 11 pt

Body Text: Arial 10 pt

Captions: Arial Narrow 9 pt

Declaration of Conformity

Declaration of Conformity for CI CoCo-80, Handheld Data Acquisition System

Manufacturer: Crystal Instruments Corporation, 4699 Old Ironsides Drive, Suite 100, Santa Clara, CA 95054

Statement of Conformity:

EC Declaration of Conformity

Council Directive 2004/108/EC on Electromagnetic Compatibility

WE, Crystal Instruments

4699 Old Ironsides Drive, Suite 100, Santa Clara, CA 95054, USA.

Product Name: CoCo-80 (Handheld Data Acquisition System)

Model No.: CoCo-80

Assessment of compliance of the product with the requirement relating to Electromagnetic Compatibility Directive .The product has been assessed by the application of the following standards:

EN 61326:1997+A1:1998+A2:2001
EN61000-3-2: 2000
EN61000-3-3: 1995+A1:2001

The tests have been performed in a typical configuration.

This Conformity is indicated by the symbol, i.e. CE "Conformité Européenne".

CoCo-80 Advanced User's Manual

Version 1.01

Crystal Instruments Corporation

4633 Old Ironsides Drive, Suite 304

Santa Clara, CA 95054, USA

This page is left blank intentionally

Table of Contents

Table of Figures

1. INTRODUCTION

The CoCo Advanced Users Manual discusses the theory, EDM software and CoCo operation for the optional advanced CoCo features including:

- Swept Sine Analysis
- Acoustic Data Acquisition: Octave Analysis and Sound Level Meter
- Order Tracking
- Sock Response Spectrum Analysis
- Automated Test and Limit Test
- Real Time Digital Filters
- Histogram and Statistics Measures
- Miscellaneous Operations.

Each topic includes a detailed description of the general theory including mathematical formulation application topics, instructions on how to create a CSA file using the EDM software, and detailed instructions on how to setup the CoCo hardware and make a measurement.

This document references the CoCo Basic Users Manual. This separate document gives details on the basic operation of the CoCo hardware and details on basic frequency spectrum measurements including theory, EDM software setup and CoCo operations. We strongly recommend that you read the CoCo Basic Users Manual first before proceeding to this document.

2. SWEPT SINE MEASUREMENTS

This section describes the swept sine measurement capabilities of the CoCo. It includes both theoretical background and application information. The Swept Sine Testing option of CoCo-80 has several unique advantages over similar products in the market, including:

- The measurement channels with very high dynamic range ensure continuous test over high dynamic range UUT (Unit Under Test). It is common to achieve 130~150dB dynamic range with CoCo-80.
- Special tracking filters realized based on TVDFT (Time Variant Discrete Fourier Transform) provide excellent spectrum estimation.
- Special algorithm enables test in wide frequency range. The result of both low and high frequency testing is excellent.
- Time domain signals are always available for viewing and recording.
- Log, Linear sweep modes are available.
- Auto-gain adjustment with closed-loop control capability to prevent input range overloading.

Sine Signal Used for Testing

Broadband random, sine, step or transient signals are widely used as excitation signals in test and measurement applications. Figure 1 illustrates that an excitation signal x, can be applied to a UUT (Unit Under Test) and generate one or multiple responses denoted by y. The relationship between the input and output is known as the transfer function or frequency response function and represented by H(y,x). In general a transfer function is a complex function that modifies the input signal magnitude and phase as the excitation frequency changes.

■ Figure 1 Left: a UUT with one response; Right: a UUT with two responses.

With swept sine excitation, the characteristics of the UUT system can be measured experimentally. These characteristics include:

- Frequency Response Function (FRF), which is described by:
- Gain as a function of frequency
- Phase as a function of frequency
- Resonant Frequencies
- Damping factors
- Total Harmonic Distortion
- Non-linearity
- Others

Frequency response can be measured using the FFT, cross power spectral method with broadband random excitation. Broadband excitation can be a true random noise signal with Gaussian distribution, or a pseudo-random signal of which the amplitude distribution can be defined by the user. The term "***broadband***" may be misleading, as a well implemented random excitation signal should be frequency band-limited and controlled by the upper limit of the analysis frequency range. That is, the excitation need not excite frequencies above that which can be measured by the instrument. The CoCo random generator will only generate random signals up to the analysis frequency range. This will also concentrate the excitation energy on the useful frequency range.

The advantage of using broadband random excitation is that it can excite the whole frequency range in a short period of time so the total testing time is less. The drawback of broadband excitation is that its frequency content is spread over a wide range within a short duration. The energy contribution of the excitation at each frequency point will be much less than the total signal energy (roughly, it is -30~ -50dB less than the total). Even with a large number of averaging in the FRF estimation, the broadband signal will not effectively measure the extreme dynamic characteristics of the UUT.

Swept sine measurements, on the other hand, can optimize the measurement at *each frequency point.* Since the excitation is a sine wave, all of its energy is concentrated at a single frequency, eliminating the dynamic range penalty in a broadband excitation. In addition, if the frequency response magnitude drops, the tracking filter of the response can help to pick up extremely small sine signals. Simply optimizing the input range at each frequency can extend the dynamic range of the measurement to beyond 150 dB.

Introducing Sweeping Sine

A sine signal with a fixed frequency f_0 can be expressed as:

$$\mathrm{x(t)} = \sin(2\pi f_0 t)$$

where t represents time. A sweeping sine signal has a changing frequency that is usually bound by two limits. The frequency change can be either in the linear scale or logarithmic scale based on different user requirements. The swept sine signal can be defined by the following parameters:

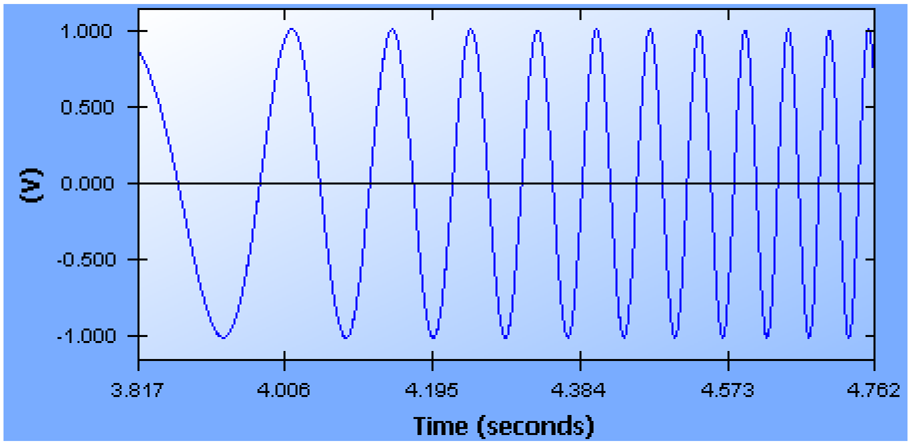
- The low frequency boundary, which is simply called Low Frequency or f_{Low}
- The high frequency boundary, which is simply called High Frequency or f_{High}
- The sweeping mode, either logarithmic or linear
- The sweeping speed, in either octave/min if the sweep mode is logarithmic, or in Hz/Sec if the sweeping mode is linear
- The amplitude of the sine signal, A(f, t), which can be a constant or a variable of time and frequency.

$$\mathrm{x(t)} = \mathrm{A(f,t)} * \sin(2\pi \left(f\left(f_{Low}, f_{High}, Speed\right)\right) t)$$

The instantaneous frequency $f\left(f_{Low}, f_{High}, Speed\right)$ represents the current frequency of the sweeping sine. It is a changing variable and usually displayed on the screen as ***Sweeping Frequency***.

The sweeping frequency can also be manually controlled during the test with the ***Hold, Resume, Jump or Pause*** **controls**.

Unlike some DSA products which use swept sine test with ***multiple discrete stepped sine tones in a sequence***, the CI swept sine test uses a true digital synthesizing technique to generate sine sweeps with extreme analog-like smooth transition from one frequency to another. This ensures that there are no sharp transitions during the test that might "shock" the UUT. The picture below shows a typical swept sine signal with 1.0 Vpk.



■ Figure 2. Typical digitally synthesized swept sine signal.

Sweeping Mode: Logarithmic or Linear

A swept sine can sweep in either linear or logarithmic mode. Linear sweep means the frequency will change at a constant speed, with units of *Hz/sec*. In this case the sweep rate is constant and the same at all frequencies.

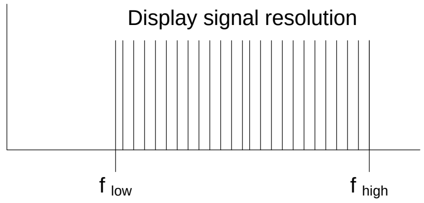
Alternatively, the sweeping mode can be set as logarithmic or Log. In Log mode, the sweeping speed is slower at low frequencies and fast at higher frequencies. In Log mode, the sweeping speed units are in *Octave/Min*. *1.0 Oct/min* means that the frequency will take one minute to double from 1kHz to 2kHz, or from 100Hz to 200Hz, or from 0.5Hz to 1.0Hz.

Most testing specifications ask for logarithmic sweeping for two reasons. The first is due to the fact that it takes longer to measure one or multiple sine cycles at low frequency than at high frequency. The second is that most mechanical and electrical systems exhibit characteristics that are better described in logarithmic frequency scale. This is because dynamics such as resonant frequencies occur over large frequencies spans: some at low frequencies and some at high frequencies. If linear sweeping is adopted, you may find that whatever the speed you choose, it is either too slow in the high frequency end or too fast in the low end. With Log sweeping mode, this problem is solved.

On the CoCo, once the sweeping mode is set to either Linear or Log in a test, the frequency distribution of the display signals will be set to linear or logarithmic accordingly. This will be discussed in the following section about the display resolution. The sweeping speed unit will also be set to either Hz/Sec. or Oct/Min automatically.

Resolution of Display Signals

In the CoCo the sweeping sine signal is point-by-point digitally synthesized. It has "infinitely" fine resolution in frequency transition. It does not jump from one frequency to another. The user may wonder how the sweeping signal is displayed. The user first needs to set the size of the displaying signals, say 1024 or 2048. The CoCo will distribute the frequency bins between f_{Low}, f_{High}. In Linear mode, the frequency spacing between two adjacent lines is represented by the frequency resolution; In Log mode, the frequency spacing between two adjacent lines of the signal will be represented by a ratio.



For example, if a linear sweep is defined with $f_{Low} = 100Hz;\ f_{High=}1000Hz,\ Signal\ Size = 1024$, then the first line of the signal will be allocated to 100Hz, the last to 1000Hz. The frequency bins of the signals will be evenly distributed with frequency resolution of

(1000-100)/(1024-1) = 0.879765Hz

If a logarithmic sweep is defined with **Log Sweep Mode**: $f_{Low} = 100Hz;\ f_{High=}1000Hz,$ $Signal\ Size = 1024$ Then the first line of the signal will be allocated to 100Hz, the last to 1000Hz. The Frequency resolution will be represented by a ratio as:

$$1000\ \text{Hz} = 100\ \text{Hz} * \text{ratio}^{(1024-1)}$$

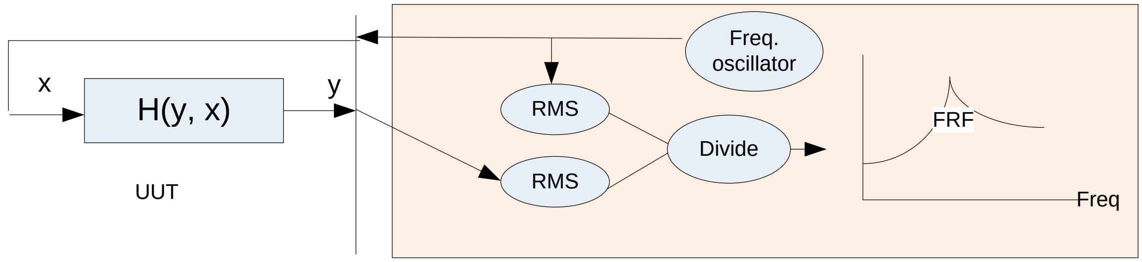
$$ratio = \left(\frac{1000}{100}\right)^{\frac{1}{1024-1}} = 1.00225335$$

This means that if the first line is at frequency 100Hz, the second line will be at 100.225335Hz, the third at 1.00451178Hz and the 1024th line at 1000Hz.
Once allocated, the display signals will keep the history of each calculated result. The CoCo will update the points that are near the instantaneous frequency of the sweep. This is how the display signals are created. With this design, the user should understand that increasing the resolution of the display signals will not increase or decrease the quality of the swept sine.

Tracking Filters

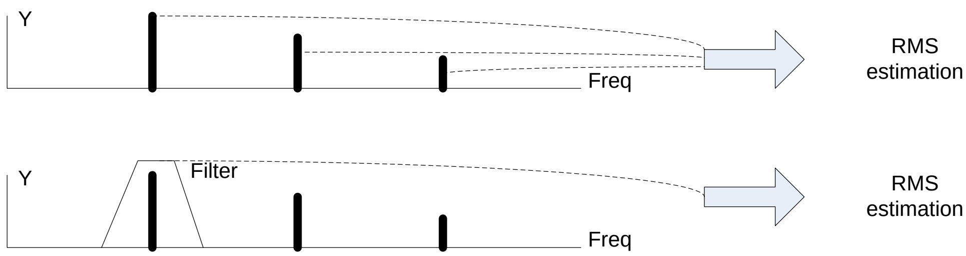
Historically, swept sine tests were originally conducted using analog technology where sine generator and measurement were all implemented in the analog domain. A very simple swept sine testing instrument consists of the following components:

- A sine oscillator of which the frequency can be changed
- An RMS estimator to the output source
- An RMS estimator to the input signal
- A divider that divides the RMS measurements between input and output
- A display or plotter to show the divided results.



- Figure 3. Analog swept sine implementation.

In many cases the UUT response is not linear. With very pure sine excitation, the response signal may contain strong harmonics. For example with a sine tone excited at 100Hz, the response signal may contain content at 200Hz, 300Hz and so on. A simple RMS estimator will not be able to distinguish the amplitudes at these content therefore the FRF calculation will not be accurate.
To overcome this problem, a tracking filter can be applied that centers at the sweeping frequency, and the RMS estimator can be applied to the output of the tracking filter as shown in Figure 4.



- Figure 4. Tracking filter implementation.

With a filter in place, the RMS estimator will accurately measure the frequency amplitude at the sweeping frequency. The energy at other frequencies will be filtered out.

The challenge of realizing such a filter in the swept sine test is that the filter has to track the center frequency of the sine frequency. Not only does the center frequency of the filter need to change, but also the bandwidth. To give an example, when the sweeping frequency is at 100Hz, it is reasonable to use a filter bandwidth of 50~100Hz. When the sweeping frequency goes down to 10Hz, the next harmonics will be at 20Hz. A filter with bandwidth of 50~100Hz will be too wide to use. To address the problem, a so called the tracking filter is used. Tracking filter changes both its center frequency and bandwidth according to the sweeping frequency. In the analog-made swept sine equipment, this is realized using mixing frequency technology with expensive electronic components. With digital technology, the digitally synthesize tracking filters are implemented in software at no additional hardware cost.

The bandwidth of the tracking filter is a key control parameter. In CoCo, it is defined as a percentage of the sweeping frequency. The user can select a percentage between 100% and 7%. A percentage of 100% means the equivalent bandwidth of the tracking filter is the same as its sweeping frequency. 50% means its bandwidth is ½ of the sweeping frequency.

CoCo uses a proprietary digital filter that allows very fast response and clean detection of the sine RMS value.

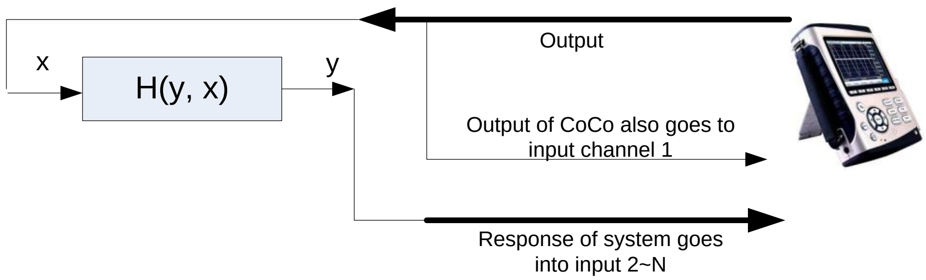
Measurement Quantities

Measurement quantities that can be monitored during the swept sine test include: time stream of each channel (raw data), spectrum of each channel, frequency responses, coherence, and phase between responses to the reference channel.

Time streams: time streams appear the same as any other applications on CoCo. Time streams are always available for viewing and recording. It is a very useful tool to observe whether the input signals are in the valid range. The recorded sine wave can be used for further post-processing. In CoCo, the time streams are often denoted as ch1, ch2 etc.

Spectra: The term spectrum is used to refer to the measurement trace in the frequency domain of each channel. It is represented in 0~Peak. The engineering unit of the spectrum is determined by the sensor used by the input channel. The resolution of spectra does not affect the quality of sine wave. In CoCo, the spectra are often denoted as Spec(ch1), Spec(ch2) etc..

Frequency Response Functions (FRF): FRF of UUT can be measured using input channel 1 as reference channel and other channels as response channels. The connection should as shown in Figure 5.



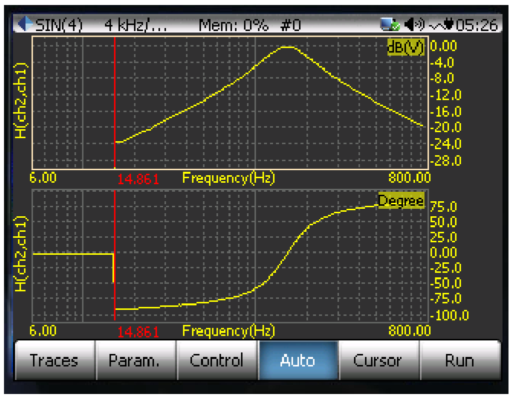
■ Figure 5. Frequency response measurement with CoCo.

The CoCo will provide the FRF functions of each response channels to the output channel. FRF signals include both phase and magnitude information. In CoCo, the FRF are often denoted as Hyx

The number of FRF signals that can be monitored depend on the number of input channels on the CoCo hardware. For example, a CoCo-8, with 4 input channels can monitor 3 FRFs: H(ch2,ch1), H(ch3,ch1) and H(ch4,ch1).

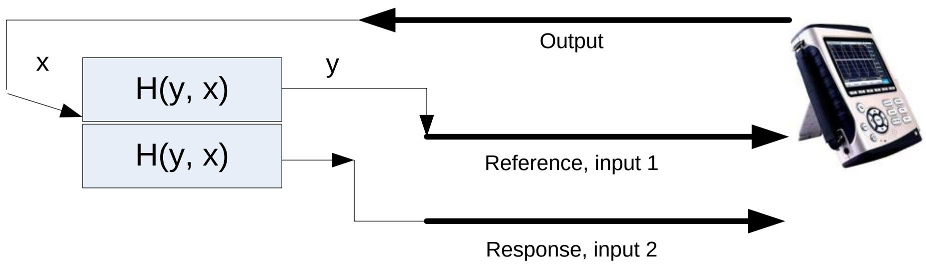
To connect the signal source to the input of UUT and back to input channel 1, you can use a BNC T-connector.

■ Figure 6 BNC T-Connector



■ Figure 7. Frequency response function.

When you measure the ratio between two response channels, it is more accurate to refer to the signals as ***Transmissibility functions*** instead of FRF because the reference signal is not really the excitation to the UUT. Figure 8 shows how to connect and measure the transmissibility between two response channels.



■ Figure 8. Typical transmissibility measurement.

Transmissibility measurements are used in many applications. For example, it can be used in "back-to-back" transducer calibration where an accurate reference transducer is used to calibrate a less accurate one.

Output Control Modes

Recall that the sine tone amplitude can be a variable of time and frequency.

$$x(t) = A(f,t) * \sin(2\pi\left(f\left(f_{Low}, f_{High}, Speed\right)\right)t)$$

We have discussed how the frequency can be changed and controlled by the sweeping mode, sweeping range and sweeping speed. This section discusses how the output amplitude $A(t)$ is controlled.

There are three Output Modes provided in the CoCo:

- Constant Output Level
- Output Level Profile
- Input Profile with Auto Gain Control

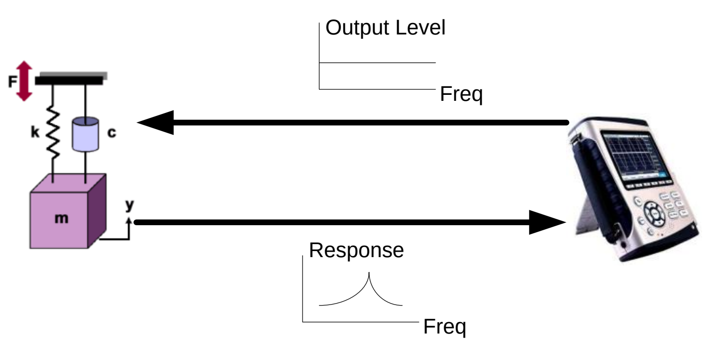
Constant Output Level

$A(f, t) = \text{constant}$

The Constant Output Level is the simplest way to generate the output. It uses a constant output level that is usually defined in the 0~peak volt. For example a 1Vpk means the output swept sine is in 1V in 0~peak.

When the constant output level is used, the response may show peaks or valleys.

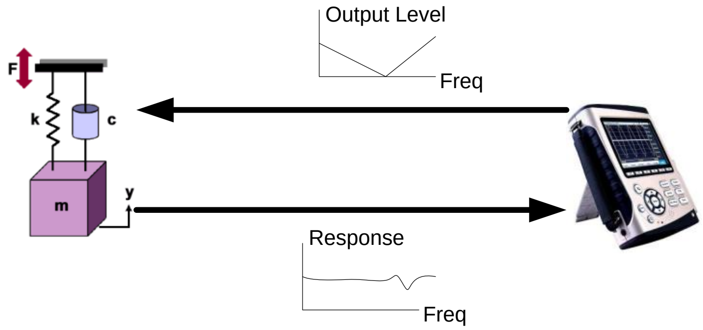
The picture below shows that a constant level sweep is applied to a Single Degree of Freedom (SDOF) device. The voltage output of the CoCo will be converted to force using a mechanical excitation system. We would expect the response measured in either displacement, velocity or acceleration will show a resonant peak.



The drawback of using constant level mode is that sometimes the dynamics of the system vary so extreme that the response signal may exceed the input range. This is very common with systems that have light damping. For example a UUT with 60dB dynamic range, which is quite common, will show the magnitude of the response change 1000 times over the test.

Output Level Profile

With output level profile control the output level $A(f, t) = A(f)$ is defined by the user. To overcome the problems with large range of variation of the response, it is possible to attenuate the excitation signal at certain frequency ranges. In the example above, because the resonance frequency is likely known to the operator, we can set the output to a lower level in that specific sweeping frequency range. We call this frequency dependent output level control the Output Level Profile.



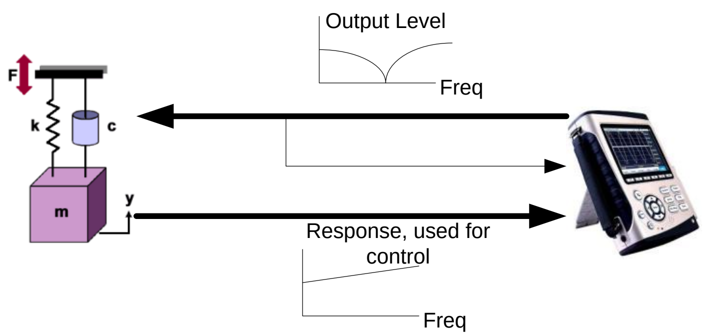
■ Figure 9. Output level profile.

Figure 9 shows that we purposely create a notch in the output level profile so that the response signal is attenuated in the resonance area.

Making the output a frequency dependent signal may help to improve the FRF or transmissibility measurement. It is much better than a constant level output. The drawback of this method is that the UUT dynamics must be known before the test. Another issue is that the output level profile may not be created accurately to match the dynamic characteristics of the UUT. To overcome this difficulty, the CoCo also includes a close-loop control method to allow the auto-gain control.

Auto Gain Control

With auto gain control, $A(f, t)$ is calculated in real time based on the target input and close-loop control gain. This advanced method can be explained in Figure 10.



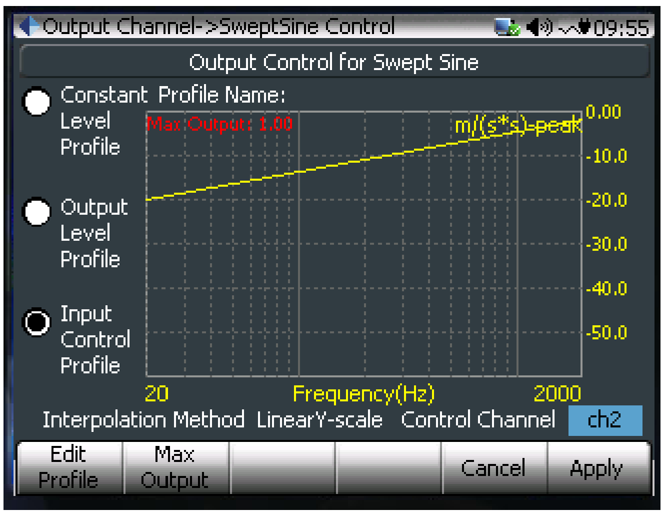
■ Figure 10. Auto gain control mode.

First the user must set up the target profile for one of the response channels (input to the CoCo). The shape of this target profile (Input Profile) does not need to be a straight line. Then during the sweep, the CoCo measures the transfer function between the response and the output. Taking this transfer function into consideration, the CoCo automatically adjusts its output so that the magnitude of the measured input signal matches the input control profile. Because the transfer function changes with frequency, this method requires a close-loop control logarithm.

The input profile with auto-gain control is the most effective way to excite the system. It can maximize the dynamic range of the input channels. However, care must be taken so that the output channel does not get too large and the input channel is saturated, or the output channel gets too small and the input channel reduces to the background noise level.

It must be noticed that the Output and Input Profiles have different engineering units. The Output Profile always has the engineering unit of Vpk for the sine wave. The Input Profile will have the engineering unit of whatever is measured by that channel. For example if the response sensor is a displacement sensor, then the Input Profile will have displacement units, in 0~Peak. If it is an accelerometer, then it will have acceleration units, in 0~Peak.

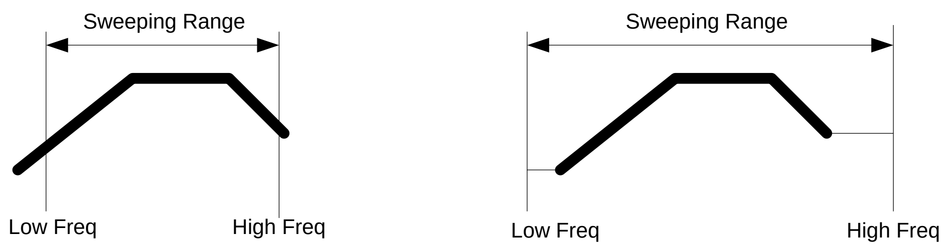
When Input Control Profile is selected, in default we use channel 2 as control channel. You can select any channel other than the reference channel (channel 1) for control.



■ Figure 11. Input auto gain control mode profile.

Sweeping Range and Profile

The sweeping range is controlled by the two boundaries of the frequency range. If the profile setting is not at the same range of the sweeping range then the CoCo will automatically adjust the range.



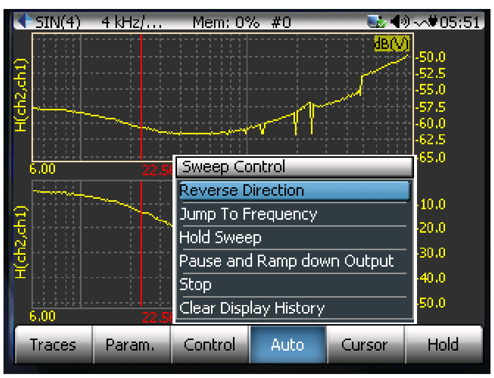
■ Figure 12. Sweeping range and profile.

In Figure 12, the thick line represents the profile. If the Low Freq or High Freq limits do not match the Profile, the software extends the ends to the left and right so there are always valid profile value points when the output sweeps.

Sweep Control

The swept sine output is controlled by ***sweeps***. One sweep indicates that the output will generate the sine frequency from the Low Frequency to the High Frequency, or high to low. In addition the user can control the sweep with manual controls including:

- Start Output
- Stop Output: this action will abort the test
- Reverse Direction
- Jump to Frequency
- Hold Sweep: this action will not ramp down the output voltage amplitude. The frequency will be fixed
- Resume Sweep
- Clear Display History

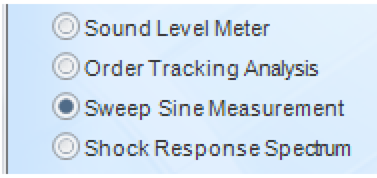


■ Figure 13. Sweep control options.

To avoid shocking the UUT, a sine output will never start or stop abruptly. Instead, the sine wave amplitude slowly ramps up from zero to the desired level. The ramping rate is defined as dB/sec. a 40dB/sec means the sine wave will ramp up or ramp down for 100 times in magnitude in a second. This is a user-defined advanced value.

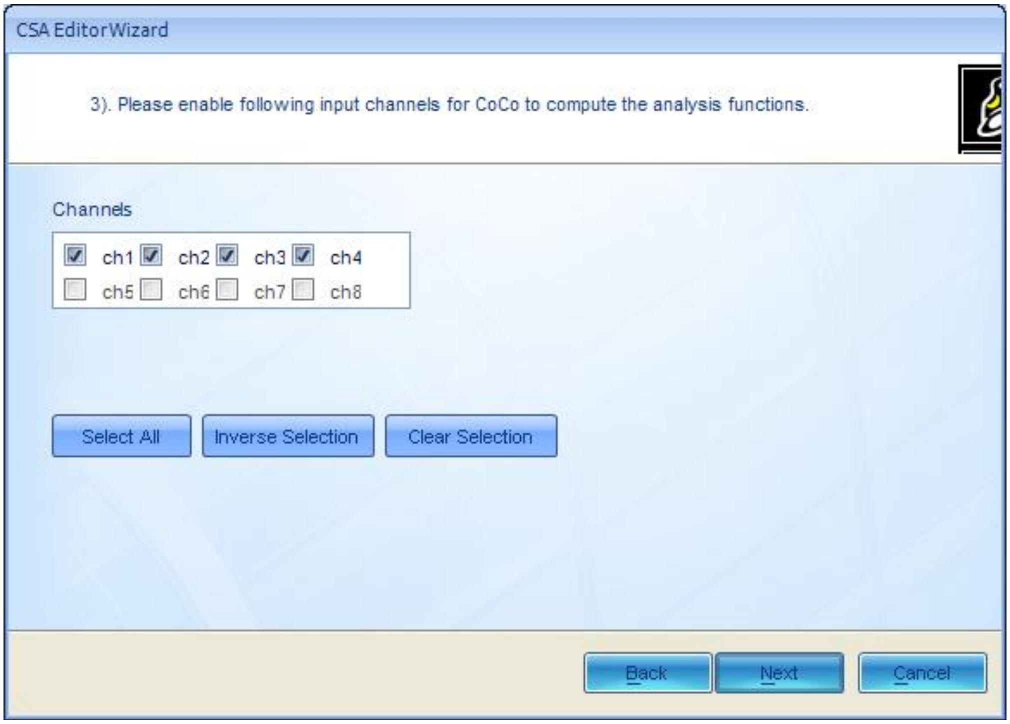
CSA Editor Operations

This section describes the operation of the CSA Editor that is related to swept sine testing. For general operation of the CSA Editor, refer to the CSA User's Manual.
In the CSA Editor Wizard, select the Swept Sine Measurement template.

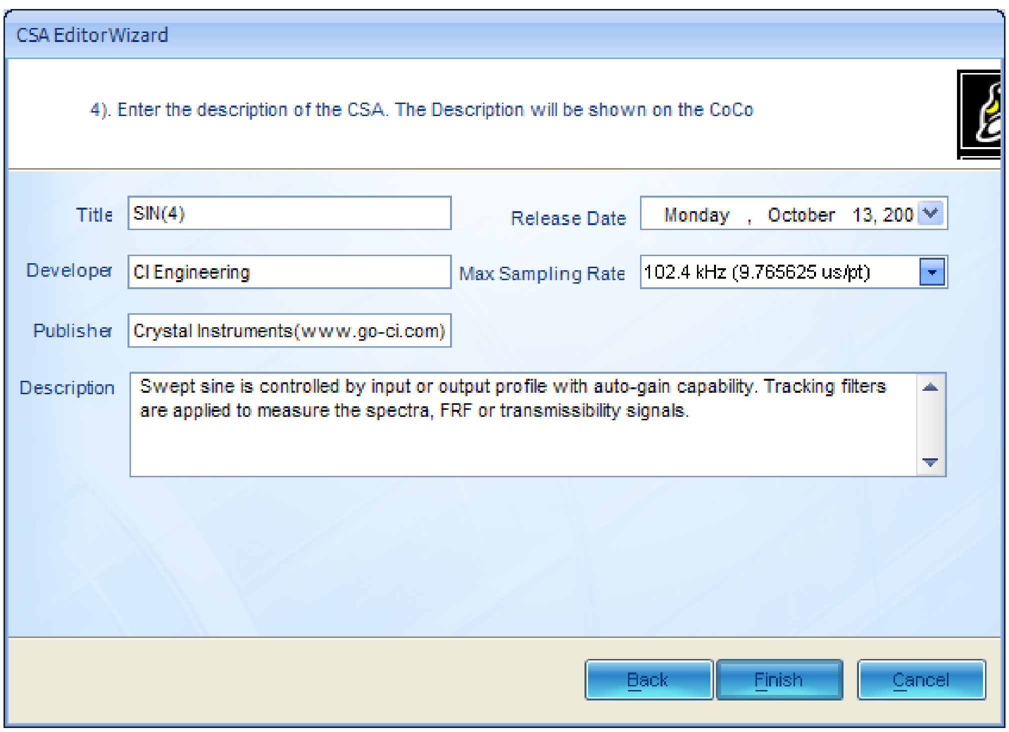


■ Figure 14. CSA Editor Wizard template selection

Next enable the minimum number of input channels that will be required for the measurement then click on the Next button.



■ Figure 15. Input channel selection.



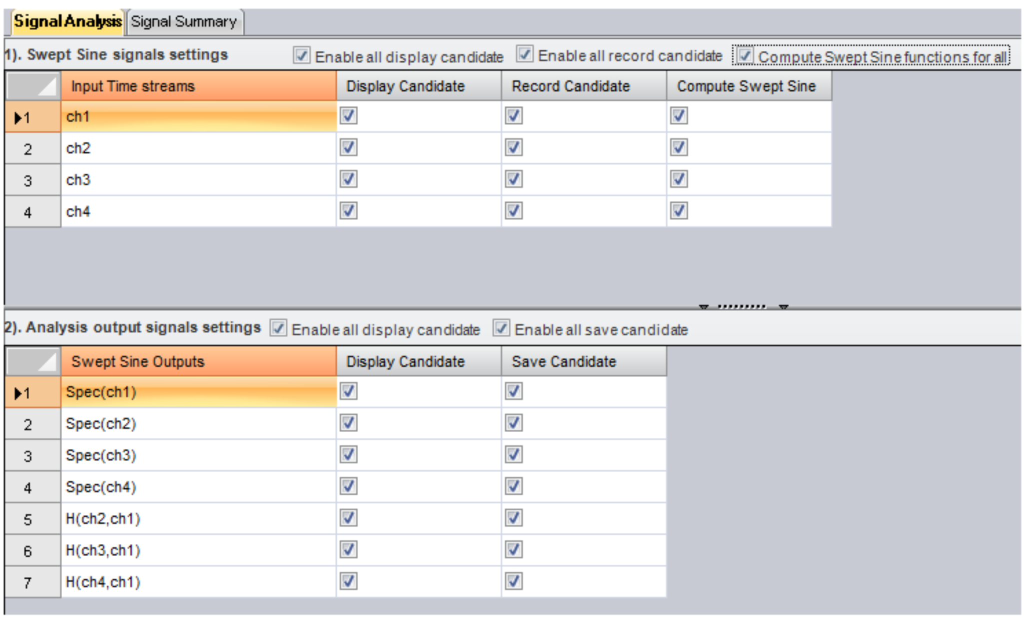
■ Figure 16. CSA description screen.

Next enter the CSA description. This information will be saved with the CSA file for future reference.

Next the software will show the Swept Sine Signal Settings display. In this screen you can select the save candidates, record candidates and Compute Swept Sine candidates. Display and record candidates are the same as other CSA types.

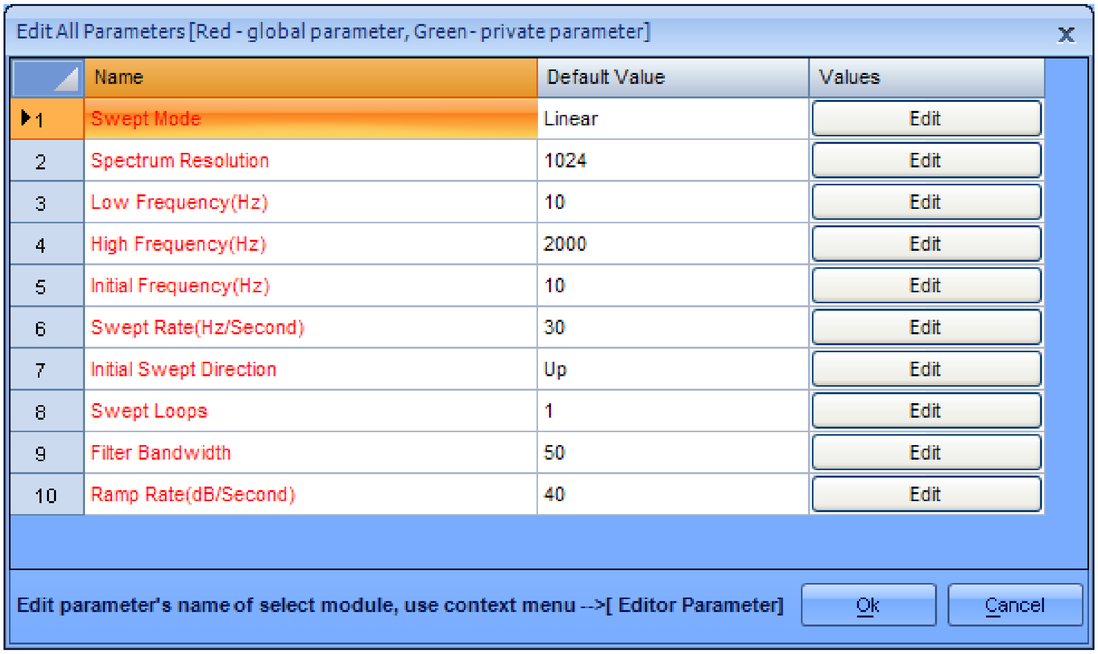
Compute Swept Sine candidates option allows you to choose which input channels will be used to compute the swept sine analysis calculations. As always you should enable the minimum number of measurements necessary to optimize computation resources and simplify the user interface on the CoCo.

The bottom of the screen shows the Analysis output signal settings. In this screen you can specify which signals will be available for display and for saving on the CoCo.



■ Figure 17. Swept Sine CSA Signal Analysis Screen.

Next click on the Parameters icon on the ribbon to open the Analysis Parameters screen. This screen allows you to modify the default parameters such as swept mode, spectrum resolution, frequency range, etc. These default values will be used on the CSA when it is loaded on the CoCo but can also be modified during run time.



■ Figure 18. Swept Sine CSA Parameters Screen.

You must validate and upload the CSA to the CoCo after editing.

CoCo Operations for Swept Sine

Select a CSA Project

This section discusses the CoCo settings that are specifically related to Swept Sine applications. For a complete explanation of these settings refer to the Basic CoCo Operation section. To run a Swept Sine CSA press the Analysis button and select the Swept Sine and SRS CSA Application Group. Then choose an Analysis Function from the CSA files on the CoCo to run.

Analysis Parameters - Window Type

The Analysis Parameter default values are defined by the CSA. If you want to change any parameters then press the F2 Param Button and select Analysis Parameters. Use the up, down and Enter buttons to modify the parameters, then press the F6 Apply button to save the new parameters.

■ Figure 19. Swept Sine Analysis Parameters.

Output Control

Select Output Control Method from the F2 Param button menu. This screen allows you to modify the output profile.

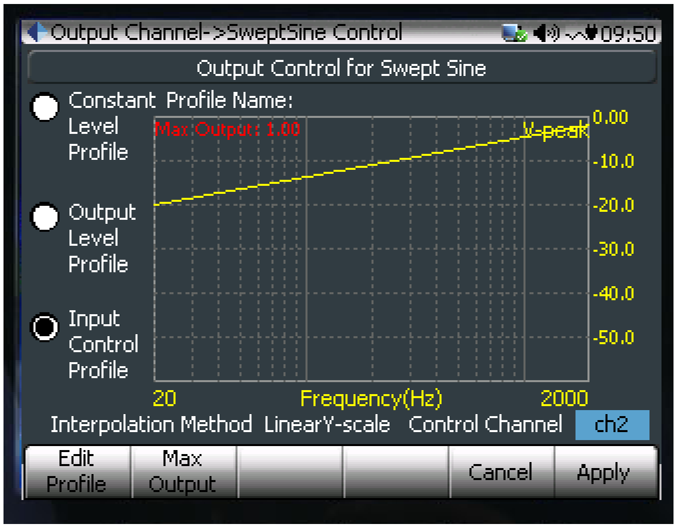
Constant Level Profile uses a sine tone with a fixed amplitude. Press the F1 Set Level button to define the amplitude level. The output level is displayed on the screen.

Output Level Profile uses a frequency variable amplitude. Press the F1 Edit Profile button to edit the profile. Use the up and down buttons to add, insert or delete points from the table and enter the amplitude peak values for each point, then press the Apply button. The output profile is displayed.

Input Control Profile uses an input channel to modify the output level. Select Input Control Profile and press the Enter Button. Then select which input channel will be used to compute the output level from the pop up menu. Next press the F1 Edit Profile Button to define the profile. The CoCo will modify the output level to try to force the input channel to match the profile during the sine sweep.

Max Output defines a maximum output level for the measurement to ensure the output does not exceed a safe level. This level will not be exceeded regardless of the output profile or control settings. Press F2 Max Output to define the maximum output level.

Press the F6 Apply Button to accept the Output Channel setting.



■ Figure 20. Output Channel Settings with an output level profile.

Manual Control

The **F3 Control Button** lets you control the operation during a swept sine measurement.

Reverse Direction changes the direction of the sweep from increasing to decreasing frequency or vise a versa.

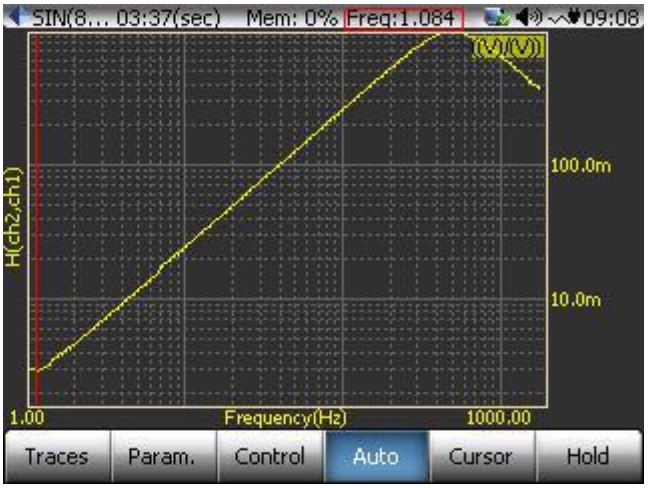
Jump to Frequency lets you enter a specific frequency to instantly change the source frequency.

Hold Sweep stops the advancement of the source frequency and dwells at the current frequency indefinitely. After the sweep is held you can use the **Release Sweep Control** command to continue the sweep measurement.

Pause and Ramp Down Output stops the sweep at the current frequency and gradually reduces the amplitude to zero. After the output is paused you can select Resume and Ramp Up Input to continue the measurement.

Start is used to begin the sine sweep measurement. After a measurement is started you can use the **Stop Control** command to stop the measurement.

Clear Display History deletes all data from the display but does not affect the progress of the output signal.



- Figure 21. Swept Sine Display showing current frequency on status bar.

3. ACOUSTIC DATA ACQUISITION: OCTAVE ANALYSIS

The Acoustics Data Acquisition software option includes Fractional Octave Filter Analysis, Sound Level Meters and Microphone Calibration functions.

The Fractional Octave Filter Analysis applies a bank of real-time octave filters to the input time streams and generates two types of signals at the same time: fractional frequency band signals, i.e., *octave spectra*, and the *RMS time history* of each filter band. The output of each real-time filter bank is in fact a 3D waterfall signal that is arranged in the x-axis as logarithmic frequency and z-axis as time. In the frequency direction, frequency weighting is applied. In the time axis, the time-weighting is applied.

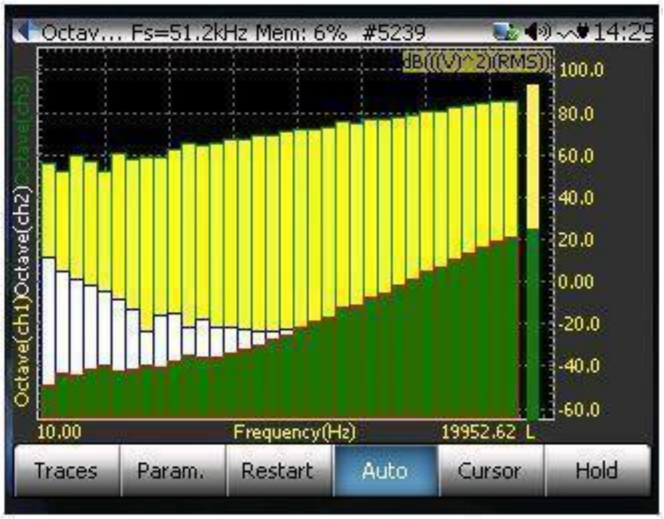
The Sound Level Meter (SLM) is a similar application to octave filters in the acoustic data acquisition. This application is also referred to as an Overall Level Meter. The SLM applies ONE frequency weighting filter to the input signal and time weighting to the output. Various measures are then extracted from both the input and output signals of this frequency weighting filter.

Fractional Octave Filter Analysis

Acoustics Analysis provides 1/N octave analysis using true real-time digital filters that conform to Conforms to ANSI std. S1.11:2004, Order 3 Type 1-D and IEC 61260-1995 specifications. A, B and C weighting filters can be applied to the input data. Output results are weighted or un-weighted RMS values. The output can be normalized with a calibration value. The results can be plotted on log or linear axes and exact or preferred frequency values are supported.

The Acoustics Analysis provides 1/N octave analysis using true real-time digital filters that conform to ANSI S1.11 and IEC 61260 specifications. Each band filter is designed in accordance to ANSI S1.11 and IEC 61260 specifications by transforming the original analog transfer function to the digital domain by means of the bilinear transform. The filter order can be specified and the frequency ratio can be calculated using the binary or decimal system.

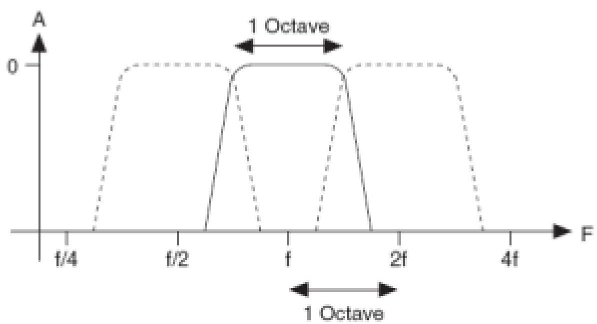
The RMS reading of each octave filter is usually represented by a “bar” in the spectrum plot. Keep in mind that the octave filters have “skirts” on both sides. They are not as straight as the bars depicts. The adjacent filters always overlap. Due to this reason, a sine tone at 1kHz will not only excite the filter with center frequency at 1kHz, but also all other filters. Figure 22 shows how the energy in each band is displayed on the octave spectrum plot using bars.



■ Figure 22. 1/3 octave filter banks.

Full Octave Filters

An octave is a doubling of frequency. For example, frequencies of 250 Hz and 500 Hz are one octave apart, as are frequencies of 1 kHz and 2 kHz.



■ Figure 23. Full octave filter shape.

Full octave analysis, i.e., 1/1 octave, displays the frequency characteristics of a signal by passing the signal through a bank of band-pass filters where the center frequency of each filter is one octave apart. If the lower and upper cutoff frequencies of a band-pass filter are f_L and f_H, then the center frequency, f_c can be determined with:

$$f_c = \sqrt{(f_L * f_H)}$$

The nominal frequency ratio G is determined by:

$$G = f_H/f_L$$

Two systems are used in the industry, Base-two or Base-ten systems. For base-two systems, G = 2. For base-ten systems, G = $10^{3/10}$. Base-ten system is used in CoCo.

The proportional bandwidth property will divide the frequency information uniformly over a log scale. It is very useful in analyzing a variety of natural systems. For example, the human response to noise and vibration is very non-linear and many mechanical systems have a behavior that is best characterized by proportional bandwidth analysis.

Fractional Octave Filters

To gain finer frequency resolution, the frequency range can be divided into proportional bandwidths that are a fraction of an octave. For example, with 1/3 octave analysis, there are 3 band-pass filters per octave where each center frequency is $10^{1/10}$ the previous center frequency.

In general, for 1/N octave analysis, there are N band pass filters per octave such that:

$$\frac{f_H}{f_L} = \left(10^{3/10}\right)^{1/N}$$

$$f_{c\,j+1} = f_{c\,j} * \left(10^{3/10}\right)^{1/N}$$

where 1/N is called the fractional bandwidth resolution.

For CoCo the equation and table below define the center frequency of each fractional filter.

$$f_c = 10^{3X/10N}$$

For example for 1/1 Octave (N =1) the first center frequency (index X = 1) is computed as

$$f_c = 10^{\frac{3*(-3)}{10*1}} = 0.125 \text{ Hz}$$

- Table 1. Octave center frequencies.

	1/1-Octave	1/3-Octave	1/6-Octave	1/12-Octave
Standard	IEC 225-1966 DIN 45651 ANSI S1.11-2004 Order 7 Type 1-D	IEC 225-1966 DIN 45651 ANSI S1.11-2004 Order 3 Type 1-D	N/A	N/A
X (index)	-3 □ ~□ 14	-10 □ ~ □ 43	-20 □ ~ □ 86	-40 □ ~ □ 172
Total number of Filters	18	54	107	213
f_c (Hz)	0.125 – 16k	0.1 – 20k	0.1 – 20k	0.1 – 20k

Nominal center frequencies (midband frequencies)
Nominal center frequencies are "round" numbers that were inherited from the old analog octave filters. They are rounded midband frequencies for the designation of band pass filters. The nominal midband frequencies for 1/1-octave and 1/3-octave are listed in the ANSI S1.11-2004 Annex A. The standard also describes how to decide the nominal midband frequencies for other fractional octave bands.

The exact center frequency of the filter band is usually not the same as that of nominal frequency. For example, in a 1/3 octave, the exact center frequencies 794.33Hz, 1000Hz and 1258.9Hz are used to correspond to the filters with nominal frequencies 800Hz, 1000Hz and 1250Hz.

Band Edge Frequencies of Fractional Filters
The low and high edge frequencies of a filter can be calculated based on the frequency ratio, G and the fractional octave resolution N (=1, 3, 6, 12…)

$$Lower\ Edge\ Frequency \quad f_L = f_c * \left(10^{3/10}\right)^{-1/2N}$$
$$Upper\ Edge\ Frequency \quad f_H = f_c * \left(10^{3/10}\right)^{1/2N}$$

The bandwidth of the filter is: BW = $f_H - f_L$

When starting or resetting the filtering operation of the fractional-octave filters, a certain time is required before the measurements are valid. This time is called the *settling time* and is related to the bandwidth of any particular filter. The lowest frequency band has the smallest bandwidth and defines the settling time required before you can consider the complete fractional-octave measurement valid. A good rule of thumb is that the settling time is approximately five divided by the bandwidth.

$$Settling\ time = \frac{5}{BW} = \frac{5}{f_H - f_L}$$

Note the settling time depends on the bandwidth which changes with center frequency. A narrower filter and a lower frequency band requires a longer settling time.

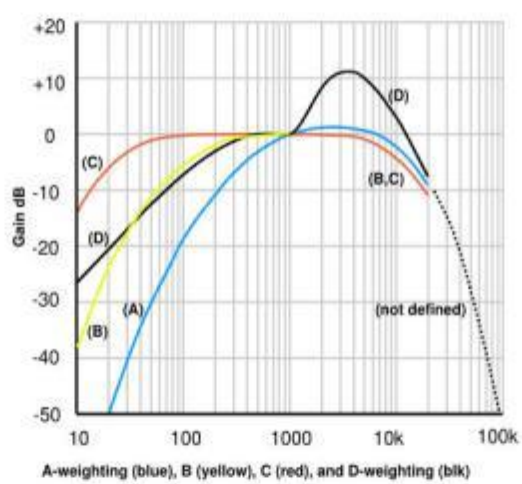
Analysis Frequency Range

In CoCo, the user can decide the analysis range by changing the lowest and highest f_c as the Analysis Parameters:

Analysis Range	1/1 Octave	1/3 Octave	1/6 Octave	1/12 Octave
Lowest f_c (Hz)	0.125 1 8	0.1 1 10 100	0.1 1 10 100	0.1 1 10 100
Highest f_c (Hz)	1000 4000 16000	1000 2000 5000 10000 20000	1000 2000 5000 10000 20000	1000 2000 5000 10000 20000

Frequency Weighting

The human hearing system is more sensitive to some frequencies than others, and its frequency response varies with level. In general, low frequency and high frequency sounds appear to be less loud than mid-frequency sounds, and the effect is more pronounced at low pressure levels, with a flattening of response at high levels. Octave analysis and sound level meters therefore incorporate weighting filters, which reduce the contribution of low and high frequencies to produce a measurement that corresponds approximately to what we hear.



- Figure 24. Frequency weighting filter shapes.

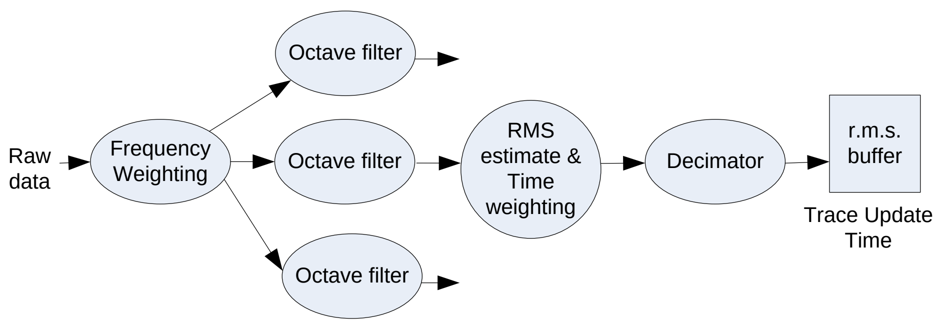
CoCo provides A, C, Z weightings conforming to IEC 61672-1 2002 and B weighting conforming to IEC 60651 in both of Octave analysis and Sound Level Meter. The Frequency weighting in the octave filters will affect the results of all filter bands.

Time or RPM based RMS Trace of the Octave Filters

The ANSI and IEC standard do not require storing the time history of the band pass filter output. However the user may be interested in viewing this information. On the CoCo the RMS history of all the band pass filters are stored, in the RMS quantity. Below is the description about how the RMSs. history is calculated.

The RMS history can be stored against one of two variables: Time or RPM.

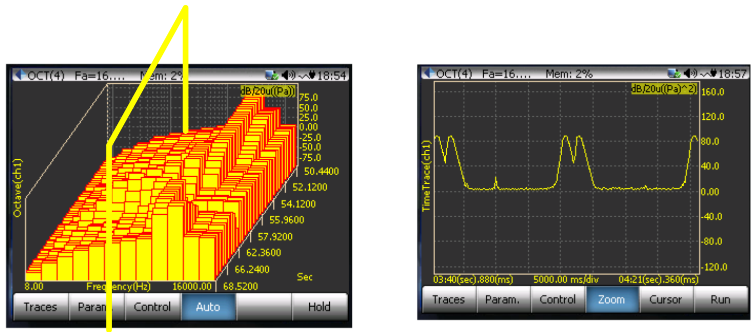
Both the input and output of a digital filter are a series of data points. While it requires excessive memory to keep all the time data of all the filters, it is useful to keep the so called RMS history of each filter output. The RMS time history is computed after the time weighting averaging operation as shown in Figure 25.



■ Figure 25. RMS time history calculation.

The *Decimator* is used so the user can choose the length of time to save the RMS data. For example, given a buffer length of 1024, a *Trace Update Time* of 5 ms will keep about 5 seconds of RMS history; if this update time is set to 5 seconds, it will record 1.4 hours of RMS history.

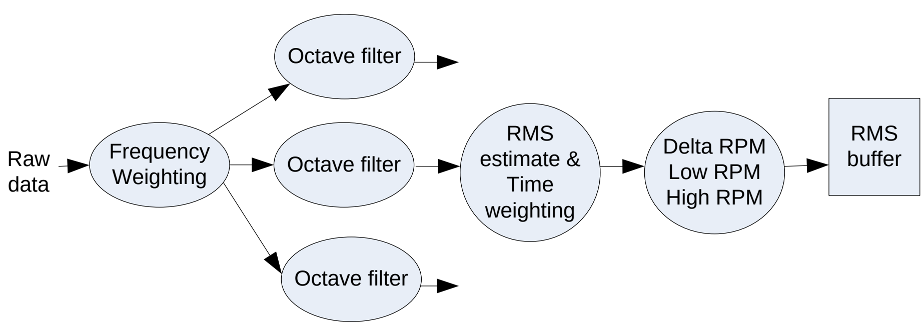
Figure 26 shows the 3D waterfall display of a 1/1 octave filter output. If *a cut* is made in the Z axis direction, the result will be an octave spectrum. If *a cut* is made in the X-axis, the result will be called a *Time Trace*.



■ Figure 26. A cut of 3D Waterfall of octave filter output (left) maps to an RMS time trace (right)

The *Time Trace* stores the history of the RMS of each filter output. The spacing between two points of the *Time Trace* is called *Trace Update Time, in seconds*. On CoCo, one *Time Trace* is allocated for each channel for display. Keep in mind that this buffer of Time Trace is the output of a specific filter, the user can change the center frequency of the filter for the Time Trace during run time. In the other words, this time trace display buffer will change its content completely when the user switches the *Time Trace Frequency* from one to another.

Alternatively the RMS trace can be stored using RPM as a variable. This method is particularly useful in the automotive NVH applications. The picture below shows how one of the outputs of filters can be stored in RPM trace.



■ Figure 27. Store RPM based RMS traces.

Exponential and Linear Averaging

Linear average: The Linear average method uses a fixed time period to sum up the historical power value of each filter and then takes the square-root to calculate the averaged RMS value. In Linear average, the RMS trace update time is governed by the time period of the averaging. For each time period of averaging, one RMS value per frequency bin is produced.
Exponential average: Exponential average applies an exponential time constant to the historical power values of each filter and takes the square-root of the averaged power value. A time constant of 0.125 seconds is equivalent to "Fast" averaging and 1.0 second is equivalent to "Slow"

averaging in the acoustics. In exponential average, the RMS trace update time is independent of the time constant.
Peak Hold: Peak Hold retains the maximum value in each frequency bin over the period of time since last "start" or "restart".

As we discussed previously, each filter may have different settling time.

Measurements available to CoCo in Octave Analysis mode

The following measurement quantities are available on the CoCo in the octave measurement mode.

Time streams of input channels
In CoCo, time domain data is always available in the form of long time history. The user can view and record the time signals. The limitation is that the sampling rate of the time signals cannot be arbitrarily changed. It is always set internally by the system based on analysis frequency range, i.e., the highest center frequency of the filter bands.

Octave Spectra
Each input channel will have an octave spectrum.

RMS Trace
Each input channel will have one RMS trace to display the RMS history. This RMS is the output of the filter for a specific band. The RMS trace is defined as the Time-RMS trace or RPM-RMS trace at the CSA Editor level. You cannot change between Time and RPM based for a specific CSA.

CSA Editor Operation

This section describes the operation of the CSA Editor related to octave analysis. For general operation of the CSA Editor refer to the CSA Users Manual.

CSA Editor Wizard

This section summarizes how to create a CSA project for octave analysis in the CSA Editor. We strongly recommend that you read the CSA Edit User's Manual to gain more detailed information before proceeding with this chapter.

To start, click on the CSA Editor icon in the upper-right corner in EDM and start the CSA Editor. The CSA Editor Wizard dialog box will be displayed. First select the CoCo type and number of channels of the CoCo that this CSA project will support. The number of channels must be equal to or less than the number of physical input channels on the CoCo hardware. Click on the Next Button.

Next select the Octave Analysis template and either time or RPM based mode. Time based octave analysis will compute the results based on time averages. This is the most common application. RPM based mode computes sound level versus the RPM measured by a tachometer for rotating equipment. Click on the Next Button.

Next select the number of input channels to compute the analysis functions. In general it is best to select the minimum number of channels to conserve computational resources for the CSA. This number can be less than or equal to the number of physical input channels on the CoCo hardware. Click on the Next Button.

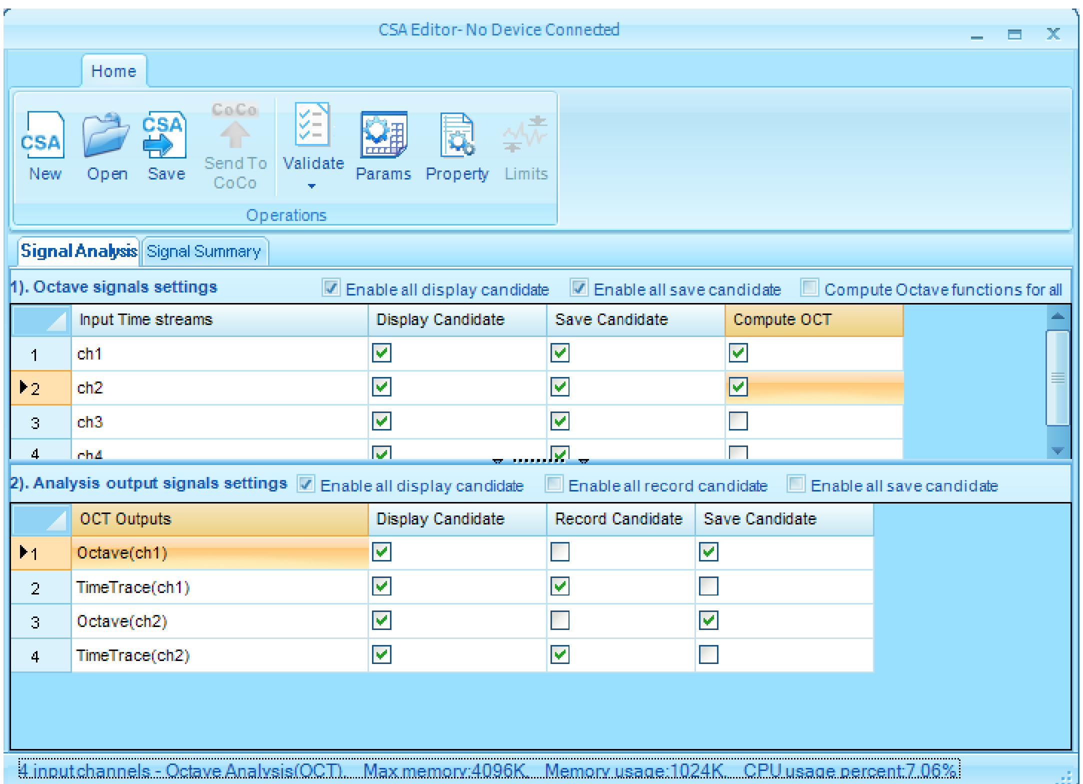
Next enter the description of the CSA. This data will be saved with the CSA and be shown on the CoCo device when the CSA is opened. Click on the Next Button.

Select Signal Candidates

Next the Signal Analysis Display is shown. This display lets you select which signals will be display candidates, save candidates and which will be used to compute the octave analysis. Always select the minimum number of candidates required to conserve computational resources. Note that when a signal is selected as a candidate then it is available on the CoCo hardware, meaning that it can be displayed or save, etc. If the signal is not specified as a candidate in the CSA then it will not be available on the CoCo hardware.

Octave signal settings lets you specify the display, save and octave computation candidates for the input time streams.

Analysis output signal settings let you specify the display, record and save candidates for the computed octave data.



■ Figure 28. Signal Analysis tab in CSA Editor.

After all the candidate selections are made, a summary can be viewed by selecting the Signal Summary.

Signal Analysis | **Signal Summary**

	Name	Display Candidate	Record Candidate	Save Candidate	Description
1	ch1	True	True	True	Time stream of ch1
2	ch2	True	True	True	Time stream of ch2
3	ch4	True	True	True	Time stream of ch4
4	ch3	True	True	True	
5	Octave(ch1)	True	False	True	Octave Spectrum
6	TimeTrace(ch1)	True	True	False	Filter Output r.m.s. Trace
7	Octave(ch2)	True	False	True	Octave Spectrum
8	TimeTrace(ch2)	True	True	False	Filter Output r.m.s. Trace

■ Figure 29. Signal Summary tab in CSA Editor.

Analysis Parameters

After the signals are selected the next step is to specify the Parameters by clicking on the Param icon in the CSA Editor Ribbon. The Parameters display lets you change the computation parameters such as octave resolution, low and high frequency, averaging type, etc. Click on the Edit Button next to each item to edit the parameter. Click on Ok to save the updated parameters.

Time Based Acquisition Analysis Parameters.

The parameters are different for time and RPM based acquisition. Figure 30 shows the parameters for time based acquisition. These parameters become the default values when the CSA is uploaded to the CoCo hardware. To change a parameter click on the Edit button.

Edit All Parameters [Red - global parameter, Green - private parameter]

	Name	Default Value	Values
1	Octave Resolution	1/3	Edit
2	Low Frequency(Hz)	10	Edit
3	High Frequency(Hz)	10000	Edit
4	Average Type	Exponential	Edit
5	Time Weighting(s)	0.125	Edit
6	Frequency Weighting	Z	Edit
7	Time Trace Frequency(Hz)	Overall	Edit
8	Trace Update Times(s)	0.02	Edit

Edit parameter's name of select module, use context menu -->[Editor Parameter] Ok Cancel

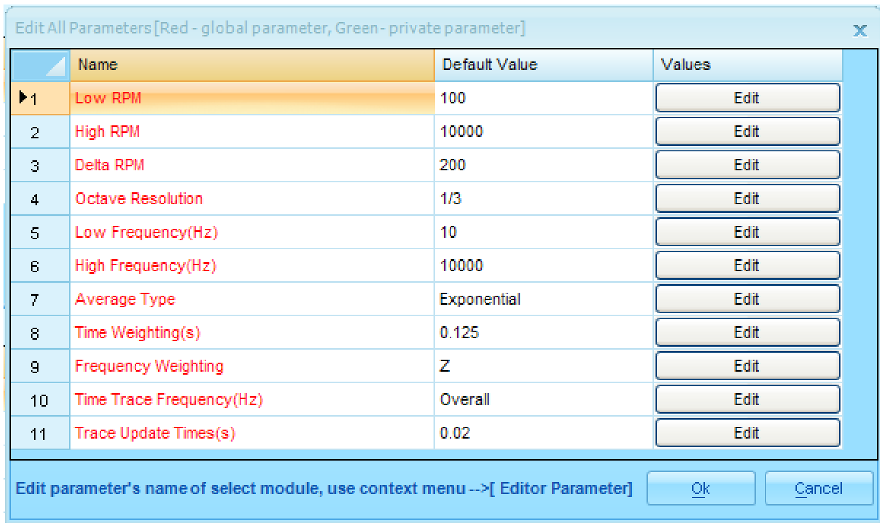
■ Figure 30. Edit Parameter window in CSA Editor for time based acquisition.

Octave Resolution – defines the octave resolution including: 1/1, 1/3, 1/6 and 1/12
Low/High Frequency (Hz) – defines the low and high frequency (span) of the measurement in Hz.
Average Type – defines the averaging type including: exponential, linear and peak hold.

Time Weighting – defines the time weighting for exponential averaging.
Frequency Weighting – defines the frequency weighting including: A, B, C or Z.
Time Trace Frequency (Hz) – defines which center band frequency or overall or frequency weighted band is used to plot time traces.
Trace Update Time – defines the time trace display duration. Select a larger update time to create a longer time trace display duration.

RPM Based Acquisition Analysis Parameters

Figure 31 shows the rpm based acquisition analysis parameters.



Edit All Parameters [Red - global parameter, Green - private parameter]

	Name	Default Value	Values
1	Low RPM	100	Edit
2	High RPM	10000	Edit
3	Delta RPM	200	Edit
4	Octave Resolution	1/3	Edit
5	Low Frequency(Hz)	10	Edit
6	High Frequency(Hz)	10000	Edit
7	Average Type	Exponential	Edit
8	Time Weighting(s)	0.125	Edit
9	Frequency Weighting	Z	Edit
10	Time Trace Frequency(Hz)	Overall	Edit
11	Trace Update Times(s)	0.02	Edit

Edit parameter's name of select module, use context menu -->[Editor Parameter] Ok Cancel

■ Figure 31. RPM based acquisition analysis parameters.

Low RPM – defines the low RPM level for display limits and also the limits for trigger mode.
High RPM – defines the high RPM level for display limits and also the limits for trigger mode.
Delta RPM – defines the resolution of RPM measurements. Note that a large RPM span and high resolution (small Delta RPM) require more computation resources.

The other parameters are the same as the time based acquisition parameters.

Validation, Save and Upload

After the CSA Wizard is complete and the CSA file is created, connect the host PC to the CoCo device and press the Validate icon to validate the CSA. It may take a few minutes to finish the validation.

■ Figure 32. Validate and Send to CoCo icons in the CSA Editor Ribbon.

The validation process analyzes the CSA file for internal consistency and estimates the required DSA resources required to run the CSA file on the CoCo device.

If the Validation passes, then press *Send to CoCo* command in the Validation dialog box to send the CSA project file to CoCo. Alternatively you can manually upload it to CoCo. The uploaded CSA will be classified into the Acoustic Analysis Application Group and can then be opened on the CoCo hardware.

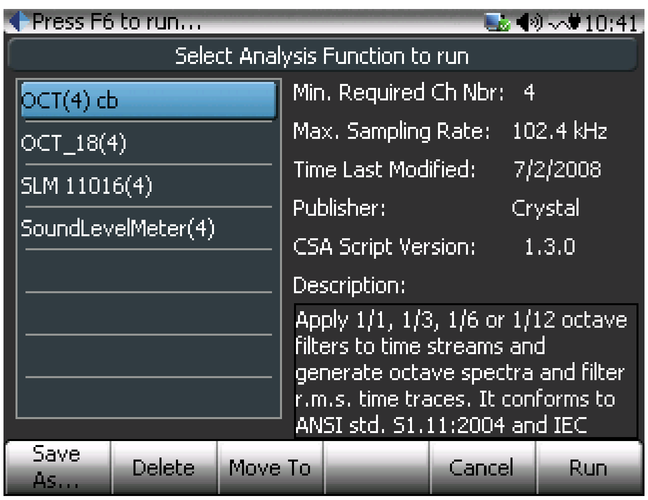
CoCo-80 Operation

This section describes the operations of CoCo that are specifically related to octave analysis. For general operations of CoCo refer to the Basic Users Manual.

Select a CSA Project

After an Octave Analysis CSA is created from the CSA Editor and downloaded to the CoCo Hardware then it can be opened on the CoCo. Alternatively you can also open the default Octave Analysis CSA scripts on the CoCo Hardware. To open a CSA press the Analysis Button, then select the Acoustic Analysis Application Group. All Octave Analysis CSA files are automatically placed in this group to help organize the CSA files on the CoCo.

Next select an octave analysis CSA from the list. Octave analysis CSA files have the name OCT() by default.



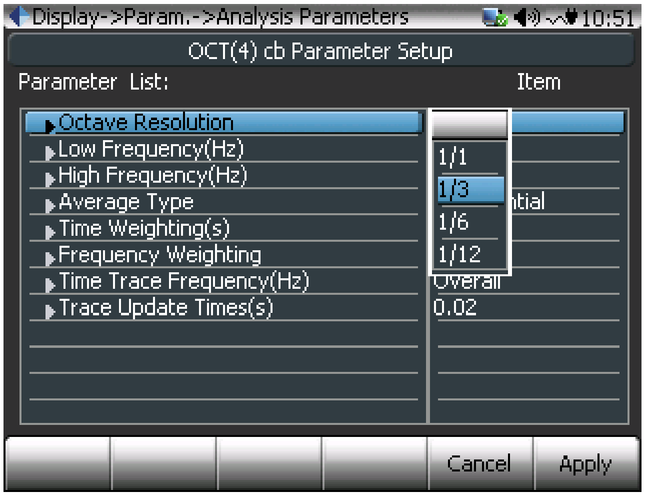
■ Figure 33. Select Analysis Function display.

Analysis Parameters for Octave Analysis

To set the parameters press the Param. Button in the signal display window.

Analysis Parameters

Analysis Parameters lets you modify the settings for the data acquisition. The default parameters are set in the CSA Editor. To change a parameter move the cursor to the parameter using the Up and Down Buttons and press the Enter Button, then set the parameters.



■ Figure 34. Analysis Parameters.

Octave Resolution:1/1, 1/2, 1/6, or 1/12
Low Frequency (Hz): set the low frequency of the fist band.
High Frequency (Hz): set the high frequency of the last band.
Average Type: linear, exponential or peak hold
Time Weighting(s): 0.035 to 1000 seconds
Frequency Weighting: A, B, C or Z
Time Trace Frequency (Hz): overall, weighting, 0.125 to 16000 Hz.
Trace Update Time(s): 0.005 to 600 seconds. Conforms to weighting time when the averaging type is linear.
The time weighting parameter can be selected so it corresponds to "Fast" or "Slow" weighting mode. For "Fast" weighting mode, set the time weighting constant to 0.125 seconds; for "Slow" mode, set it to 1.0 seconds.

Acquisition Mode

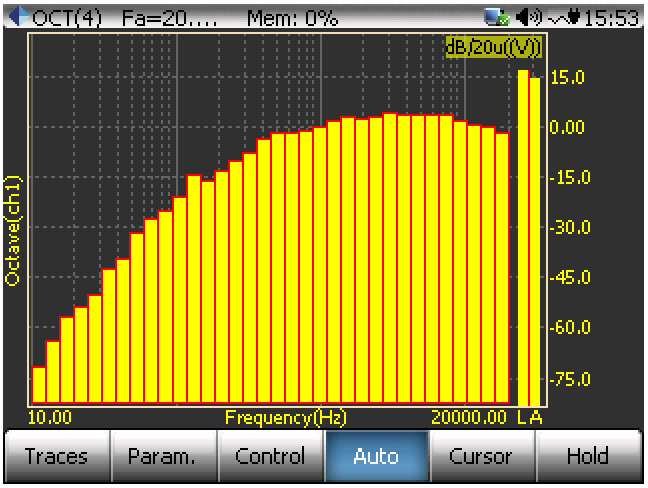
Acquisition Mode controls the conditions under which the data is acquired. It can refer to triggered mode for time based acquisition or RPM mode for RPM based measurement. Refer to the CoCo Basic User Manual for trigger mode acquisition. Refer to the Order Tracking section which follows this section for a detailed description of the tachometer RPM setup.

Displays

This section describes the displays that are unique to the Octave Analysis template. Refer to the Basic User Manual for a description of displays that are common to the other templates.

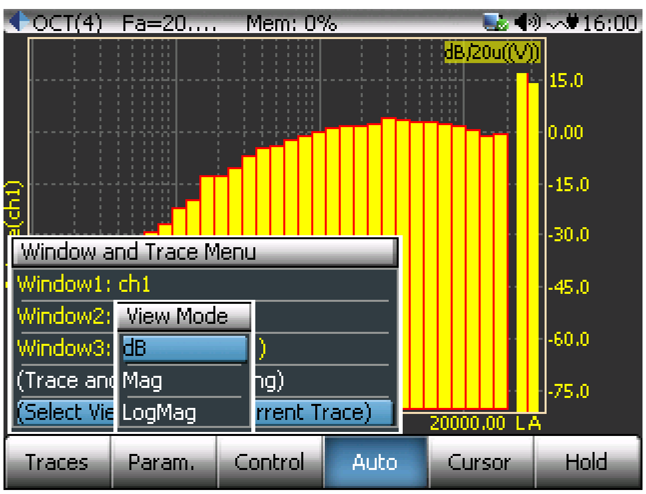
Octave Display

The octave display shows the octave bands with the overall level indicated with 'L' and the frequency weighted level indicated with 'A, B, C or Z'.



■ Figure 35. Octave display.

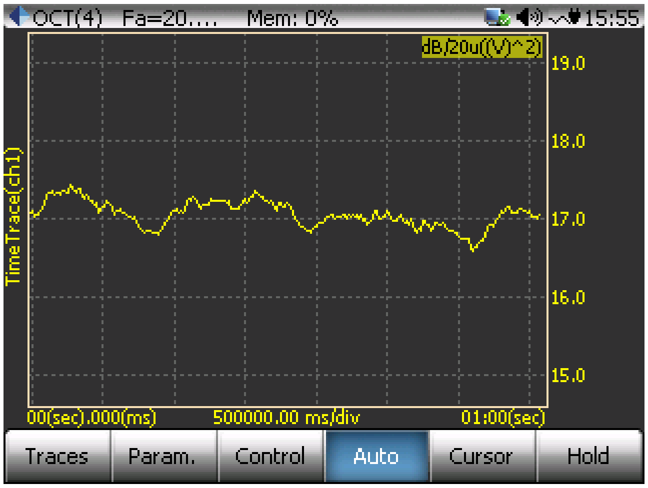
The vertical units can be change by selecting Select View Mode for Current Trace under the Trace Button.



■ Figure 36. Trace View Mode.

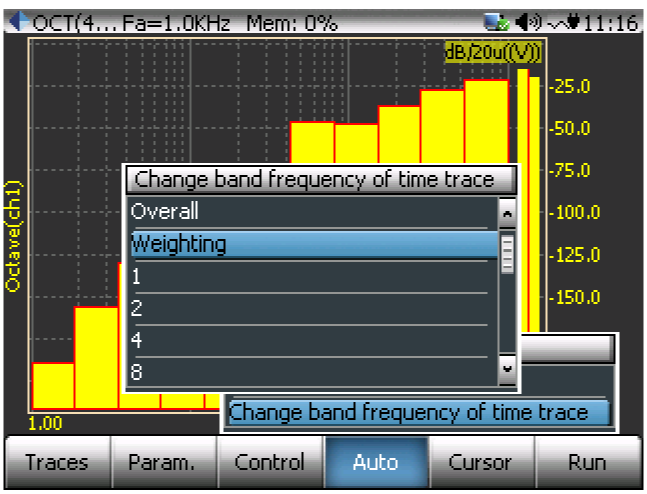
Time Trace Display

A time trace display shows the sound level of one of the octave bands including the overall or frequency weighted level vs. time. The trace update time can be changed under the Analysis Parameters to show a longer or shorter duration.



■ Figure 37. Time trace display.

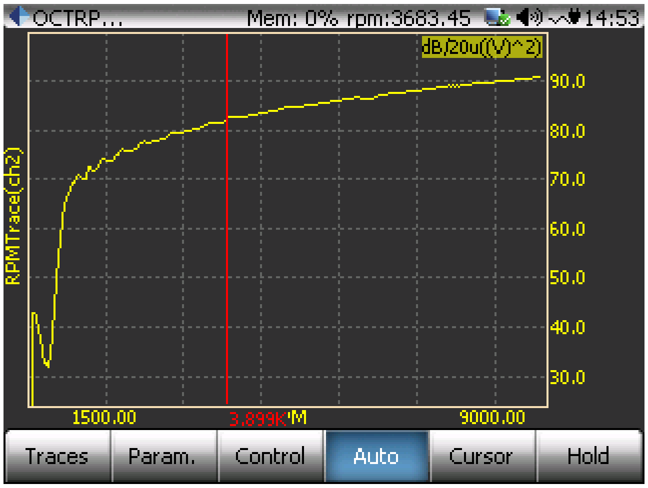
The time trace frequency band can be changed by pressing the Control Button and selecting Change band frequency of time trace. Next chose the overall, weighted or any of the octave band center frequencies. The time trace will then display the selected band.



■ Figure 38. Change time trace frequency band.

RPM Based Level Display

An RPM based level display can be added by selecting RPMTrace. It shows the level of one octave band or the overall or frequency weighted band vs. RPM. This trace type is only available for RPM based Acquisition Mode.



■ Figure 39. RPM based level display.

To change the frequency band of the time trace press the **Auto Button** and select **Change band frequency of time trace** then select from Overall, Weighting or one of the band center frequencies.

■ Figure 40. Change band frequency of time trace.

Make a Measurement

After the Analysis Parameters and displays are configured you are ready to make a measurement. Press the Run Button. To stop the display update press the Hold Button. To reset the averaging press the Control Button and select Restart.

You can use the Cursor Button to add cursors for analysis and use the Save Button to save the results. Refer to the CoCo Basic User Manual for a description of these features.

4. ACOUSTIC DATA ACQUISITION: SOUND LEVEL METER

An analog sound level meter measures the sound pressure level. The standard sound level meter is more correctly called an *exponentially averaging sound level meter* as the AC signal from the microphone is converted to DC by a RMS circuit and thus it must have a time-constant of integration. This is referred to as time-weighting. Three of these time-weightings have been standardized, 'S' (1s) originally called Slow, 'F' (125 ms) originally called Fast and 'I' (35 ms) originally called Impulse. The output of the RMS circuit is linear in voltage and is passed through a logarithmic circuit to give a readout in decibels (dB). This is 20 times the base 10 logarithm of the ratio of a given root-mean-square sound pressure to the reference sound pressure. Root-mean-square (RMS) sound pressure is obtained with a standard frequency weighting and standard time weighting.

With the advent of digital technology and increasing accuracy of the electronic circuits, the sound level meter functions are more recently calculated in the digital domain. One of the important factors of such implementation is that the instrument must provide very high dynamic range so that both weak and strong signals can be calculated and observed. CoCo provides 130dB dynamic range. High dynamic range is one of the most important measures of the quality of an acoustic analyzer.

A traditional sound level meter only includes the 1/1 and 1/3 octave filters. In the CoCo system octave analysis is provided in addition to the other analysis functions providing more flexibility and computation power than a traditional sound level meter.

You should use Octave Analysis as the template to create a CSA projects when fractional octave analysis is needed. In both the Octave Analysis and Sound Level Meter templates the user can see the frequency weighted readings (such as dBA). But the reading results may be slightly different when comparing Octave Analysis and Sound Level Meter results. This is because the data processing flow in octave filter analysis and sound level meter is computed differently. In the octave analysis, the dBA, i.e., the A-weighted sound level is computed by applying the frequency weighting function to the output of each individual filter bank; while in SLM, the A-weighted sound level is created by applying the A-weight filter in the entire time domain. The SLM template should be used to obtain the dBA or similar overall readings for most sound studies that might be compared to results taken with a traditional sound level meter because the computation is more similar to that of a traditional sound level meter.

Terms and Definitions

In this section we will define the terminology used in the SLM software options.

Reference sound pressure It is conventionally chosen as 20 µPa. This is the threshold of hearing of the average person and is used to compute the sound pressure level in the dB scale.

Sound pressure level (in dB)

Sound pressure level (dB) is defined as twenty times the logarithm to the base ten of the ratio of the RMS of a given sound pressure to the reference sound pressure. Sound pressure level is expressed in decibels (dB); symbol L_p.

Peak sound pressure
The peak sound pressure is the greatest absolute instantaneous sound pressure during a stated time interval.

Peak sound level (in dB)
The peak sound level (dB) is defined as twenty times the logarithm to the base ten of the ration of a peak sound pressure to the reference sound pressure, peak sound pressure being obtained with a standard frequency weighting. (example letter symbols are L_{peak}, L_{Cpeak})

Frequency weighting
Frequency weighting is the difference between the level (dB) of the signal indicated on the display device and the corresponding level of a constant-amplitude steady-state sinusoidal input signal, specified in the IEC or ISO standards as a function of frequency. It accounts for A, B and Z frequency weighting discussed in the previous section.

Time weighting
Time weighting is an exponential function of time, of a specified time constant, that weights the square of the instantaneous sound pressure. This is the same as exponential averaging in the time domain to the instantaneous sound pressure.

It is a continuous averaging process that applies to the output of a frequency weighting filter or one of the fractional filters. The amount of weight given to past data as compared to current data depends on the exponential time constant. In exponential averaging, the averaging process continues indefinitely.

In a sound level meter the time weighting exponential averaging mode supports the following time constants:

Slow uses a time constant of 1,000 ms. Slow averaging is useful for tracking the sound pressure levels of signals with sound pressure levels that vary slowly.
Fast uses a time constant of 125 ms. Fast averaging is useful for tracking the sound pressure of signals with sound pressure levels that vary quickly.
Impulse uses a time constant of 35 ms if the signal is rising and 1,500 ms if the signal is falling. Impulse averaging is useful for tracking sudden increases in the sound pressure level and recording the increases so that you have a record of the changes.
User Defined allows you to specify a time constant suitable for your particular application.

Time-weighted sound level (in dB)
This is twenty times the logarithm to the base ten of the ratio of a given RMS sound pressure to the reference sound pressure, RMS sound pressure being obtained with a standard frequency weighting and standard time weighting. (example letter symbols are L_{AF}, L_{AS}, L_{CF}, L_{CS})

Maximum and minimum time-weighted sound level (in dB)
This is the greatest and lowest time-weighted sound level within a stated time interval. (example letter symbols are L_{AFmax}, L_{ASmax}, L_{CFmax}, L_{CSmax}, L_{AFmin}, L_{ASmin}, L_{CFmin}, L_{CSmin})

Time-average sound level (equivalent continuous sound level) (in dB)

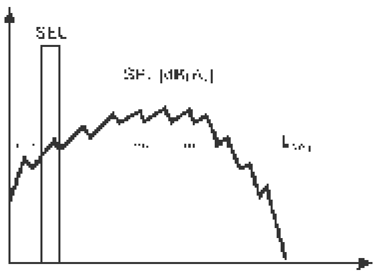
This is twenty times the logarithm to the base ten of the ratio of a RMS sound pressure during a stated time interval to the reference sound pressure, sound pressure being obtained with a standard frequency weighting. (example letter symbols are L_{Aeq}, L_{Ceq})

Sound exposure

This is the time integral of the square of sound pressure over a stated time interval or event. Sound exposure is used to measure high-level, short duration noises and to study their effects on humans.

Sound exposure level (in dB)

Sound exposure level is the total sound energy of a single sound event and takes into account both its intensity and duration. Sound exposure level is the sound level you would experience if all of the sound energy of a sound event occurred in one second. This normalization to duration of one second allows the direct comparison of sounds of different durations.



■ Figure 41. Sound exposure level illustration.

Figure 41 shows the relationship between the Sound Exposure Level (SEL), the Sound Pressure Level (SPL), and the Leq. The Leq is the constant level needed to produce the same amount of energy as the actual varying sound (the SPL).

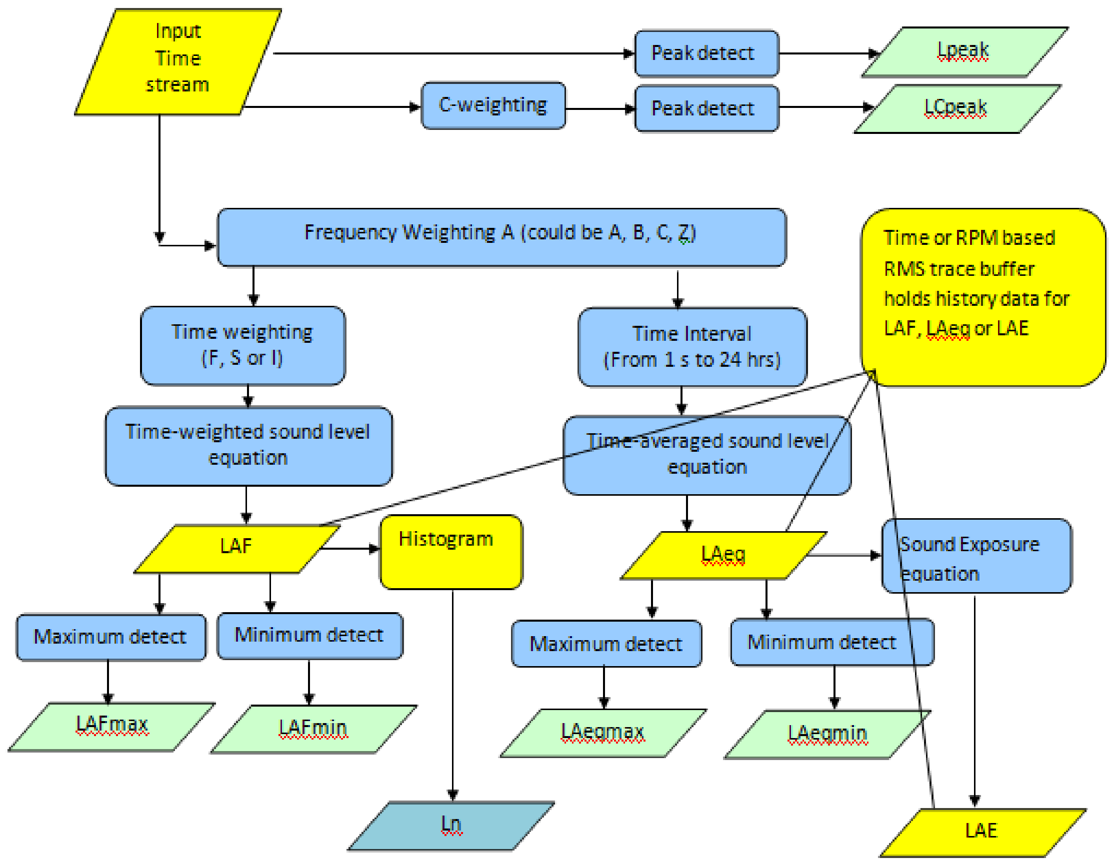
The SEL is the Leq normalized to 1 second. It is what the Leq would be if the event occurred over a one second duration.

Statistical Level (L_N)

LN is defined as the sound pressure level which is exceeded N% of the time over the duration of a measuring time interval. L0 is the maximum level over the duration of the measurement. L100 is the minimum.

Data Processing Diagram

Figure 42 shows the data processing diagram for ONE input channel for all the SLM measurements when A-weight is applied.



■ Figure 42. Sound level meter computation diagram.

In the SLM measurement, after the digitized data comes in, it is split into three paths: one goes to frequency weighting A, B, C or Z and one goes to C weighting or no weighting. The peak detection is computed from the output of C weighting or no weighting. The output of frequency weighting (A, B, C or Z) is further split into two paths. The first will go to a *time weighting function* which is more or less equivalent to an exponential averaging mode to calculate LAF; the second path goes to a time averaging function, which is equivalent to a linear averaging mode to calculate Leq.

With A-weighted applied as shown in the example, the list of symbol to be used by this instrument is:

Symbol of Measured Values	**Description**
LAF	A-weighted, F time-weighted sound level
LAFmax	Maximum A-weighted, F time-weighted sound level
LAFmin	Minimum A-weighted, F time-weighted sound level
LCpeak	Peak C sound level, greatest absolute instantaneous C-weighted sound pressure level
Lpeak	Peak sound level, greatest absolute instantaneous sound pressure level
LAeq	A-weighted, time-average sound level (equivalent continuous sound level)
LAeqmax	Maximum A-weighted, time-average sound level (equivalent continuous sound level)
LAeqmin	Minimum A-weighted, time-average sound level (equivalent continuous sound level)
LAE	A-weighted sound exposure level
L_N (N = any integer between 0~100)	Statistical Level general term
L1, L5, L50, L95....	Statistical Levels with specific N values. The sound level exceeds this level 1, 5, 50 or 95 percent of the time for the duration of the measurement.

Measurements available to CoCo in SLM mode

There are two ways to view sound level measurements: instantaneous SLM measures and RMS history. Instantaneous SLM measures represent the most current value of the measurement. RMS history not only shows the most current value, but also a record of historical values against time or RPM. Some of the measures allow only instantaneous values others allow both.

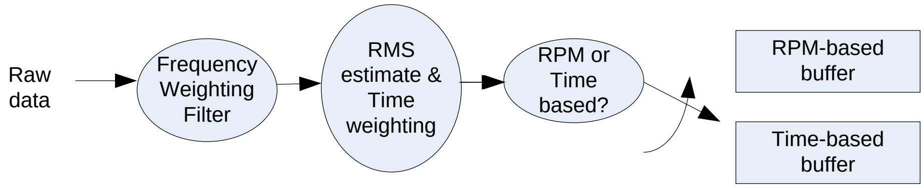
The following measurement quantities are available to CoCo in the measurement.

SLM Measures

The following SLM measures are available for real-time reading and can be saved as a data structure for future review.

Time Weighted Sound Levels

In CoCo, time weighted sound level is the output of frequency-weighting and then time weighting filters. Time weighting serves an exponential averaging operator. The computation is illustrated in Figure 43.



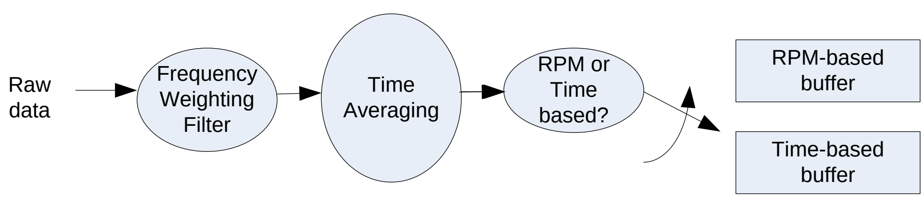
■ Figure 43. Time weighting sound level computation and storage against RPM or Time

The table below shows the symbols for the time-weighted sound level.

Symbol used for time weighted value		**Frequency Weighting**			
		Z	**A**	**B**	**C**
Time Weighting	**F(Fast)**	L_{ZF}	L_{AF}	L_{BF}	L_{CF}
	S(Slow)	L_{ZS}	L_{AS}	L_{BS}	L_{CS}
	I(Impulse)	L_{ZI}	L_{AI}	L_{BI}	L_{CI}
	Custom	L_{ZC}	L_{AC}	L_{BC}	L_{CC}

Time Averaged Sound Levels

In CoCo, time averaged sound level is the output of frequency-weighting and then time average operation. Time average serves a linear averaging operator. Figure 44 illustrates the computation.



■ Figure 44. Time averaged sound level computation.

The table below shows the symbols for time-average sound level. In the time averaging sound level measurement, Frequency weighting can be selected as A, B, C or Z. The time interval for time averaging can be set to any value between 1 second and 24 hours.

Frequency Weighting	Z	A	B	C
Symbol	L_{eq}	L_{Aeq}	L_{Beq}	L_{Ceq}

Peak sound level

Only C-weighted and un-weighted are available for peak sound level. This is required by the standards.

Symbol	L_{peak}	L_{Cpeak}

Sound exposure level

Sound exposure level and time-average sound level have the same frequency weighting and same time interval.

Frequency Weighting	Z	A	B	C
Symbol	L_E	L_{AE}	L_{BE}	L_{CE}

Statistical level: value reading

Any statistical level L_N is the sound level which is exceeded for N% of the defined measurement duration.

Symbols for L_N, N = 1, 5, 50, 95	L1	L5	L50	L95

Input Channel Time Streams

In CoCo, time domain data is always available in the form of long time history. The user can view and record the time signals. The limitation is that the sampling rate of the time signals cannot be arbitrarily changed. It is always set internally by the system based on analysis frequency range.

RMS trace of weighted level, time averaged level or sound exposure

CoCo records an RMS trace of the sound level. The user must choose between the time weighted level L_{AF}, the equivalent time averaged level L_{AEQ} or sound exposure level L_{AE}. Only one can be recorded at a time.

The RMS trace must be selected using one of Time or RMS as variable at the CSA Editor stage.

Histogram of Time Weighting

CoCo also records a signal containing a histogram of the dB values of the time weighted signal. This signal is used to compute the Ln data.

CSA Editor Operation for Sound Level Meter

This section describes the operation of the CSA Editor related to sound level meter measurements. For general operation of the CSA Editor refer to the CSA Users Manual.

CSA Editor Wizard for Sound Level Meter

This section summarizes how to create a CSA project for sound level meter measurements in the CSA Editor. We strongly recommend that you read the CSA Edit User's Manual to gain more detailed information before proceeding with this chapter.

To start, click on the CSA Editor icon in the upper-right corner in EDM and start the CSA Editor. The CSA Editor Wizard dialog box will be displayed. First select the CoCo type and number of channels of the CoCo that this CSA project will support. The number of channels must be equal to or less than the number of physical input channels on the CoCo hardware. Click on the Next Button.

Next select the Sound Level Meter template and either time or RPM based mode. Time based analysis will compute the results based on time averages. This is the most common application. RPM based mode computes results versus the RPM measured by a tachometer for rotating equipment. Click on the Next Button.

Next select the number of input channels to compute the analysis functions. In general it is best to select the minimum number of channels to conserve computational resources for the CSA. This number can be less than or equal to the number of physical input channels on the CoCo hardware. Click on the Next Button.

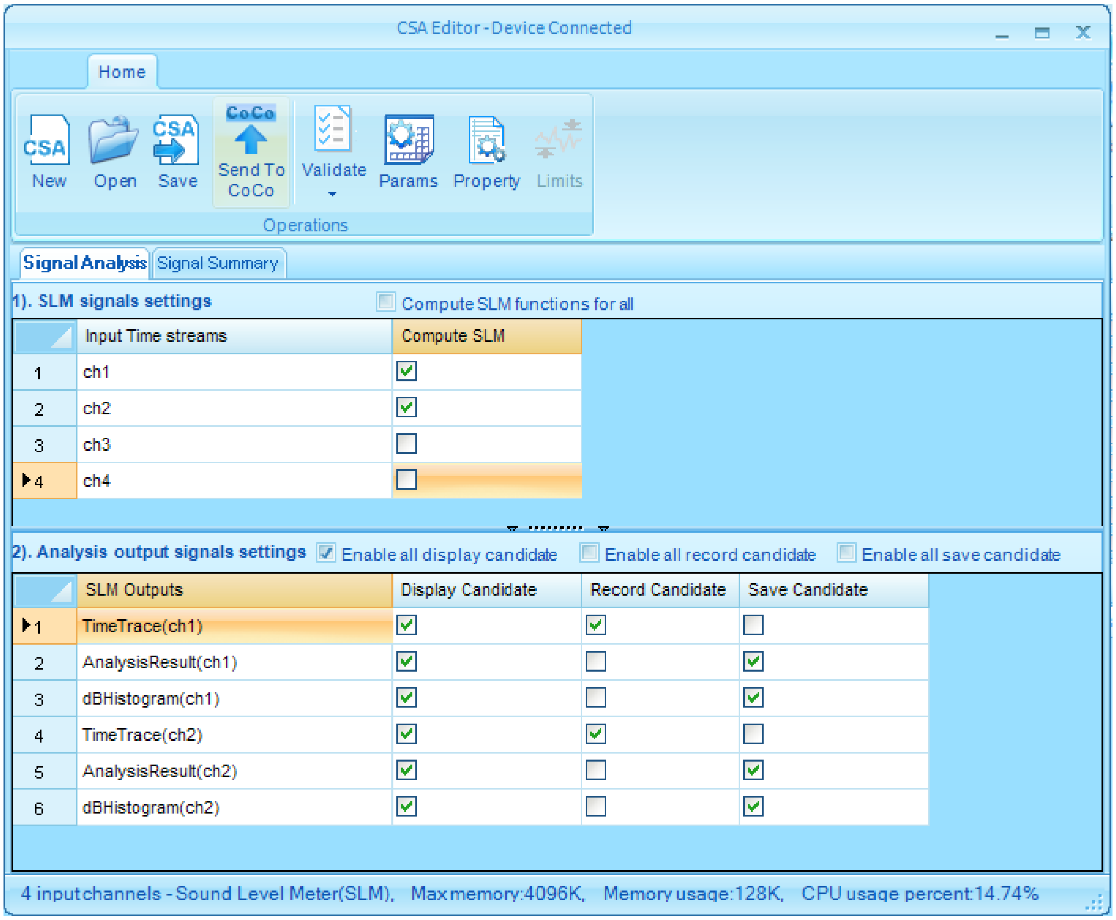
Next enter the description of the CSA. This data will be saved with the CSA and be shown on the CoCo device when the CSA is opened. Click on the Next Button.

Select Signal Candidates

Next the Signal Analysis Display is shown. This display lets you select which signals will be display candidates, save candidates and which will be used to compute the octave analysis. Always select the minimum number of candidates required to conserve computational resources. Note that when a signal is selected as a candidate then it is available on the CoCo hardware, meaning that it can be displayed or save, etc. If the signal is not specified as a candidate in the CSA then it will not be available on the CoCo hardware.

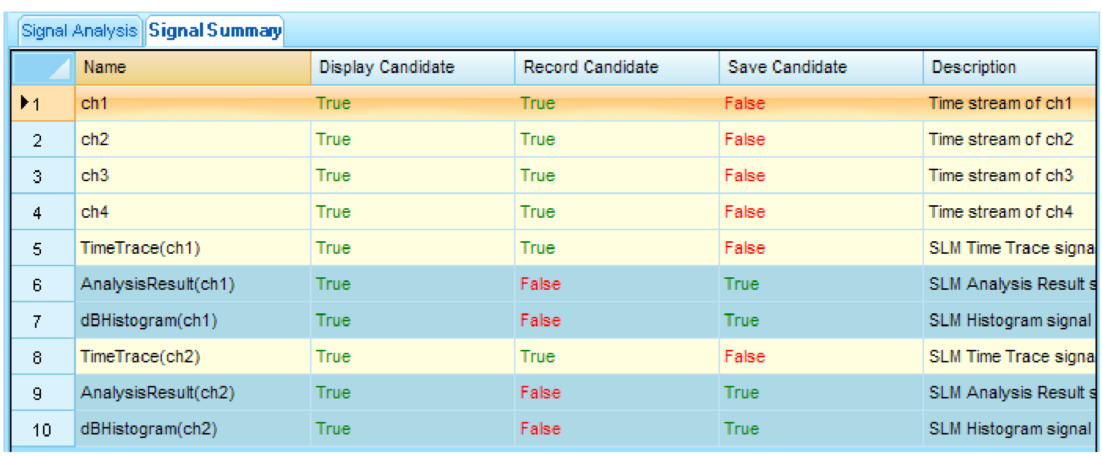
SLM Signal Settings lets you specify the SLM computation candidates for the input time streams.

Analysis output signal settings let you specify the display, record and save candidates for the computed data.



■ Figure 45. Signal Analysis tab in CSA Editor.

After all the candidate selections are made, a summary can be viewed by selecting the Signal Summary.



	Name	Display Candidate	Record Candidate	Save Candidate	Description
1	ch1	True	True	False	Time stream of ch1
2	ch2	True	True	False	Time stream of ch2
3	ch3	True	True	False	Time stream of ch3
4	ch4	True	True	False	Time stream of ch4
5	TimeTrace(ch1)	True	True	False	SLM Time Trace signa
6	AnalysisResult(ch1)	True	False	True	SLM Analysis Result s
7	dBHistogram(ch1)	True	False	True	SLM Histogram signal
8	TimeTrace(ch2)	True	True	False	SLM Time Trace signa
9	AnalysisResult(ch2)	True	False	True	SLM Analysis Result s
10	dBHistogram(ch2)	True	False	True	SLM Histogram signal

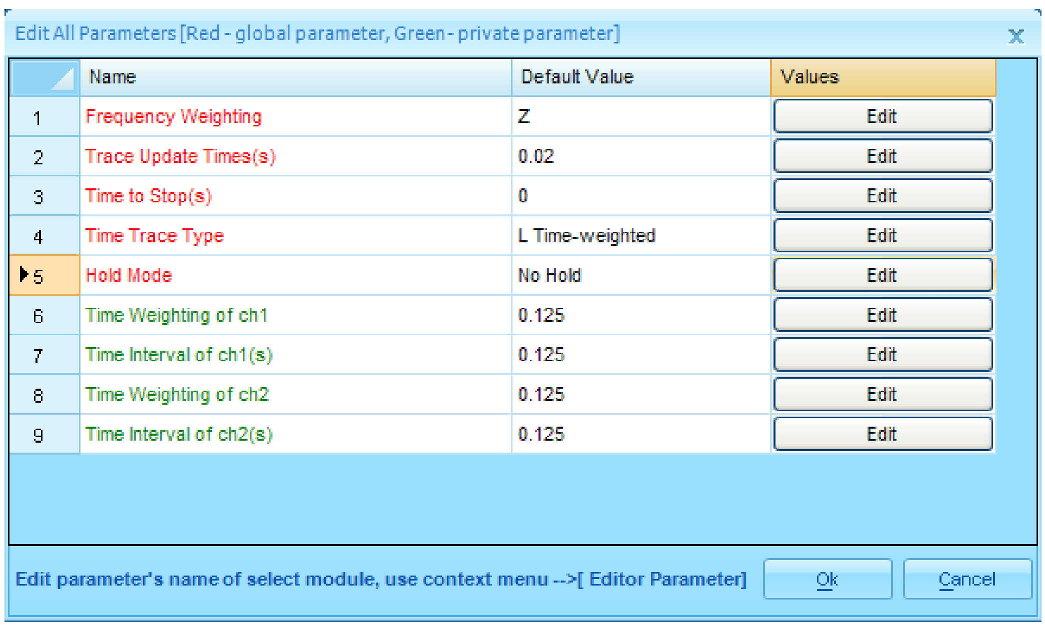
■ Figure 46. Signal Summary tab in CSA Editor.

Analysis Parameters

After the signals are selected the next step is to specify the Parameters by clicking on the Param icon in the CSA Editor Ribbon. The Parameters display lets you change the computation parameters such as frequency weighting, trace update time, time to stop, etc. Click on the Edit Button next to each item to edit the parameter. Click on Ok to save the updated parameters.

Time Based Acquisition Analysis Parameters

The Analysis Parameters are different if the Acquisition mode is set to Time or Frequency based analysis. The time based acquisition mode parameters are Figure 47.

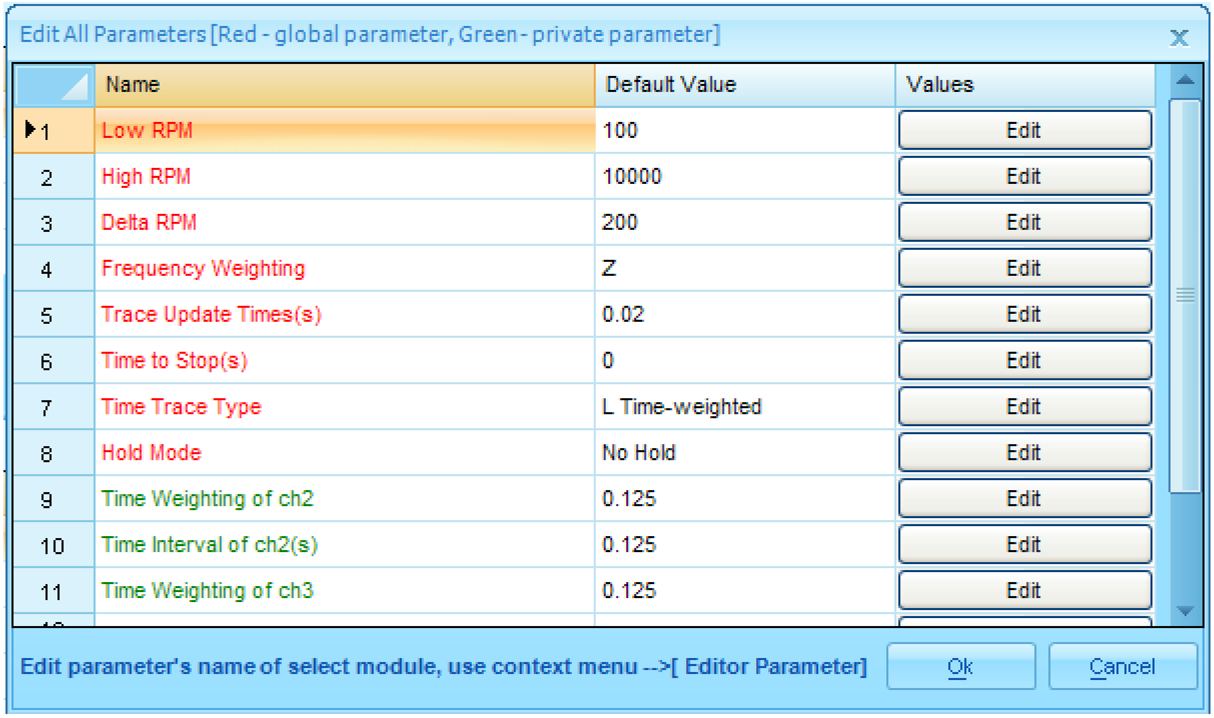


- Figure 47. Edit Parameter window in CSA Editor for time based acquisition.

Frequency Weighting – defines the frequency weighting including A, B, C or Z.
Trace Update Times – defines the update rate of the time trace. Enter a larger update time for a longer display duration.
Time to Stop – defines the length of time for data acquisition. When the parameter is set to No Hold then the acquisition will continue indefinitely.
Time Trace Type – defines the time weighting includes L, Leq, L and LE.
Hold Mode – This parameter only has the impact to the measurement of Lmax, Lmin, LEQmax, LEQmin, Lpeak and LCpeak. If the Hold Mode is off, for every period of Time Interval, these measurement values will be reset and calculate again. If the Hold Mode is ON, these measurements will be taken within the whole elapse time.
Time Weighting – Exponential averaging decay factor used to calculate the time-weighted sound level for that individual channel.
Time Interval – This is the time period of calculating Leq for each individual channel.

RPM Based Acquisition Analysis Parameters

The RPM based acquisition parameters are shown in Figure 48.



■ Figure 48. Edit Parameter window in CSA Editor for RPM based acquisition.

Low RPM – defines the low RPM level for display limits and also the limits for trigger mode.
High RPM – defines the high RPM level for display limits and also the limits for trigger mode.
Delta RPM – defines the resolution of RPM measurements. Note that a large RPM span and high resolution (small Delta RPM) require more computation resources.

The other parameters are the same as the time based acquisition parameters.

Validation, Save and Upload

After the CSA Wizard is complete and the CSA file is created, connect the host PC to the CoCo device and press the Validate icon to validate the CSA. It may take a few minutes to finish the validation.

■ Figure 49. Validate and Send to CoCo icons in the CSA Editor Ribbon.

The validation process analyzes the CSA file for internal consistency and estimates the required DSA resources required to run the CSA file on the CoCo device.

If the Validation passes, then press *Send to CoCo* command in the Validation dialog box to send the CSA project file to CoCo. Alternatively you can manually upload it to CoCo. The uploaded CSA will be classified into the Acoustic Analysis Application Group and can then be opened on the CoCo hardware.

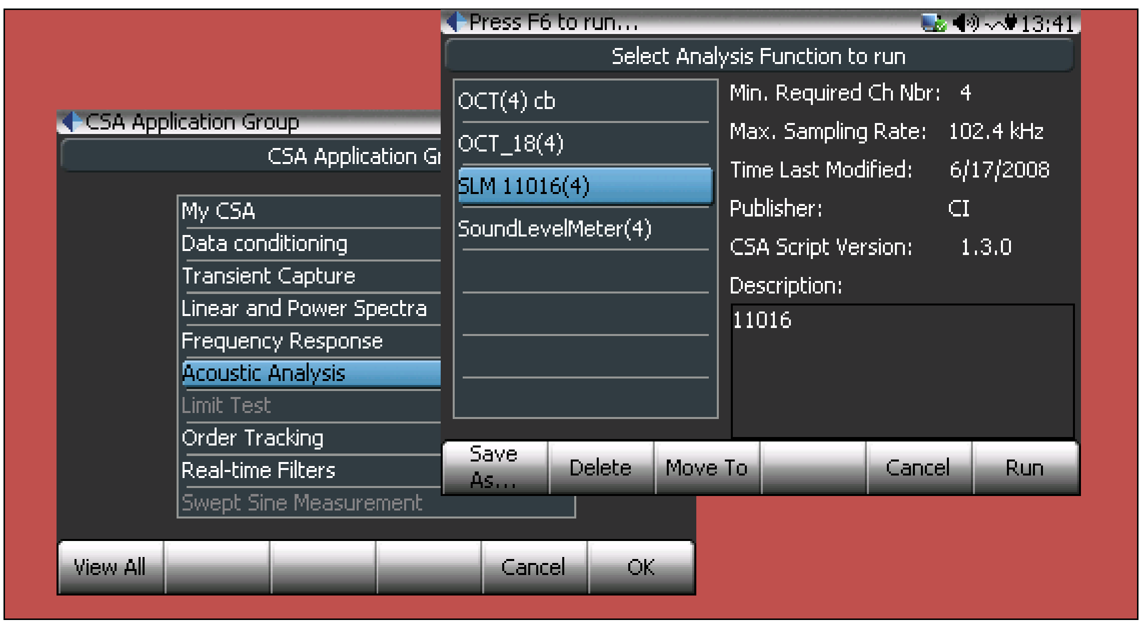
CoCo Operation for Sound Level Meter

This section describes the operations of CoCo that are specifically related to sound level meter analysis. For general operations of CoCo refer to the Basic Users Manual.

Select a CSA Project

After an SLM CSA is created from the CSA Editor and downloaded to the CoCo Hardware then it can be opened on the CoCo. Alternatively you can also open the default SLM CSA scripts on the CoCo Hardware. To open a CSA press the Analysis Button, then select the Acoustic Analysis Application Group. All SLM CSA files are automatically placed in this group to help organize the CSA files on the CoCo.

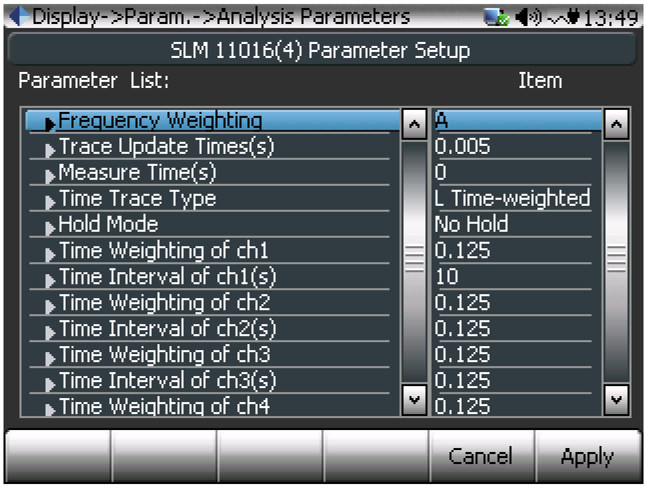
Next select SLM CSA from the list. SLM analysis CSA files have the name SLM() by default.



■ Figure 50. Select Analysis Function for SLM.

Analysis Parameters for SLM

To set the parameters press the Param. Button in the signal display window, select Analysis Parameters, then set the parameters. Note the default values are defined in the CSA Editor.

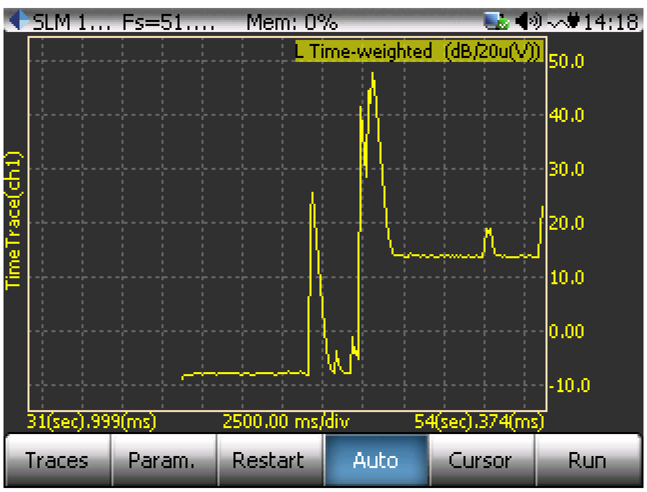


■ Figure 51. SLM Analysis Parameters.

The time weighting parameter can be selected so it corresponds to "Fast" or "Slow" weighting mode. For "Fast" weighting mode, set the time weighting constant to 0.125 seconds; for "Slow" mode, set it to 1.0 seconds.

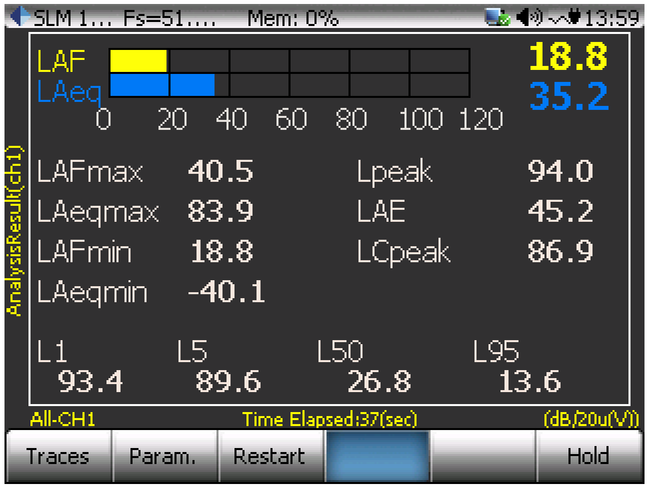
SLM Displays

SLM includes traces of the native time streams and the time weighted sound levels,



■ Figure 52. Native or time weighted sound levels can be viewed in a time trace.

SLM also includes a display called Analysis Results. The Analysis Results display is similar to a typical sound level meter display and shows the numerical values for the measures and a bar meter for the time and frequency weighted level and equivalent level.



■ Figure 53. SLM Analysis Results display.

Notice that the statistics display, L1=93.4, L5=89.6, L50=26.8 and L95=13.6 means that since the measurement started, among the historical time weighted measurement, 1% exceeded 93.4dB; 5% exceeded 89.6dB; 50% exceeded 26.8dB and 95% exceeded 13.6dB.

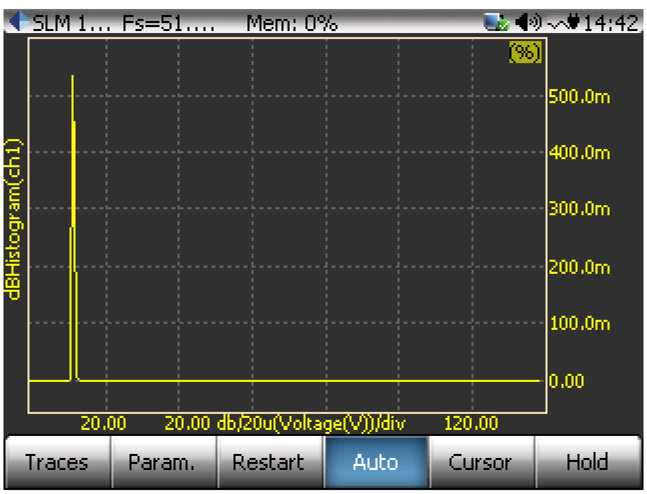
Alternatively, you can select to view individual analysis measures by setting up ***(Select View Mode for Current trace)***:

For example, when four analysis measurement signals are selected in a Trace and Leq is selected, following picture will be shown:

■ Figure 54. Numerical display for SLM.

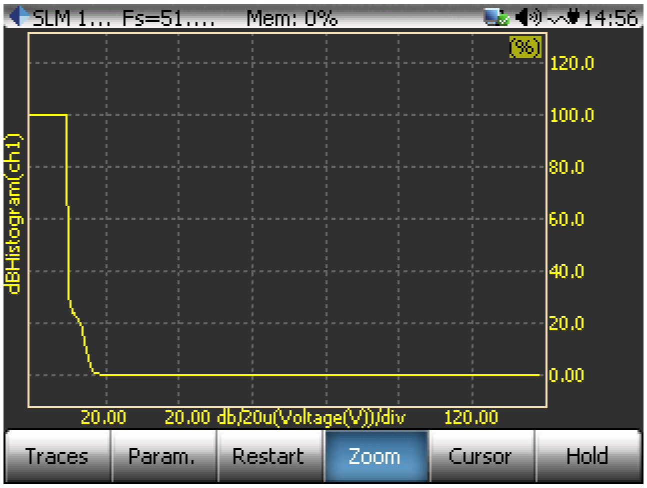
It shows the Leq values of each of the four input channels.

Another common display for SLM measurements is a histogram which shows a statistical distribution of the frequency of occurrence versus amplitude.



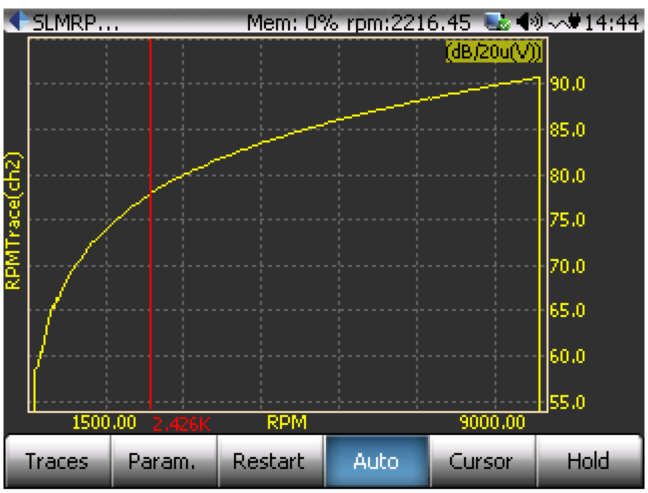
■ Figure 55. Linear-Normalized histogram display for SLM measurements.

The histogram can also be viewed in cumulative format by selecting Select View Mode for Current Trace from the Traces Button and choosing Cumulative.



■ Figure 56. Cumulative histogram display for SLM measurements.

RPM Based SLM measurements can be made by selecting an RPMTrace. Figure 57 shows the sound level in dB vs. RPM. This is one of the most popularly used measurement in automotive NVH application.



■ Figure 57. RPM based SLM measurement.

5. ORDER TRACKING

Order Tracking is a general term describing a collection of software functions used for analyzing the mechanical dynamic behavior of rotating or reciprocating machinery for which the rotational speed can change over time. Unlike the power spectrum and other frequency-domain analysis standards where the changing variable is the frequency, Order Tracking functions present the data related to the variable rotating speed, i.e., RPM (revolution per minute).

The most useful measurements are order spectra and order tracks. An order spectrum gives the amplitude of the signal as a function of harmonic order of the rotation frequency. This means that a harmonic or sub-harmonic order component remains in the same analysis line independent of the speed of the machine.

The technique that observes the changes of any quantity vs. RPM is called ***tracking***, as the rotation frequency is being tracked and used for analysis. Most of the dynamic forces exciting a machine are related to the rotation frequency so interpretation and diagnosis can thus be greatly simplified by use of order analysis.

Order tracks are simply the observations to the amplitude of the components with fundamental frequency or harmonics. It is one typical type of *tracking*. There are other types of tracking. For example, the user can track the FFT-based PSD spectra, a fixed band or an octave band etc.; all these can be called tracking.

With the CI Order Tracking package, the instrument can:

- Process a tachometer signal and give a high fidelity RPM measurement
- Measure the order spectra
- Measure the order tracks
- Measure the RPM FFT spectrum
- Measure the energy in fixed bands vs. RPM
- Measure the amplitude and phase of an order relative to the tachometer.

There are several different applications for order tracking. A discussion of some is given below.

The first application, often referred to Run Up/Run Down, is used to evaluate the noise or vibration dynamic response when RPM is used as a changing variable. In this case, the RPM range can be very large, from a few RPM to 10,000 RPM. Typical application tests are used in the automotive or aircraft engines. The measurements can be any physical quantities such as sound, displacement, velocity, acceleration, torque, etc. The analysis measure can be the amplitude or the power of an order, the energy over a fixed frequency band, a bin of octave filter, etc. The phase information of the responses to tacho is less important in this type of application. In fact the rotating element might be hidden inside of mechanical system. The primary result for this type of measurement is the magnitude of the responses vs. RPM.

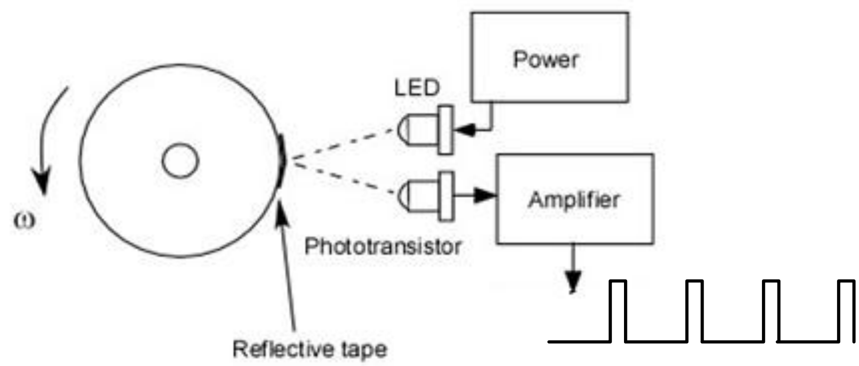
The second application is rotating machine analysis that focuses on the measurement of displacement or velocity of the rotors while it is rotating. The instrument measures the amplitudes of specific orders and their relative phase to a reference signal. The phase is calculated relative to the tachometer input or a separate reference input. This application is common for machine diagnosis and balancing. In this case, the RPM is stable or quasi-stable. Order tracking technology is useful to increase the accuracy of the estimation of orders.

Order signals with phase are useful in the test of rotating machine in the Run Up/Run Down process. This is often presented as a "Bode Plot". The Bode Plot is a borrowed concept from control theory; it is a collection of Amplitude and Phase data over a changing speed range (i.e. Run Up or Coast Down). Some of the setup information depends on the rate of change of the RPM. The Run Up or Coast Down could take anywhere from a few minutes to a few hours (such as for a cold startup on a turbine). Other displays such as the orbit plot is useful as well.

The CoCo includes the ability to measure RPM based octave analysis and sound levels. This feature is similar to order tracks except that spectra are recorded in octave bands with A, B, C or Z frequency weighting. This feature is included in the Acoustic Analysis and Sound Level Meter CSA Templates instead of the Order Tracking Template. Refer to these sections for more details.

Tachometer Signal Processing and RPM Measurement

A **tachometer (tacho)** converts the angular velocity of a rotating shaft into an electrical signal, typically a voltage. It is common for calibrated instruments to provide a measurement of the shaft in units of revolutions per minute (RPM) or revolutions per second (RPS). Many modern rotating machines (electric motors, generators, pumps, turbines, IC engines, etc.) have integrated tachometers that can measure shaft angular velocity. An example of an optical tachometer is shown in the figure below.



■ Figure 58. Optical tachometer setup.

The goal of tacho signal processing is to get a clean and stable RPM reading. The tacho signal must be carefully processed to provide a base of tracking. Any order tracking results can only be thought of as being as accurate as the tachometer signals that were used to estimate the instantaneous frequency of the order in the analysis process. If the quality of the tachometer channel is poor, the results from all other channels will be poor or even unreliable.

In old analog methods, tacho channels were conditioned with a tracking ratio tuner with a phase lock loop. The disadvantage of this method is the limited slew rate and the use of complex hardware. To overcome these limitations various digital tacho processing methods have been developed.

From hardware design point, there are two ways to implement a tacho input channel: use a dedicated tacho channel with a digital counter, or use an analog input channel.

Dedicated tacho channel using counter
Using a dedicated tacho channel, usually without A/D converter, has been popular. This hardware approach contains its own tacho clock which runs at a much higher speed; typically in MHz. This tacho hardware also contains special counters which maintain a continuous counter reading to avoid skipping any triggered cycles of the tacho signal. There is also an option to allow these counters to "average" several tacho periods for cases when the input tacho frequency is very high.

Using Analog Input Channel
Alternatively, some systems use an analog input data channel as a tacho channel. In this case, the tacho clock is actually the sample rate of the data channels. This sample rate usually limits the tacho frequency range since the tacho range is now set by the input data frequency range requirement. In addition, due to the "frame processing" nature of some not-so-well designed input sampling processes, some instruments may be limited to how they acquire the tacho signal. This restriction usually means they get several tacho cycles in every data frame. The result is often an "averaged" value which is okay unless the tacho signal is changing frequency during the data frame event, which is often the case.

With the advances of electronics and lower cost of electronic components cost is less of a concern. The approach of dedicated tacho channel with a digital counter, without A/D, may or may not be the best choice.

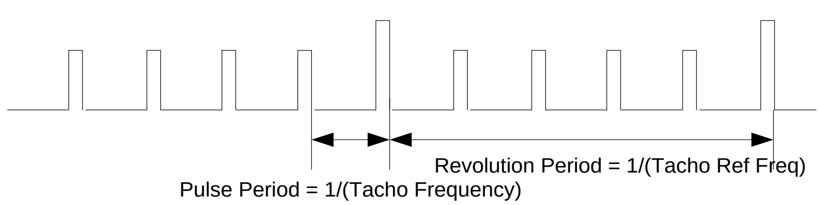
The CoCo-80/90 can use any of the data channel as tacho channel. For simple interface design, usually channel 1 is used for the tacho. While the data input channel is used as a tacho measurement, the special hardware circuitry allows this data channel to sample at the highest possible sampling rate. In the other words, the accuracy of tacho speed measurement is depending on the current range of the analysis frequency. This technique has several obvious advantages:

- The time domain signal of the tacho input is transformed by A/D converter into a digital signal. The user can observe the pulse trains of the tacho signal and set threshold arbitrarily.
- Accurate phase information can be obtained relative to each data channel because the tacho channel, which is fed by high frequency sampling counter, is synchronized with data channels.
- The RPM estimation is not influenced by the current data sampling rate.

High Pulse per Rev

Pulse per Rev is defined as the number of pulses per revolution. Pulse per Rev. must be defined by the user so the instrument can calculate the *reference frequency* of tacho using *tacho frequency*. The relationship is:

Tacho Reference freq = Tacho freq / Pulses per Rev



In most rotor tests, especially in balancing, the *Pulses per Rev* is simply 1. However, in other cases, such as in flywheel or geared data measurement, the *Pulses per Rev* can be as high as hundreds. To deal with this situation, a dedicated tacho channel with high speed counter might work better.

In the CoCo-90, in addition to using any data channel as tacho input, a dedicated tacho channel is installed to measure a high speed RPM, or deal with high Pulses/Rev or digital TTL trigger. The counter speed is about 25MHz. This second choice provides a more versatile solution to the user for their applications.

The special tacho hardware design on the CoCo system with the Order Tracking package offers the most accurate possible approach for performing a wide range of Real-Time machinery-related vibration and noise analysis.

Pulse Detection

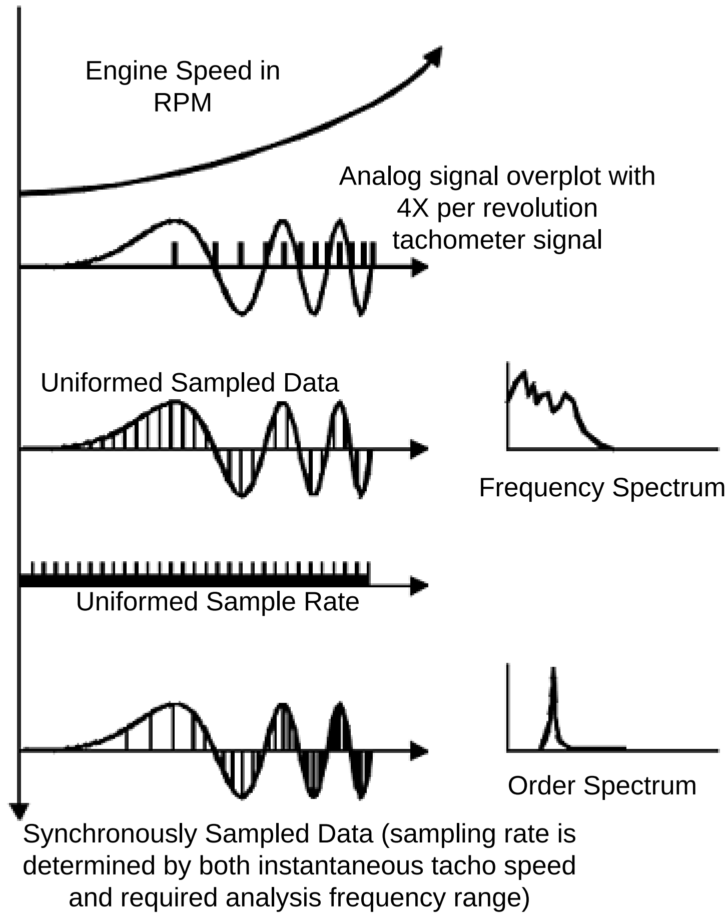
A good tacho processing device should allow the user to see the tacho signal in its original time waveform visually, and set the *Pulse per Rev*., the threshold of pulse detection. This will help setup the tachometer and diagnose any problems quickly. In the CoCo hardware, a special display window is created so the user can switch between the RPM trace and the tacho original time waveform displays conveniently. The pulse detection threshold can easily be controlled by using Up/Down buttons.

Order Tracks and Order Spectrum

Knowledge of the rotating speed allows presentation of measurement results in the angle and order domains, corresponding to the time and frequency domains. *An order is a frequency normalized with some reference frequency, e.g. the shaft frequency*. This means that the order of a vibration component in the order spectrum indicates the number of vibration cycles per shaft revolution. The magnitude, which can be measured using EU_{pk}, EU_{rms} or EU_{rms}^2 , of an order is the measurement extracted through a tracking filter with the center frequency located at this frequency. Multiple measurements of a range of orders will construct a so called *Order Spectrum*. An order power spectrum measurement gives a quantitative description of the amplitude, or power, of the orders in a signal. It provides a good view of all order components of a signal. This can help you find significant orders and compare the level of different order components

There are two methods to perform rotationally coherent sampling, phase-locked frequency multipliers and digital re-sampling. Phase-locked frequency multipliers were mainly used in early work. They generate sampling pulses based on a rotational reference signal. These sampling pulses control the sampling process. Note that the sampling frequency will depend on the rotational speed, and thus an adjustable anti-aliasing filter is needed. This complicates the method considerably. In the digital re-sampling technique, the time signal is conventionally sampled together with some rotational reference signal. The time signal is then digitally re-sampled to the angle domain by interpolation techniques. The rotational reference signal can be acquired with a tachometer or an incremental pulse encoder.

The following picture shows conceptually how angle data re-sampling can be used to analyze vibrations from an engine during start up. Once the signal has been transformed into its angle domain, the FFT can be applied to analyze the order spectrum of the vibrations.

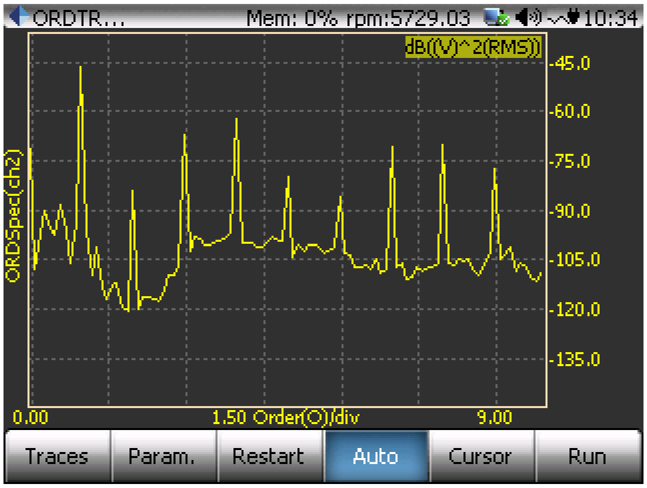


■ Figure 59 Angular Data Re-sampling of a Chip Signal

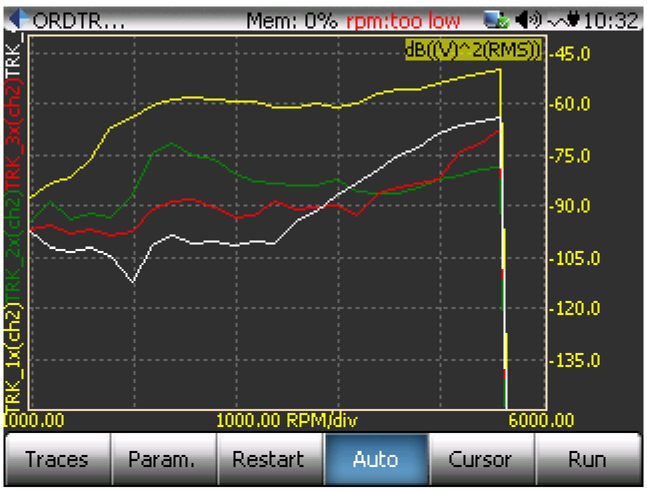
The last plot in the picture shows that the sampling rate will be determined by both instantaneous tacho speed and required analysis frequency range.

In the CoCo system, the order tracks and order spectrum are computed with a proprietary technology that combines digital re-sampling, data decimation, and interpolation, DFT and FFT calculations.

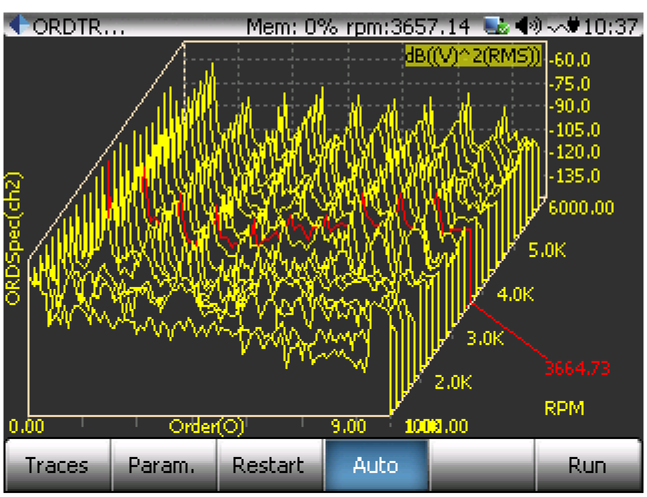
Three measurements can be generated from order tracking computation: Order Spectrum shown in ,**Error! Reference source not found.** Order Tracks shown in **Error! Reference source not und.**, and the 3D RPM Order Spectrum shown in **Error! Reference source not found.**. The 3D RPM Order Spectrum is simply the a 3-dimensional view of the other two types of measurement.. Another way to visualize these types of plots is that the order spectrum is a cross section of the 3D plot along a fixed RPM value while the order track is a cross section along a fixed order number. The relationship of them is:



■ Figure 60. Typical order spectrum.



■ Figure 61. Typical order tracks.



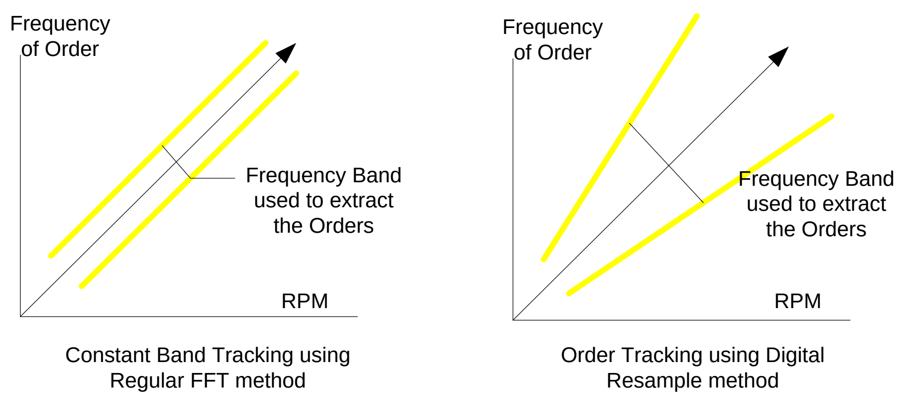
■ Figure 62. Typical 3D order track waterfall plot.

To be completed.

■ Figure 63 This picture must be replaced with CoCo measurement result

An important concept that must be introduced now is called delta order, *ΔOrder*. In the FFT based frequency spectrum analysis, the frequency span and frequency resolution are fixed. The capability of discriminating frequency components is equal in both low and high frequency. In rotating machine analysis, we want to have better analysis resolution in the low frequency than that in high frequency. For example, if the rotating speed is at 60 RPM, we definitely care if the instrument can tell the difference between 1Hz (order 1) and 2Hz (order 2); on the contrary, if the rotating speed is at 6000 RPM, the user probably won't care if the instrument can discriminate the measurement between 100Hz (order 1) and 101Hz.

With the digital re-sampling technique, the order tracks and order spectrum are extracted based on a filter with equal *ΔOrder* instead of equal *ΔFrequency.* The concept is illustrated in the following figure:

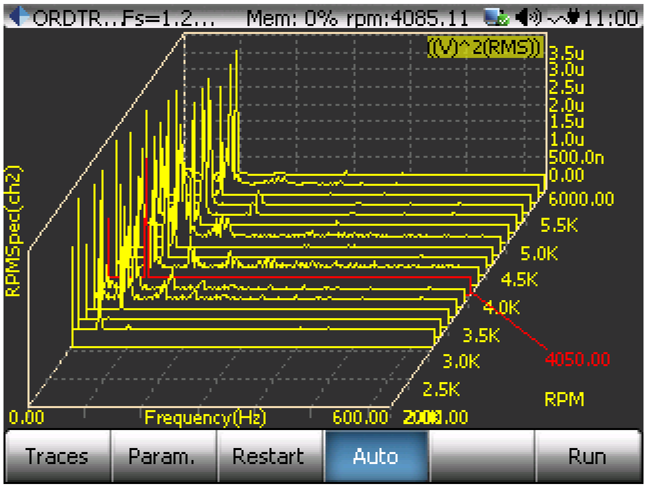


■ Figure 64. Comparison of constant band tracking and digital re-sampling method.

In this figure, the left side shows when the order tracks are extracted using conventional FFT method with fixed resolution, the *ΔFrequency* of tracking filter will be fixed; the right side shows that if the order tracks are extracted using digital re-sampling, the *ΔFrequency* tracking filter will be increased proportionally with the RPM. Obviously, the method of digital re-sampling is more desirable in extracting the measurement of orders.

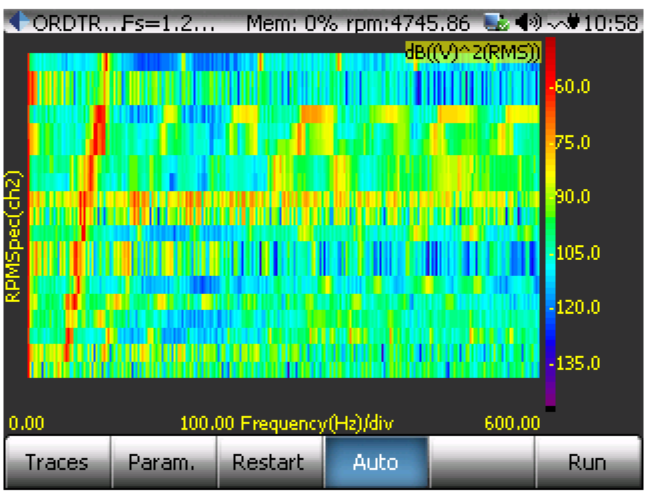
RPM Frequency Spectrum

While the order tracks and order spectra are developed to analyze the characteristics of the system on the order space, the measures of fixed bands are also helpful for analysis. Similar to the RMS time trace for a given frequency band with time as variable, the RMS trace can be extracted for a given frequency band with RPM as the independent variable. This is simply called an RPM Spectrum. An RPM Spectrum can be described as a 3D waterfall as shown below:



■ Figure 65 RPM spectrum.

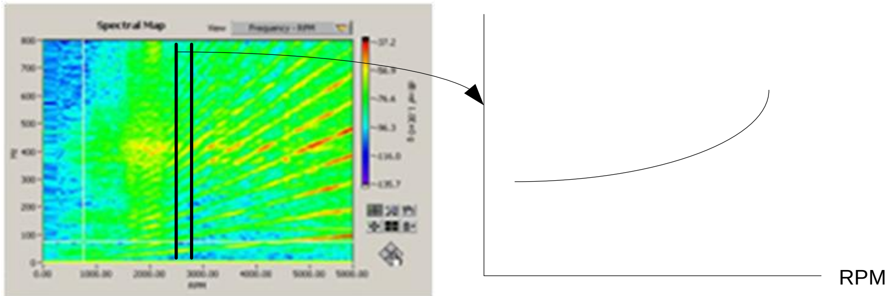
The horizontal axis of the 3D RPM Spectrum is frequency. The Z axis is RPM and the measurement unit is usually EU_{rms}^{2} or EU_{rms}. A color map can also be used to describe the magnitude of the whole range as shown below.



■ Figure 66 Color map of an RPM spectrum.

With a 3D RPM frequency spectrum, the instrument can extract the total energy over a fixed frequency band, and plot it with RPM as the independent variable. This is called *Fixed Band RPM Spectrum* as shown below.

To be completed.



Fixed Band RPM Spectrum

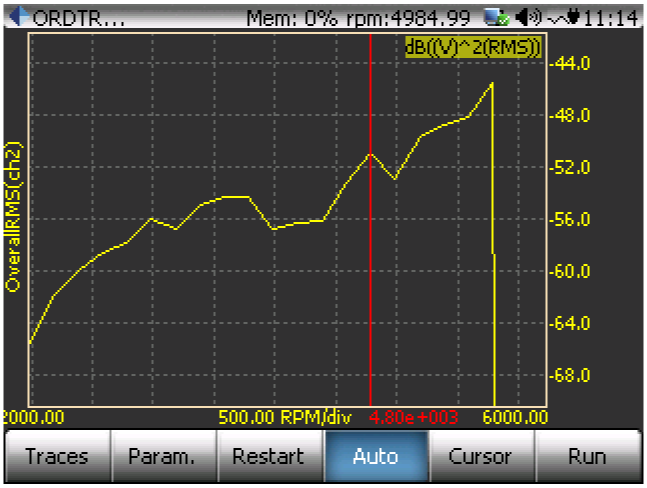
■ Figure 67 Fixed band RPM spectrum.

The measurement engineering unit of Fixed Band RPM Spectrum is EU_{rms}^2 or EU_{rms} representing the total power in a fixed band measured versus rotating speed change. This data is particularly useful to watch the total magnitude in a resonance area when the rotating speed of the shaft is changing. You can define the frequency band around the resonant frequency and perform a run up/down test. Both order tracking and order spectrum cannot extract the response magnitude of the resonance as accurately as a fixed band RPM spectrum because the bandwidth of the tracking filter of order tracking is not explicitly controlled.

Overall Level Measurement

In order tracking, about it is important to monitor the overall RMS level or power level of the measurement versus RPM. The overall level is a good reference for comparing with other signals such as order tracks or fixed band RPM spectrum.

Overall level can be in unit of RMS (EU_{rms}) or power (EU_{rms}^2). The horizontal axis is RPM. Below is a typical overall level display.



- Figure 68.Overall RMS level plot.

Raw Data Time Streams

In many other order tracking software products, the user can either conduct real-time order tracking analysis, or record the data with other tools and then post process the order tracks, but not both at the same time. The CoCo is a high performance data recorder in addition to a real time analyzer and can do both at the same time. Continuous time streams of each input channels are always available even while order track data is computed..

Order Tracks with Phase

The Phase in Rotating Machine Analysis

Many mechanical faults are associated with certain orders, analyzing order magnitude and phase can help you detect mechanical faults directly. For example, a strong first order magnitude indicates imbalance in most cases. Analyzing the first order magnitude can help you identify the imbalance. Moreover, the magnitude and phase of the first order can help you correct the imbalance by adding weights on the appropriate rotor positions. However to fix such an imbalance problem requires phase information of order tracks. A list of the sources of vibration in the rotating machine are:

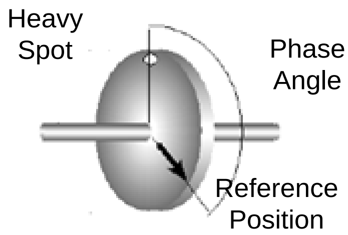
Order	Source of Problem
0.05X~0.35X	Diffuser Stall
0.43X~0.49X	Instability
0.5X	Rubbing
0.65X~0.95X	Impeller Stall
1X	Imbalance
1X+2X	Misalignment
(#Vane)X	Vane/Volute gap
(#Blades)X	Blade/Diffuser Gap

As previously discussed, an order track is the measurement taken for an order, i.e., normalized frequency, versus RPM. In most of the applications of engine related test, the phase information of order tracks are not important. In rotating machine analysis, the phase of the signal is vitally important.

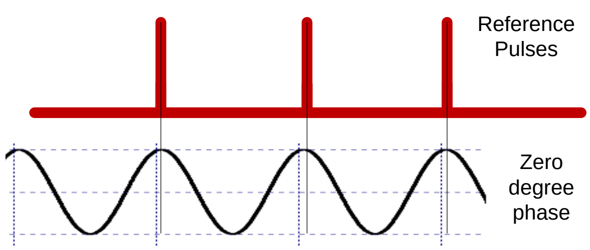
Phase is a relative measurement quantity and can only be measured with a pair of signals. It indicates the time delay at certain frequency between two signals. The phase value can be translated into the difference of relative angle, relative position or propagation time if additional information is given. When we refer to the phase information of one signal, we imply its phase is relative to a reference signal that was mentioned in context.

In rotating machine analysis, the phase of the first order of the rotor can be directly mapped to an angular difference between a signal and the reference signal. The reference signal can be another channel of measurement, or the tachometer signal. The phase difference between two waveforms is often called a phase shift or phase delay. A phase shift of 360 degrees is a time delay of one cycle, or one period of the wave, which actually amounts to no phase shift at all. A phase shift of 90 degrees is a shift of 1/4 of the period of the wave, etc. Phase shift may be considered positive or negative, i.e., one waveform may be delayed relative to another one, or one waveform may be advanced relative to another one. These conditions are called phase lag and phase lead respectively.

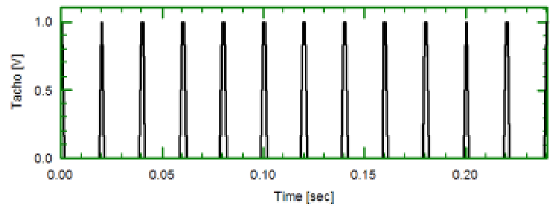
An example of this is the phase of an imbalance component in a rotor with reference to a fixed point on the rotor, such as a key way. To measure this phase, a trigger-pulse must be generated from a certain reference point on the shaft. This trigger can be generated by a tachometer or some type of optical or magnetic probe that senses a discontinuity on the rotor, and is sometimes called a "tach" pulse.

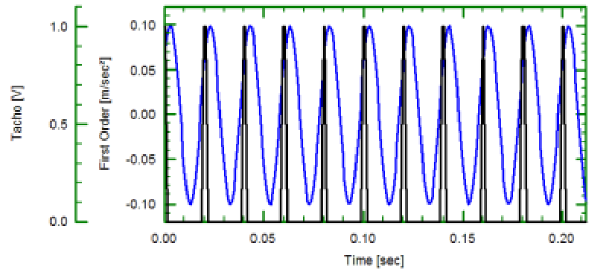


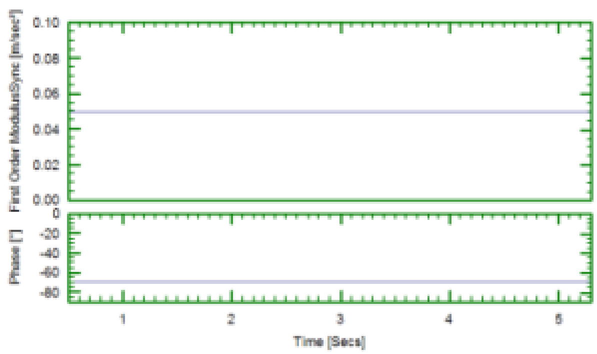
A zero degree phase delay at a frequency can be depicted as a series of pulses overlaid with a sine wave where the pulse edge is exactly located in peak position of the sine wave.



In Figure X a section of the tacho signal is shown on its own and then overlaid on the vibration signal. The tacho signal in this example crosses the vibration signal at exactly the same point on each cycle. If the phase of the vibration signal were to change then its position relative to the tacho pulse would also change. Extracting the first order modulus and phase, as before, gives the curves shown in Figure Y. The phase is now constant near -60^o as it should be for such a signal. Because the rotating period of the signal is about 20ms, -60^o corresponds to a 20*60/360=3.3ms delay.







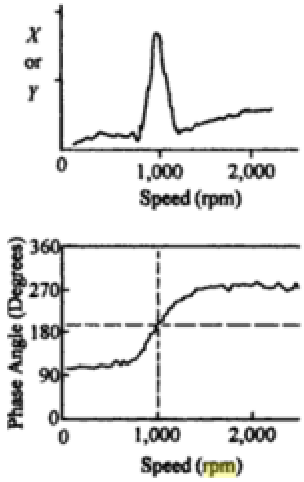
The phase measurement at higher orders will have the similar physical interpretation although they are difficult to comprehend intuitively.

It must be noted that the order tracks with phase, or Complex Order Tracks by name, are not regular complex signals as frequency response or cross spectrum. They are really auto-spectra with assigned phase. These synthesized signals can certainly be viewed as a complex signal using tools including Bode Plot, polar and orbit diagram. However the user must keep in mind that the magnitude and phase of a complex order track are calculated separately.

In the following sections, we will present how the order tracks with phase can be presented graphically with the Bode, polar and orbit plot.

Bode Plot

The term Bode Plot is borrowed from the field of control theory, referring to a plot of magnitude and phase angle between the input and output verses frequency of a control system. Many in the rotating machine vibration industry have adopted this term to describe the steady-state vibration response amplitude and phase angle versus rotational speed (RPM). It turns out that the Bode Plot is the best way to describe order tracks with phase. You typically use Bode plots for transient analysis in both start-up and coast-down conditions. A Bode plot can help to identify the resonance speed of a rotor or examine the rotor dynamics on an order basis. A typical Bode Plot for an order track is shown below:

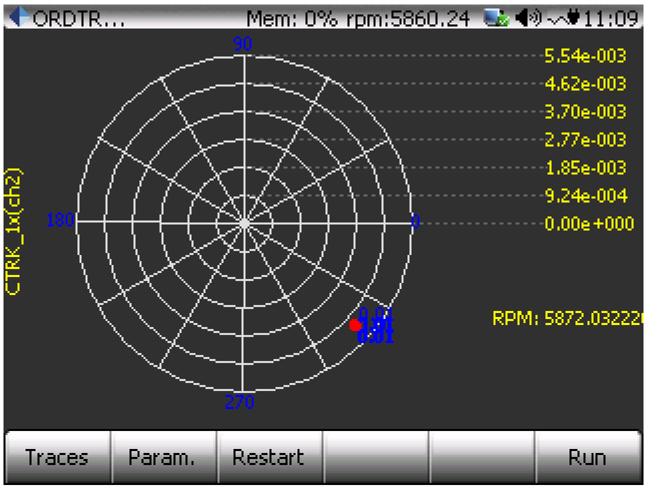


■ Figure 69 This picture must be replaced

In the CoCo system, after the order tracks are acquired together with their phase information then the Bode Plot can show one or multiple tracks.

Polar Plot

The Polar Plot is another useful tool to view the order tracks for both amplitude and phase information. A polar plot draws the amplitude and phase on a polar coordinate. A typical polar plot is shown below.



■ Figure 70. Polar plot shows magnitude and phase on a polar axis.

In the polar plot, the dot shows the current order track value. The distance between the dot and the center indicates the magnitude of the order track while the angle corresponds to the phase measurement. The polar plot only shows the instantaneous measurement. It cannot keep the history versus RPM.

The Polar plot is often used to visually indicate the imbalance of the rotor. Polar plots and Bode plots often are combined to describe the rotating speed vector signal locus during speed changes. A Bode plot provides excellent change visibility with respect to speed, while the polar plot shows improved phase variation resolution.

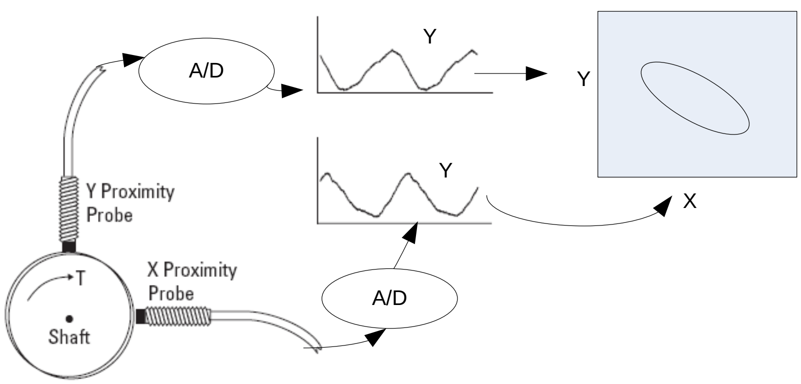
Orbit Display

The Orbit display uses two data channels in the time domain. The signals from two channels are drawn on an X and Y plane to display the shaft position change versus angle of rotation. Orbit displays give a two dimensional visual picture of the motion of a rotating shaft.

A well balanced shaft with no movement in any direction and would produce a dot in the middle of the plot. The shaft movement can give an indication of the vibration source e.g. if there is a lot of up/down movement it may be that the machine feet are not bolted down tightly enough.

To create an orbit plot you need to take a dual channel simultaneous measurement to capture data at the horizontal and vertical axes at the same time. The displacement or acceleration sensors must be placed 90° apart from each other.

Orbit display uses a pair of measurement in time domain. It does not need the technique of *order tracking*.



The orbit display is similar to the Polar display in that it only displays the instantaneous status at the current RPM. In theory, the orbit display does not need a tachometer or another reference signal because X and Y reference to each other.

Summary

With the Bode, polar and orbit plots, the order tracks with phase can be presented visually. These are effective tools used for rotating machine analysis. Bode plot is mostly used in the Run Up/Coast Down process. Polar plot and orbit, which only show the instantaneous status of an order at current RPM, are adequate for applications at steady or quasi-steady rotating speed such as dynamic balancing.

CSA Editor Operation for Order Tracks

This section discusses how to create a CSA project for order tracking measurements in the CSA Editor. We strongly recommend that you read the CSA Edit User's Manual to gain more detailed information before proceeding with this chapter.

With the previous discussion, the Order Tracking software option is categorized into three sub groups based on different purpose of application. In each sub-group, different measurement quantities can be obtained. The measurements available in each group are:

Order Tracking Analysis Sub-Groups	Description	Measured Signals and View Mode
Normalized Order Spectra and Order Tracks	Use digital re-sampling and FFT or DFT to calculated normalized order spectra and order tracks. Order tracks have no phase information. This is mainly used in engine test, Run Up and Run Down.	• Raw time streams of data channels • Normalized Order Spectrum viewed in $EUrms^2$, EUrms, EUpk vs. Orders • Normalized Order Spectrum Waterfall viewed vs. order and RPM • Order Tracks viewed in EUrms, EUpk vs. RPM • Overall in EUrms or $EUrms^2$ vs. RPM
Constant Frequency Bands	Use FFT method to calculate the RPM spectrum either in a fixed frequency band or whole frequency range. This is mainly to observe the response of specific frequency range or to gain overall picture of frequency spectrum when RPM changes.	• Raw time streams of data channels • Waterfall RPM Spectrum viewed in any spectrum unit vs. frequency horizontally and RPM in Z direction • Fixed Band Spectrum in EU_{rms}^2 or EU_{rms} vs. RPM • Overall in EUrms or $EUrms^2$ vs. RPM
Order Tracks with Phase	Use digital re-sampling and FFT or DFT to calculate the order tracks, in considering the phase reference to tacho input signal. This is mainly used in Run Up/Run Down, or dynamic balancing for rotating machine	• Raw time streams of data channels • Order tracks vs. RPM viewed in Bode Plot • Order tracks displayed in Polar plot • Dual time signals displayed in orbit plot • Overall in EUrms or $EUrms^2$ vs. RPM

To start, click on the CSA Editor icon in the upper-right corner in EDM and start the CSA Editor. The CSA Editor Wizard dialog box will be displayed. First select the CoCo type and number of channels of the CoCo that this CSA project will support. The number of channels must be equal to or less than the number of physical input channels on the CoCo hardware. Click on the Next Button.

Next select the Order Tracking Analysis template and either Normalized, Constant Frequency Bands or Order Tracks with Phase and click on the Next Button.

Next select the number of input channels to compute the analysis functions. In general it is best to select the minimum number of channels to conserve computational resources for the CSA. This number can be less than or equal to the number of physical input channels on the CoCo hardware. The first channel (Ch1) is reserved for the tachometer signal. Click on the Next Button.

Next enter the description of the CSA. This data will be saved with the CSA and be shown on the CoCo device when the CSA is opened. Click on the Next Button.

Normalized Order Tracks

The setup varies somewhat between Normalized, Constant Frequency Bands or Order Tracks with Phase. This section describes the Normalized Order Tracks setup in detail. The other templates are described in the following sections with emphasis on the differences between them and the Normalized template. We recommend that you read this section before proceeding to the other template types.

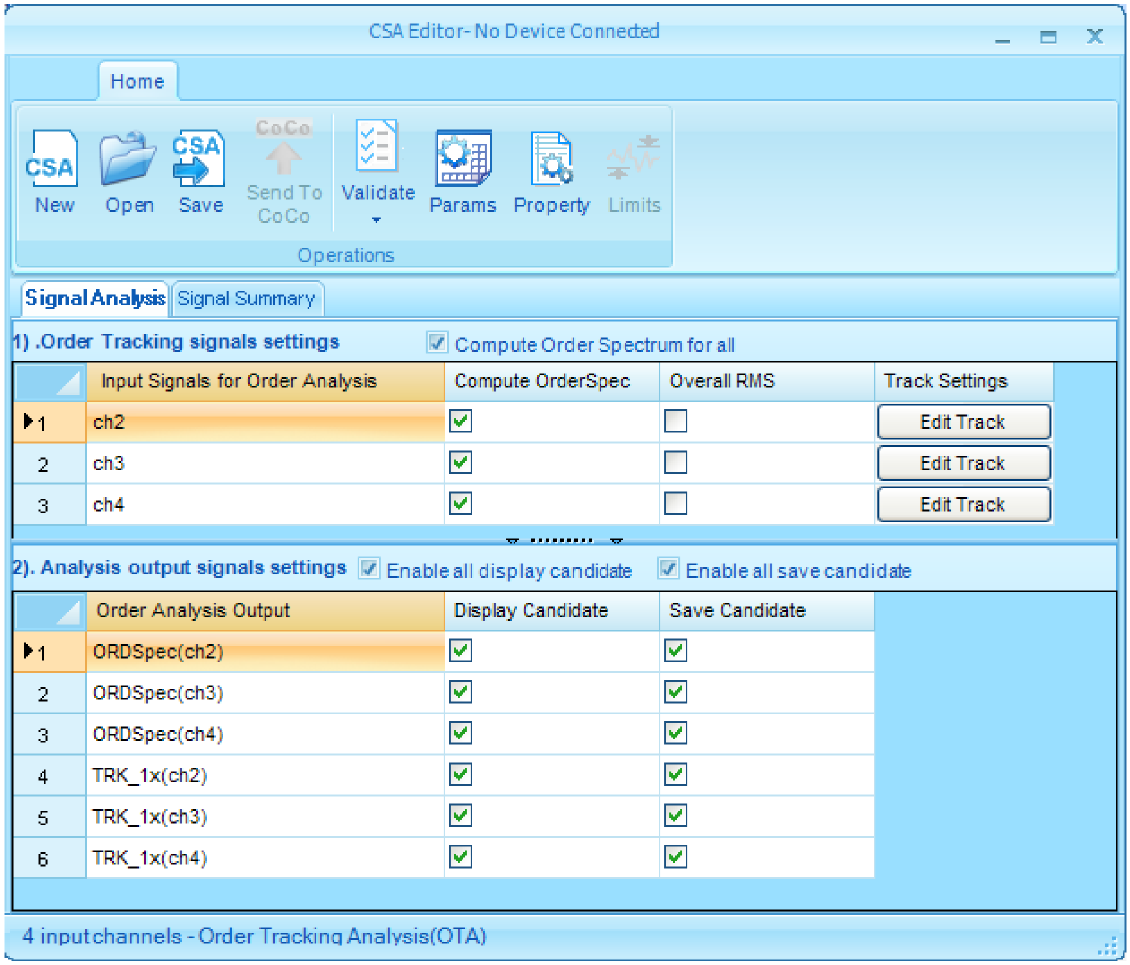
Select Signal Candidates for Normalized Order Tracks

Next the Signal Analysis Display is shown. This display lets you select which signals will be display candidates, save candidates and which will be used to compute the order track analysis. Always select the minimum number of candidates required to conserve computational resources. Note that when a signal is selected as a candidate then it is available on the CoCo hardware, meaning that it can be displayed or save, etc. If the signal is not specified as a candidate in the CSA then it will not be available on the CoCo hardware.

Order Tracking Signal Settings – this display is used to define which input channels are used to compute the order tracks.

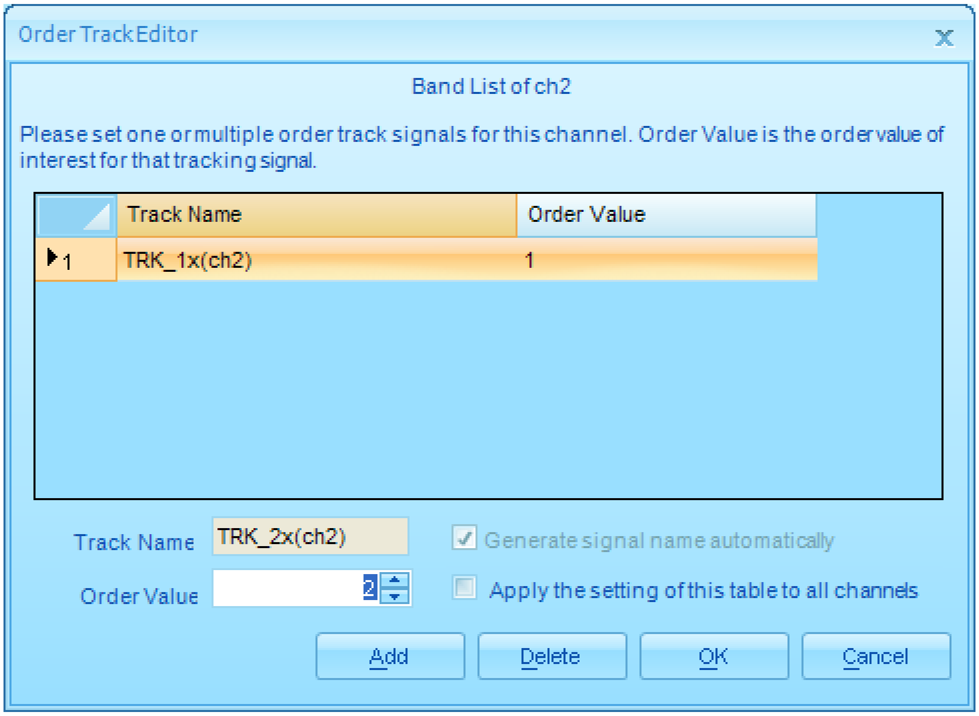
Compute OrderSpec – when the box in this column is checked then the order spectrum for the input channel will be computed.

Overall RMS – when the box in this column is checked then the overall RMS spectrum of the input channel will be computed.



■ Figure 71. Select signal candidates for order tracks.

Track Settings – click on the Edit Track button to edit the track settings for each input channel. Track Settings allows you to define which order tracks will be computed.



■ Figure 72. Edit order track settings.

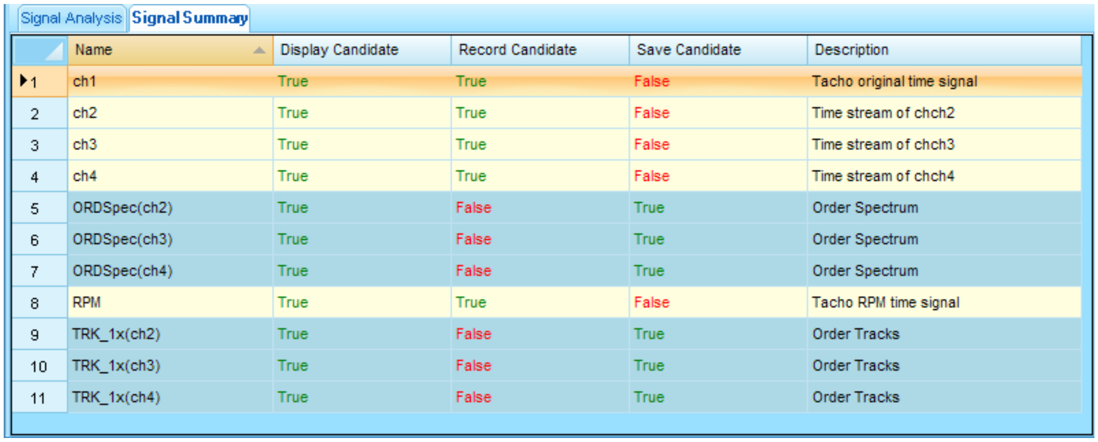
To add an order track enter the Order Value. Use the up and down arrows to choose the order in increments of 0.5X. The Track Name is automatically generated. You can automatically create a similar track for all of the input channels by checking the Apply the settings of this table to all channels button. Next click on the Add Button to create the track. The new track will appear on the Signal Analysis tab.

Analysis output signal settings are used to define which signals can be viewed and saved. As a rule you should select the minimum number of display and save candidates to conserve computation resources.

Display Candidate – check the box in this column to enable the signal as a display candidate.

Save Candidates – check the box in this column to enable the signal as a save candidate.

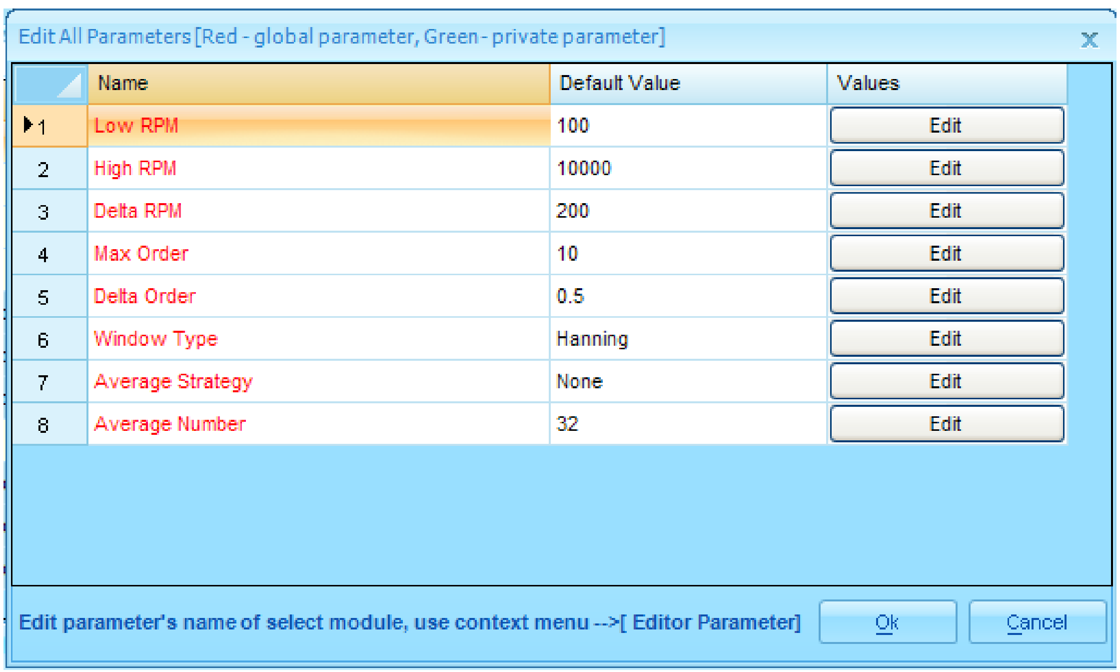
After all the candidate selections are made, a summary can be viewed by selecting the Signal Summary.



■ Figure 73. Signal Summary.

Parameters for Normalized Order Tracks

After the signals are selected the next step is to specify the Parameters by clicking on the Param icon in the CSA Editor Ribbon. The Parameters display lets you change the computation parameters such as low, high and delta RPM, Max Order, etc. Click on the Edit Button next to each item to edit the parameter. Click on Ok to save the updated parameters.



■ Figure 74. Normalized order track parameters.

Low/High RPM: The Low and High RPM define the range of RPM for any order signals, RPM waterfall or RPM traces to be analyzed. If the CoCo detects the current RPM is between the Low and High RPM, it will take the measurement and display it. Otherwise, the system will display the status as RPM High or RPM Low on the top status bar and the required signals will not get computed or displayed.

Delta RPM: The Delta RPM defines the resolution of the RPM trace or the resolution of the waterfall in the RPM axis. The higher the delta RPM, the finer the signals will be stored and displayed in the RPM axis and more storage will be required. Typically the Delta RPM is chosen between 25 and 100.

Max Order: The Max Order defines the highest order number for an order spectrum. The CoCo uses this value to determine the analysis frequency range. You should define the minimum Max Order that your application needs. If you define a Max Order too high, the system must sample at a very high frequency to cover the whole frequency range and the accuracy of lower order estimations may be poor.

Delta Order: The delta Order defines the resolution of the order spectra. The Max Order and Delta Order together define the number of points in a normalized order spectrum. You should define the minimum delta order required for your application to conserve computation resources.

Window Type – defines the data windowing function used in the FFT order track computation.

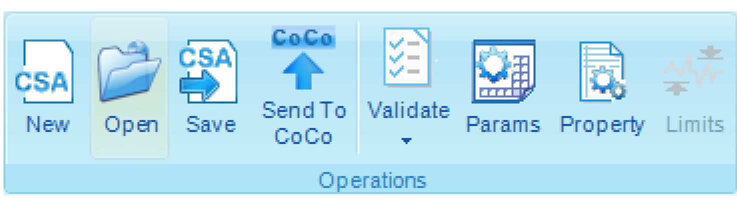
Average Strategy – defines the averaging strategy used in the order track computation: exponential, linear or peak hold.

Average Number – defines the number of averages in a linear average or the weighting factor for exponential averaging.

After the parameters are set click the OK button.

Validation, Save and Upload

After the CSA Wizard is complete and the CSA file is created, connect the host PC to the CoCo device and press the Validate icon to validate the CSA. It may take a few minutes to finish the validation.



■ Figure 75. Validate and Send to CoCo icons in the CSA Editor Ribbon.

The validation process analyzes the CSA file for internal consistency and estimates the required DSA resources required to run the CSA file on the CoCo device. If the DSA exceeds the computational resources of the device then an error message will indicate that you must modify the CSA.

If the Validation passes, then press *Send to CoCo* command in the Validation dialog box to send the CSA project file to CoCo. Alternatively you can manually upload it to CoCo. The uploaded CSA will be classified into the Order Tracking Application Group and can then be opened on the CoCo hardware.

Click the Save Icon on the ribbon to save the CSA file to the host PC. All the signal settings and parameters will be saved with the file. The CSA can be opened later and modified by clicking on the Open icon on the ribbon.

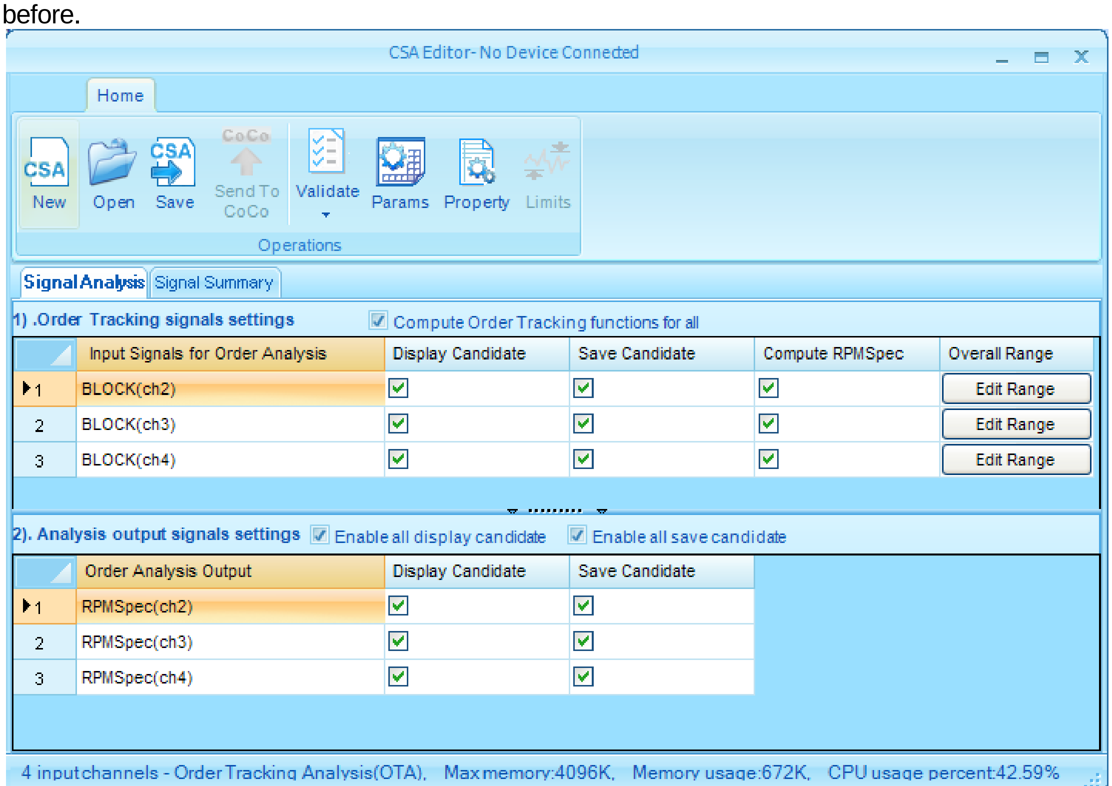
The final step is to upload the CSA file fro the host PC to the CoCo Hardware. The CoCo must be connected first. Click the Sent To CoCo icon on the Ribbon. You will be prompted to enter a name for the CSA file. Then click OK. A message will indicate that the file was successfully sent to the CoCo hardware. After the CSA is saved then the CSA Editor can be closed. Now the CSA can be opened on the CoCo and used to make order track measurements.

Constant Frequency Order Tracks

The initial steps for creating a CSA with Constant Frequency Order Tracks are identical to the previous section. This section describes the differences between the Constant Frequency and Normalized template. We recommend that you read the previous section before reading this one.

Select Signal Candidates for Constant Frequency Order Tracks

The difference occurs on the Signal Analysis tab. Display and save candidates are the same as before.

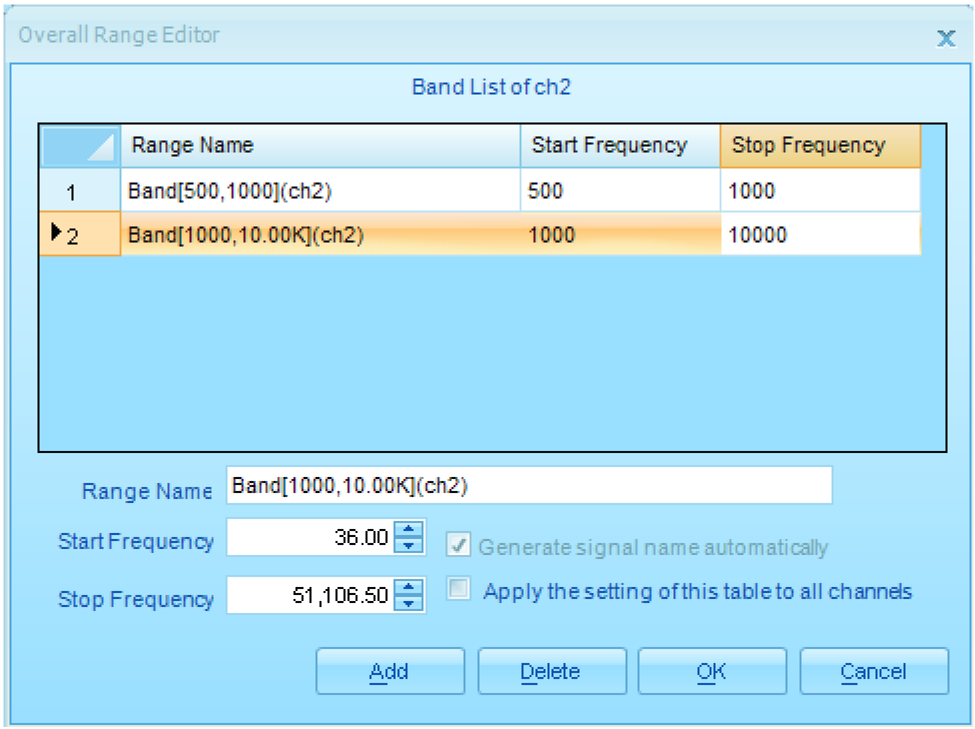


■ Figure 76. Select signal candidates.

Compute RPMSpec – check the box in this column to compute the RPM spectrum for this input channel.

Overall Range – click on the Edit Range button in this column to edit the frequency range settings used to compute the order tracks for this input channel.

To add a Fixed Band RPM Spectrum, first edit the Start and Stop Frequencies using the up and down arrows and then click the Add Button. The frequencies can be edited in the table after the range is created by clicking on the frequency values and editing them. The Range Name is automatically generated and include the start and stop frequencies. You can remove a Fixed Band RPM Spectrum by highlighting it and clicking the Delete Button. After you have created all the desired spectra click the OK Button.



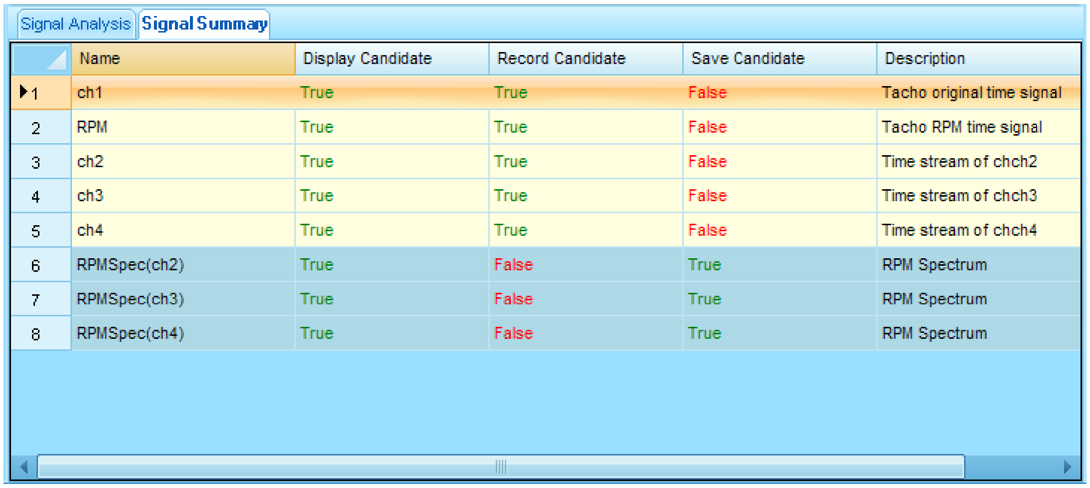
■ Figure 77. Edit order track settings.

Analysis output signal settings are used to define which signals can be viewed and saved. As a rule you should select the minimum number of display and save candidates to conserve computation resources.

Display Candidate – check the box in this column to enable the signal as a display candidate.

Save Candidates – check the box in this column to enable the signal as a save candidate.

After all the candidate selections are made, a summary can be viewed by selecting the Signal Summary.



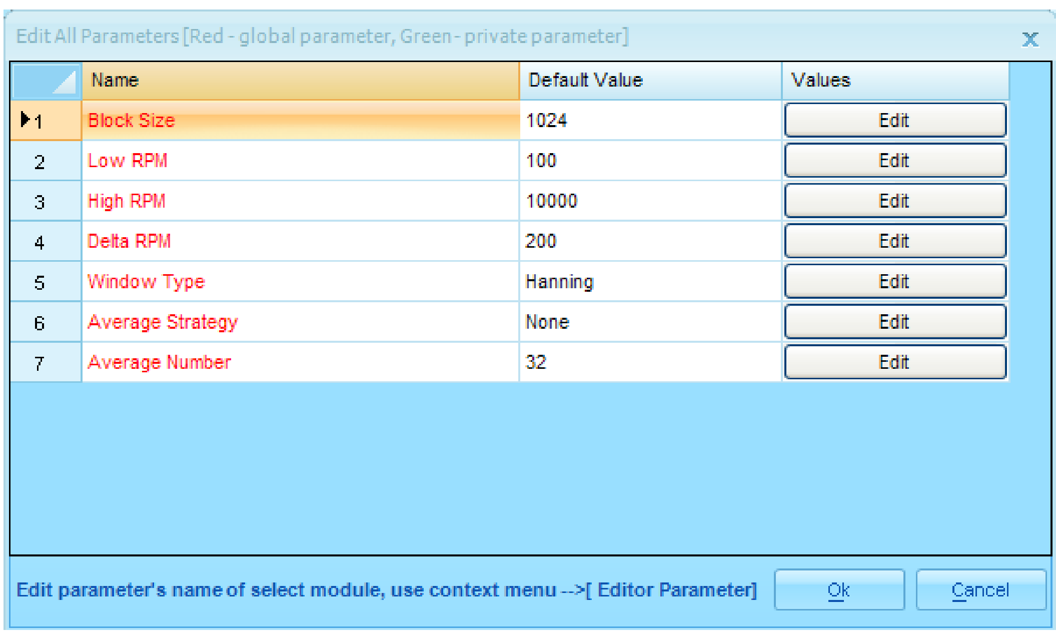
Signal Analysis | Signal Summary

	Name	Display Candidate	Record Candidate	Save Candidate	Description
1	ch1	True	True	False	Tacho original time signal
2	RPM	True	True	False	Tacho RPM time signal
3	ch2	True	True	False	Time stream of chch2
4	ch3	True	True	False	Time stream of chch3
5	ch4	True	True	False	Time stream of chch4
6	RPMSpec(ch2)	True	False	True	RPM Spectrum
7	RPMSpec(ch3)	True	False	True	RPM Spectrum
8	RPMSpec(ch4)	True	False	True	RPM Spectrum

■ Figure 78. Signal Summary.

Parameters

After the signals are selected the next step is to specify the Parameters by clicking on the Param icon in the CSA Editor Ribbon. The Parameters display lets you change the computation parameters such as block size, low, high and delta RPM, etc. Click on the Edit Button next to each item to edit the parameter. Click on Ok to save the updated parameters.



Edit All Parameters [Red - global parameter, Green - private parameter]

	Name	Default Value	Values
1	Block Size	1024	Edit
2	Low RPM	100	Edit
3	High RPM	10000	Edit
4	Delta RPM	200	Edit
5	Window Type	Hanning	Edit
6	Average Strategy	None	Edit
7	Average Number	32	Edit

Edit parameter's name of select module, use context menu -->[Editor Parameter] Ok Cancel

■ Figure 79. Parameter settings.

Block Size defines the number of data points used in the FTT computation.
Low/High RPM: The Low and High RPM define the range of RPM for any order signals, RPM waterfall or RPM traces to be analyzed. If the CoCo detects the current RPM is between the Low

and High RPM, it will take the measurement and display it. Otherwise, the system will display the status as RPM High or RPM Low on the top status bar and the required signals will not get computed or displayed.

Delta RPM: The Delta RPM defines the resolution of the RPM trace or the resolution of the waterfall in the RPM axis. The higher the delta RPM, the finer the signals will be stored and displayed in the RPM axis and more storage will be required. Typically the Delta RPM is chosen between 25 and 100.

Window Type – defines the data windowing function used in the FFT order track computation.

Average Strategy – defines the averaging strategy used in the order track computation: exponential, linear or peak hold.

Average Number – defines the number of averages in a linear average or the weighting factor for exponential averaging.

After the parameters are set click the OK button.

Validation, Save and Upload

After all the Signal Settings and Parameters are defined then you should validate the CSA, save it and upload it to the CoCo hardware.

After the CSA is saved then the CSA Editor can be closed

Order Tracks with Phase

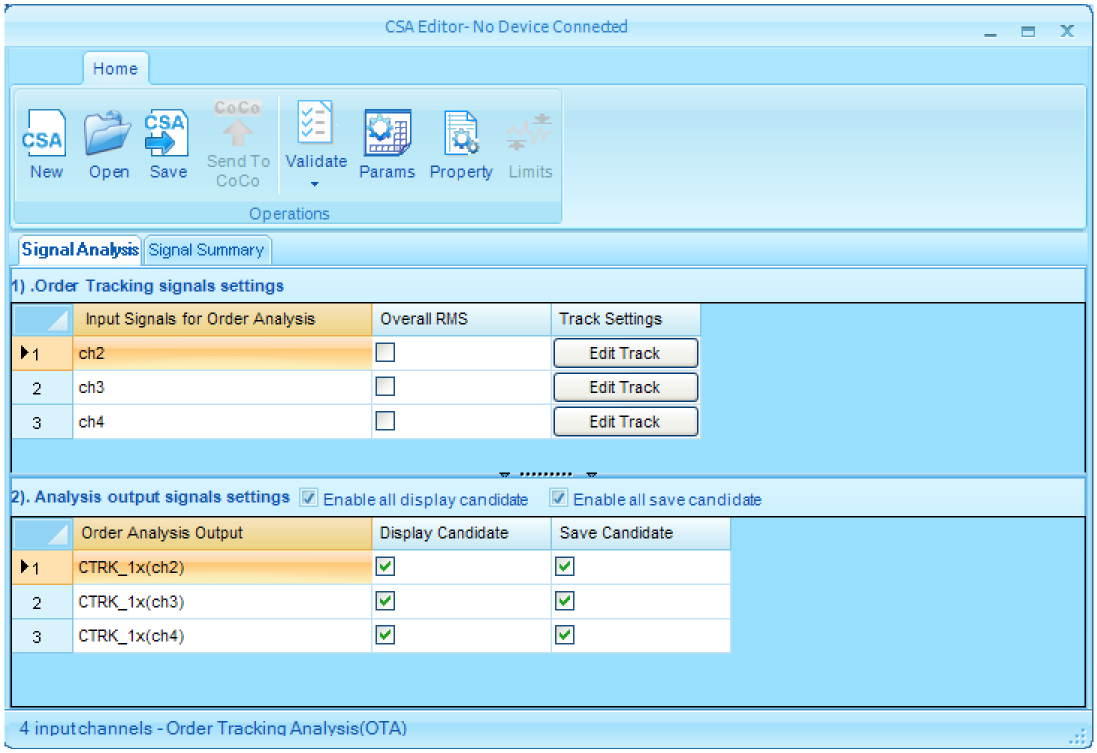
The initial steps for creating a CSA with Order Tracks with Phase are identical to the previous sections. This section describes the differences between the Order Tracks with Phase and the Normalized template. We recommend that you read the previous sections before reading this one.

The initial steps for creating a CSA with Order Tracks with Phase are identical to the previous section. The difference is on the Signal Analysis tab.

Select Signal Candidates for Order Tracks with Phase

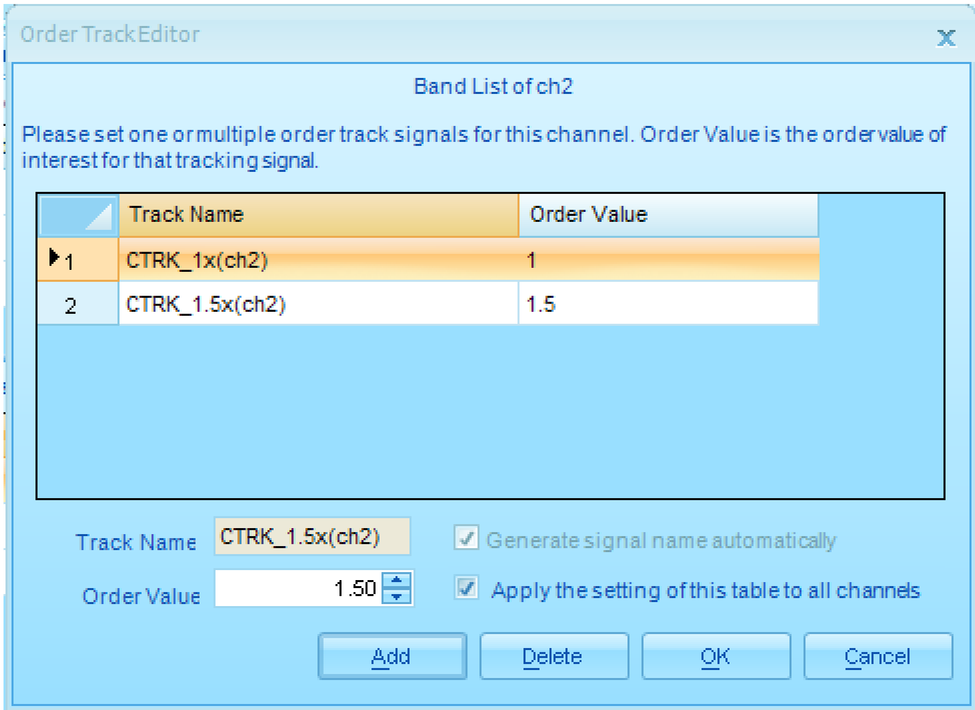
Display and save candidates are the same as before.

Overall RMS – when the box in this column is checked then the overall RMS spectrum of the input channel will be computed.



■ Figure 80. Select signal candidates for order tracks with phase.

Track Settings – click on the Edit Track button to edit the track settings for each input channel. Track Settings allows you to define which order tracks will be computed.



■ Figure 81. Order track settings.

To add an order track enter the Order Value. Use the up and down arrows to choose the order in increments of 0.5X. The Track Name is automatically generated. You can automatically create a similar track for all of the input channels by checking the Apply the settings of this table to all channels button. Next click on the Add Button to create the track. The new track will appear on the Signal Analysis tab.

Analysis output signal settings are used to define which signals can be viewed and saved. As a rule you should select the minimum number of display and save candidates to conserve computation resources.

Display Candidate – check the box in this column to enable the signal as a display candidate.

Save Candidates – check the box in this column to enable the signal as a save candidate.

After all the candidate selections are made, a summary can be viewed by selecting the Signal Summary.

Signal Analysis | Signal Summary

	Name	Display Candidate	Record Candidate	Save Candidate	Description
1	ch1	True	True	False	Tacho original time signal
2	RPM	True	True	False	Tacho RPM time signal
3	ch2	True	True	False	Time stream of chch2
4	ch3	True	True	False	Time stream of chch3
5	ch4	True	True	False	Time stream of chch4
6	CTRK_1x(ch2)	True	False	True	Order Tracks
7	CTRK_1x(ch3)	True	False	True	Order Tracks
8	CTRK_1x(ch4)	True	False	True	Order Tracks

■ Figure 82. Signal summary.

Parameters

After the signals are selected the next step is to specify the Parameters by clicking on the Param icon in the CSA Editor Ribbon. The Parameters display lets you change the computation parameters such as low, high and delta RPM, etc. Click on the Edit Button next to each item to edit the parameter. Click on Ok to save the updated parameters.

Edit All Parameters [Red - global parameter, Green - private parameter]

	Name	Default Value	Values
1	Low RPM	100	Edit
2	High RPM	10000	Edit
3	Delta RPM	200	Edit
4	Delta Order	0.5	Edit
5	Max Order	10	Edit
6	Window Type	Hanning	Edit
7	Average Strategy	None	Edit
8	Average Number	32	Edit

Edit parameter's name of select module, use context menu -->[Editor Parameter] Ok Cancel

■ Figure 83. Order track parameters.

Low/High RPM – defines the range of RPM for any order signals, RPM waterfall or RPM traces to be analyzed. If the CoCo detects the current RPM is between the Low and High RPM, it will take the measurement and display it. Otherwise, the system will display the status as RPM High or RPM Low on the top status bar and the required signals will not get computed or displayed.

Delta RPM – defines the resolution of the RPM trace or the resolution of the waterfall in the RPM axis. The higher the delta RPM, the finer the signals will be stored and displayed in the RPM axis and more storage will be required. Typically the Delta RPM is chosen between 25 and 100.

Max Order – defines the highest order number for an order spectrum. The CoCo uses this value to determine the analysis frequency range. You should define the minimum Max Order that your application needs. If you define a Max Order too high, the system must sample at a very high frequency to cover the whole frequency range and the accuracy of lower order estimations may be poor.

Delta Order – defines the resolution of the order spectra. The Max Order and Delta Order together define the number of points in a normalized order spectrum. You should define the minimum delta order required for your application to conserve computation resources.

Window Type – defines the data windowing function used in the FFT order track computation.

Average Strategy – defines the averaging strategy used in the order track computation: exponential, linear or peak hold.

Average Number – defines the number of averages in a linear average or the weighting factor for exponential averaging.

After the parameters are set click the OK button.

Validation, Save and Upload

After all the Signal Settings and Parameters are defined then you should validate the CSA, save it and upload it to the CoCo hardware.

After the CSA is saved then the CSA Editor can be closed
f

CoCo Operation for Order Tracks

The CoCo operation for order tracks is similar for all three templates: Normalized, Constant Frequency and Order Track with Phase. In the following sections the Normalized will be described in detail first. Then the other templates will be discussed with emphasis on the differences between them and the Normalized template. We recommend you read the Normalized section first before proceeding to the others.

Normalized Order Tracking

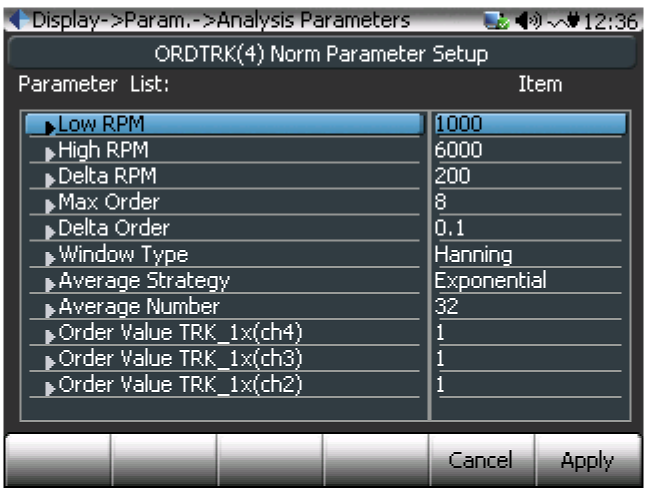
This section describes the CoCo operation for Normalized Order Tracking in detail. The operation consists of selecting a CSA project, setting the Analysis Parameters, setting up the displays and making the measurement.

Selecting a CSA Project

The first step in setting up the CoCo to make an order track measurement is to select a CSA file. An order track file must be uploaded from the host PC to the CoCo first. Press the Analysis Button then select the Order Tracking template and choose a CSA file to run and press the Enter Button. Order tacking CSAs are automatically saved in the Order Track Application Group to help organize the files on the CoCo.

Analysis Parameters

The next step is to set the Analysis Parameters. Note that the default values are defined when the CSA is created in the CSA editor but can be modified after the CSA is uploaded to the CoCo. Select Analysis Parameters under the Param. Button. To modify any of the parameters move the cursor to the desired parameter using the Up and Down Buttons and press the Enter Button. Then select from the menu or edit the value using the key pad display.



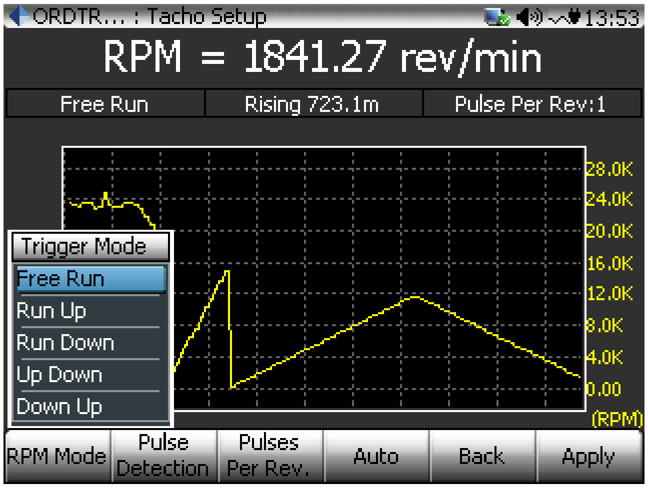
■ Figure 84. Set the Analysis Parameters.

After you have set all the necessary Analysis Parameters press the Apply Button so save the new parameters with the CSA. Next the Display Window will be shown.

Acquisition Mode

In order tracking mode, Acquisition Mode controls the processing of the tachometer signal including the trigger mode, pulse detection level and the pulses per revolution. The current RPM is displayed in numerical and graphical format. Also the trigger mode, level and pulses per rev settings are indicated on the display. The settings can be changed to ensure the most accurate RPM processing possible.

Trigger Mode includes the following settings:



■ Figure 85. RPM trigger mode.

Free Run – data is acquired regardless of the direction of the RPM as long as the RPM is between low and high RPM limits.

Run Up – data is only acquired when RPM starts below low RPM limit and then increases. Acquisition stops when RPM exceeds high RPM limit.

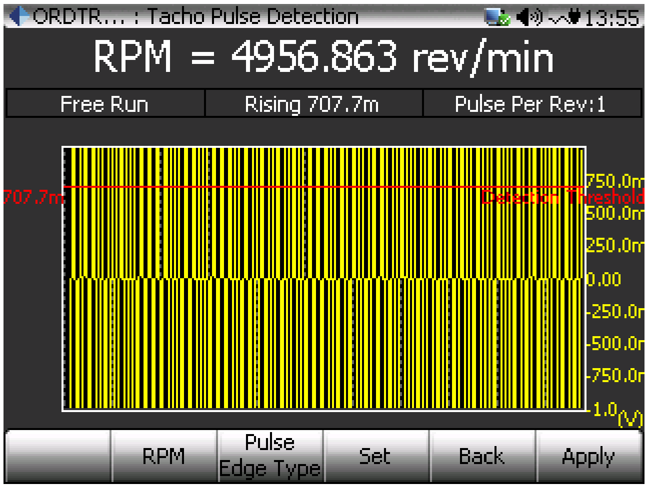
Run Down – data is only acquired when RPM starts above high RPM limit and then decreases. Acquisition stops when RPM exceeds low RPM limit.

Up Down – data is only acquired when RPM starts below low RPM limit and increasing. Acquisition continues as RPM increases past high RPM limit and then as RPM decreases past the low RPM limit.

Down Up – data is only acquired when RPM starts above high RPM limit and decreasing. Acquisition continues as RPM decreases past low RPM limit and then as RPM increases past the high RPM limit.

Pulse Detection is used to set the voltage and edge to detect the tachometer signal.

RPM – use the Up and Down Arrow Buttons to set the RPM level. The horizontal line indicates the current level. Set the level so that the tachometer signal consistently crosses the red line on every pulse.



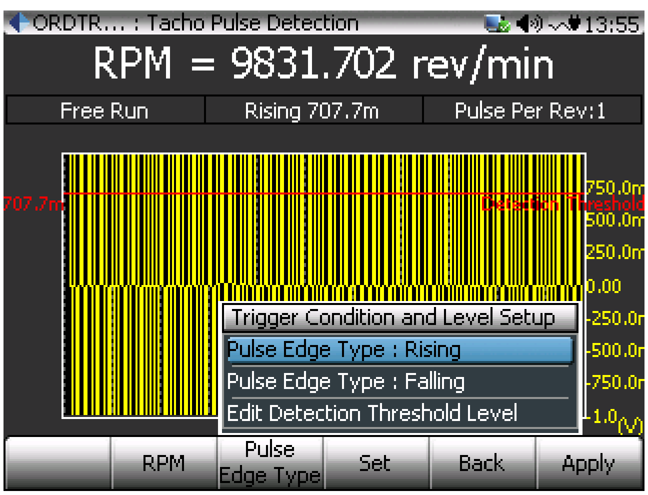
■ Figure 86. Tacho pulse detection settings.

Pulse Edge Detection – controls the pulse trigger edge conditions.

Rising – sets the trigger condition to a rising slope.

Falling – sets the trigger condition to a falling slope.

Edge Detection Threshold Level – lets you enter a RPM level threshold using the keypad instead of the up and down arrow buttons.

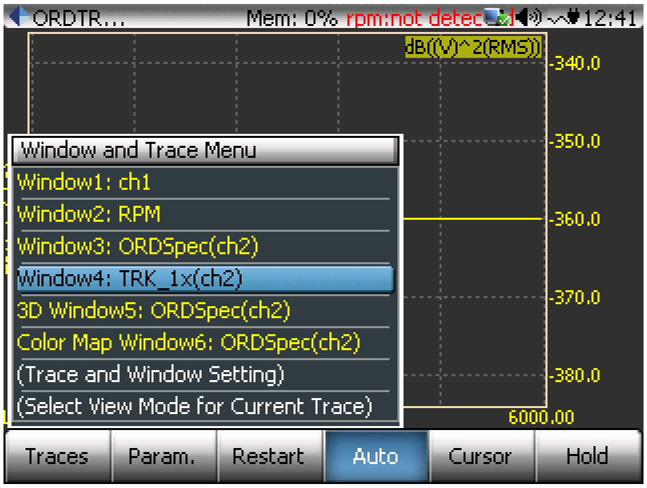


■ Figure 87. Tacho pulse edge type settings.

After all the settings are complete press the Apply button to save the settings then press the Back Button to view the order track measurement window.

Displays

After the Analysis Parameters are set, the next step is to define the necessary displays. Press the Traces Button to view the existing displays or to modify the traces.



■ Figure 88. Modify the traces.

You can change from one window to another by selecting it from the menu and pressing the Enter Button.

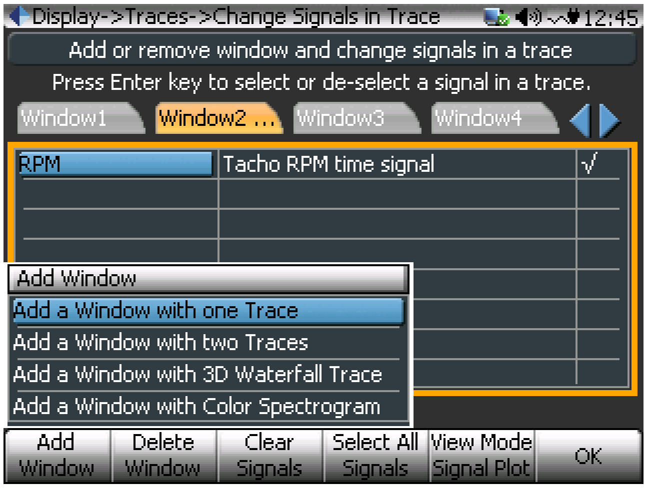
The time trace of **Channel 1** is reserved for the tachometer input. It can be used to view the raw tachometer signal vs. time. This can be useful to diagnose any problems with the tachometer signal.

The **RPM trace** shows the tachometer signal after it is processed and converted into RPM vs. time.

Select View Mode for Current Trace lets you modify the format of the current trace. The options depend on the quantity that is defined in the trace.

Trace and Window Settings let you add, delete or modify an existing window.

Add Window includes windows that are specific to order tracking: 3D Waterfall Trace and Color Spectrogram.



■ Figure 89. Add a window for order tracking.

3D Waterfall Trace is used to display order spectra and order tracks vs. RPM in three dimensions.

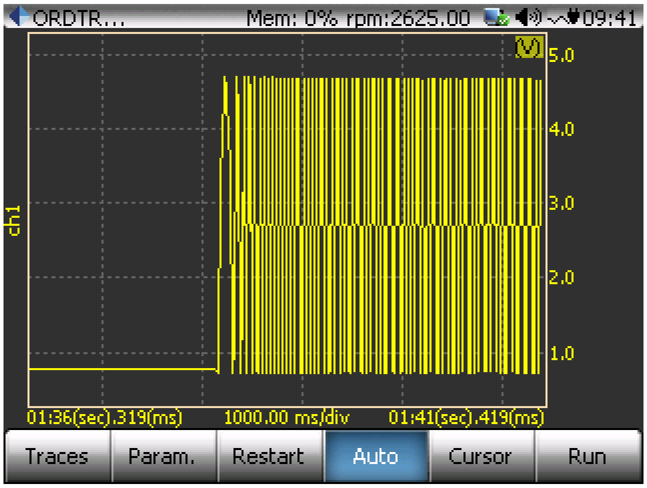
Color Spectrogram is used to display order spectra and order tracks vs. RPM in 2 dimensions using color to represent the magnitude of the signal.

The limits of the 3D waterfall and color spectrogram depend on the Analysis Parameters such as low and high RPM and max order. The resolution of the plots depends on the Analysis Parameters delta RPM and delta order. Note that a larger RPM or order span or a higher resolution will affect the acquisition of the data and the memory required. You should use the minimum span and resolution necessary for your application to conserve computation resources and get the highest quality results.

Make a Measurement

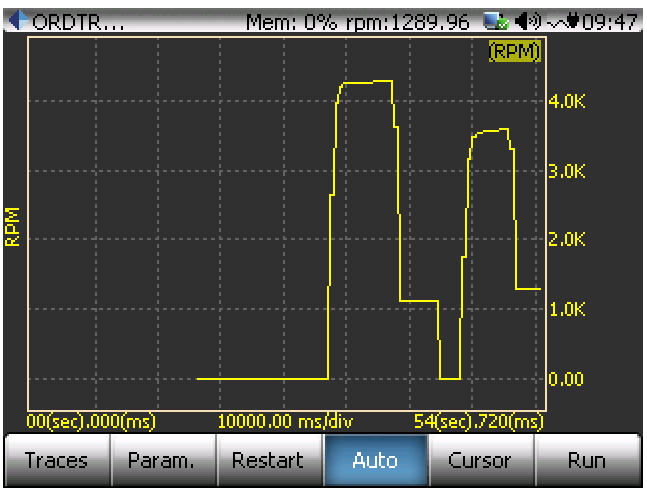
To start the measurement press the Run Button. Order track measurements require an RPM input so spectra displays may only update when an RPM signal is detected. The current RPM is displayed in the status bar at the top of the display. If no RMP signal is detected or if the RPM not between the minimum and maximum RPM parameter, then the spectrum will not be updated and the RPM status will display "rpm not detected", "rpm: too low," or "rpm: too high".

The Ch1 trace is reserved for the tachometer input signal. This shows the raw time waveform of the tachometer signal before it is processed into the RPM vs. time signal. shows a tachometer signal. The example has no data for the first few seconds until the tachometer is activated. Then the increasing tachometer speed can be seen in the increasing frequency of the sinusoidal wave.



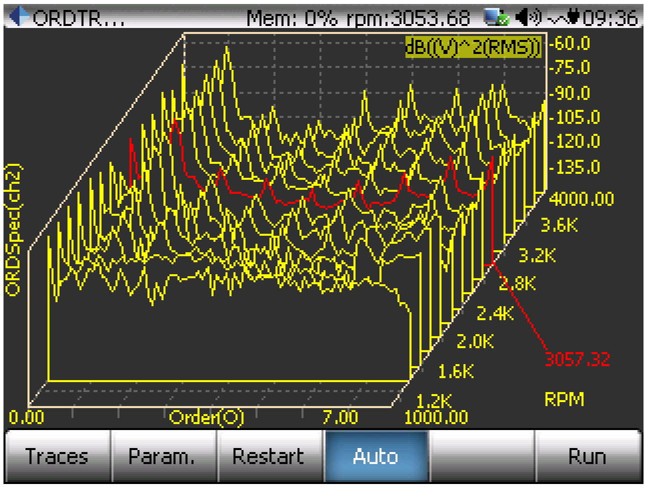
■ Figure 90. Ch1 is reserved for the tachometer input trace.

The RPM trace is generated by processing the tachometer signal and computing the RPM vs. time. Figure 91 shows an example RPM trace with two run up, run down sequences.



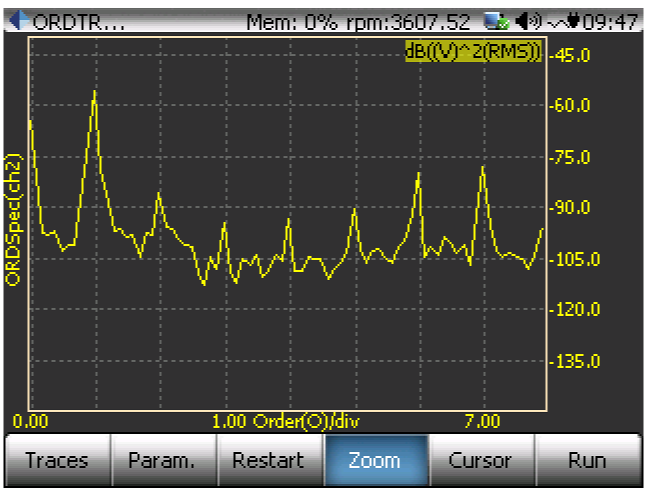
■ Figure 91. RPM trace shows the processed tachometer input signal vs. time.

A 3D Waterfall shows order spectra vs. RPM. As the RPM changes the current RPM is indicated as a red spectrum as shown in Figure 92. Note that the spectra is plotted vs. orders of the fundamental.



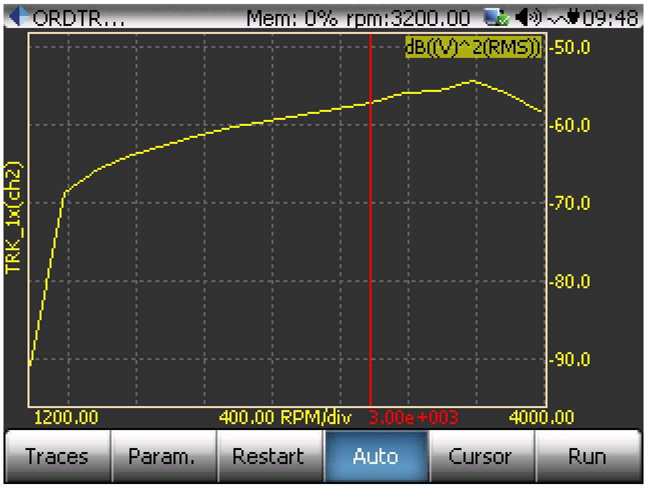
■ Figure 92. 3D Order spectrum shows current rpm in red.

An order spectrum shows the amplitude of orders at the indicated RPM. This type of trace is the same as a cross section of the 3D waterfall cut along the current RPM value.



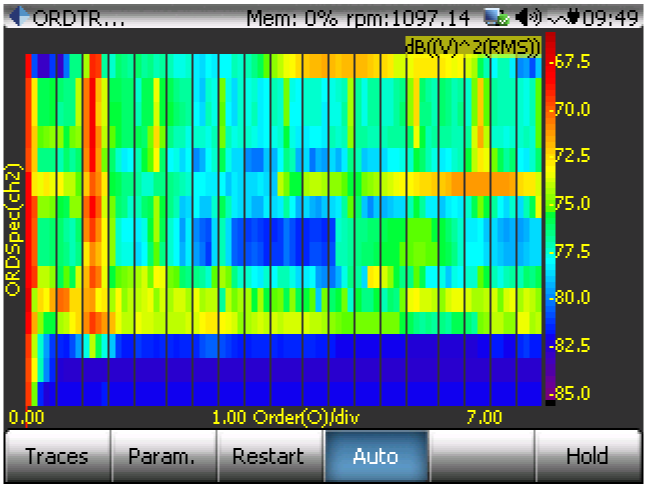
■ Figure 93. Order spectrum shows the amplitudes of orders at the current RPM.

An order track shows the amplitude of a specific order vs. RPM. This type of trace is the same as a cross section of the 3D waterfall cut along the specific order value.



■ Figure 94. Order track shows the amplitude of an order vs. RPM.

A color spectrogram is a 2D trace that shows spectra vs. RPM using color to indicate the amplitude of the spectra.



■ Figure 95. Color spectrogram shows order spectra vs. RPM.

A horizontal black gap in the spectrogram indicates that no spectra was recorded at the specific RPM. This can occur if the RPM changes too rapidly for the spectra to be acquired and processed. The same artifact can be seen in a 3D waterfall. This can be avoided by either slowing down the rate of change of the RPM (slew rate), or by modifying the Analysis Parameters such as reducing the order span or RPM span. This will reduce the computation resources required and possibly improve the results.

During a measurement you can press the Restart Button to reset the averaging. This will also reset all of the spectra in a 3D waterfall and spectrogram to zero

After you have acquired the desired data press the Hold Button to stop the measurement. You can and use the Save Button to save the results. Refer to the Basic User Manual for details on using Save, Cursors, and Traces

Constant Frequency Order Tracks

The CoCo operation for constant frequency order tracks is similar to operation for normalized order tracks. This section will describe the differences. We recommend that you read the previous section before proceeding with this one. This section describes the Analysis Parameters and displays.

Analysis Parameters

Most of the Analysis Parameters for constant frequency order tracks are the same as for normalized order tracks including: low, high and delta rpm, window type and average strategy and number. However since constant frequency order tracks use a fixed sampling rate you must specify the sampling rate and block size. Note that this is the not the case for normalized order tracks which use digital re-sampling to automatically change the sampling rate depending on the RPM.

Sampling Rate - Constant frequency order tracks use a fixed sampling rate FFT to compute spectra. Therefore you must manually set the Sampling Rate under the Param. Button. Select a sampling rate that is high enough to capture the highest frequency of interest for your analysis. However you should not select a sampling rate that is higher than necessary as it will require more computational resources and may reduce the quality of lower frequency data.

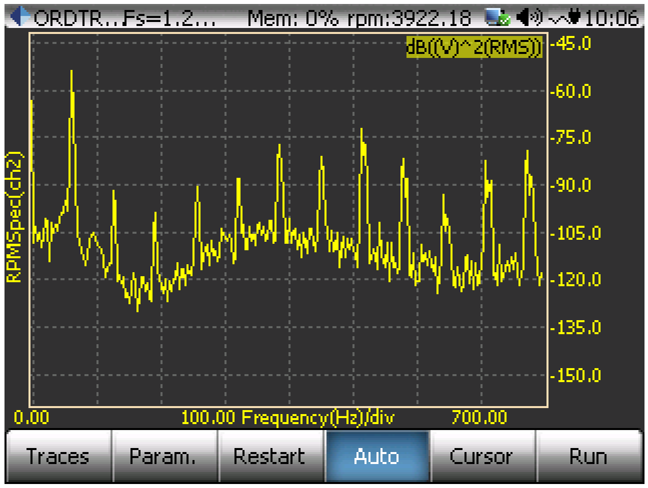
Block Size – The block size specifies the number of data points in an FFT window and also the FFT frequency resolution. You should choose an optimal block size for your application to give the best results and also conserve computational resources.

Make a Measurement

The operation is the same as the normalized order tracks. To start the measurement press the Run Button. Order track measurements require an RPM input so spectra displays may only update when an RPM signal is detected. The current RPM is displayed in the status bar at the top of the display. If no RMP signal is detected or if the RPM not between the minimum and maximum RPM parameter, then the spectrum will not be updated and the RPM status will display "rpm not detected", "rpm: too low," or "rpm: too high".

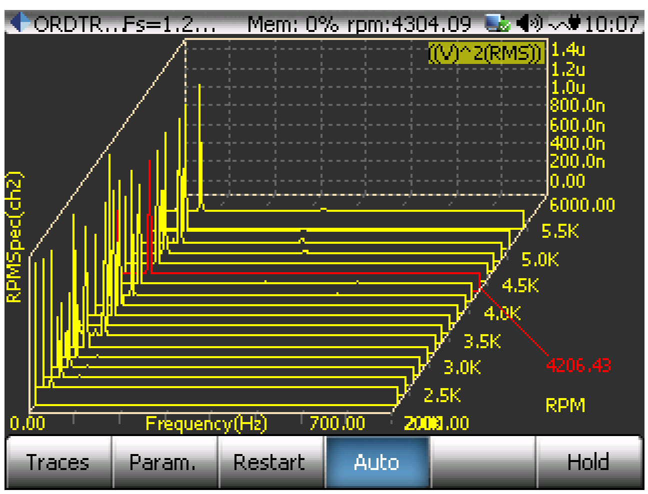
The trace types are similar to the normalized order tracks. The Ch1 is reserved for the tachometer input signal. The RPM trace shows the RPM vs. time.

The most significant difference between constant frequency and normalized order tracks is that constant frequency order track traces are plotted vs. frequency instead of orders. Figure 96 shows an RPM spectrum vs. frequency.



■ Figure 96. RPM spectra shows the spectra vs. frequency.

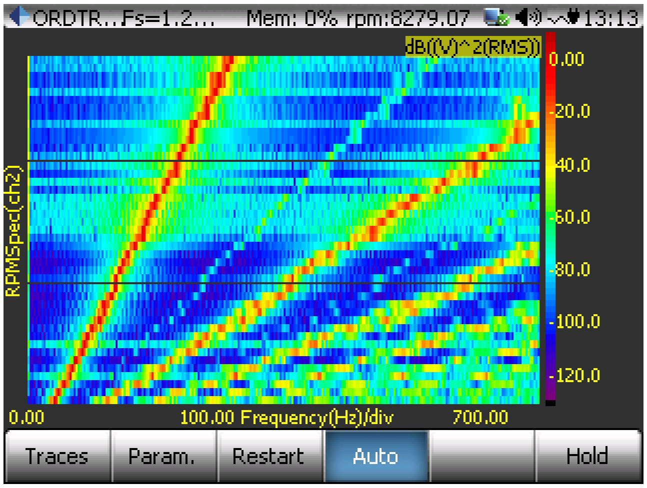
Figure 97 shows a 3D waterfall of an RPM spectra. Note the spectra are plotted vs. frequency not orders as is done in the normalized order tracks. The consequence of this is that orders are observed as ridges that appear at skewed angles relative to the perpendicular axes of the figure. This is more apparent in the color spectrogram.



■ Figure 97. 3D waterfall of RPM spectra shows spectra vs. RPM.

Figure 98 shows a color spectrogram of an RPM spectra. Note the spectra are plotted vs. frequency instead of order as is done in the normalized order tracks. The first order can clearly be seen as a red line at an angle. Higher orders will appear as lines radiating from the origin at increasing angles.

A horizontal black gap indicates that no spectra was recorded at the specific RPM. This can occur if the RPM changes too rapidly for the spectra to be acquired and processed. The same artifact can also be seen in a 3D waterfall. This can be avoided by either slowing down the rate of change of the RPM (slew rate), or by modifying the Analysis Parameters such as reducing the sampling rate, block size or RPM span. This will reduce the computation resources required and possibly improve the results.



■ Figure 98. Color spectrogram RPM spectra.

Most other aspects of the operation are similar to normalized order tracks. Refer to the previous section and to the Basic User Manual for more detail.

Order Tracks with Phase

The CoCo operation for order tracks with phase is similar to operation for normalized order tracks. This section will describe the differences. We recommend that you read the previous sections before proceeding with this one.

Analysis Parameters

Order tracks with phase use the same digitally re-sampled method as the normalized order track template. Therefore, the Analysis Parameters for order tracks with phase are the same as for normalized order tracks

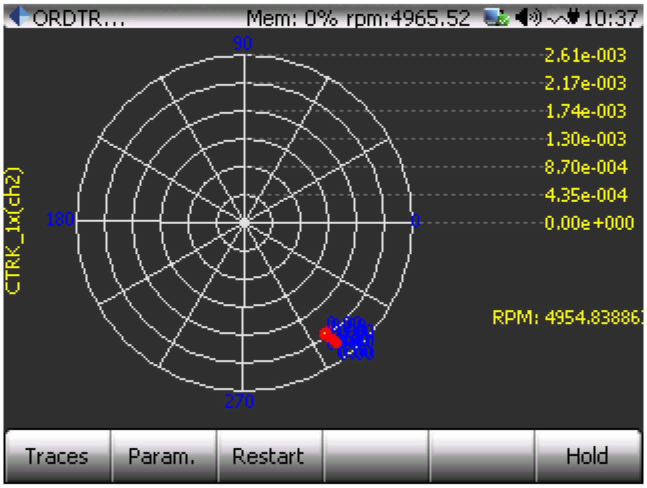
Make a Measurement

The operation is the same as the normalized order tracks. To start the measurement press the Run Button. Order track measurements require an RPM input so spectra displays may only update when an RPM signal is detected. The current RPM is displayed in the status bar at the top of the display. If no RMP signal is detected or if the RPM not between the minimum and maximum RPM parameter, then the spectrum will not be updated and the RPM status will display "rpm not detected", "rpm: too low," or "rpm: too high".

The trace types are similar to the normalized order tracks. The Ch1 is reserved for the tachometer input signal. The RPM trace shows the RPM vs. time.

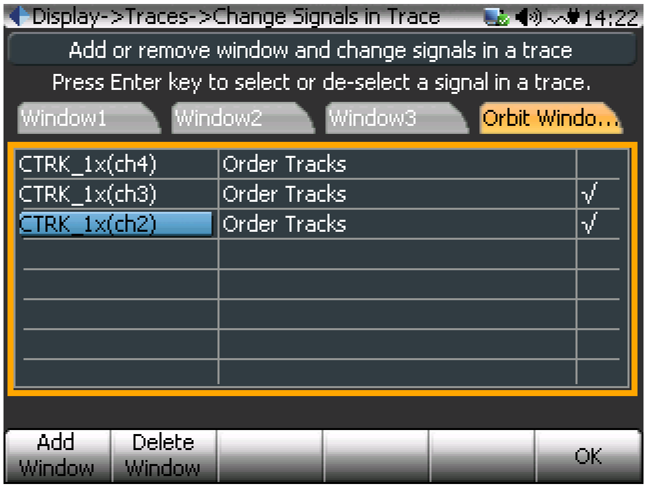
Order tracks with phase allow additional traces that are not available in the normalized order tracks template including: polar, orbit and bode traces.

A polar trace shows the magnitude and phase in a polar axis of the current signal relative to the tachometer signal. The data is shown as a red dot. The magnitude scale is shown on the right.

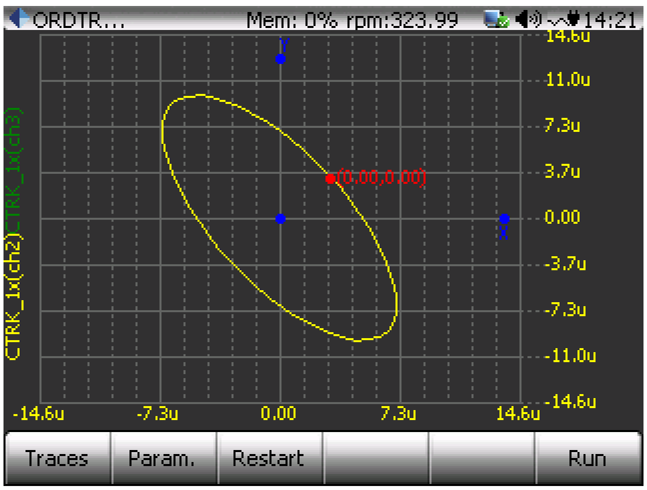


■ Figure 99. Polar trace shows the magnitude and phase of the measured signal relative to the tachometer reference.

The orbit trace shows a parametric plot of two input signals. It is usually used to visualize the displacement of a rotating machine. Note that two time stream input channels must be selected for an orbit plot.

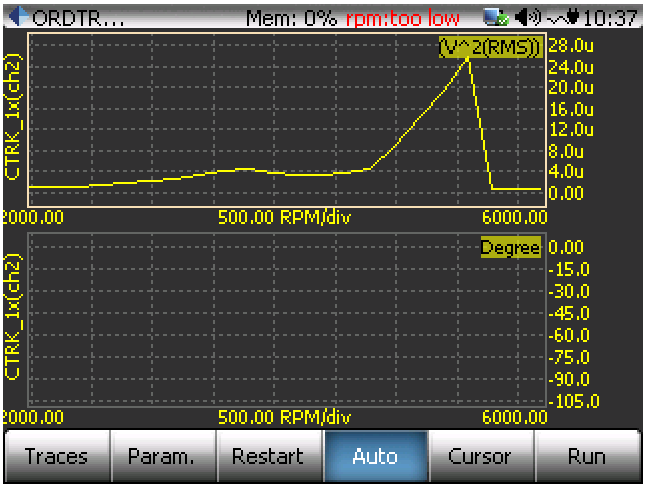


■ Figure 100. Orbit window setup requires two input time streams.



■ Figure 101. Orbit plot shows.

The Bode trace shows the magnitude and phase of an input signal relative to the tachometer reference vs. RPM. The top trace is the magnitude and the bottom is phase.



■ Figure 102. The Bode trace shows the amplitude and phase relative to the tachometer reference plotted vs. RPM.

Most other aspects of the operation are similar to normalized order tracks. Refer to the previous section and to the Basic User Manual for more detail.

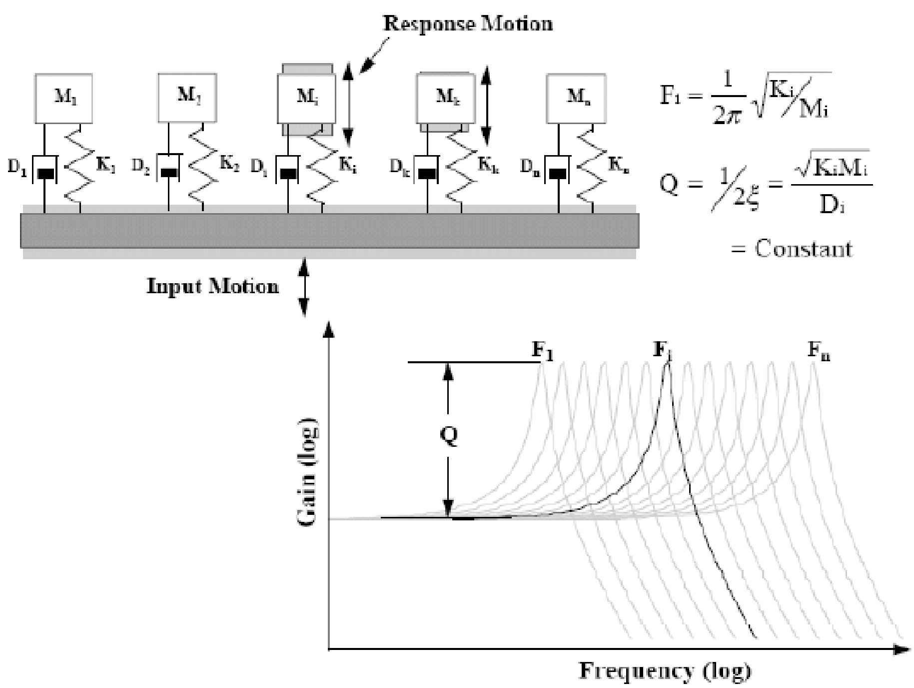
6. SHOCK RESPONSE SPECTRUM ANALYSIS

A **Shock** Response Spectrum (SRS) is a graphical representation of an arbitrary transient acceleration input, such as shock in terms of how a Single Degree Of Freedom (SDOF) system (like a mass on a spring) responds to that input. Actually, it shows the peak acceleration response of an infinite number of SDOFs, each of which have different natural frequencies. Acceleration response amplitude is represented on the vertical axis, and natural frequency of any given SDOF is shown on the horizontal axis.

A SRS is generated from a shock waveform using the following process:

1. Pick a damping ratio for the SRS
2. Assume a hypothetical Single Degree of Freedom System (SDOF), with a damped natural frequency of x Hz
3. Calculate (by time base simulation or something more subtle) the maximum instantaneous absolute acceleration experienced by the mass element of your SDOFs at any time during (or after) exposure to the shock in question. Plot this in g's (g's are standard, but pick any unit of acceleration you want) against the frequency (x) of the hypothetical system.
4. Repeat steps 2 and 3 for other values of X, for example, logarithmically up to 1000x.

The resulting plot of peak acceleration vs. test system frequency is called a Shock Response Spectrum, or SRS. This process can be depicted in the following picture:



■ Figure 103. Illustration of multi-degree of freedom system model used to compute SRS.

A SDOF mechanical system consists of the following components:

- Mass, whose value is represented with the variable, M
- Spring, whose stiffness is represented with the variable, K
- Damper, whose damping coefficient is represented with the variable C.

The resonance frequency, Fi, and the critical damping factor, ζ, characterize a SDOF system, where:

$$Fi = \frac{1}{2\pi}\sqrt{\frac{K}{M}}$$

$$\zeta = \frac{\mathrm{c}}{2\sqrt{\mathrm{KM}}}$$

For light damping ratio where ζ is less than or equal to 0.05, the peak value of the frequency response occurs in the immediate vicinity of Fi and is given by the following equation, where Q is the quality factor:

$$Q = \frac{1}{2\zeta}$$

Any transient waveform can be presented as an SRS, but the relationship is not unique; many different transient waveforms can produce the same SRS (something one can take advantage of through a process called "Shock Synthesis"). The SRS does not contain all the information about the transient waveform from which it was created because it only tracks the peak instantaneous accelerations.

Different damping ratios produce different SRSs for the same shock waveform. Zero damping will produce a maximum response. Very high damping produces a very flat SRS. The level of damping is demonstrated by the "quality factor", Q which can also be thought of transmissibility in sinusoidal vibration case. A damping ratio of 5% results in a Q of 10. An SRS plot is incomplete if it doesn't specify the assumed Q or damping ratio value.

Frequency Spacing of SRS Bins

Usually the SRS spectrum consists of multiple bins distributed evenly in the logarithmic frequency scale. The frequency distribution can be defined by two numbers: a reference frequency and the fractional octave number, such as 1/1, 1/3 or 1/6.

An octave is a doubling of frequency. For example, frequencies of 250 Hz and 500 Hz are one octave apart, as are frequencies of 1 kHz and 2 kHz.

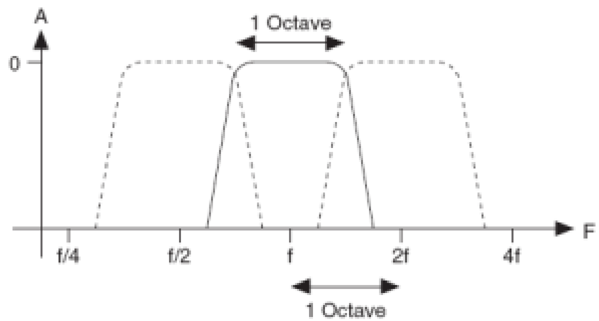


Figure 104. Full octave filter shape.

The proportional bandwidth property will divide the frequency information uniformly over a log scale. It is very useful in analyzing a variety of natural systems. For example, the human response to noise and vibration is very non-linear and many mechanical systems have a behavior that is best characterized by proportional bandwidth analysis.

To gain finer frequency resolution, the frequency range can be divided into proportional bandwidths that are a fraction of an octave. For example, with 1/3 octave spacing, there are 3 SDOF filters per octave. In general, for 1/N fraction octave, there are N band pass filters per octave such that:

$$f_{c\,j+1} = f_{c\,j} * 2^{1/N}$$

where 1/N is called the fractional octave number. f_r, the ***reference frequency***, is simply any of the frequencies $f_{c\,j}$. All other center frequencies of SDOF filters reference to this frequency. When the reference frequency and the fractional octave number are fixed, the frequency distribution over the whole frequency range is determined.

SRS Measurement Quantities

Measurement quantities available to the CoCo SRS test are: time stream of each channel (raw data), block captured time signals and three SRS of each channel.

Time streams: this is the same as any other applications on the CoCo. Time streams are always available for viewing and recording. It is a very useful tool to observe whether the input signals are in the valid range. The recorded sine wave can be used for further post-processing. In CoCo, the time streams are often denoted as ch1, ch2 etc.

Block time signals: These are the block captured signals that are used for SRS analysis. Acquisition Mode will control how the block time signals are acquired.

SRS: Shock Response Spectra will be calculated for each block time signals. The engineering unit of the spectrum is determined by the sensor used by the input channel. In CoCo, the spectra are often denoted as three types: Maximum Positive spectrum; Maximum Positive spectrum and Maximum-Maximum spectrum.

Maximum Positive Spectrum: This is the largest positive response due to the transient input, without reference to the duration of the input.

Maximum Negative Spectrum: This is the largest negative response due to the transient input, without reference to the duration of the input.

Maximax Spectrum: this is the enveloge of the absolute values of the positive and negative spectra. It is the most often used SRS type. The log-log Maximax is the universally accepted format for SRS presentation.

Other common SRS measures include the so called Primiary SRS, Residue SRS and Composite SRS. The CoCo only calculates the Composite SRS.

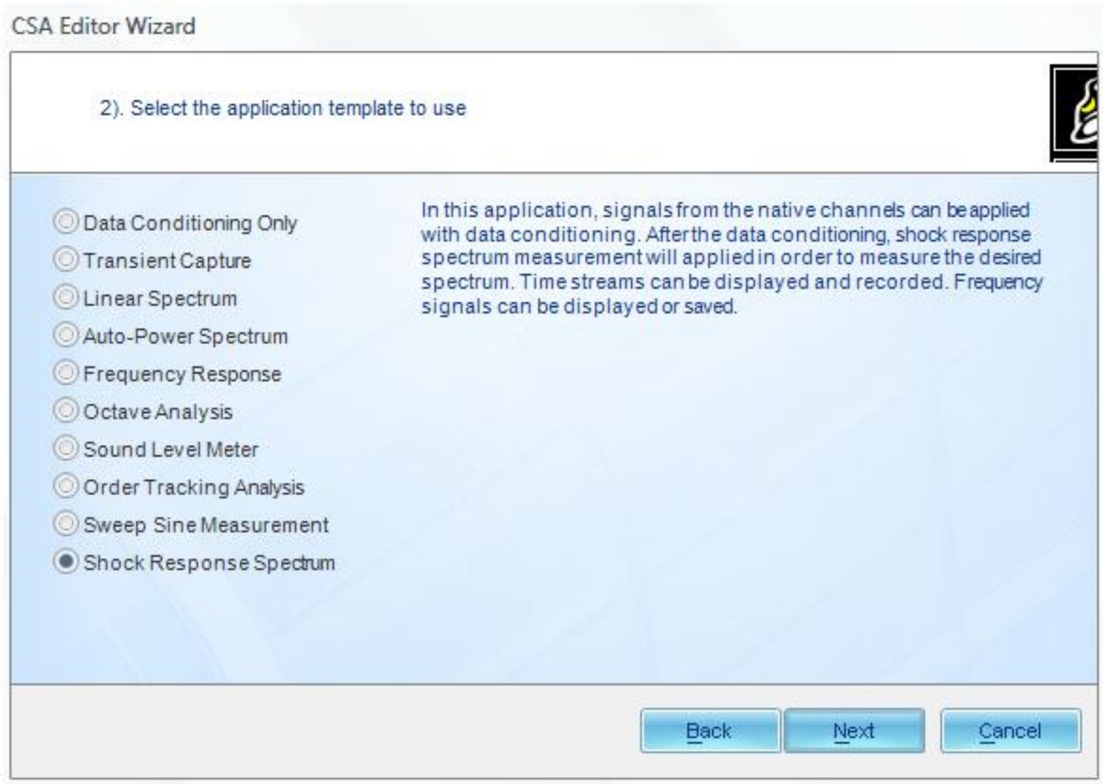
SRS CSA Editor Operation

This section describes the operation of CSA Editor related to SRS analysis. For general operation of CSA Editor, refer to the CSA User's Manual.

CSA Editor Wizard

This section summarizes how to create a CSA project for SRS analysis in the CSA Editor. We strongly recommend that you read the CSA Editor User's Manual to gain more detail information before proceeding with this chapter.

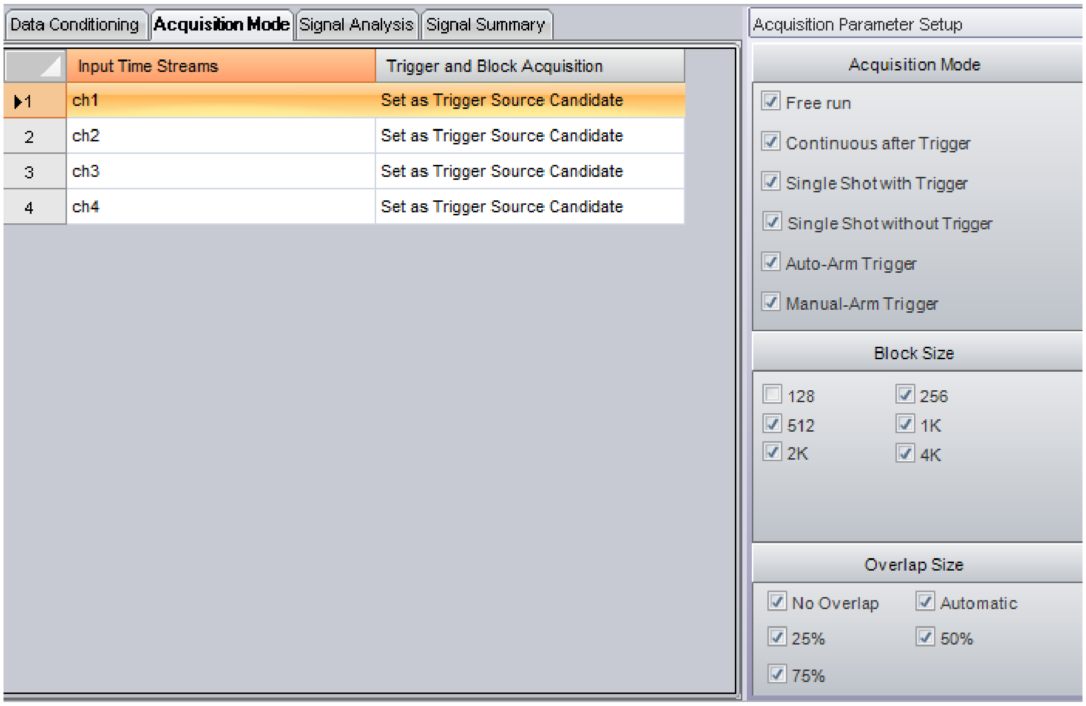
To start, click on the CSA Editor icon in the upper-right corner in EDM and start the CSA Editor. The CSA Editor Wizard dialog box will be displayed. Select the number of input channels on your CoCo and then click next. Then select Shock Response Spectrum from the application template list and click Next.



■ Figure 105. Application template list for SRS.

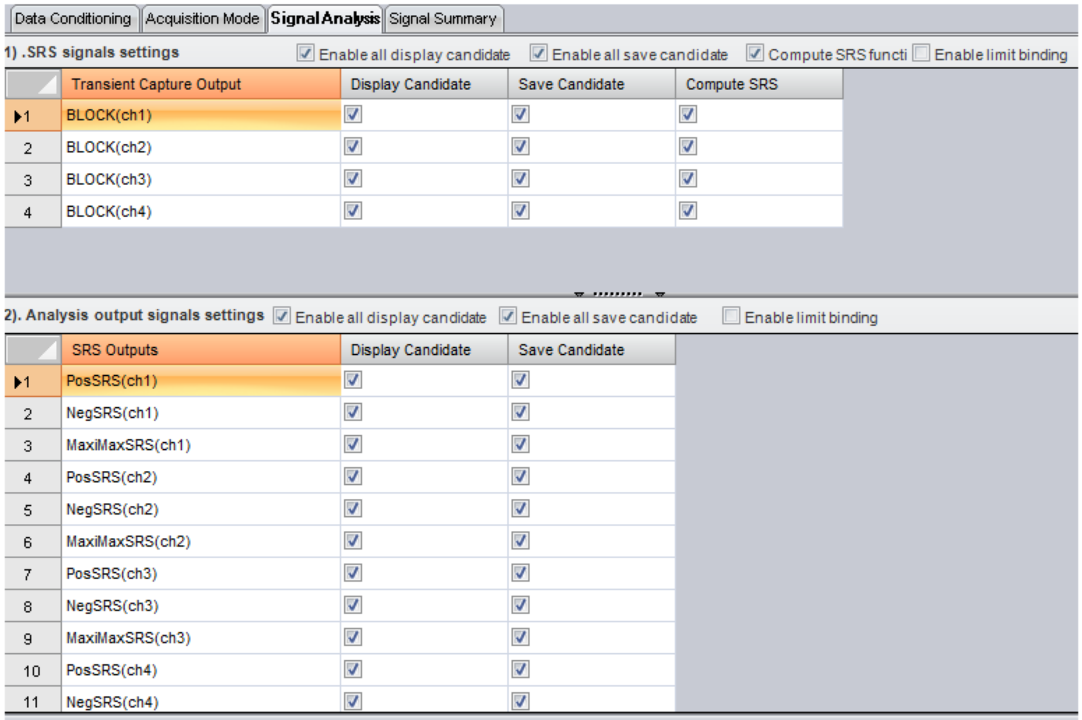
Next enable the minimum number of input channels required for your measurement and click Next. Enter the CSA description information and click Finish.

SRS is typically used to process transient captured signals. A trigger is normally required. Click on the Acquisition Mode tab to modify the trigger settings.



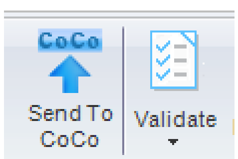
- Figure 106 Enable Acquisition Modes

Click on the Signal Analysis tab to enable the display, save and compute options. The Analysis output settings at the bottom of the screen include the signals that are unique to SRS.



■ Figure 107 Create Signals to be displayed and saved

After the CSA has been configured it must be validated and uploaded to the CoCo device before it can be used.



■ Figure 108 Validate and then Send to CoCo

SRS CoCo Operation

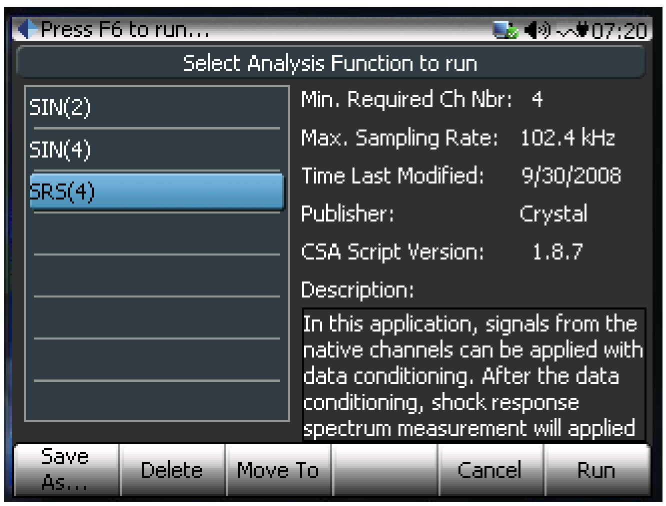
This section describes the operations of CoCo that are specifically related to SRS analysis. For general operations of CoCo, refer to previous Chapters of this manual.

Select an SRS CSA Project

To run a SRS CSA press the Analysis button and select Swept Sine and SRS application group then press the F6 OK Button.

■ Figure 109. SRS CSAs are saved in the Swept Sine and SRS application group.

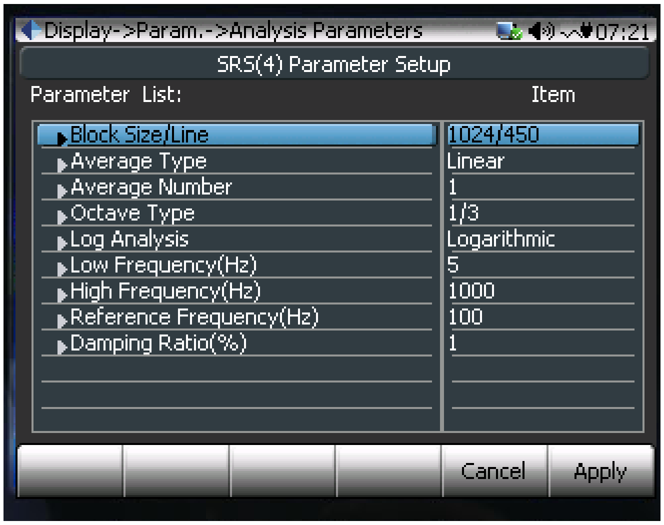
Then use the up and down arrow buttons to highlight a SRS CSA and press the F6 Run Button to load it.



■ Figure 110. Press the F6 Run button to load the SRS CSA.

SRS Analysis Parameters

Analysis parameters that are unique to SRS CSAs can be modified with Analysis Parameters from the Param menu.



■ Figure 111. SRS Analysis parameters screen.

Block Size: is the size of the time block buffer. The time block signals will be used as input signals of SRS analysis.

Average Type: Average can be applied to SRS spectrum. But in most cases no average is applied. Setting the Averaging Number as 1 will result in no averaging.

Octave Type: defines the fractional octave number which defines the spectral resolution.

Low Frequency: defines the lowest frequency boundary of the SRS spectrum.

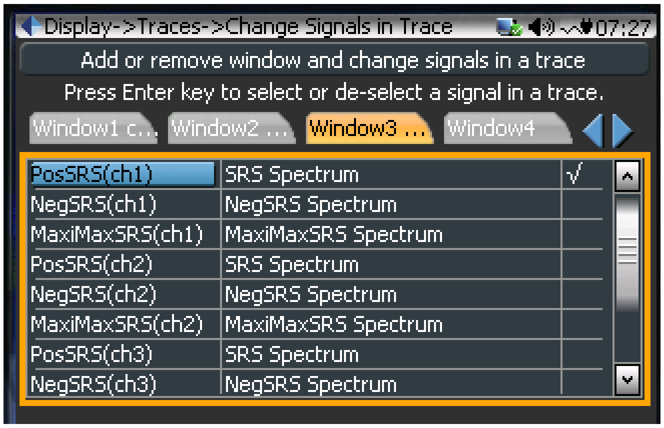
High Frequency: defines the highest frequency boundary of the SRS spectrum.

Reference Frequency: defines the reference frequency the SRS spectrum.

Damping Ratio (%): defines the percentage of the damping factor.

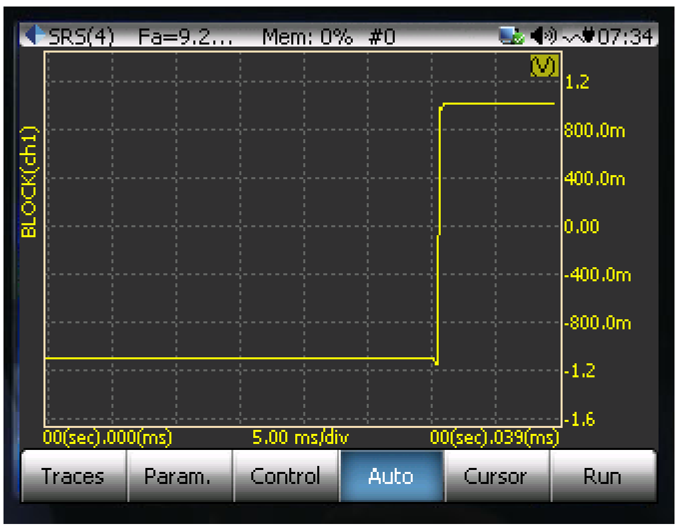
SRS Signal Display

Signals that are unique to SRS analysis can be viewed in the same way as any other signal.



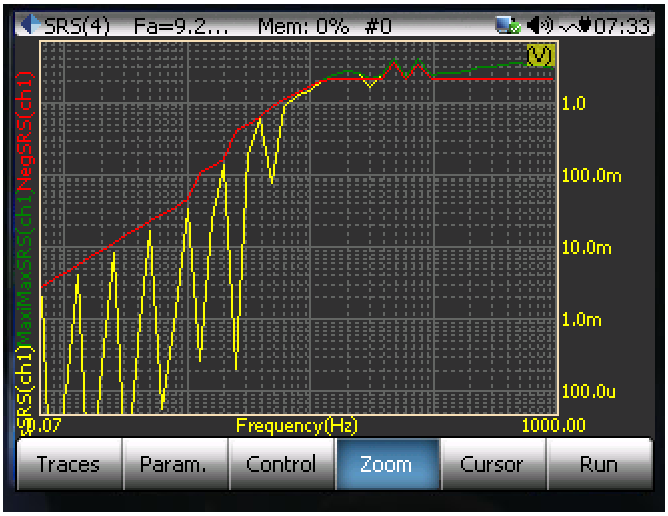
■ Figure 112 Select the signals for display

Figure 113shows a transient signal captured and displayed as a block signal.



■ Figure 113 Block signal that SRS to be applied

Figure 114 shows the SRS, Maximax and NegSRS signals overlaid in a single window.



■ Figure 114 Three overlaid SRS Signals

7. AUTOMATED TEST AND LIMIT CHECK

Automated limit testing allows engineers and technicians to set up a pass/fail measurement on Any measured signal. This feature automates the process of determining whether an acquired signal meets, or is within a given set of criteria.

A limit test typically consists of comparing a waveform to upper and lower boundaries which the measured waveform must not cross. These boundaries are typically defined by the user to specify a tolerance band around a waveform. If any part of the waveform falls outside the limit, the software returns a failure message and the location of the failure on the waveform.

Application Examples

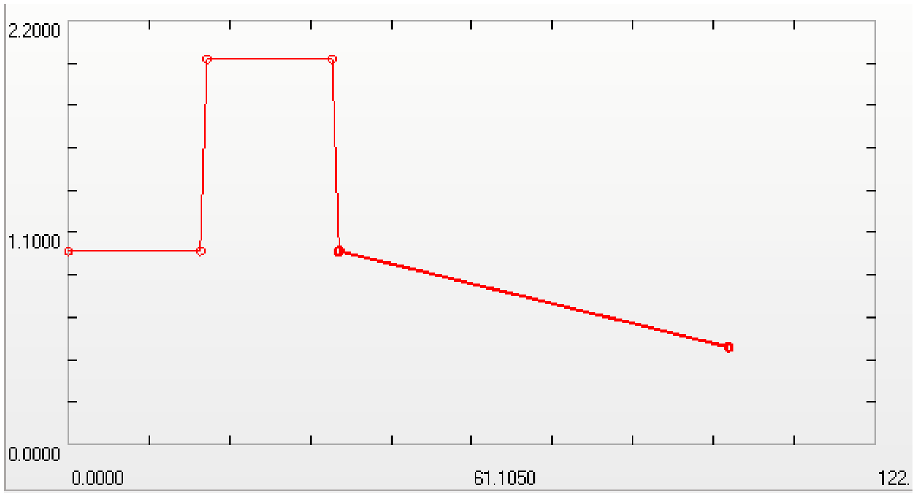
A common example for automated testing is related to structural testing. When excited, a structure will resonate at its natural frequencies. Structures can be excited through impact or by other means. Structural defects can result in a shift in resonant peaks. Therefore, in structural tests, frequency 'alarms' are used to monitor the frequency response in areas of spectral interest.

Another example of automated testing is related to rotating machinery. Rotating or moving assemblies produce vibration and noise patterns that can be examined to identify the fingerprint of 'good quality.' Product defects will cause additional spectral peaks, or changes in peak levels. Therefore, in 'self-excited' product testing, 'level alarms', which can be set to trigger on peak or RMS values, are placed around the areas of spectral interest. Unwanted signals, from background noise, are therefore ignored.

Testing Limit Signals and Testing Schedule

Automated testing can be performed on a wide variety of signals including a time domain capture, an auto-power spectrum, an octave spectrum, an order track signal or a frequency response. The CoCo instrument compares the limits to the live measured signal in real-time, after every single frame of measurement. If the limits are exceeded, the CoCo takes the appropriate actions based on the user setup.

An upper and or lower limit can be applied to a signal to be tested. Limit signals are constructed by defining breakpoints. A breakpoint is controlled by a pair of X/Y values. Figure 115 shows a typical automated test limit signal with 4 breakpoints.



■ Figure 115. Typical automated test limit signal with 4 breakpoints.

To automatically control the limit checking test, a ***testing schedule*** is developed for CoCo. The testing schedule defines the various operations to automate the process. For example the testing schedule can tell the instrument when the limit checking will be turned on, when it will be turned off and for how long the test will be conducted.

To record the events of the test a Testing Log and a summary report are needed. The Testing Log records the important events, including whether the limits are exceeded, in chronicle order. The Summary Report provides the status of limiting check since the last time when the test was started.

When the limit signal is exceeded then a user defined limit alarm event will be triggered. This can include an audible beep, Save Signals, send messages and so on.

To summarize, an automated limit checking test requires the following building blocks:

- At least one test signal
- At least one limit signal applied to the measured signals
- A testing schedule
- A testing log and summary report
- A setup for the limit alarm events

Figure 116 illustrates the automated testing process

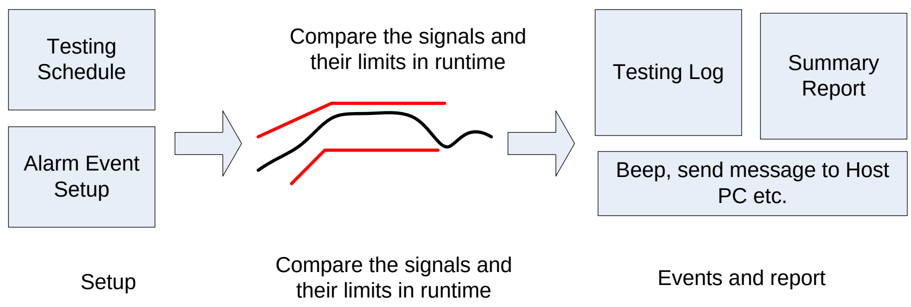


Figure 116. Illustration of automatic testing process.

Test Signals: Any block signal can be used for testing. Typically the test signals are time captured blocks, auto power spectra, frequency response, octave spectra or order tracks. Time streams are not used for limiting test. **Limit signals**: Limit signals including upper and/or lower limits are defined in the CSA Editor. Limit signals are applied to testing signals. Up to a maximum of 64 segments can be defined for each limit signal. The maximum number of limit signals is 64.

Testing Schedule: The testing schedule automatically controls the test using an event driven process. Multiple testing schedules can be developed and one is executed at a time. Testing schedule event entries include: Loop/End-Loop, Set Sampling Rate, Set All Input Mode, Run Duration, Hold, Limit Check on, Limit Check off, Start Recording, Stop Recording, Save Signals, Turn Signal Source On and Turn Signal Source Off.

Testing Log: A log file is automatically created for each run of the schedule to record major events.

Limit Check Alarm Event Setup: Events include an audible beep from the CoCo, CoCo screen flashing, entry into the Testing Log, send message to host PC via EDM software and Save Signals.

When a limit is exceeded, the predefined events are triggered. For example, the CoCo may beep, flash the screen, save the signals to the storage device, or send the message to the host PC.

Networked CoCo used for Automated Test

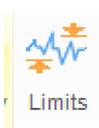
CoCo has an Ethernet network interface that provides a unique advantage that multiple CoCo units can be connected remotely using an Ethernet network. This is particularly useful for vibration monitoring or production test that requires long distance access.

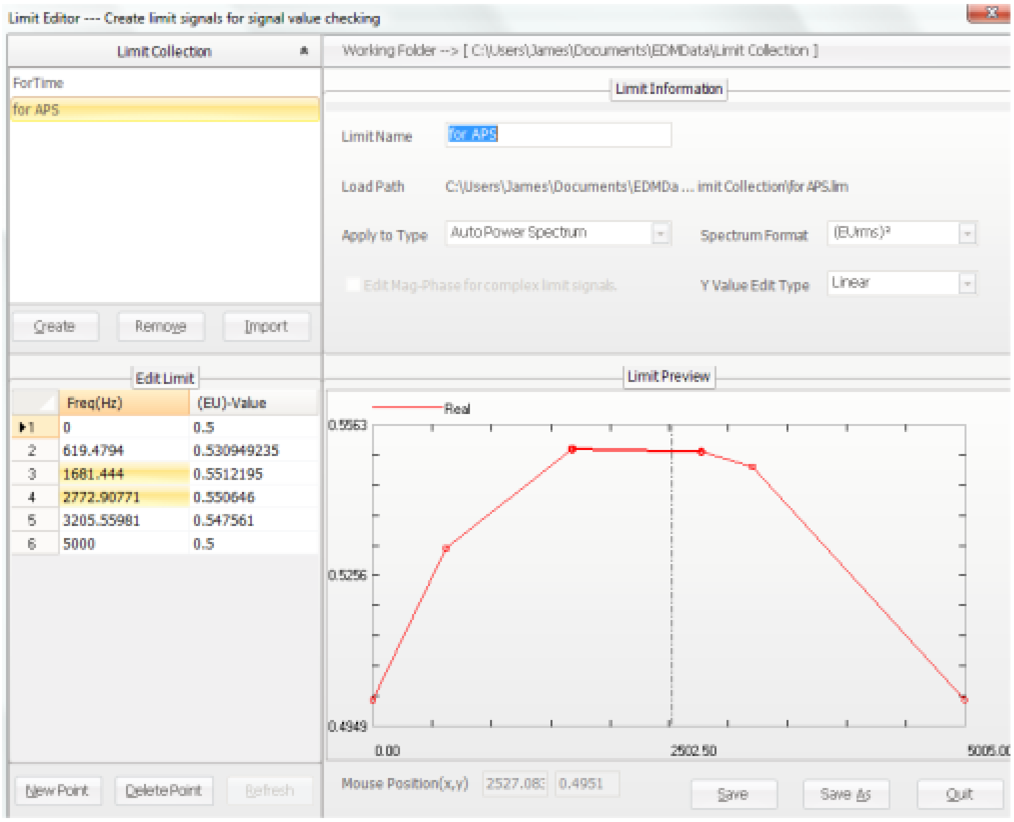
■ Figure 117. Remote operation for multiple CoCo units with automated testing.

When multiple CoCo units are managed through the EDM software, the host computer can record or react to the alarm events from each remote CoCo unit.

CSA Editor Operation

Limit signals are defined on the host computer with the CSA Editor software. After the limit signals are defined, they are kept in a Limit Collection pool. Next, you define which limit signal is compared which measured signal. To define limit signals, complete the CSA Wizard process then click on the Limits icon in the CSA Editor.

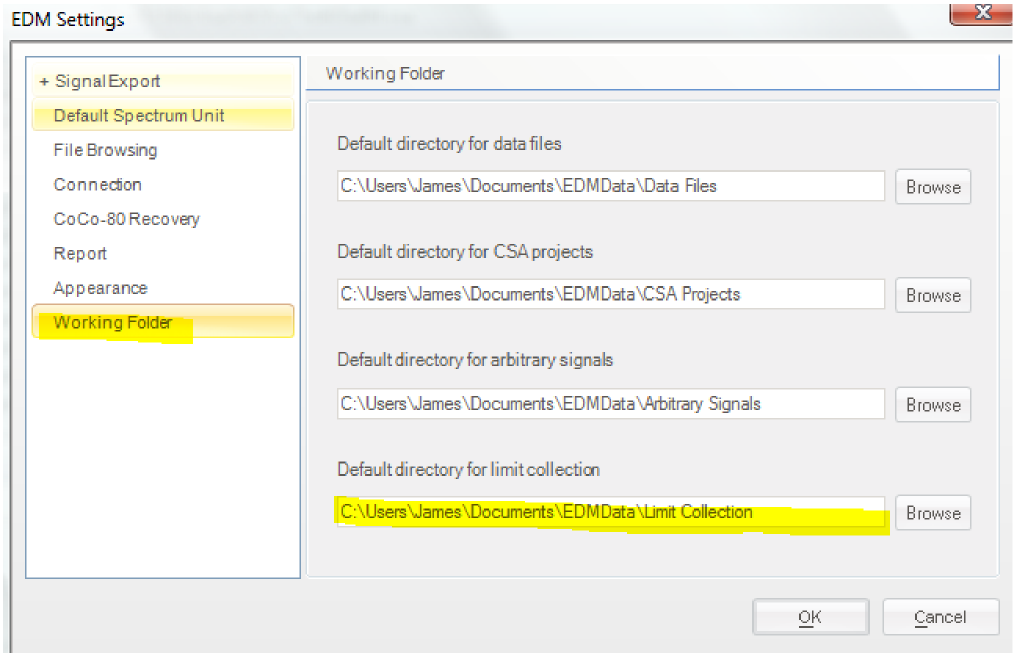




■ Figure 118. Limit Editor is used to define a limit signal on the host computer.

Figure 118 shows the Limit Editor dialog box which is divided into four areas. The ***Limiting Collection*** area holds all the limit signals that you create. Later you will bind one or two limit signals to any signal that you want to test on CoCo.

The ***Limit Information*** area defines the name of the limit signal, the type of the signal and other attributes. For example if you want to apply the limits to an auto power spectrum, you must select auto spectrum type under the *Apply to Type* field, and select the appropriate spectrum type. Spectrum type selection is critical for any auto spectrum test. The default directory indicates where these limit signals are saved on the host computer. The path is defined in the *Settings* of the Home tab, under the Working Folder item as shown in Figure 119.

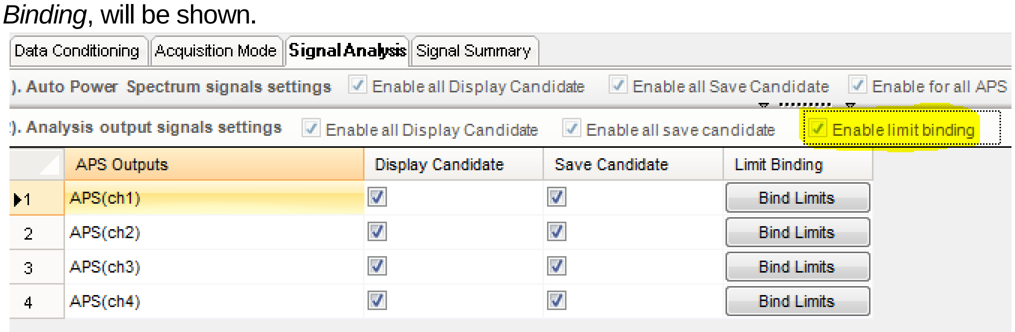


■ Figure 119. Limit signals are saved in the location defined under EDM Settings.

The ***Edit Limit*** and ***Limit Preview*** areas are for editing the limit signal. A limit signal consists of multiple break points. With the table, you can manually add or remove the breakpoints. Alternatively you can use the mouse to drag and draw breakpoints.

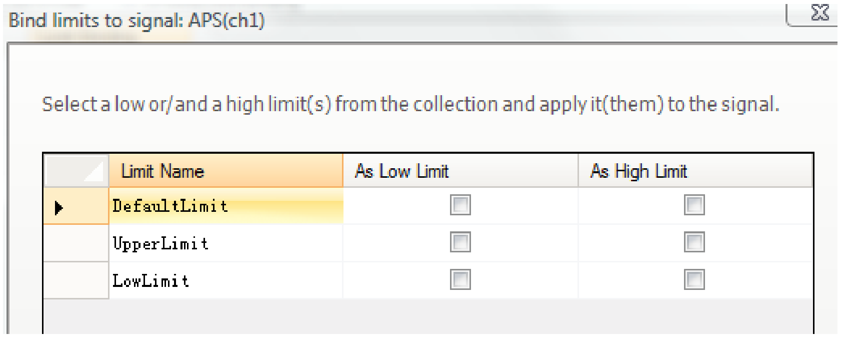
After the limit definition is done, use the Save button to save the limit signals on the host computer. The Save As button lets you save the limit signal into different folder or with a different name. The Quit button exits the dialog box.

The next step is to apply a limit signal to a measured signal. This process is referred to as binding. Click on the Signal Analysis tab and check the *Enable limit binding box*. A new column, *Limit Binding*, will be shown.



■ Figure 120. Bind the limit signal from the Signal Analysis tab.

Click on the Bind Limits button next to each APS Output signal. This will display the following dialog box.



■ Figure 121. Define the limit signal for a specific measured signal.

To apply a limit signal to the measured signal, check either the Low and/or High limit box next to the desired limit name. Then click OK.

After the binding process, you should validate the CSA project and send it to CoCo for run-time execution. The limit signals will be sent together with the CSA project file.

CoCo-80 Operation for Limit Test

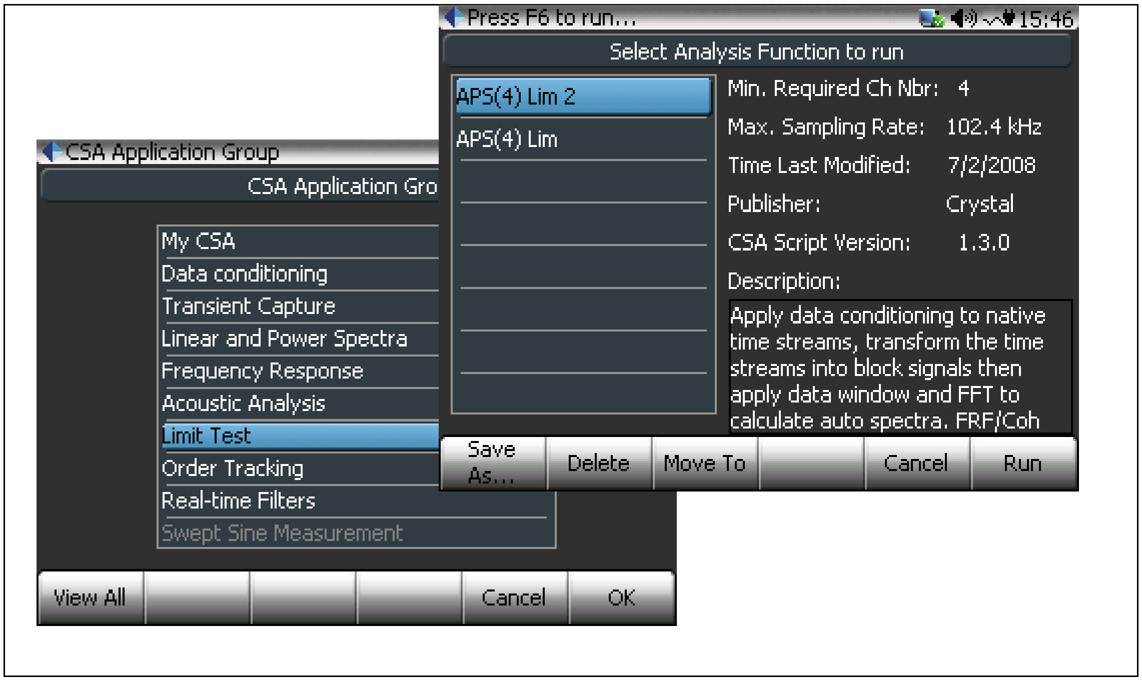
This section describes how to run a limit test on the CoCo hardware. To run an automated limit test, you need to do the following on the CoCo:

1. Load a CSA with a limit signal
2. Make a testing schedule and enable the limit test in the testing schedule
3. Set up the limit alarm actions
4. Create windows, and choose the appropriate limit signals for display
5. Activate the testing schedule
6. After the test is done, view the report

Select a CSA Project

After a CSA with a limit signals is downloaded from the host PC to the CoCo, the CSA script will be available under the Limit Test Application Group. To open a CSA press the Analysis Button, then select the Limit Test Application Group. All Limit Test CSA files are automatically placed in this group to help organize the CSA file on the CoCo.

Next select a CSA file from the list.



- Figure 122. Select a CSA file with a Limit Signal from the Limit Test Application group.

Make a Testing Schedule and Enable the Limit Test

The next step is to make a testing schedule and enable the limit test. First select Schedule Setup from the Param. Button.

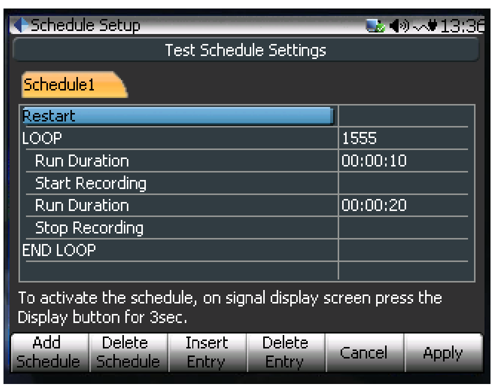
- Figure 123. Schedule Setup window.

Next insert schedule events using the Insert Entry Button. Events can also be deleted from the schedule.

■ Figure 124. Test Schedule Settings window.

The testing schedule automatically controls the test duration and automates the testing process. Multiple testing schedules can be saved but only on can be executed at a time. The testing schedule event entries include: Loop/End-Loop, Run Duration, Hold, Limit Check on, Limit Check off, Start Recording, Stop Recording, Save Signals, Turn Signal Source On and Turn Signal Source Off.

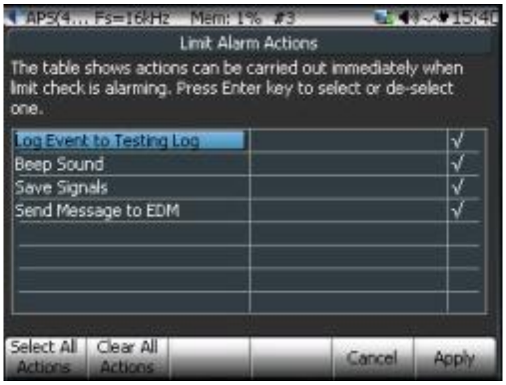
The Loop/End Loop function is useful for automating a repetitive function. For example, the following schedule will loop 1555 times. In each loop, the system will pause for 10 seconds and then record data for data for 20 seconds then stop the recording and pause again.



■ Figure 125. A loop can be used to repeat a sequence of events.

Set up Limit Alarm Actions

Limit alarm actions determine the event that occurs when a measured signal exceeds the limit signal. To enable actions select Limit Alarm Setup from the Param. Button.



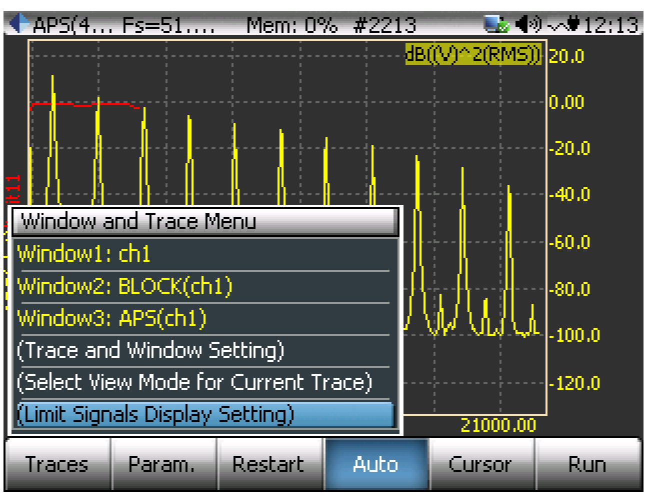
■ Figure 126. Limit Alarm Actions window.

The Limit Alarm Actions display is only available for a CSA that includes at least one limiting check function. The CoCo will trigger the events that are selected including:

Log Event to Testing Log: log limit exceeded information to a text file.
Beep Sound: generate a beep sound from the CoCo unit.
Save Signals: save the frame into a file
Send Message to EDM: send a message contains limit exceeded information to connected host computer with EDM software. The EDM software then can be setup to make a beep sound or save the message to a file.

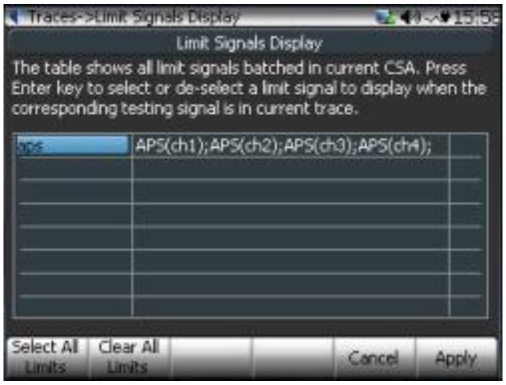
Display Limit Signals

A common display for limit testing includes the measured signals on the same window as the limit signal. This display can be created by selecting Limit Signals Display Setting from the Traces Button. This setup item will only be available for a CSA that includes at least one limiting check function.



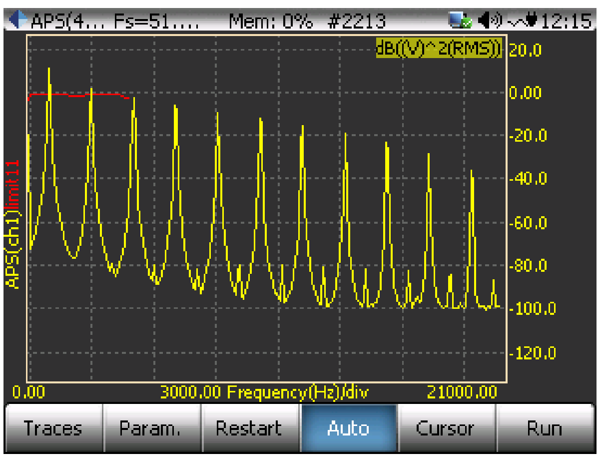
■ Figure 127. Limit signals can be displayed with live data.

Next select the limit signal from the Limit Signal Display window. When you select a limit signal then it is automatically added to any window that contains the associated measured signal.



■ Figure 128. The Limit Signal Display dialog lets you choose which limit signals to display.

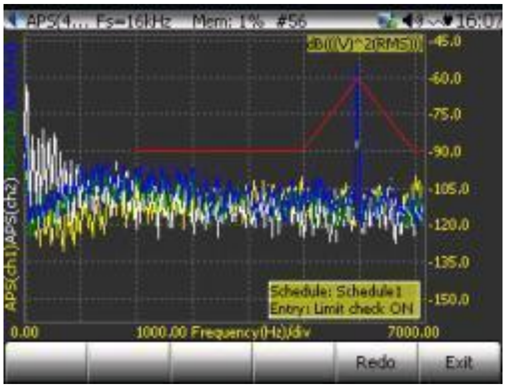
Figure 129 shows a typical limit signal in red on the same trace as a measured signal in yellow. Note that the measured signal exceeds the limit signal in this case because the first and second harmonic peaks are too large.



■ Figure 129. Limit signal on the same trace as a measured signal.

Activate the Testing Schedule

After the limit test is set up, the next step is to activate the test. When the signal display widow is shown as below, press the Display button for three seconds and release the button, the testing schedule will be activated. Once the testing schedule is activated, all the function buttons will be disabled. You can not switch to other screens during the test schedule.

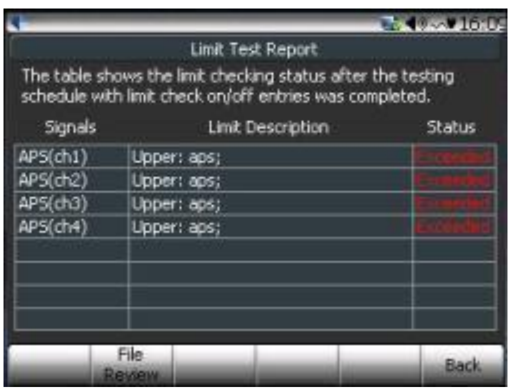


■ Figure 130. Typical display with limit signal during a test schedule.

During a test schedule you can stop the test by pressing the Exit Button. After the test is complete you can start it again manually by pressing the Redo Button.

View the Limit Report

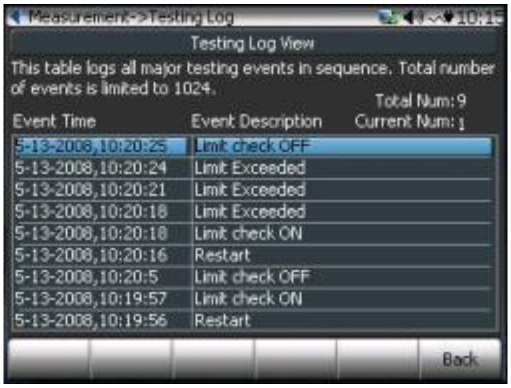
The Limit Report shows the status of each limit signal. When the limit test finished, Press F1 to show the limit report.



■ Figure 131. Limit Report.

View the Testing Log

The Test Log records every event during the test schedule.

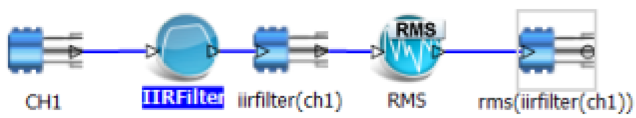


- Figure 132. Testing Log View.

8. REAL TIME DIGITAL FILTERS

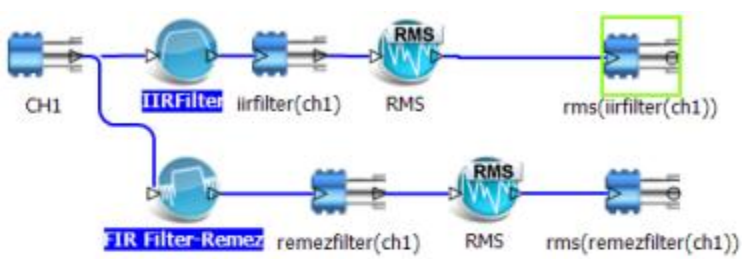
Real Time Digital Filters is a powerful analysis tool that can be used to filter a measured signal in real time and then apply the FFT and time based analysis built into the CoCo. You can precisely define the filter characteristics to meet your specific application. The filter definition is performed in the EDM software and the filter is included in the CSA file that is downloaded to the CoCo. This capability in a small portable unit makes the CoCo a very powerful analysis tool.

For example, a user might want to look at the energy distribution over time, for a specific band of frequencies instead of the entire frequency spectrum from zero to the maximum sampling rate. This can be done by creating a band-pass filter then applying an RMS estimator to the output of the filter. Figure 133 shows the graphical representation of this process which is used to define the real time filter in the EDM software. The icon on the left, CH1 represents the native measured time stream. It is connected to an IIR Filter which computes a signal named iirfilter (ch1) which is connected to an RMS estimator. The output of the RMS estimator is a signal named rms(iirfilter(ch1)). The EDM software will be discussed in more detail later.



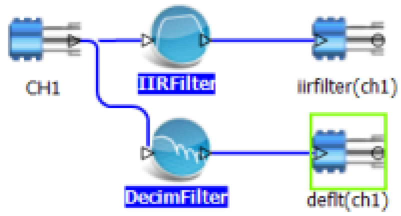
■ Figure 133. Example real time digital filter application.

Another example is that the user might want to look at the frequency energy over 100Hz to 200Hz and 1000Hz to 2000Hz separately. This can be done by deriving two output streams from the native channel 1, then applying the band-pass filter to each path as shown in Figure 134.



■ Figure 134. Digital Real Time Filter example with two output streams.

In another example, a user might want to look at the very fast time characteristics of a channel at high frequency, and the same channel at a very low sampling rate. This can be done by applying a decimation filter to the native time stream as shown in Figure 135 . The native channel time stream is split into two streams so the signal from the same channel is recorded at both high and lower sampling rates.



■ Figure 135. Example computing high and low sampling rate with a decimation filter.

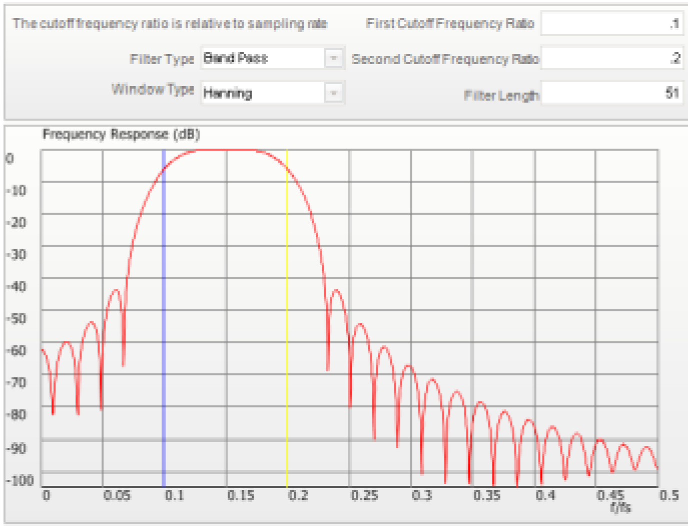
The Real Time Digital Filters option includes three types of digital filters: FIR, IIR and decimation filters. For FIR and IIR filter, you can specify low-pass, high-pass, band-pass or band-stop types with several different methods. This chapter first explains the theory about the filter design, and then introduces the operations within the CSA Editor and CoCo hardware.

Real Time Filter implementation can be divided into two steps that include the filter definition on the EDM software, and secondly download and run the CSA on the CoCo hardware.
You design a filter based on certain criteria such as cut-off frequency, pass band ripple, attenuation level and so on. The EDM software walks you through this process. The outcome of this design process is simply a number of filter coefficients that represent the filter which are included in a CSA. The software will upload the CSA including the filter coefficients to the CoCo hardware.

After filter is defined and the coefficients downloaded to the CoCo hardware you can run the CSA. When the CoCo is running, the filter coefficients created in the filter design process will be used. The time streams will pass through the filters and generate new time stream signals.

The goal of filter design is to calculate a series of filter coefficients, also known as “taps” based on the user specified criteria. The criteria are often described by following variables:

- **Number of filter coefficients**: this is also known as the order of the filter. The filter order defines how many coefficients are required to define the filter. A lower order filter consists of a fewer number of coefficients. A low order filter responds relatively faster than a higher order filter, that is there is less of a time lag in the output of the filter.
- **Cutoff frequencies**: For low pass or high pass filters, only one cutoff frequency is needed. Band pass or band stop filters require two cutoff frequencies to fully define the filter shape. Figure 136 shows a typical band pass filter design with the two cutoff frequencies set to approximately 0.1 and 0.2 Hz as indicated by the blue and yellow vertical lines.
- **Attenuation of stop band in dB**: This defines how much of the input signal is cut out of the output at the rejected frequencies. In theory the higher the attenuation the better. In Figure 136 the stop ban attenuation is > 40 dB as seen from the highest side lobe just below 0.25 Hz.
- **Pass band ripple**: Ripple is an unavoidable characteristic if a digital filter. It refers to the fluctuation in the filter shape at transition frequencies. If a very flat filter is required then it can be specified by choosing a very low ripple. In Figure 136 ripple is seen in the stop band and no ripple is evident in the pass band. Ideally the pass band should be very flat and some ripple is tolerable in the stop band.
- **Width of transition bands**: This refers to the filter shape between a band pass and a band stop region. Ideally this transition band should be very small. However, a very narrow transitional band requires a higher order filter which affects the filter response time and can also affect ripple. In Figure 136 the transition bands are between 0.05 to 0.1 and 0.2 to 0.25.



■ Figure 136. Filter design shows cutoff frequencies, ripple, band stop attenuation.

In most cases filter design includes making tradeoffs between minimizing the filter order, ripple, transition band width and response time. Not all can be satisfied at the same time. Filter design can be an iterative process and experience is helpful.

FIR Real Time Digital Filters

Finite Impulse Response (FIR) filters have the distinctive trait that their impulse response lasts for a finite duration of time as opposed to, an Infinite Impulse Response (IIR) filters whose impulse response is infinite in duration. This trait is due to the fact that there are no feedback paths in the FIR filter. FIR filters offer several advantages over IIR filters:

- Completely constant group delay throughout the frequency spectrum. Group delay refers to the time delay between when a signal goes into the filter and when it comes out. Constant group delay means that an input signal will come out of the filter with all parts delayed the same amount with no distortion.
- Complete stability at all frequencies regardless of the size of the filter.

FIR filters also have some disadvantages as well:

- The frequency response is not as easily defined as it is with IIR filters
- The number of coefficients required to meet a frequency specification may be far larger than that required for IIR filters.

CoCo allows up to 128 taps (orders) for the real-time FIR filter.

A digital filter can be understood by considering the difference equation which defines how the input signal is related to the output signal as

$$y[n] = b_0 x[n] + b_1 x[n-1] + \cdots + b_N x[n-N]$$

where $x[n]$ is the current input signal sample, $x[n-1]$ is the previous signal sample and $x[n-N]$ is the last sample in the series. The series multiplies the most recent N+1 samples with associated the N+1 filter coefficients. , $y[n]$ is the current output signal and b_i are the filter coefficients. The number N is known as the *filter order*; an N^{th}-order filter has $(N + 1)$ terms on the right-hand side and N+1 filter coefficients also referred to as "*taps"*.

This equation illustrates why a higher order filter has a slower response time. It takes more samples and therefore more time for an event to work its way though the series until the output is no longer affected by the event as compared to a lower order filter with fewer coefficients.

The previous equation can also be expressed as a convolution of the filter coefficients and the input signal.

$$y[n] = \sum_{i=0}^{N} b_i x[n-i]$$

The Impulse Response of the filter shows how the historical data affect the current filtered value. The longer the impulse response, the farther the old data will affect the current filtered value. To find the impulse response we set

$$x[n] = \delta[n]$$

where $\delta[n]$ is the Kronecker delta impulse. The equation below shows that the impulse response for an FIR filter is simply the set of coefficients b_n, as follows

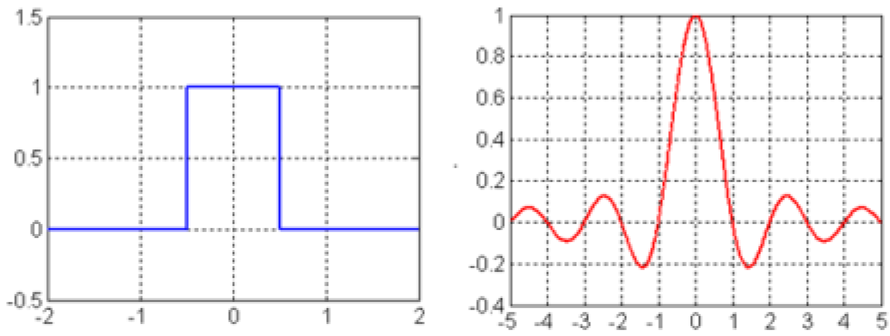
$$h[n] = \sum_{i=0}^{N} b_i \delta[n-i] = b_n \qquad for\ n = 0\ to\ N$$

FIR filters are clearly stable, since the output is a sum of a finite number of finite multiples of the input values, so can be no greater than $\sum_{n=0}^{N} |b_n|$ times the largest value appearing in the input.

Data Windows FIR Filter Design

In the academic world, hundreds of methods are available to design an FIR filter to meet various criteria. The EDM includes the most popular filter design methods: Data Window and Remez. Both methods are discussed below.

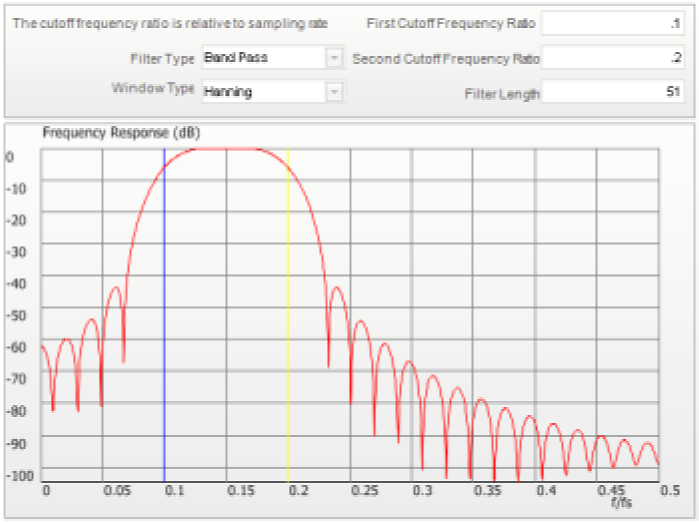
The Data Window FIR Filter Design method is the easiest to understand. The name "Window" comes from the fact that these filters are created by scaling a *sinc* (SIN(X)/X) function with a window such as a Hanning, Flat Top, etc. to produce the desired frequency effect.



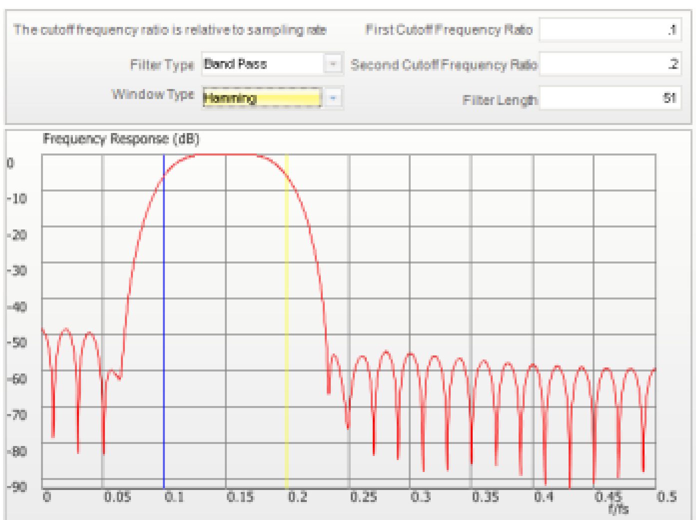
■ Figure 137 Sinc function is the Fourier transform of a square shape.

A data window FIR filter is generated by starting with an ideal "brick-wall" shaped filter, that is a filter with vertical edges or zero transition band width as shown on the left in Figure 137. The brick-wall filter is specified by the cutoff frequencies and has a band pass amplitude of 1 and a stop band amplitude of zero. The problem with the ideal brick-wall filter is that the time response oscillates forever and it requires an infinite number of filter coefficients. This ideal filter can be modified by applying a data window to force the time response to decay in a finite time. Of course this degrades the shape of the ideal brick-wall filter performance. It introduces ripple, increases the transition band width and increases the stop band attenuation. However it allows the filter to be defined by a finite number of filter coefficients. The filter performance can be modified by using different data windowing functions and making the tradeoff between filter order and response time. The user must choose these settings during the filter design.

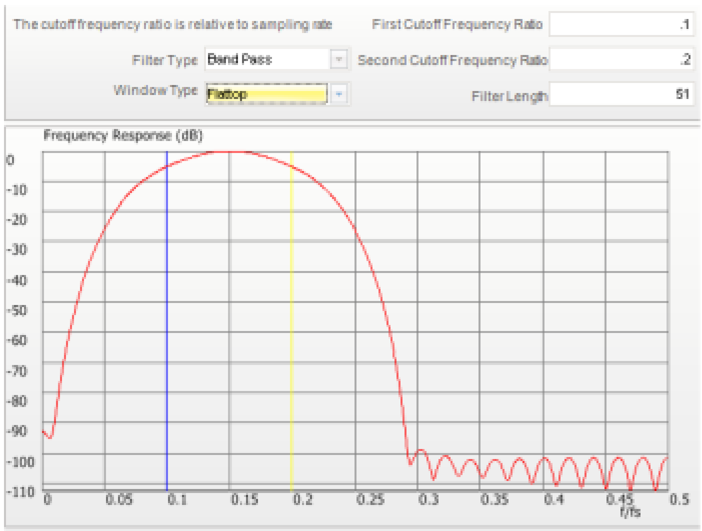
Figure 138 through Figure 143 show a comparison of different data window choices for the same filter settings. In all cases the low and high cutoff frequencies are 0.1 and 0.2 relative to the sampling frequency. The number of filter taps is 51.



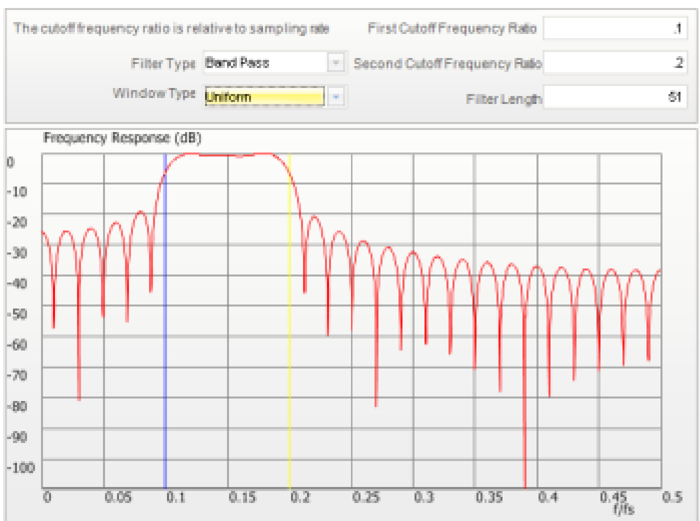
■ Figure 138 Hanning window method.



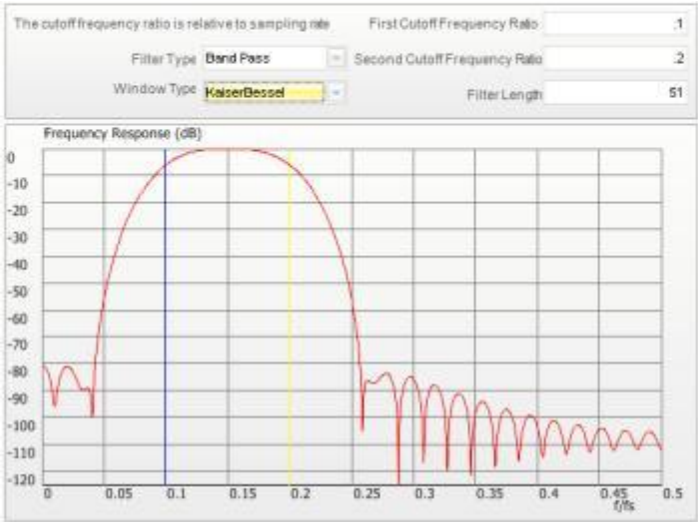
■ Figure 139 Hamming window method.



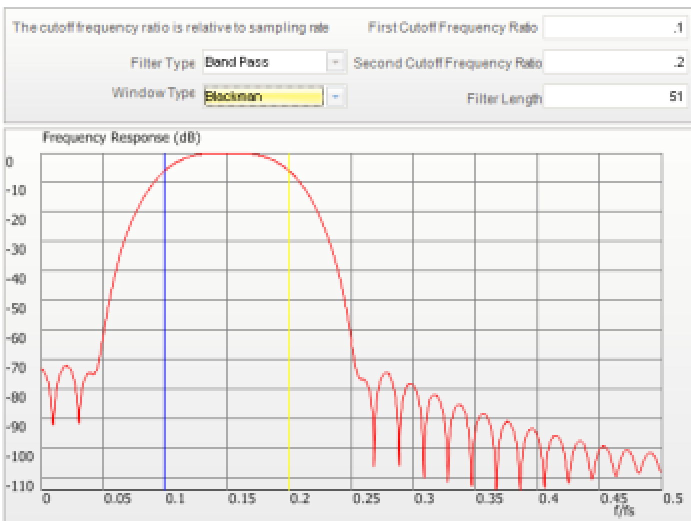
- Figure 140 Flattop window method.



- Figure 141 Uniform window method.



■ Figure 142 Kaiser Bessel window method.



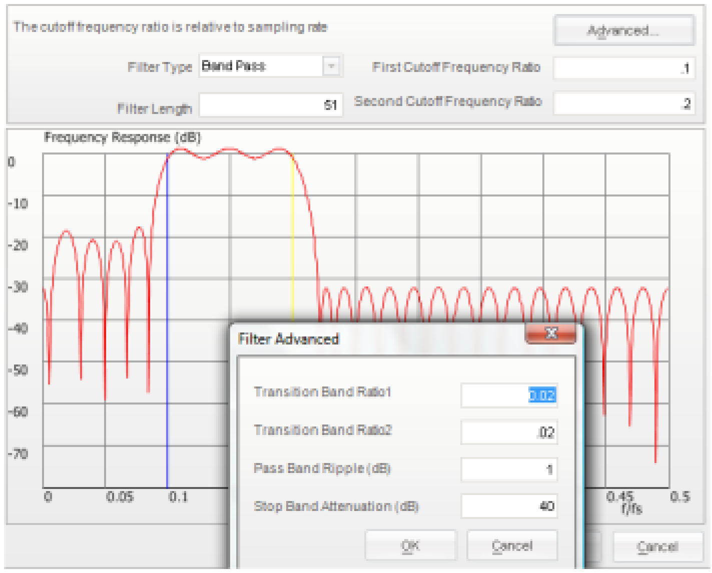
■ Figure 143 Blackman window method.

As shown in the pictures, different window methods produce different filter performance, i.e., different attenuation of the main lobe and side lobes. The best data window choice depends on your specific application. Refer to the Basic Spectral Analysis section for a comparison of windowing functions.

Remez Filter Design

The Remez Filter is a different method for designing an FIR filter. It is more computationally intensive than the data window method. A Remez filter is generated with iterative error-reducing algorithms designed to reduce the pass band error. In addition to allowing stop band ratio and frequency definition, the Remez filter allows the "Ripple Ratio" to be defined as a user specified parameter.

Figure 144 shows an example of a filter design using the Remez method in the EDM software. The low and high cutoff frequencies are 0.1 and 0.2 relative to the sampling frequency. The number of filter taps is 51.



■ Figure 144. Ramez FIR Filter design dialog.

The software is intelligent enough to automatically calculate the total FIR filter length based on these criteria. For example if the user asks for very high attention, very small ripple or very sharp transition band, the filter length will go very high. The user must make tradeoffs between these parameters so that appropriate filter length can be generated and used.

IIR Real Time Digital Filters

Infinite impulse response (IIR) filters have the trait that their impulse response decays very slowly but theoretically lasts forever. This is due to the fact that the filter input includes the measured signal and also the filter output creating a feedback path which results in the infinite impulse duration. This is in contrast to finite impulse response filters (FIR) which have fixed-duration impulse responses.

The design procedures for IIR filters is somewhat more complicated than FIR filter design because there is no direct design method like the data window method for FIR filters. Instead IIR filters are typically designed by starting with an ideal analog filter in terms of the frequency response characteristics such as the Chebyshev, Butterworth, or Bessel filter. Then the analog filter is converted into a digital filter using a method known as the Bilinear transformation or impulse invariance method.

An IIR digital filter can be understood by considering the difference equation which defines how the input signal is related to the output signal as

$$y[n] = b_0 x[n] + b_1 x[n-1] + \cdots + b_p x[n-P] - a_1 y[n-1] - \cdots - a_q y[n-Q]$$

where P is the feed-forward filter order, b_i are the feed-forward filter coefficients, Q is the feedback filter order, a_i are the feedback filter coefficients, x[n] is the input signal and y[n] is the output signal.

The previous equation can also be expressed as a convolution of the filter coefficients and the input signal.

$$y[n] = \sum_{i=0}^{P} b_i x[n-i] - \sum_{j=0}^{Q} a_j y[n-j]$$

which, when rearranged, becomes:

$$\sum_{j=0}^{Q} a_j y[n-j] = \sum_{i=0}^{P} b_i x[n-i] \quad if\ we\ let\ a_0 = 1$$

To find the transfer function of the filter, we first take the Z-transform of each side of the above equation, where we use the time-shift property to obtain:

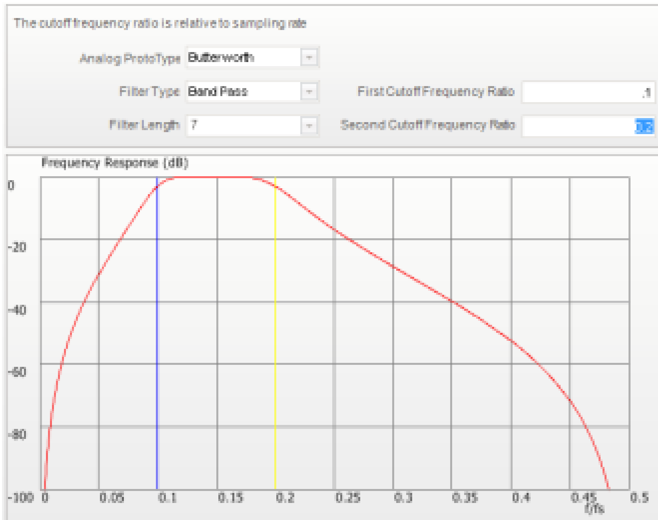
$$\sum_{j=0}^{Q} a_j z^{-j} Y(z) = \sum_{i=0}^{P} b_i z^{-i} X(z)$$

We define the transfer function to be:

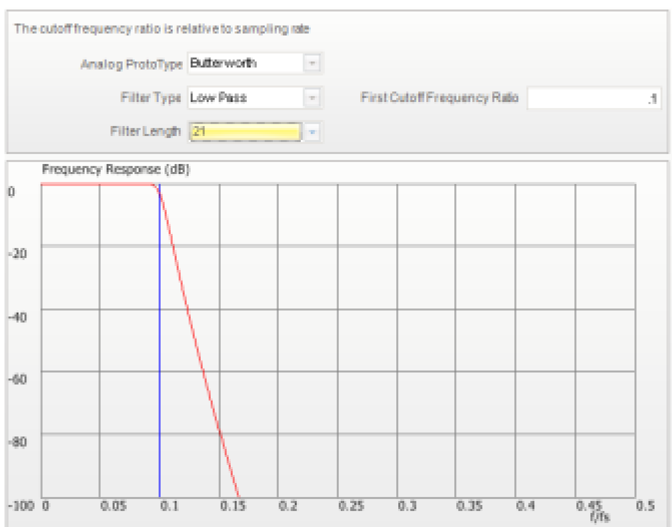
$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{i=0}^{P} b_i z^{-i}}{\sum_{j=0}^{Q} a_j z^{-j}}$$

The transfer function gives the frequency response that relates the input to the output magnitude and phase relationship.

Various analog filter types can be used as the basis for the IIR filter. The Butterworth Filter is the filter type that results in the flattest pass band and contains a moderate group delay. Below are examples of Butterworth low pass and band pass filters.

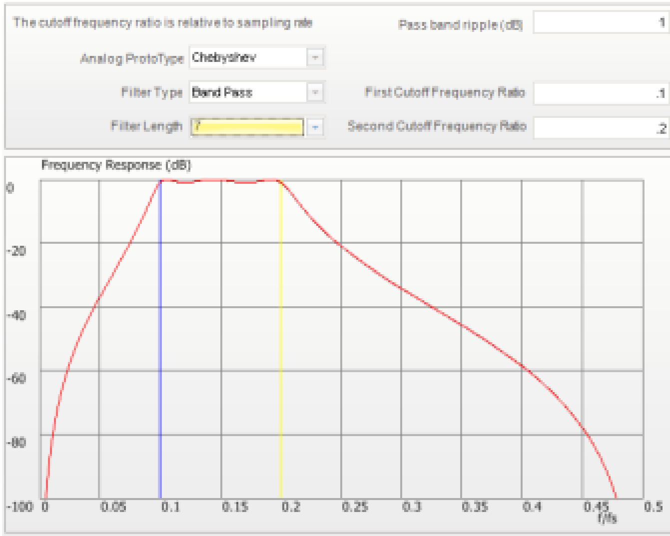


■ Figure 145. Butterworth band pass filter.

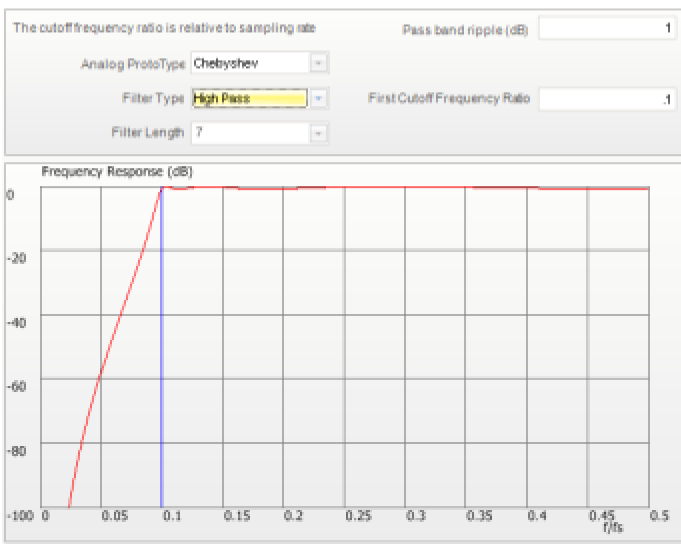


■ Figure 146. Butterworth low pass filter.

The Chebyshev Type I Filter results in the sharpest pass band cut off and contains the largest group delay. The most notable feature of this filter is the significant ripple in the pass band magnitude. A standard Chebyshev Type I Filter's pass band attenuation is defined to be the same value as the pass band ripple amplitude. Below are examples of Chebyshev Type I band pass and high pass filters.

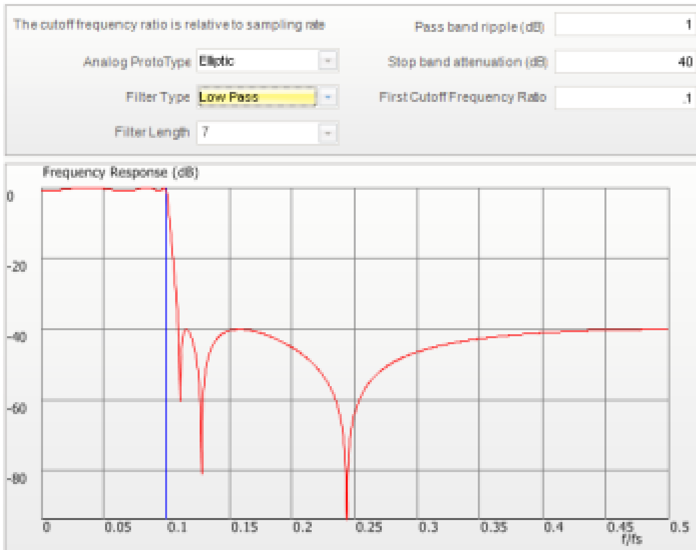


■ Figure 147. Chebyshev type I band pass filter.

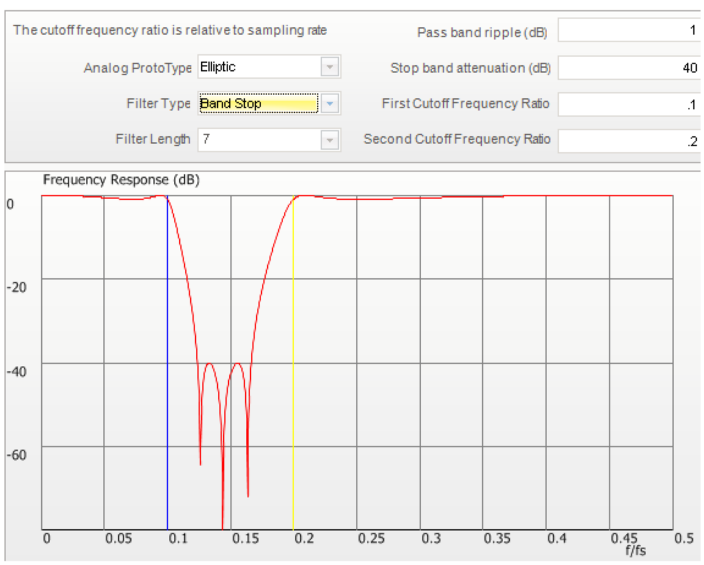


■ Figure 148. Chebyshev type 1 high pass filter.

The Elliptic Filter contains a Chebyshev Type I style equi-ripple pass band, an equipped stop band, a sharp cutoff, high group delay, and the greatest possible stop band attenuation. Below are examples of 7th order Elliptic low pass, band stop filters.



■ Figure 149. Elliptical low pass filter.



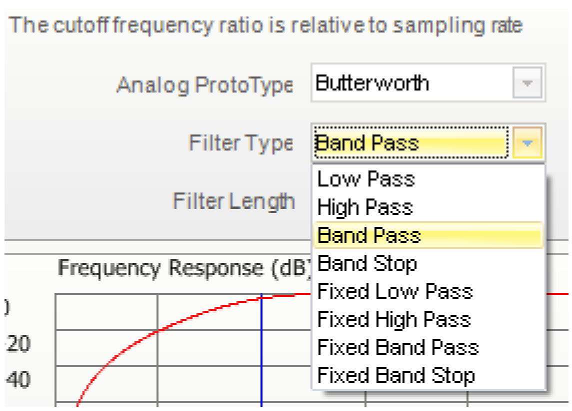
■ Figure 150. Elliptical band stop filter.

Filter Design Using Fixed instead of Relative Frequency

Filter design can be accomplished using either fixed or relative frequency methods. In the relative frequency method the cutoff frequencies are defined relative to the maximum sampling rate. For example if the sampling rate is 1000 Hz and a low pass filter is defined with a cutoff frequency of 0.5 with the relative frequency method then the cutoff frequency is 500 Hz. Note that if the sampling rate is changed to 500 Hz and the cutoff frequency is no changed from 0.5, then the cutoff frequency will change to 250 Hz. The relative frequency method is the preferable method because the filter performance such as ripple or transition band width will not change when the sampling rate is changed.

The alternative is the fixed frequency method where the cutoff frequency is defined by a fixed frequency. With this method the cutoff frequencies do not need to be changed when the sampling rate is changed. While this method is more user friendly than the relative frequency method, it is not the recommended method because the filter performance, such as ripple or transition band width can vary when the sampling rate is changed.

To change between fixed or relative frequencies, go to Filter Type menu and select a filter type with Fixed. The filter types without Fixed are relative by default. :

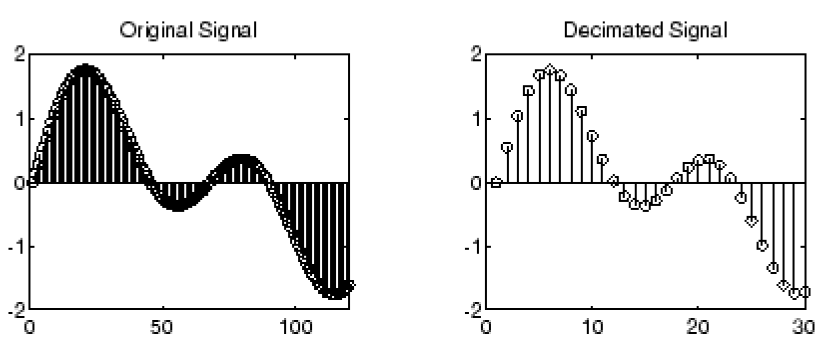


■ Figure 151. Fixed or relative frequency setting.

Decimation Filters

The decimation filter is a special filter available on the CoCo. Decimation reduces the original sampling rate for a sequence to a lower rate. The decimation process filters the input data with a low-pass filter and then re-samples the resulting smoothed signal at a lower rate.

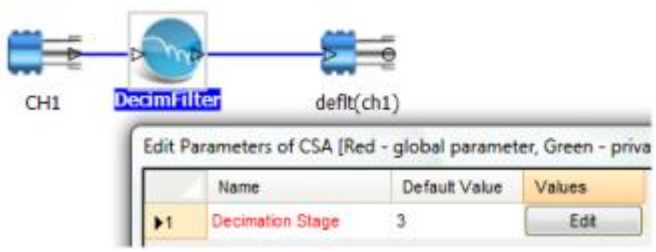
Figure 152. Illustration of a decimation filter. shows how a decimation filter reduce the number of sampled points from 150 to 30 while the signal shape which is dominated by the low frequency components is still retained.



■ Figure 152. Illustration of a decimation filter.

Low pass filtering is important in the decimation process to ensure no aliasing occurs. Aliasing refers to the effect of under sampling a high frequency signal and misrepresenting the high frequency behavior by a lower frequency. When aliasing occurs there is no way to distinguish the erroneous aliased signal from the actual signal. In the CoCo hardware the decimation filter uses a fixed proprietary low pass FIR filter with excellent ripple performance in the pass-band and very high attenuation in the stop band.

In the CoCo hardware, the decimation filter module contains multiple stages of decimation filters. In each stage the data is decimated by a factor of two. After N stages of decimation, the data will be reduced to its $1/2^N$. In the example below, since the decimation stage is set to 3, the data will be reduced to $1/2^3$ = 1/8 of its input points after this decimation module.

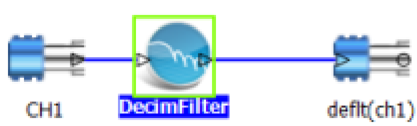


- Figure 153. Decimation filter in the EDM software.

The decimation filter is widely used to view, analyze and record low frequency signals. For example, a system may be used to acquire the vibration and pressure or temperature data simultaneously. While the signals are all sampled at the high data rate, the pressure channel and the temperate channel should be viewed and recorded at a much lower rate because these types of signals typically do not change dynamically (at a high frequency). In this case, we can simply apply decimation filters to the channels that measure and record pressure or temperature.

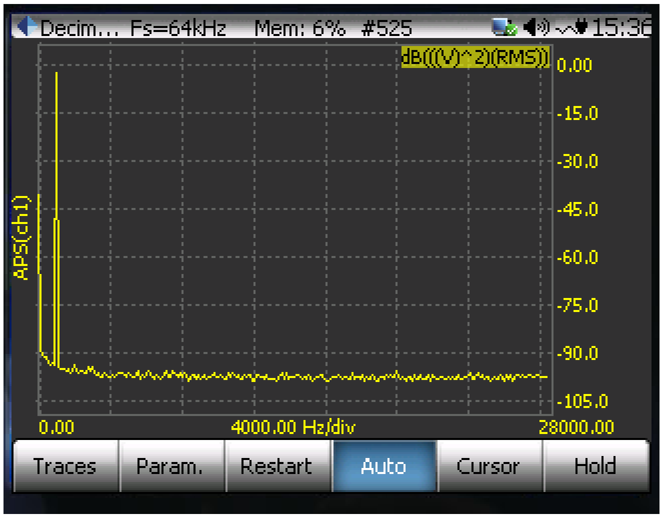
Integrating decimation filter with other filter techniques allows the user to analyze high frequency and lower frequency signals with different frequency resolution simultaneously. This capability is unique to the CoCo system.

For example, in a CSA file you can apply a 3-stage decimation filter to channel 1 as shown in Figure 154.



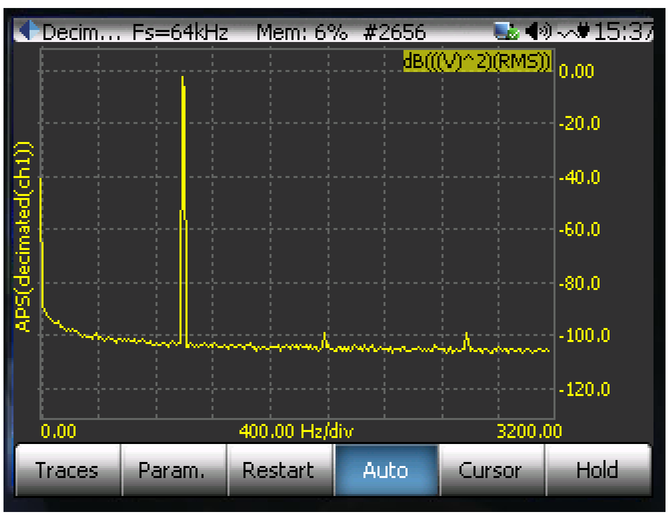
- Figure 154. Decimation filter example.

If you connect a signal source with 1 kHz sine output to channel 1 and set the sampling rate to 64 kHz, you can see the broad spectrum up to 28 kHz as shown in Figure 155.



■ Figure 155 The auto spectrum of an 1kHz sine wave when sampled at 64kHz

However you can also **simultaneously** show the spectrum of the decimated signal, *decimated(ch1)*, with 8 times frequency resolution as shown in Figure 156. Note that in this example the two spectra are of the exact same time signal not of two samples acquired at different times.

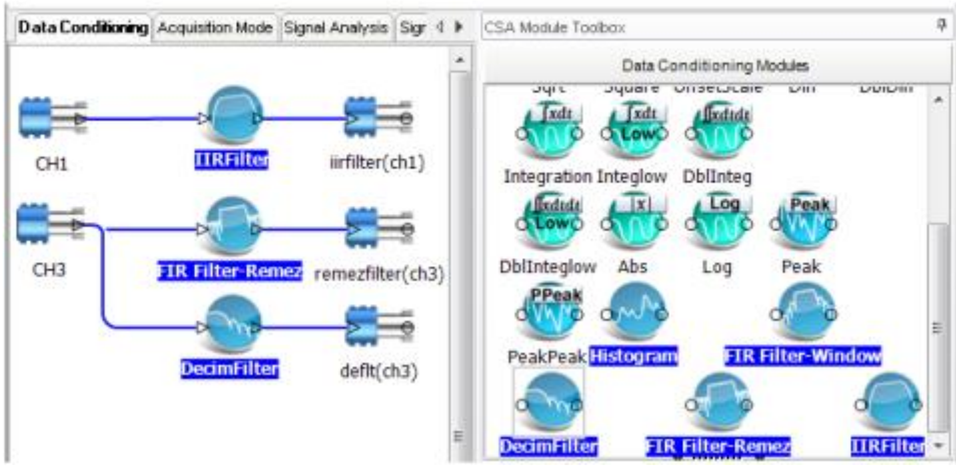


■ Figure 156 The same sine wave at 1kHz, after decimated 8:1, shows at different location on the spectrum

CSA Editor Operation for Real Time Filters

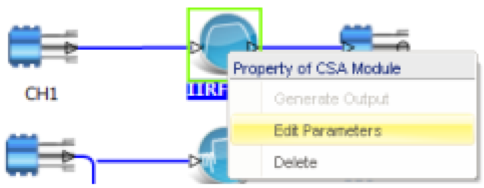
Applying a real time filter in the CSA Editor is very easy. The Real Time Filters can be applied to a CSA created with any of the first five CSA templates including: Data Conditioning Only, Transient Capture, Linear Spectrum, Auto-Power Spectrum and Frequency Response. It can not be applies to Octave Analysis, Sound Level Meter or Order Tracking Analysis.

In the Data Conditioning Tab, drag and drop the filter module to after any of data time streams and connect them. Note that the time stream can be split into multiple paths so for example you can monitor the time stream with two different filters at once as shown below or the filtered and unfiltered signals can be monitored at the same time.



■ Figure 157. Add filters in the Data Conditioning tab.

To enter the filter design dialog box, right click on the filter icon in the Data Conditioning Window, then select Edit Parameters. This opens the filter design window where you can edit the filter settings.



■ Figure 158. Edit filter parameters.

Validation, Save and Upload

After the CSA design, you should validate the CSA by pressing the Validate icon. If the validation passed, you can upload it to CoCo and run it there in real-time.

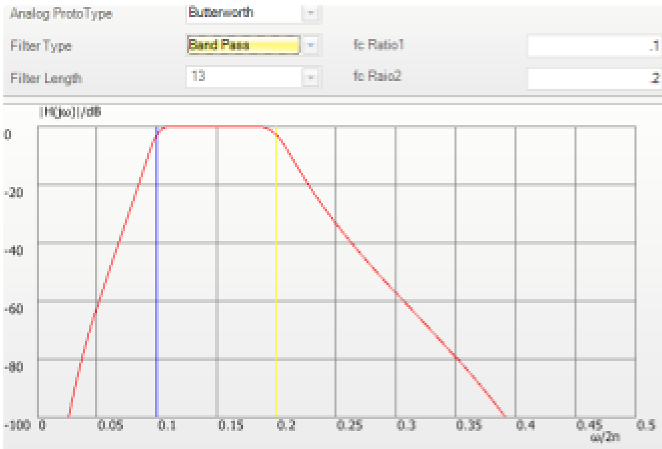
CoCo-80 Operation

There is no special operation on the CoCo hardware when real-time filters are applied. Each of the filter outputs are available as a regular time stream. You can display, record or analyze the time streams in the same way as you do with the native input time streams.

It is important to keep in mind that the cutoff frequencies are relative to the sampling rate, when the sampling rate changes, the cutoff frequencies in absolute Hz will be changed. For example, when the sampling rate is 1000Hz, a 0.1 cutoff frequency means 100Hz. If the sampling rate is changed to 102.4kHz, the cutoff frequency will be moved to 1.024kHz.

An Example

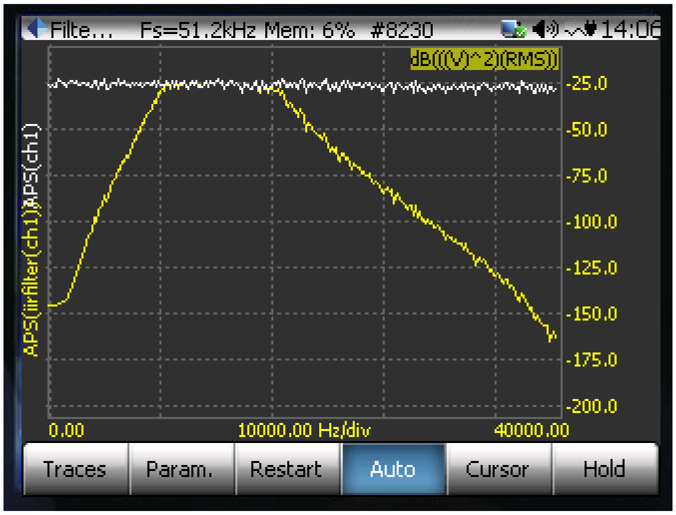
The following figures show an example from design to run-time results. Figure 162 shows a band-pass Butterworth filter with cutoff frequency from 0.1 to 0.2. The Filter order is 13.



■ Figure 159. CSA filter design window.

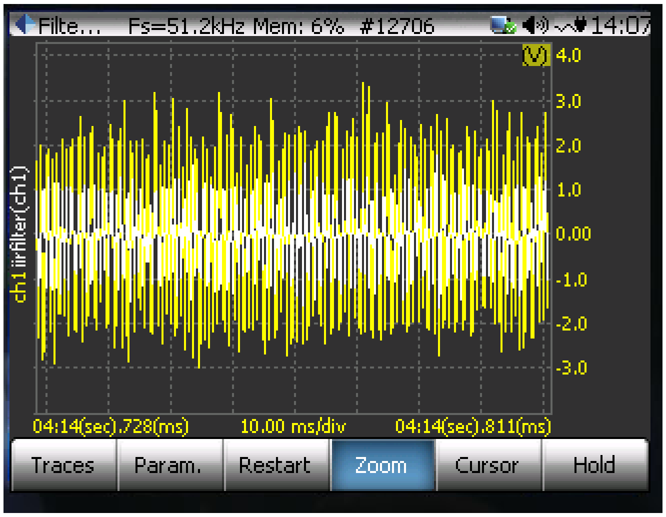
After the CSA is created, validated and uploaded to the CoCo, the signal source is set as white noise and connect to the channel with this filter applied.

Figure 164 shows the auto spectra before and after the filtering process. The white plot shows a flat line, which is the spectrum of the white noise. The yellow plot is the spectrum of the signal after this filter.



■ Figure 160. Filtered and unfiltered white noise spectra.

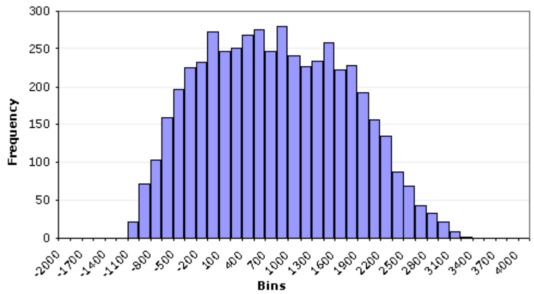
Figure 165 shows the time domain display for the signals before in yellow and after the filter in white.



■ Figure 161. Filtered and unfiltered white noise in the time domain.

9. HISTOGRAM AND STATISTIC MEASURES

A histogram is a graphical display that shows the number (or frequency) of events that fall into each of several or many specified categories. Figure 162 shows a typical histogram.



■ Figure 162. Typical Histogram.

Mathematically, a histogram is a mapping m_i that counts the number of observations that fall into various disjoint categories (known as *bins*). Thus, if we let n be the total number of observations and k be the total number of bins, the total number of events can be found by adding the frequency in all the bins as

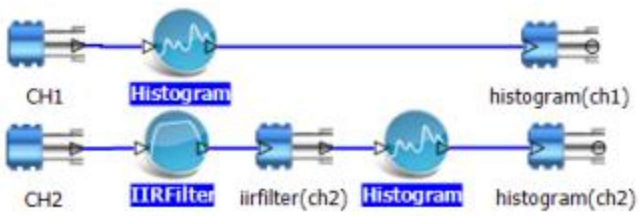
$$n = \sum_{i=1}^{k} m_i$$

Cumulative Histogram

A cumulative histogram is a mapping that counts the cumulative number of observations in all of the bins up to the specified bin. That is, the cumulative histogram M_i of a histogram m_i is defined as:

$$M_i = \sum_{j=1}^{i} m_j$$

In CoCo implementation, the Histogram and Statistics function is a single CSA module that can be applied to *any* time stream. The output of the Histogram and Statistics module is a histogram signal and the associated statistics results. You can change the display format on CoCo.



- Figure 163. Histogram example.

CoCo provides the following measurement parameters for a histogram: bin number for the bar chart and amplitude ranges. It provides the following display formats for the histogram graph: normalized-linear, normalized -logarithmic, un-normalized and cumulative. While the histogram is measured, it also provides the following statistics values: *mean, max, min, RMS, variance, skewness, crest factor* and *kurtosis*. The definitions of these statistics measures for N samples are:

$$Mean = \mu_x = \bar{x} = E(x) = \left(\sum_{i=1}^{N} x_i\right)/N$$

$$Variance(x) = \left(\sum_{i=1}^{N} (x_i - \bar{x})^2\right)/N$$

$$standard\ deviation\ \sigma = \left[\left(\sum_{i=1}^{N} (x_i - \bar{x})^2\right)/N\right]^{\frac{1}{2}}$$

$$rms(x) = \left[\left(\sum_{i=1}^{N} (x_i)^2\right)/N\right]^{\frac{1}{2}}$$

The *rms(x)* is equal to the standard deviation when the mean is 0.

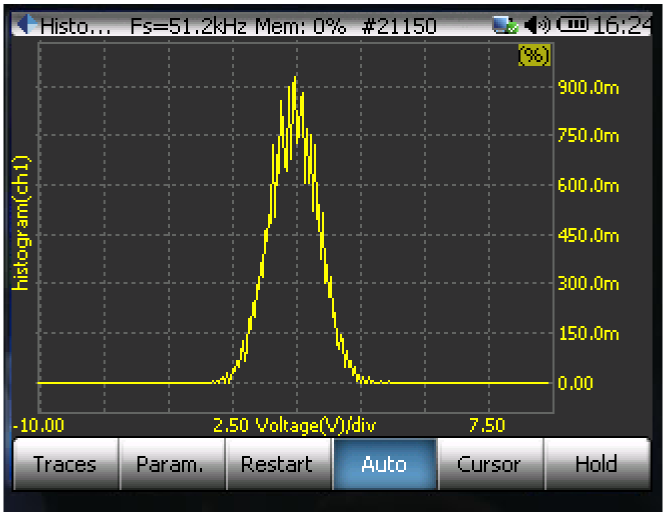
Skewness is a measure of the asymmetry of the data around the sample mean. If the skewness is negative, the data are spread out more to the left of the mean than to the right. If the skewness is positive, the data are spread out more to the right. The skewness of the normal distribution (or a perfectly symmetric distribution) is zero.

$$Skewness(x) = \left(\sum_{i=1}^{N} (x_i - \bar{x})^3\right)/N\sigma^3$$

Kurtosis is a measure of how outlier-prone a distribution is. The kurtosis of the normal Gaussian distribution is 3. Distributions that are more outlier-prone than the normal distribution have kurtosis greater than 3; distributions that are less outlier-prone have kurtosis less than 3.

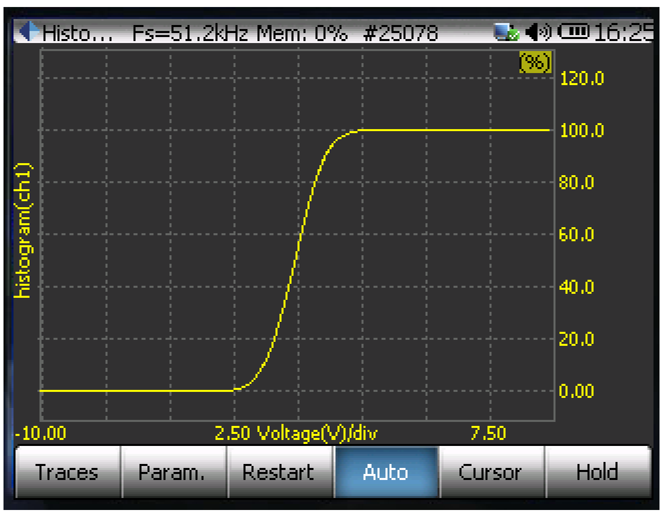
$$Kurtosis(x) = \left(\sum_{i=1}^{N} (x_i - \bar{x})^4\right)/N\sigma^4$$

As an example, a histogram of a Gaussian random noise is displayed in Figure 164.



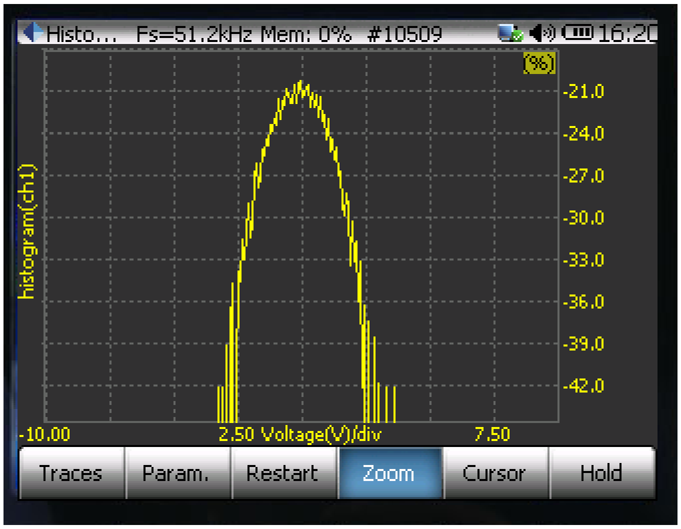
■ Figure 164. Histogram of a Gaussian distributed random signal.

Its cumulative histogram is displayed in Figure 165.



■ Figure 165. Cumulative Histogram of random Gaussian signal.

The display format can be changed to analyze the histogram. For example, a signal distribution with high Kurtosis can be observed using a logarithmic vertical scale with dB units, in the unit of dB, as shown in Figure 166. Here the low frequency outliers can more easily be seen.



■ Figure 166. Histogram with dB scale.

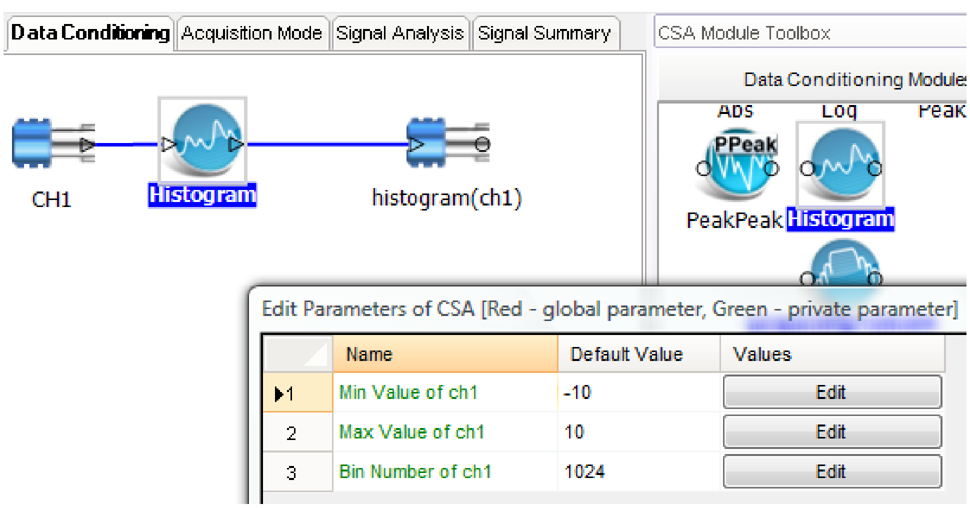
CSA Editor Operation

Open CSA Editor

Click on the CSA Editor icon in EDM.

Apply Histogram Module

Drag the Histogram module to any time streams that you want to analyze, and then connect the time stream to that histogram module. The output signal of Histogram will be display automatically on the screen.



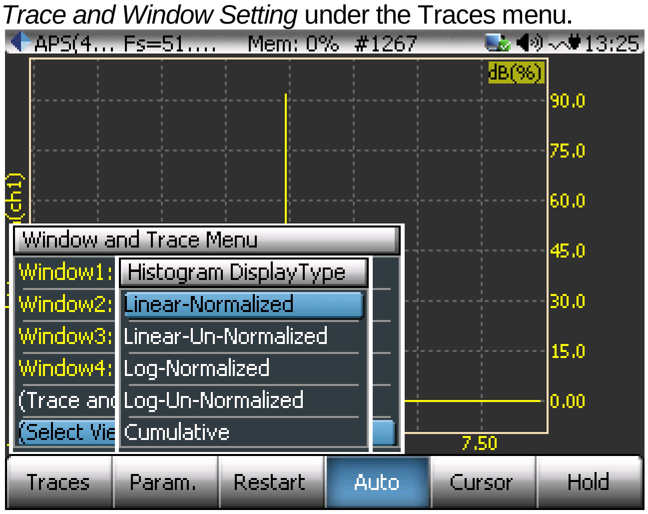
Right-click on the Histogram module, select item of Edit Parameters, the dialog box will be prompted to all you to set the range and the default values of three parameters:

- **Min Value**: minimum value in the amplitude, in engineering unit, for this histogram measurement
- **Max Value**: maximum value in the amplitude, in engineering unit, for this histogram measurement
- **Bin Number**: number of bins within the range,

Finally validate and Upload the CSA to the CoCo hardware.

CoCo-80 Operation

The CoCo can display the histogram in five different modes: normalized, un-normalized, log normalized, log un-normalized, and cumulative. To make such a selection, select View Mode in the *Trace and Window Setting* under the Traces menu.

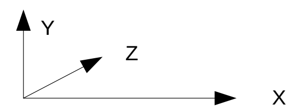


■ Figure 167. Change the histogram display format.

10. MISCELLANEOUS OPERATIONS

3-Dimensional Signal Display

A 3D signal consists of a sequence of 2D signals taken over a period of time. In the CoCo, many signals can be displayed in the three dimensional format. A 3D signal display helps the user to understand the signal with one more axis, Z-axis. This display function is available to many signal types. As long as the 2D signals are block by block, the 3D signal can be used. Time streams cannot be displayed in 3D style.



■ Figure 168 Three axis of 3D display

The engineering unit of Z-axis can be:

Time: when the Z-axis is time, the 2D signals are continuously cascaded one by one in the time direction to form a 3D signal.

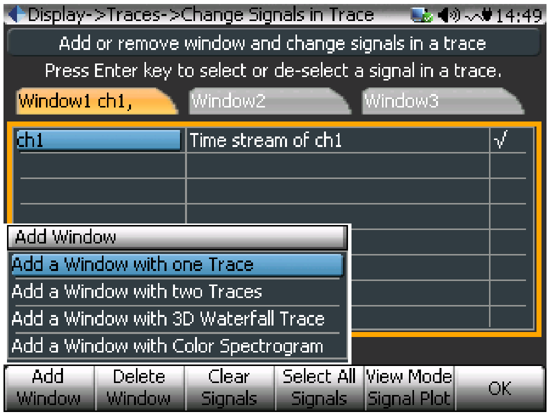
RPM: when the Z-axis is RPM, the CoCo will first look up the RPM attribute of the current measured 2D signal and then place this measurement into the pre-allocated RPM signal slot. Order spectrum and narrowband frequency spectrum are often used. The RPM 3D signal is only available in the Order Tracking software option.

Sequence Number: when the Z-axis is sequence number, the Z-axis displays each frame number of the signals. It is particularly useful for comparing a series of transient capture signals.

3D signals can be displayed in either waterfall or color spectrogram.

CoCo-80 Operation

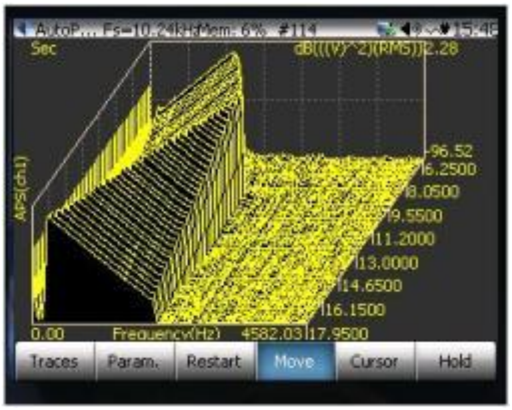
To create a 3D signal display, when you are in the Trace and Window Setting page, press F1 Add Window. The following selection items will be prompted:



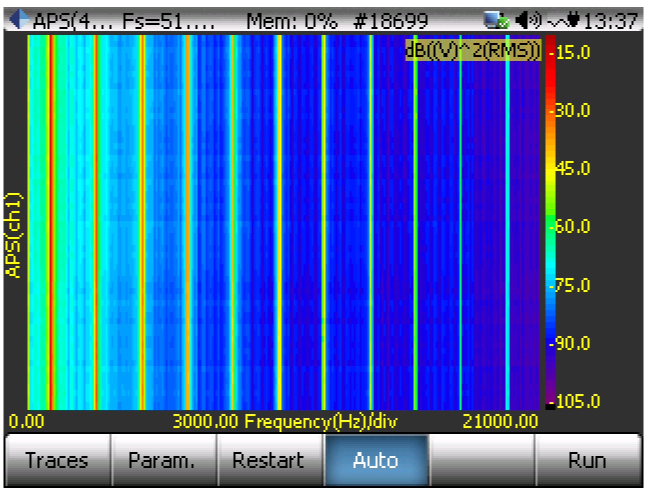
■ Figure 169. Add a 3D window.

The last two choices are for 3D signal display. You can select either 3D waterfall display or color spectrogram.

Below are typical waterfall and color spectrogram displays.



■ Figure 170 A power spectrum time waterfall measuring a swept sine signal



■ Figure 171. Typical color spectrogram.

Transducer Calibration

In the channel table of CoCo, the user will ask for to type in the sensitivity of the transducer. Typically, the sensitivity quantity is described as Engineering Unit per millivolts, or ***EU/mV*** in abbreviation. After the instrument multiplies the input voltage by the sensitivity for that input channel, the displayed unit will always be in the engineering unit that sensor measures.

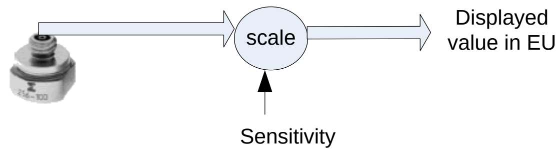


Figure 172 Engineering Unit Scaling to the Input Channel

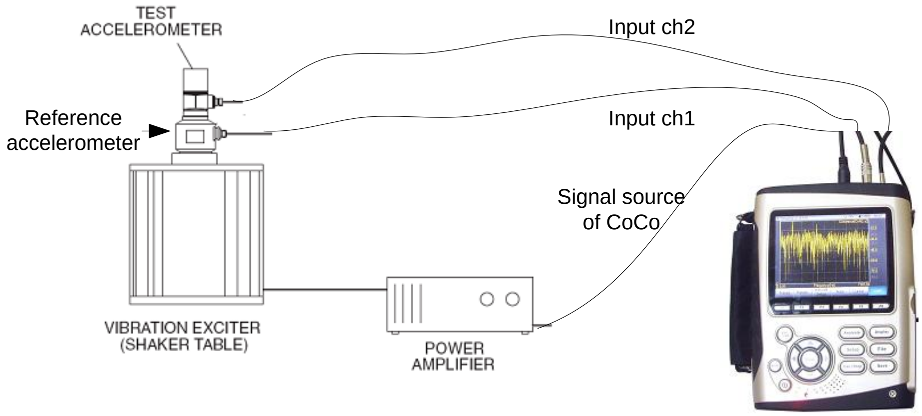
Usually, the user specifies the sensitivity that comes with the transducer from the manufacturer. However as time goes by, the actual sensitivity may deviate from this value causing calibration errors is measurements. You can either send the transducer to a calibration service company to re-calibrate it, i.e., to get the sensitivity value updated, or use the CoCo to calibrate the transducer.

For many transducers, the sensitivity that is used is called nominal sensitivity. It is a single value that represents the most typical average sensitivity values over some frequency range. In reality, the sensitivity of any kind of transducer varies with frequency.

Transducer calibration in CoCo uses a reference source to calibrate the sensitivity of the sensors, such as microphones and accelerometers. The reference source can be an exciter that can generate an accurate and reliable physical quantity that the transducer can measure, or use an exciter with a more accurate sensor to conduct a "back-to-back" calibration. In both cases, the Coco will ask for a reference input signal with a reliable measurement quantity, record the transducer signal, and calculate the sensitivity of the transducer.

Back-to-Back Calibration

This method is mainly used for calibrating the accelerometers. Accelerometer calibration determines the sensitivity at various frequencies of interest. The Instrumentation, Systems, and Automation Society
(ISA) approved back-to back comparison method is probably the most convenient and least expensive technique. Back-to-back calibration involves coupling the test accelerometer directly to a (NIST) traceable double-ended calibration standard accelerometer and driving the coupled pair with a vibration exciter at various frequencies and acceleration (g) levels. The assumption here is that since the accelerometers are tightly coupled together, both will experience exactly the same motion, thus the calibration of the back-to-back standard accelerometer can be precisely "transferred" to the test accelerometer. A back-to-back vibration calibration system used in conjunction with a small electrodynamic shaker, a signal generator, a frequency meter and several other pieces of equipment provides an inexpensive means to set up a calibration facility.



■ Figure 173. Typical back to back calibration setup.

CoCo is an ideal platform to conduct the back-to-back test because it has an accurate signal source. Following these procedures:

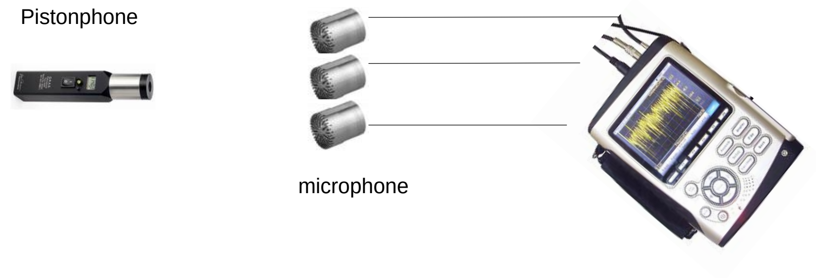
1. Turn down the gain of the amplifier of the shaker so there is not output to the shaker when we connect them
2. Connect the amplifier, the reference accelerometer and the test accelerometer as shown in the diagram
3. Connect the signal source of CoCo to the amplifier
4. Use sine wave as reference signal output. Select a typical frequency, say 1kHz.
5. Connect the outputs of reference accelerometer and test accelerometer to channel 1 and 2 of CoCo.
6. Type in the sensitivity of reference accelerometer into channel 1. This sensitivity is believed to be accurate and can be used for comparison.

Pistonphone Microphone Calibration

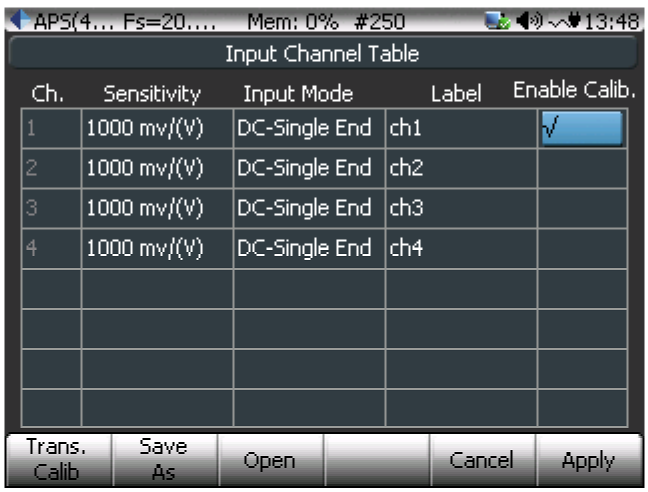
A pistonphone is used for calibrating a microphone. A pistonphone is usually a battery-operated, precision sound source for accurate and reliable calibration of measurement microphones, sound level meters and other sound measuring equipment. With a microphone placed in the coupler of the pistonphone, the calibration level and frequency is some sound level such as 114dB, reference to 20µPa, at around 250Hz. The actual sound pressure level, corrected for static ambient pressure, can also be shown on the display of the pistonphone.

■ Figure 174 A pistonphone used for calibrating the microphone

To calibrate the microphone, simply connect one or multiple microphones to the input channels of CoCo, as shown below.



Select the Input Channels from the Param. Button. Then move the cursor to the Enable Calibration column and press the Enter Button to select the calibration channel. You can calibrate any one or all of the input channels. Next press the Trans. Calib. Button.



APS(4... Fs=20.... Mem: 0% #250 13:48

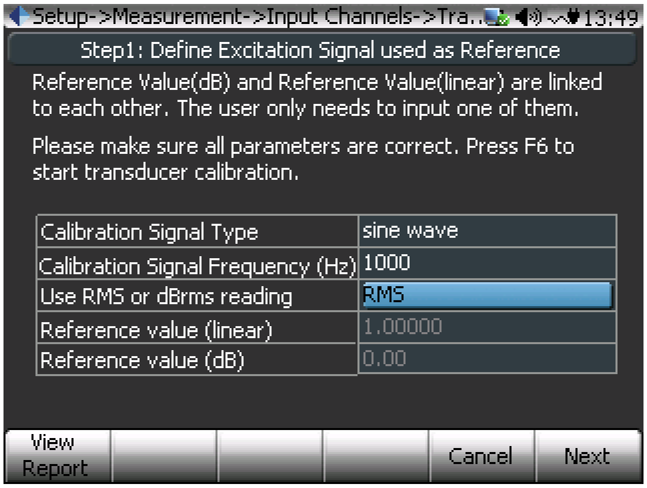
Input Channel Table

Ch.	Sensitivity	Input Mode	Label	Enable Calib.
1	1000 mv/(V)	DC-Single End	ch1	√
2	1000 mv/(V)	DC-Single End	ch2	
3	1000 mv/(V)	DC-Single End	ch3	
4	1000 mv/(V)	DC-Single End	ch4	

Trans. Calib | Save As | Open | Cancel | Apply

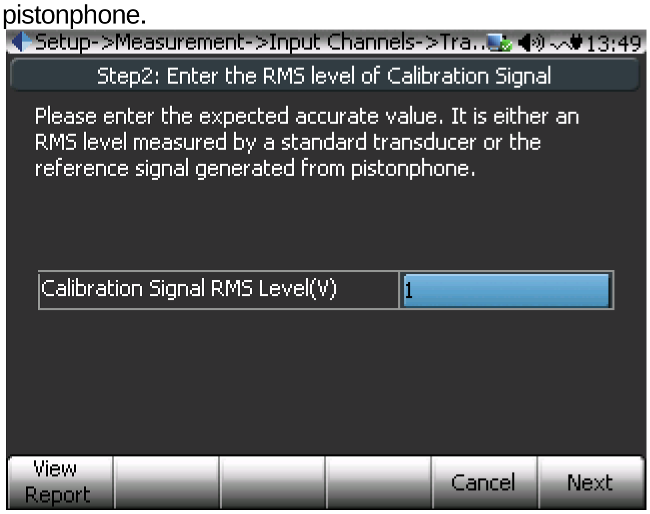
■ Figure 175. Select which channels will be calibrated.

Next enter the calibration signal frequency in Hz and either the linear or dB reference value then press the Next Button.



■ Figure 176. Set the calibration reference values.

Next enter the calibration signal RMS level in volts. This value is typically given by the pistonphone.

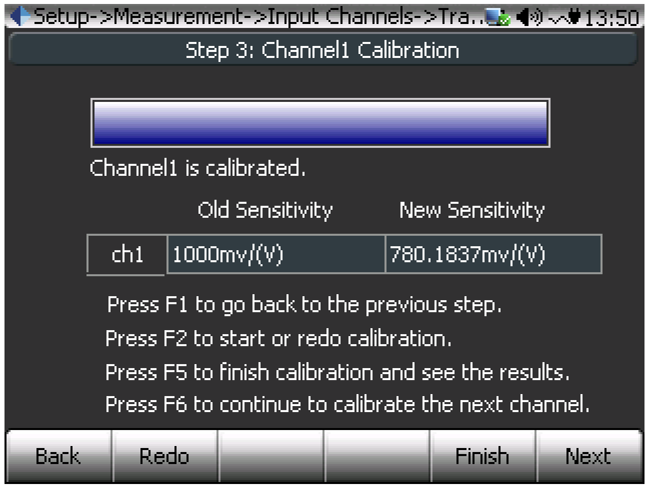


■ Figure 177. Enter the calibration signal level.

For many calibration procedures the calibration is made in the dB scale. This is very common in microphone calibration. In this case you should se the Use RMS or dBrms reading to dB. Then in the Figure 106 you must enter a dB value, for example, 94 dB.

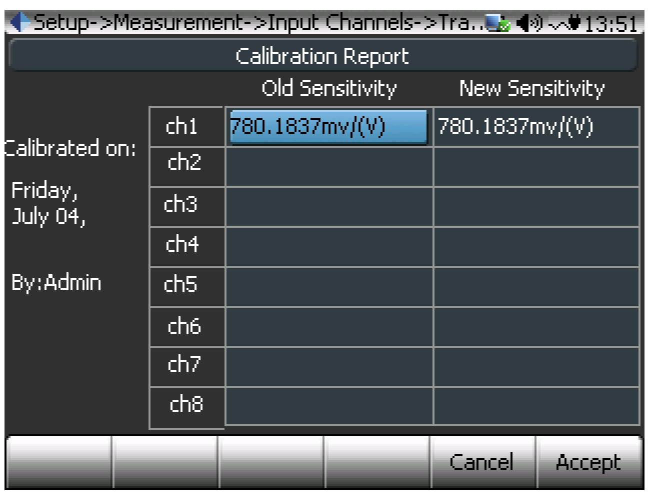
Connect the microphone to the pistonphone and then press the Start Calib. Button. The display indicates which channel is currently being calibrated. Note the calibration starts with the first channel and then proceeds to the second channel etc. The display shows "Waiting for signal…" until it detects the calibration signal on the input channel. When the calibration signal is detected, then the progress bar shows the status of the process and indicates when the calibration is complete. The old and new sensitivities are then displayed.

At this point you can go back to the previous step by pressing the Back Button, redo the calibration by pressing the Redo Button, finish the calibration and see the results by pressing the Finish Button, or continue to calibrate the next channel by pressing the Next Button.



■ Figure 178. Calibration results.

When you finish calibrating all the input channels press the Finish Button to view the Calibration Report which shows the old and new sensitivity values for each channel. Finally press the Accept Button to accept the new calibration values. Then press the Apply Button to accept the new Input Channel values.



■ Figure 179. Calibration report.

After the Input Channel Table is modified, you can save it for use with other projects so that you do not have to redo the calibration with every additional project. Press the Save Button. Then select Save Input Channel Table. Next type in a name for the file and press the OK Button. This can also be done from the Input Channel Table window.

To load a saved Input Channel Table select Input Channel Table from the Param. Button and then press the Open Button. Select an input channel table file from the list and press the Open Button.

11. Appendix

Version

Version	Release Date	Comments
???	7/15/08	First Draft includes octave analysis, sound level meter, order tracking, automated test, real time filter, histogram, integration, and misc.
0.994	10/14/08	Added Swept Sine and SRS
1.01	2/2009	Move Digital Integration Chapter to Basics, pictures compressed

Users' Manual Typeface

Headings Arial Black 12 and 11 pt

Body Text: Arial 10 pt

Captions: Arial Narrow 9 pt

Declaration of Conformity

Declaration of Conformity for CI CoCo-80, Handheld Data Acquisition System

Manufacturer: 4633 Old Ironsides Drive, Suite 304, Santa Clara, CA 95054, USA

Statement of Conformity:

EC Declaration of Conformity

Council Directive 2004/108/EC on Electromagnetic Compatibility

WE, Crystal Instruments

4633 Old Ironsides Drive, Suite 304, Santa Clara, CA 95054, USA

Product Name: CoCo-80 (Handheld Data Acquisition System)

Model No.: CoCo-80

Assessment of compliance of the product with the requirement relating to Electromagnetic Compatibility Directive .The product has been assessed by the application of the following standards:

EN 61326:1997+A1:1998+A2:2001
EN61000-3-2: 2000
EN61000-3-3: 1995+A1:2001

The tests have been performed in a typical configuration.

This Conformity is indicated by the symbol, i.e. “Conformité Européenne”.

Dynamic Signal Analysis Basics

James Zhuge, Ph.D., President
Crystal Instruments Corporation
4633 Old Ironsides Drive, Suite 304
Santa Clara, CA 95054, USA
www.go-ci.com

(Part of CoCo-80 User's Manual)

Table of Contents

Table of Figures

FREQUENCY ANALYSIS

Basic Theory of FFT Frequency Analysis

Introduction

DSA, often referred to Dynamic Signal Analysis or Dynamic Signal Analyzer depending on the context, is an application area of digital signal processing technology. Compared to general data acquisition and time domain analysis, DSA instruments and math tools focus more on the dynamic aspect of the signals such as frequency response, dynamic range, total harmonic distortion, phase match, amplitude flatness etc.. In recent years, time domain data acquisition devices and DSA instruments have gradually converged together. More and more time domain instruments, such as oscilloscopes, can do frequency analysis while more and more dynamic signal analyzers can do long time data recording.

DSA uses various different technology of digital signal processing. Among them, the most fundamental and popular technology is based on the so called the Fast Fourier Transform (FFT). The FFT transforms the time domain signals into the frequency domain. To perform FFT-based measurements, however, you need to understand the fundamental issues and computations involved. This Chapter describes some of the basic signal analysis computations, discusses antialiasing and acquisition front end for FFT-based signal analysis, explains how to use windowing functions correctly, explains some spectrum computations, and shows you how to use FFT-based functions for some typical measurements.

In this Chapter we will use standard notations for different signals. Each type of signal will be represented by one specific letter. For example, “G” stands for a one-side power spectrum, while “H” stands for a transfer function.

The following table defines the symbols used in this Chapter:

Cyx	Coherence function between input signal x and output signal y
Gxx	Auto-spectral function (one-sided) of signal x
Gyx	Cross-spectral function (one-sided) between input signal x and output signal y
Hyx	Transfer function between input signal x and output signal y
k	Index of a discrete sample
Rxx	Auto-correlation function of signal x
Ryx	Cross-correlation function between input signal x and output signal y
Sx	Linear spectral function of signal x
Sxx	Instantaneous auto-spectral function (one-sided) of signal x
Syx	Instantaneous cross-spectral function (one-sided) between input signal x and output signal y
t	Time variable
x(t)	Time history record
X(f)	Fourier Transform of time history record

Fourier Transform

Digital signal processing technology includes FFT based frequency analysis, digital filters and many other topics. This chapter introduces the FFT based frequency analysis methods that are widely used in all dynamic signal analyzers. CoCo has fully utilized the FFT frequency analysis methods and various real time digital filters to analyze the measurement signals.

The Fourier Transform is a transform used to convert quantities from the time domain to the frequency domain and vice versa, usually derived from the Fourier integral of a periodic function when the period grows without limit, often expressed as a Fourier transform pair. In the classical sense, a Fourier transform takes the form of

$$X(f) = \int_{-\infty}^{\infty} x(t)e^{-j2\pi ft}\,dt$$

where

x(t)	continuous time waveform
f	frequency variable
j	complex number
X(f)	Fourier transform of *x(t)*

Mathematically the Fourier Transform is defined for all frequencies from negative to positive infinity. However, the spectrum is usually symmetric and it is common to only consider the single-sided spectrum which is the spectrum from zero to positive infinity. For discrete sampled signals, this can be expressed as

$$X(k) = \sum_{n=0}^{N-1} x(k)e^{-j2\pi kn/N}$$

where

x(k)	samples of time waveform
n	running sample index
N	total number of samples or “frame size”
k	finite analysis frequency, corresponding to “FFT bin centers”
X(k)	discrete Fourier transform of *x(k)*

In most DSA products, a Radix-2 DIF FFT algorithm is used, which requires that the total number of samples must be a power of 2 (total number of samples in FFT = 2^m , where m is an integer).

Data Windowing

The Fourier Transform assumes that the time signal is periodic and infinite in duration. When only a portion of a record is analyzed the record must be truncated by a data window to preserve the frequency characteristics. A window can be expressed in either the time domain or in the frequency domain, although the former is more common. To reduce the edge effects, which cause leakage, a window is often given a shape or weighting function. For example, a window can be defined as

$$w(t) = g(t) \qquad -T/2 < t < T/2$$
$$= 0 \qquad \text{elsewhere}$$

where $g(t)$ is the window weighting function and T is the window duration.

The data analyzed, $x(t)$ are then given by

$$x(t) = w(t)\, x(t)'$$

where $x(t)'$ is the original data and $x(t)$ is the data used for spectral analysis.

A window in the time domain is represented by a multiplication and hence, is a convolution in the frequency domain. A convolution can be thought of as a smoothing function. This smoothing can be represented by an effective filter shape of the window; i.e., energy at a frequency in the original data will appear at other frequencies as given by the filter shape. Since time domain windows can be represented as a filter in the frequency domain, the time domain windowing can be accomplished directly in the frequency domain.

In most DSA products, rectangular, Hann, Flattop and several other data windows are used;

Rectangular Window

$$w(k) = 1 \qquad 0 \le k \le N\text{-}1$$

Hann Window

$$w(k) = 0.5 * (1 - \cos (2\pi k /(N-1)) \qquad 0 \le k \le N\text{-}1$$

Because creating data window attenuates a portion of the original data, a certain amount of correction has to be made in order to get an un-biased estimation of the spectra. In linear spectral analysis, an *Amplitude Correction* is applied; in power spectral measurements, an *Energy Correction* is applied. See the sections below for details.

Linear Spectrum

A linear spectrum is the Fourier transform of windowed time domain data. The linear spectrum is useful for analyzing periodic signals. You can extract the harmonic amplitude by reading the amplitude values at those harmonic frequencies.

An averaging technique is often used in the time domain when synchronized triggering is applied. Or equivalently, the averaging can be applied to the complex FFT spectra.

Because the averaging is taking place in the linear spectrum domain, or equivalently, in the time domain, based on the principles of linear transform, averaging make no sense unless a synchronized trigger is used.

Most DSA products use the following steps to compute a linear spectrum:

<u>Step 1</u>
First a window is applied:

$$x(t) = w(t)\, x(t)'$$

where $x(t)'$ is the original data and $x(t)$ is the data used for the Fourier transform.

<u>Step 2</u>

The FFT is applied to *x(t)* to compute *X(k)*, as described above.

Step 3
Averaging is applied to *X(k)*. Here Averaging can be either an Exponential Average or Stable Average. Result is *Sx'*.

Sx' = Average (X(k))

Step 4
To get a single-sided spectrum, double the value for symmetry about DC.
An Amplitude Correction factor is applied to *Sx'* so that the final result has an un-biased reading at the harmonic frequencies.

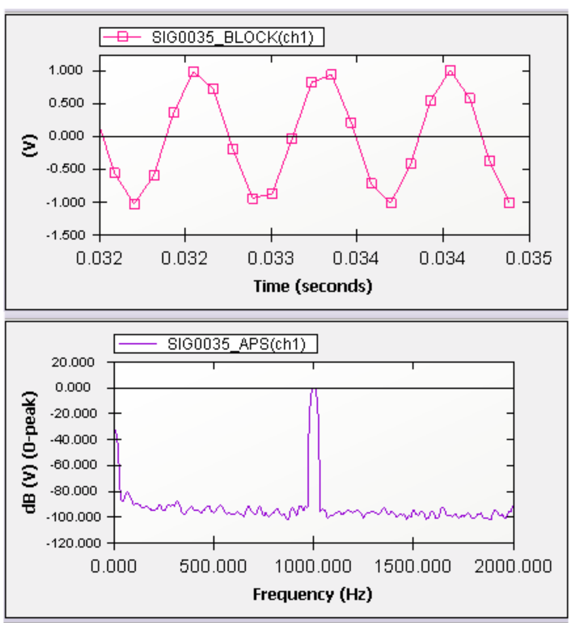
$$Sx = 2 \bullet Sx' / AmpCorr$$

where AmpCorr is the amplitude correction factor, defined as:

$$AmpCorr = \sum_{k=0}^{N-1} w(k)$$

where *w(k)* is the window weighting function.

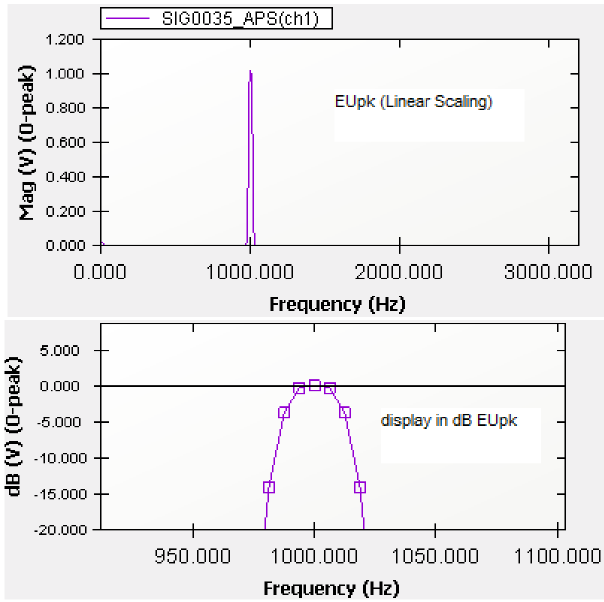
This correction will make the peak or RMS reading of a sine wave at specific frequency correct regardless of which data window is applied. For example, if a 1.0 volt amplitude 1kHz sine wave sampled at 6.4kHz is analyzed with a Linear Spectrum with Hann window, you will get following the spectral shape:



■ Figure 1. Sine wave with Hanning window applied to the spectrum.

The top picture is the digitized time waveform. The sine-wave is not smooth because of the low sampling rate relative to the frequency of the signal. However the well known Nyquist principle indicates that the frequency estimate from the FFT will be accurate as long as the sampling rate is more than twice of the signal frequency. The frequency spectrum of the period signal will show the accurate frequency and level. Note for a more accurate sample of the time waveform a higher sampling rate is required.

Figure 2 illustrates a windowing function applied to a pure sine tone.



■ Figure 2. Hanning windowing function applied to a pure sine tone.

The top picture is displayed in EUpk, i.e., the peak of the spectrum is scaled to the actual 0 peak level, which is 1.0 in this case. The bottom picture shows the same signal with the dB scale applied. Since we use 0dB as reference, the 1.0 Vpk is now scaled to 0.0 dB. With the dB display, we can see frequency points around the peak causing by the Hanning window.

The linear spectrum is saved internally in the complex data format with real and imaginary parts. Therefore, you should be able to view the real and imaginary parts, or amplitude and phase of the spectrum.

Power Spectrum

Spectral analysis is popular in characterizing the operation of mechanical and electrical systems. A type of spectral analysis, the power spectrum (and power spectral density (PSD)), is especially popular because a “power” measurement in the frequency domain is one that engineers readily accept and apply in their solutions to problems. Single channel measurements (auto-power spectra) and two channel measurements (cross-power spectra) both play important roles.

In power spectrum measurements, window ***amplitude correction*** is used to get un-biased final spectrum amplitude reading at specific frequency. In PSD or energy spectral density (ESD) measurements, window ***energy correction*** is always used to get an un-biased spectral density or energy reading.

To compute the spectra listed above, the instrument will follow these steps:

Step 1
A window is applied:

$$x(k) = w(k)\, x(k)'$$

where $x(k)'$ is the original data and $x(k)$ is the data used for a Fourier transform.

Step 2
The FFT is applied to $x(t)$ to compute Sx

$$Sx = \sum_{n=0}^{N-1} x(k) e^{-j2\pi kn/N}$$

Next the so called periodogram method is used to compute the spectra with area correction. Using Sx.

Step 3

Calculate the Power Spectrum $Sxx = Sx\, Sx^{*} / (AmpCorr)^{2}$

Or calculate the Power Spectral Density $= Sx\, Sx^{*}\, T / EnergyCorr$

Or calculate the Energy Spectral Density $= Sx\, Sx^{*}\, T^{2} / EnergyCorr$

where T is the time duration of the capture. The symbol * is for complex conjugation. *EnergyCorr* is a factor for energy correction, which is defined as:

$$EnergyCorr = \frac{1}{N} \sum_{k=0}^{N-1} w(k)^{2}$$

N is the total number of the samples and $w(k)$ is window function.

For any power spectral measurement of the three types listed above, the EU is automatically chosen as EU_{rms} because only EU_{rms} has a physical meaning related to signal power.

After the power spectra are calculated, the averaging operation will be applied. More details will be discussed in the next sections for averaging operation.

Spectrum Types

Several Spectrum Types are given for both Linear Spectrum and Power Spectrum measurements in CoCo and EDM. The concept of spectrum type is explained below in detail.

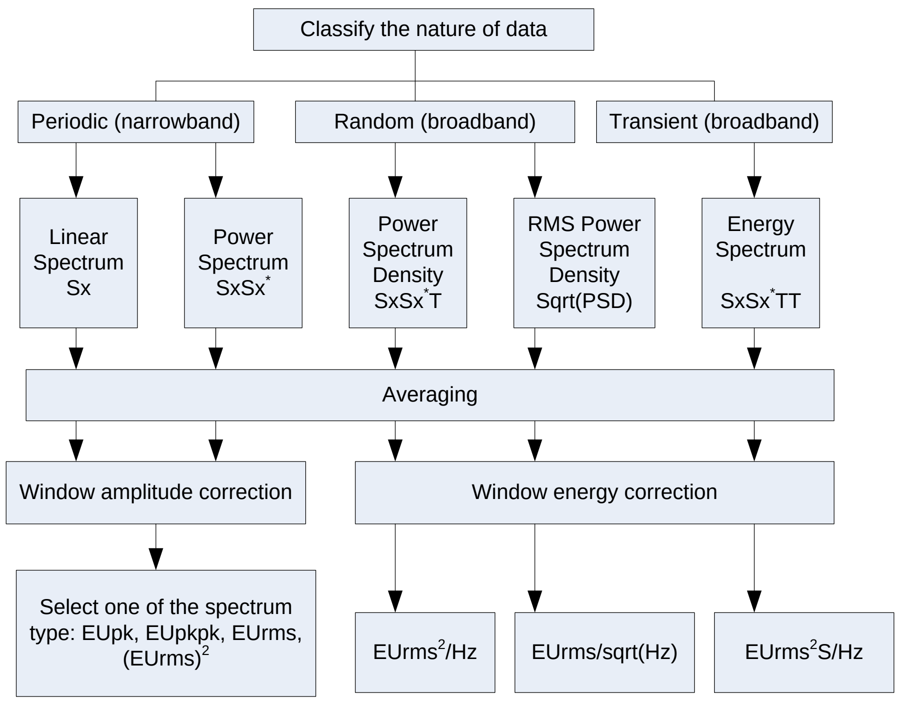
First let's consider the signals with periodic nature. These can be the signals measured from a rotating machine, bearing, gearing, or anything that repeats. In this case we would be interested in amplitude changes at fundamental frequencies, harmonics or sub-harmonics. In this case, you can choose a spectrum type of EU_{pk}, EU_{pkpk} or EU_{rms}.

A second scenario might consist of a signal with a random nature that is not necessarily periodic. It does not have obvious periodicity therefore the frequency analysis could not determine the "amplitude" at certain frequencies. However, it is possible to measure the r.m.s. level, or power level, or power density level over certain frequency bands for such random signals. In this case, you must select one of the spectrum types of EU_{rms}^2/Hz, or EU_{rms}/sqrt(Hz), which is called power spectral density, or root-mean squared density.

A third scenario might consist of a transient signal. It is neither periodic, nor stably random. In this case, must select a spectrum type as EU^2S/Hz, which is called energy spectrum.

In many applications, the nature of the data cannot be easily classified. Care must be taken to interpret the data when different spectrum types are used. For example, in the environmental vibration simulation, a typical test uses multiple sine tones on top of random profile, which is called Sine-on-Random. In this type application, you have to observe the random portion of the data in the spectrum with EU_{rms}^2/Hz and the sine portion of the data with EU_{pk}.

Figure 3 shows a general flow-chart to choose one of the measurement techniques and spectrum types for linear or auto spectrum:

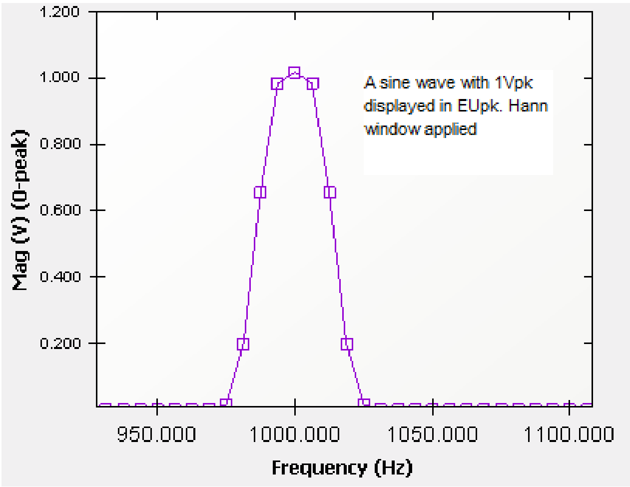


■ Figure 3. Flow chart to determine measurement technique for various signal types.

The following figures illustrate the results of different measurement techniques on a 1 volt pure sine tone. The figures include RMS, Peak or Peak-Peak value for the amplitude, or power value corresponding to its amplitude. Notice these readings can only be applied to a periodic signal. If you applied these measurement techniques to a signal with random nature, the spectrum would not be a meaningful representation of the signal.

EU_{pk} or EU_{pkpk}

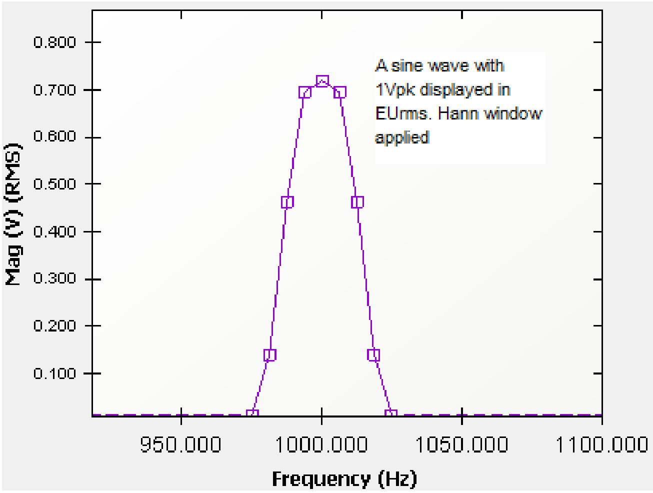
The EU_{pk} and EU_{pkpk} displays the peak value or peak-peak value of a periodic frequency component at a discrete frequency. These two spectrum types are suitable for narrowband signals.



■ Figure 4. A sine wave is measured with EUpk spectrum unit. The sine waveform has a 1V amplitude.

EU_{rms}

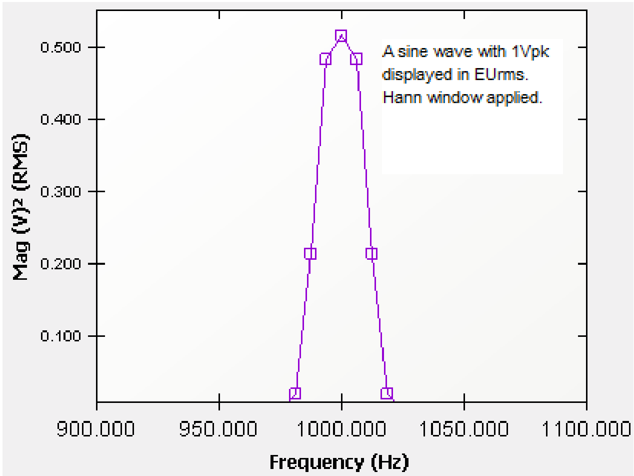
The EU_{rms} displays the RMS value of a periodic frequency component at a discrete frequency. This spectrum type is suitable for narrowband signals.



■ Figure 5. A sine wave is measured with EUrms spectrum unit. The peak reading is 0.707V. The sine waveform has a 1V amplitude.

$(EU_{rms})^2$ Power spectrum

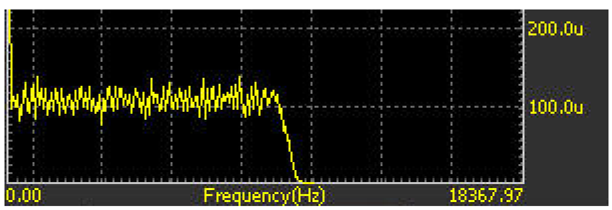
The $(EU_{rms})^2$ displays the power reading of a periodic frequency component at a discrete frequency. This spectrum type is suitable for narrowband signals.



■ Figure 6. A sine wave is measured with $(EUrms)^2$ spectrum unit. The peak reading is $0.5V^2$. The sine waveform has a 1V amplitude.

EU^2/Hz, Power Spectrum Density

The EU^2/Hz is the spectrum unit used in power spectrum density (PSD) calculations. The unit is in engineering units squared divided by the equivalent filter bandwidth. This provides power normalized to a 1Hz bandwidth. This is useful for wideband, continuous signals. EU^2/Hz really should be written as $(EU_{rms})^2$/Hz. But probably due to the limitation of space, people put it as EU^2/Hz.



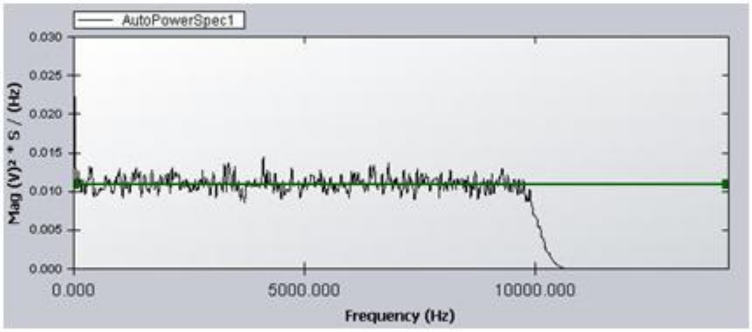
■ Figure 7. White noise with 1 volt RMS amplitude displays as 100 u V_{rms}^2/Hz.

Figure 7 shows a white noise signal with $1V_{rms}$ amplitude or $1V^2$ in power level. The bandwidth of the signal is approximately 10000 Hz and the V^2/Hz reading of the signal is around 0.0001 V^2/Hz. The 1 V RMS can be calculated as follows:

$$1\ V_{rms} = \mathrm{sqrt}\ (10000\mathrm{Hz} * 0.0001\ V^2/\mathrm{Hz})$$

EU^2S/Hz, Energy Spectrum Density

The EU^2S/Hz displays the signal in engineering units squared divided by the equivalent filter bandwidth, multiplied by the time duration of signal. This spectrum type provides energy normalized to a 1Hz bandwidth, or energy spectral density (ESD). It is useful for any signals when the purpose is to measure the total energy in the data frame. Figure 8 shows a random signal with a 1 volt RMS level in the ESD format.



- Figure 8. Random signal with 1 volt RMS amplitude and Energy Spectrum Density format.

The ESD is calculated as follows:

Values for ESD = values of PSD * Time Factor

were the Time Factor = (Block size)/Δf and Δf is the sampling rate / block size.

Notice that in **EU^2/Hz**, or **EU^2S/Hz,** EU really means the RMS unit of the EU, i.e., EU_{rms}.

It should also be noted that since a window is applied in time domain, which corresponds a convolution in the linear spectrum, we cannot have both a valid amplitude and correct energy correction at the same time. Use Figure 3 to select appropriate spectrum types.

In a Linear Spectrum measurement, a signal is saved in its complex data format which includes both real and imaginary data. Then is averaging operation applied to the linear spectrum. In a Power Spectrum measurement, the averaging operation is applied to the squared spectrum, which has only real part. Because of different averaging techniques, the final results of Linear Spectrum and Power Spectrum will be different even though the same spectrum type is used.

Spectrum Types selection only applies to Power Spectrum and Linear Spectrum signals. Spectrum Types do not apply to transfer functions, phase functions or coherence functions.

Cross Spectrum

Cross spectrum or cross power spectrum density is a frequency spectrum quantity computed using two signals, usually the excitation and response of a dynamic system. Cross spectrum is not commonly used by its own. Most often it is used to compute the frequency response function (FRF), transmissibility or cross correlation function.

To compute the cross-power spectral density *Gyx* between channel x and channel y:

Step 1, compute the Fourier transform of input signal x(k) and response signal y(k):

$$Sx = \sum_{n=0}^{N-1} x(k)\, w(k) e^{-j2\pi kn/N}$$

$$Sy = \sum_{n=0}^{N-1} y(k)\, w(k) e^{-j2\pi kn/N}$$

Step 2, compute the instantaneous cross power spectral density

$Syx = Sx^{*}\ Sy\,T$

Step 2, average the *M* frames of *Sxx* to get averaged PSD *Gxx*

Gyx' = Average (Syx)

Step 3, Compute the energy correction and double the value for the single-sided spectra

Gyx = 2 Gyx' / EnergyCorr

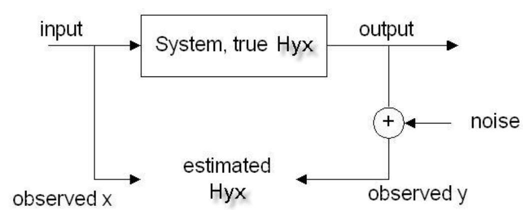
Frequency Response and Coherence Function

The cross power spectrum method is often used for estimating the frequency response function (FRF) between channel x and channel y. The equation is:

$$H_{yx} = G_{yx} / G_{xx}$$

where *Gyx* is the averaged cross-spectrum between the input channel x and output channel y. *Gxx* is the averaged auto-spectrum of the input. Either power spectrum, power spectral density or energy spectral density can be used to compute the FRF because of the linear relationship between input and output.

Using the cross-power spectrum method instead of simply dividing the linear spectra between input and output to calculate the FRF will reduce the effect of the noise at the output measurement end, as shown below.



■ Figure 9. Frequency response function computation.

The frequency response function has a complex data format. You can view it in real and imaginary or magnitude and phase display format.

The coherence function is defined as:

$$C_{yx}^{\ 2} = \frac{|G_{yx}|^2}{G_{xx}G_{yy}}$$

where *Gyx* is the averaged cross-spectrum between the input channel x and output channel y. *Gxx* and *Gyy* are the averaged auto-spectrum of the input and output. Either power spectrum, power spectral density or energy spectral density can be used here because of the linear relationship between input and output so that any multiplier factors will be cancelled out.

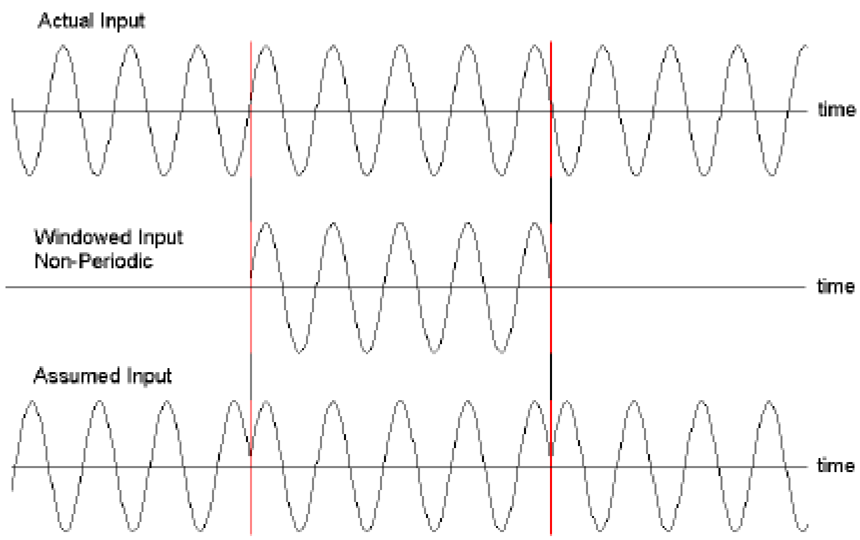
Coherence is a statistical measure of the how much of the output is caused by the input. The maximum coherence is 1.0 when the output is perfectly correlated with the input and zero when there is no correlation between input and output. Coherence is calculated by an average of multiple frames. When it is computed for only one frame, then the coherence function has a meaningless result of 1.0 due to the estimation error of the coherence function.

The coherence function is a non-dimensional real function in the frequency domain. You can only view it in the real format.

Data Window Selection

Leakage Effect

Windowing of a simple signal, like a sine wave may cause its Fourier transform to have non-zero values (commonly called leakage) at frequencies other than the frequency of this sine. This leakage effect tends to be worst (highest) near sine frequency and least at frequencies farthest from sine frequency. The effect of leakage can easily be depicted in the time domain when a signal is truncated. As shown in the picture, after data windowing, truncation distorted the time signal significantly, hence causing a distortion in its frequency domain.



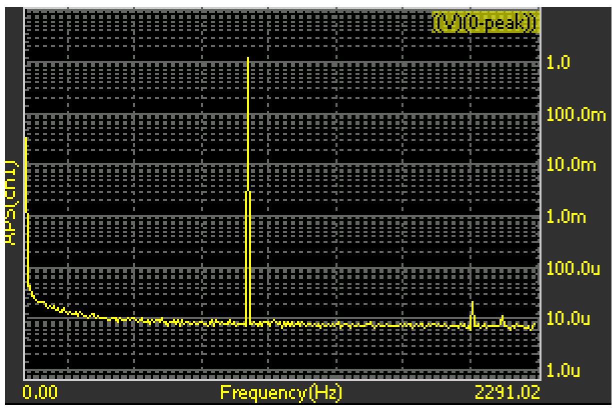
- Figure 10. Illustration of a non-periodic signal resulting from sampling.

If there are two sinusoids, with different frequencies, leakage can interfere with the ability to distinguish them spectrally. If their frequencies are dissimilar, then the leakage interferes when one sinusoid is much smaller in amplitude than the other. That is, its spectral component can be hidden or masked by the leakage from the larger component. But when the frequencies are near each other, the leakage can be sufficient to interfere even when the sinusoids are equal strength; that is, they become undetectable.

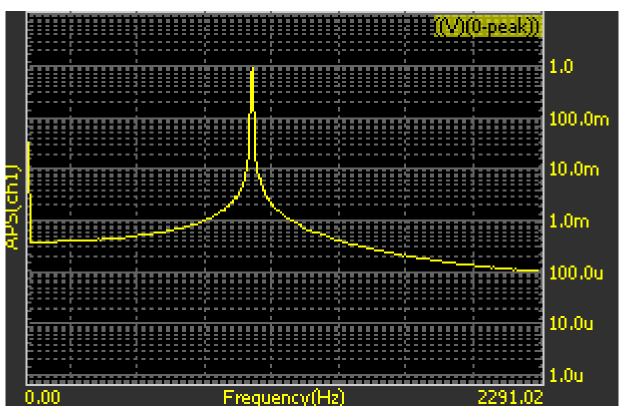
There are two possible scenarios that leakage does not occur. The first is that when the whole time capture is long enough to cover the complete duration of the signals. This can occur with short transient signals. For example in a hammer test, if the time capture is long enough it may extend to the point where the signal decays to zero. In this case, data window is not needed.

The second case is when a periodic signal is sampled at such a sampling rate that is perfectly synchronized with the signal period, so that with a block of capture, an integer number of cycles of the signal are always acquired. For example, if a sine wave has a frequency of 1000Hz and the sampling rate is set to 8000Hz. Each sine cycle would have 8 integer points. If 1024 data points are acquired then 128 complete cycles of the signal are captured. In this case, with no window applied you still can get a leakage-free spectrum.

Figure 11 shows a sine signal at 1000 Hz with no leakage resulting in a sharp spike. Figure 12 shows the spectrum of a 1010 Hz signal with significant leakage resulting in a wide peak. The spectrum has significant energy outside the narrow 1010 Hz frequency. It is said that the energy leaks out into the surrounding frequencies.

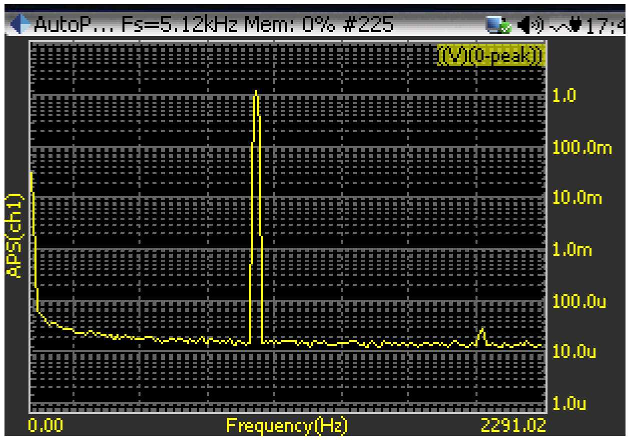


■ Figure 11. Sine spectrum with no leakage.



■ Figure 12. Sine spectrum with significant leakage.

Several windowing functions have been developed to reduce the leakage effect. The picture below shows a Flattop window applied to the same sine signal with frequency 1010Hz:



- Figure 13. Sine spectrum with Flattop windowing function.

When Flattop window is used, the leakage effect is reduced. Both the sine peak and noise floor can be seen now. However, such data windowing operation also makes the spectrum peak “fatter” and less accurate. In the rest of the sections we will discuss how to choose different data windows.

Data Window Formula

In this section, we will describe the math formula that we used for each data window.

Uniform window (rectangular)

$$w(k) = 1.0$$

Uniform is the same as no window function.

Hamming window

$$w(k) = 0.53836 - 0.46164 \cos\left(\frac{2\pi k}{N-1}\right)$$

Hann window

$$w(k) = 0.5 - 0.5 \cos\left(\frac{2\pi k}{N-1}\right)$$

The Hann and Hamming windows are in the family known as "raised cosine" windows, are respectively named after Julius von Hann and Richard Hamming. The term "Hanning window" is sometimes used to refer to the Hann window, but is ambiguous as it is easily confused with Hamming window.

Blackman window

$$w(k) = 0.84 - 0.5\cos\frac{2\pi k}{N-1} + 0.08\cos\frac{4\pi k}{N-1} \qquad for\ k = 0 \sim N-1$$

Flattop window

$$w(k) = 1 - 1.93\cos\frac{2\pi k}{N-1} + 1.29\cos\frac{4\pi k}{N-1} - 0.388\cos\frac{6\pi k}{N-1} + 0.032\cos\frac{8\pi k}{N-1} \qquad for\ k = 0 \sim N-1$$

Kaiser Bessel window

$$w(k) = 1.0 - 1.24\cos\frac{2\pi k}{N-1} + 0.244\cos\frac{4\pi k}{N-1} + 0.00305\cos\frac{6\pi k}{N-1} \qquad for\ k = 0 \sim N-1$$

Exponential Window

The shape of the exponential window is that of a decaying exponential. The following equation defines the exponential window.

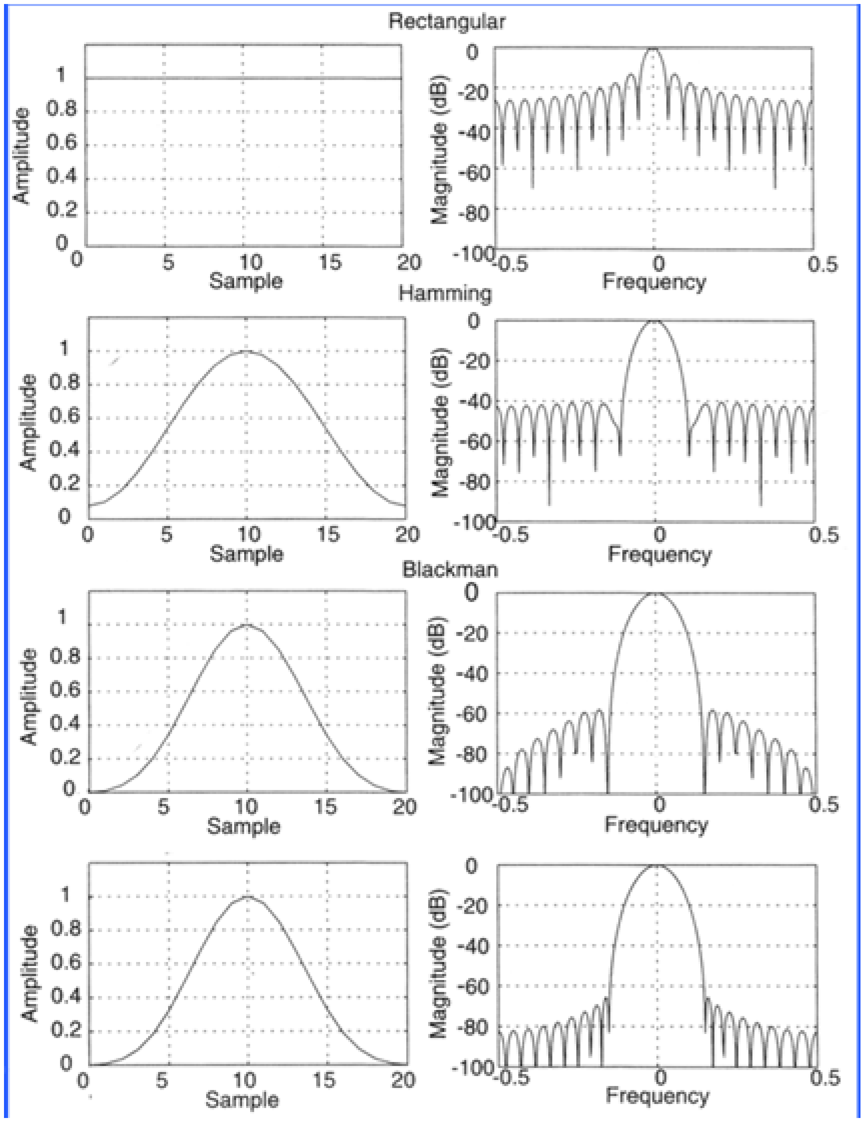
$$w(k) = e^{\left(\frac{k \ln(final)}{N-1}\right)}$$

$$for\ k = 0 \sim N-1$$

where N is the length of the window, w(k)is the window value, and ***final*** is the final value of the whole sequence. The initial value of the window is one and gradually decays toward zero.

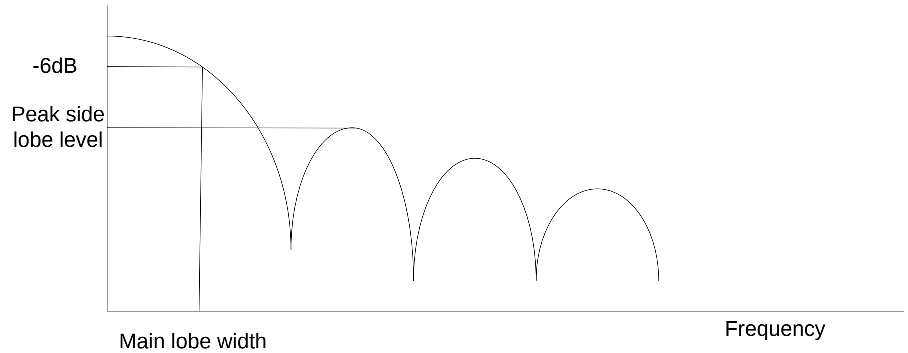
How to Choose the Right Data Window

In this section we will discuss how to choose the data window. Figure 14 shows the spectral shape of four typical windows corresponding to their time waveform.



- Figure 14. Spectral shape of common windowing functions.

It can be seen that the spectral shape of the data window is always symmetric. The spectral shape can be described as a main lobe and several side lobes.



■ Figure 15. Window frequency response showing main lobe and side lobes.

The following table lists the characteristics of several data windows.

Frequency Characteristics of Data Windows

Window	–3 dB Main Lobe Width (bins)	–6 dB Main Lobe Width (bins)	Maximum Side Lobe Level (dB)
Uniform (none)	0.9	1.2	–13
Hanning	1.4	2.0	–32
Hamming	1.3	1.8	–43
Blackman	1.6	2.3	–58
Flattop	2.9	3.6	–44

Main Lobe

The center of the main lobe of a window occurs at each frequency component of the time-domain signal. By convention, to characterize the shape of the main lobe, the widths of the main lobe at –3 dB and –6 dB below the main lobe peak describe the width of the main lobe. The unit of measure for the main lobe width is FFT bins or frequency lines.

The width of the main lobe of the window spectrum limits the frequency resolution of the windowed signal. Therefore, the ability to distinguish two closely spaced frequency components increases as the main lobe of the smoothing window narrows. As the main lobe narrows and spectral resolution improves, the window energy spreads into its side lobes, increasing spectral leakage and decreasing amplitude accuracy. A trade-off occurs between amplitude accuracy and spectral resolution.

Side Lobes

Side lobes occur on each side of the main lobe and approach zero at multiples of f_s/N from the main lobe. The side lobe characteristics of the smoothing window directly affect the extent to which adjacent frequency components leak into adjacent frequency bins. The side lobe response of a strong sinusoidal signal can overpower the main lobe response of a nearby weak sinusoidal signal. Maximum side lobe level and side lobe roll-off rate characterize the side lobes of a smoothing window. The maximum side lobe level is the largest side lobe level in decibels relative to the main lobe peak gain.

Guidelines of Choosing Data Windows

If a measurement can be made so that no leakage effect will occur, then do not apply any window (in the software, select Uniform.). As discussed before, this only occurs when the time capture is long enough to cover the whole transient range, or when the signal is exactly periodic in the time frame.

If the goal of the analysis is to discriminate two or multiple sine waves in the frequency domain, spectral resolution is very critical. For such application, choose a data window with very narrow main slope. Hanning is a good choice.

If the goal of the analysis is to determine the amplitude reading of a periodic signal, i.e., to read EU_{pk}, EU_{pkpk}, EU_{rms} or EU_{rms}^2, the amplitude accuracy of a single frequency component is more important than the exact location of the component in a given frequency bin, choose a window with a wide main lobe. Flattop window is often used.

If you are analyzing transient signals such as impact and response signals, it is better not to use the spectral windows because these windows attenuate important information at the beginning of the sample block. Instead, use the Force and Exponential windows. A Force window is useful in analyzing shock stimuli because it removes stray signals at the end of the signal. The Exponential window is useful for analyzing transient response signals because it damps the end of the signal, ensuring that the signal fully decays by the end of the sample block.

If the nature of the data is has a random nature or unknown, choose Hanning window.

Averaging Techniques

Averaging is widely used in spectral measurements. It improves the measurement and analysis of signals that are purely random or mixed random and periodic. Averaged measurements can yield either higher signal-to-noise ratios or improved statistical accuracy.

Typically, three types of averaging methods are available in DSA products. They are:

Linear Averaging, **Exponential Averaging**, and **Peak-Hold**

Linear Averaging

In linear averaging, each set of data (a record) contributes equally to the average. The value at any point in the linear average in given by the equation:

$$Averaged = \frac{Sum\ of\ Records}{N}$$

N is the total number of the records. The advantage of this averaging method is that it is faster to compute and the result is un-biased. However, this method is suitable only for analyzing short signal records or stationary signals, since the average tends to stabilize. The contribution of new records eventually will cease to change the value of the average.

Usually, a target average number is defined. The algorithm is made so that before the target average number reaches, the process can be stopped and the averaged result can still be used.

When the specified target averaging number is reached, the instrument usually will stop the acquisition and wait for the instruction for another collection of data acquisition.

Moving Linear Averaging

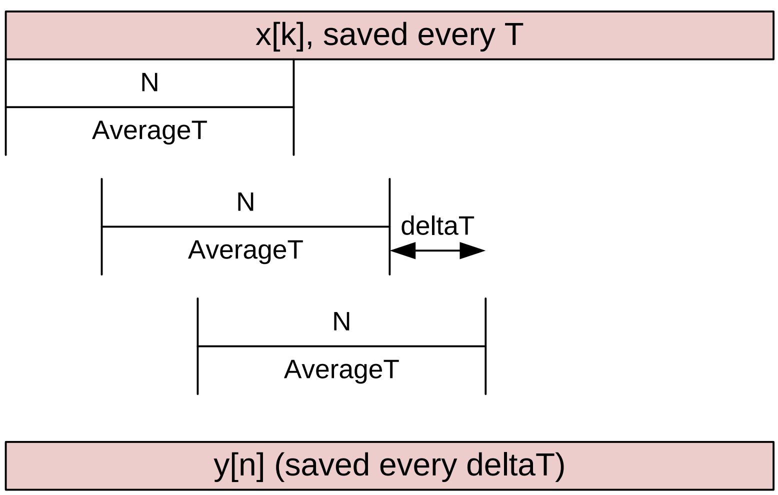
In a regular Linear Average, the data rate of the output of the averaging operator is only 1/N of that of the original signal. Therefore more averages takes longer to compute. Thus averaging will increase the time of the measurement. To reduce the time a Moving Linear Averaging can be used. Moving Linear Averaging uses overlapped input data points to generate more than 1/N results within a period of time. Moving linear average has the advantage that the resulted trace update time can be much shorter than the linear averaging period.

Moving Linear Average is computed by

$$y[n] = \frac{1}{N}\sum_{j=0}^{N-1} x[n-j]$$

Where x[k] is the input data, with sampling rate of T, y[n] is the output data, with Trace Update rate deltaT, AverageT is the period of Linear Average and ,N is the total samples used for Linear Average. N = AverageT/T

The Moving Linear Averaging is illustrated in Figure 16. Assume the averaging period is *AverageT* but the progressive time for each averaging operation is *deltaT*, the output buffer will have a data rage of *deltaT* instead of *AverageT*.



- Figure 16. Illustration of moving linear average.

The Moving Linear Average is useful in many situations. For example, in Sound Level Meter, Leq is defined as a linear averaged value over a long period of time, say 1 second to 24 hours. Assume the AverageT is 1 hour, without moving linear average, in a 24 hours period, you can only get 24 readings. This is not very useful. With moving averaging, you can get the readings in every 1 second, for the linear averaging of the past 1 hour.

Exponential Averaging

In exponential averaging, records do not contribute equally to the average. A new record is weighted more heavily than old ones. The value at any point in the exponential average is given by:

$$y[n] = y[n-1] * (1-\alpha) + x[n] * \alpha$$

where $y[n]$ is the *nth* average and $x[n]$ is the nth new record. α is the weighting coefficient. Usually α is defined as 1/(Number of Averaging). For example in the instrument, if the Number of Averaging is set to 3 and the averaging type is selected as exponential averaging, then $\alpha = 1/3$

The advantage of this averaging method is that it can be used indefinitely. That is, the average will not converge to some value and stay there, as is the case with linear averaging. The average will dynamically respond to the influence of new records and gradually ignore the effects of old records.

Exponential averaging simulates the analog filter smoothing process. It will not reset when a specified averaging number is reached.

The drawback of the exponential averaging is that a large value may embed too much memory into the average result. If there is a transient large value as input, it may take a long time for y[n] to decay. On the contrary, the contribution of small input value of x[n] will have little impact to the averaged output. Therefore, exponential average fits a stable signal better than a signal with large fluctuations.

Peak-Hold

This method, technically speaking, does not involve averaging in the strict sense of the word. Instead, the “average” produced by the peak hold method produces a record that at any point represents the maximum envelope among all the component records. The equation for a peak-hold is

$$y[n] = \mathrm{MAX}_{\mathrm{j=0}}^{\mathrm{N-1}}(x[n-j])$$

Peak-hold is useful for maintaining a record of the highest value attained at each point throughout the sequence of ensembles. Peak-Hold is not a linear math operation therefore it should be used carefully. It is acceptable to use Peak-Hold in auto-power spectrum measurement but you would not get meaningful results for FRF or Coherence measurement using Peak-Hold.

Peak-hold averaging will reset after a specified averaging number is reached.

Linear Spectrum versus Power Spectrum Averaging

Averaging can be applied to either linear spectrum or power spectrum. If you want to reduce the spectral estimation variance, use power spectral averaging. If you want to extract repetitive or periodic small signals from a noisy signal, you can use triggered capture and average them in linear spectral domain. Linear Spectrum averaging must be performed with on a triggered event so that the time signal of one average is correlated with other similar measurements. Without time synchronizing mechanism, averaging in the Linear Spectrum domain makes no sense. Linear spectrum averaging is also called Vector averaging. It averages the complex FFT spectrum. (The real part is averaged separately from the imaginary part.) This can reduce the noise floor for random signals since they are not phase coherent from time record to time record.

Power Spectrum Averaging is also called RMS Averaging. RMS averaging computes the weighted mean of the sum of the squared magnitudes (FFT times its complex conjugate). The weighting is either linear or exponential. RMS averaging reduces fluctuations in the data but does not reduce the actual noise floor. With a sufficient number of averages, a very good approximation of the actual random noise floor can be displayed. Since RMS averaging involves magnitudes only, displaying the real or imaginary part, or phase, of an RMS average has no meaning and the power spectrum average has no phase information.

Table 1 gives a summary of the averaging methods described above.

- Table 1. Summary of Averaging Methods.

Linear Spectrum Averaging	**Power Spectrum Averaging**
No statistical spectral estimate, for deterministic signals only.	Statistical spectral estimate, for signals with random characteristics.
Signal must have periodic components.	Applicable to both pure random and mixed random/periodic signals.
Improve SNR.	Does not improve SNR.
Requires a synchronized trigger in fixed relation to the signals.	Does not require a synchronized trigger.

Spectrum Estimation Error

You may wonder how much confidence we should have when we take the spectral measurement. This is a academic topic that can go very deep. First you must classify your signal types. If you are measuring a deterministic signal, with very few averaging, the spectrum estimation can be very accurate. If the signal has a random nature, with partially random, or significant measurement noise, more averaging must be used.

Assume the time data is captured from a stationary random process and we calculate various spectra using window, FFT and averaging techniques, how much we can trust the measured spectra can be measured by a statistical quantity, *standard deviation*. Here are a few useful equations to compute the standard deviation of the spectra when linear averaging is used:

Functions being estimated	**Standard Deviation**
Auto-spectrum Gxx	$\frac{1}{\sqrt{n}}$
Cross-spectrum |Gyx|	$\frac{1}{\|Cyx\|\sqrt{n}}$
Coherence Function Cyx^2	$\frac{(1-Cyx^2)\sqrt{2}}{\|Cyx\|\sqrt{n}}$

Frequency Response Function Hyx	$\dfrac{\sqrt{(1-Cyx^2)}}{\lvert Cyx \rvert \sqrt{2n}}$

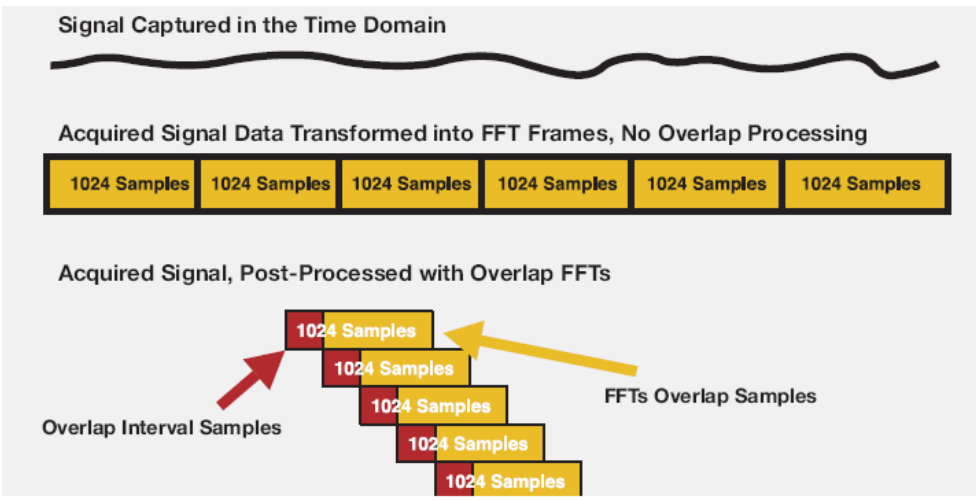
where *n* is the average number in linear averaging. The transfer function is computed in the cross-power spectrum method as presented earlier.

Assume a signal is random and has an ***expected*** power spectral density at 0.1 V^2/Hz. The goal of a measurement is to average a few power spectra and to estimate such an expected value. If the average number is 1, meaning, with no average, the standard deviation of the error of such a measurement will be 100%. When we average two frames of auto power spectra, the standard deviation of the error will become $\frac{1}{\sqrt{2}} = 70.7\%$ When the average number is increased to 100, the standard deviation of the error of the reading is 10%. This means that the reading is likely in the neighborhood of (0.1±0.01) V^2/Hz

Now if this signal has a deterministic nature, say a sine wave, the spectral estimation error will only be applied to the random portion, i.e., the noisy portion, of this signal.

Overlap Processing

To increase the speed of spectral calculation, overlap processing can be used to reduce the measurement time. The diagram below shows how the overlap is realized.



■ Figure 17. Illustration of overlap processing.

As shown in this picture, when a frame of new data is acquired after passing the Acquisition Mode control, only a portion of the new data will be used. Overlap calculation will speed up the calculation with the same target average number. The percentage of overlap is called overlap ratio. 25% overlap means 25% of the old data will be used for each spectral processing. 0% overlap means that no old data will be reused.

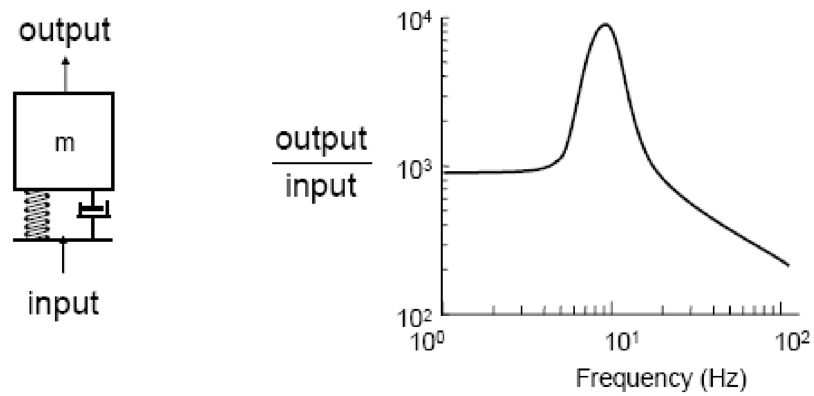
Overlap processing can improve the accuracy of spectral estimation. This is because when a data window is applied, some useful information is attenuated by the data window on two ends of each block. However, it is not true that the higher the overlap ratio the higher the spectral estimation accuracy. For Hanning window, when the overlap ratio is more than 50%, the estimation accuracy of the spectra will not be improved.

Another advantage to apply overlap processing is that it helps to update the display more quickly.

Single Degree of Freedom System

This section briefly discusses the single degree of freedom (SDOF) system as background for the frequency response function and damping estimation methods.

The vibration nature of a mechanical structure can be decomposed into multiple, relatively independent Single-Degree-Of-Freedom systems. Each SDOF system can be modeled as a mass fixed to the ground by a spring and a damper in parallel as shown in Figure 18. The frequency response function (FRF) of this mechanical system is also shown.



- Figure 18. SDOF system and their frequency response.

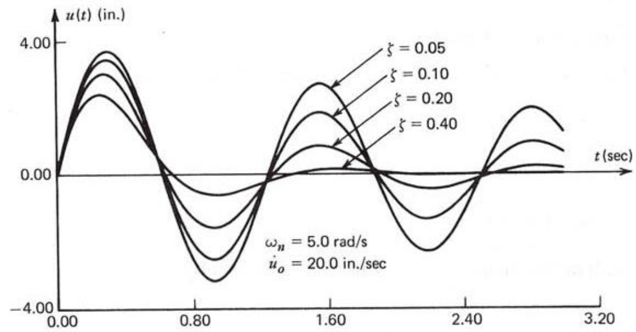
The differential equation of motion for this system is given by

$$m\ddot{x} + c\dot{x} + kx = f(t)$$

The natural frequency ω_n and damping ratio ζ can be calculated from the system parameters as

$$\omega_n^2 = \frac{k}{m}, \quad and \quad 2\zeta\omega_n = \frac{c}{m}$$

where m is the mass, k is the spring stiffness and c is the damping coefficient.
The natural frequency, ω_n , is in units of radians per second (rad/s). The typical units displayed on a digital signal analyzer are in Hertz (Hz). The damping ratio, ζ, can also be represented as a percent of critical damping – the damping level at which the system experiences no oscillation. This is the more common understanding of modal damping.. Figure 18 illustrates the response of a SDOF system to a transient excitation showing the effect of different damping ratios.



- Figure 19. Step response of a SDOF system with different damping ratios.

A SDOF system with light damping factor will have longer oscillation in a transient process. This is why the exponential window may be chosen to reduce the leakage effect in its spectral analysis.

dB and Linear Magnitude

Most often, amplitude or power spectra are shown in the logarithmic unit decibels (dB). Using this unit of measure, it is easy to view wide dynamic ranges; that is, it is easy to see small signal components in the presence of large ones. The decibel is a unit of ratio and is computed as follows.

$$dB = 10\log_{10}(Power/Pref)$$

where Power is the measured power and P_{ref} is the reference power.

Use the following equation to compute the ratio in decibels from amplitude values.

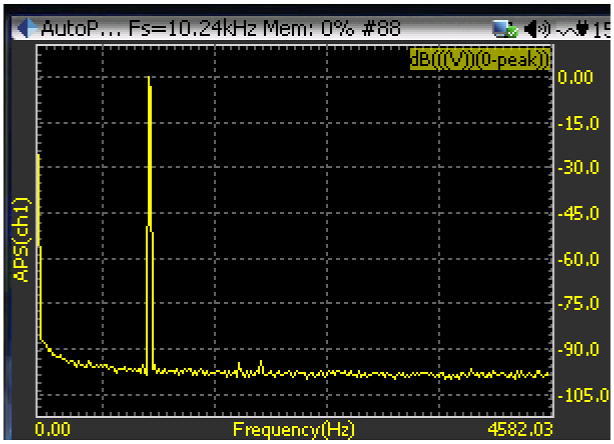
$$dB = 20\log_{10}(Ampl/A_{ref})$$

where Ampl is the measured amplitude and A_{ref} is the reference amplitude.

When using amplitude or power as the amplitude-squared of the same signal, the resulting decibel level is exactly the same. Multiplying the decibel ratio by two is equivalent to having a squared ratio. Therefore, you obtain the same decibel level and display regardless of whether you use the amplitude or power spectrum.

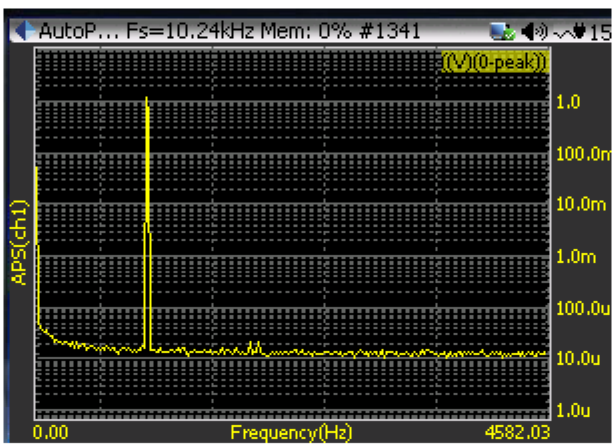
As shown in the preceding equations for power and amplitude, you must supply a reference for a measure in decibels. This reference then corresponds to the 0 dB level. Different conventions are used for different types of signals. A common convention is to use the reference 1 Vrms for amplitude or 1 Vrms squared for power, yielding a unit in dBV or dBVrms. In this case, 1 Vrms corresponds to 0 dB. Another common form of dB is dBm, which corresponds to a reference of 1 mW into a load of 50 Ω for radio frequencies where 0 dB is 0.22 Vrms, or 600 Ω for audio frequencies where 0 dB is 0.78 Vrms.

The picture below shows a sine wave with 1V amplitude displayed in dB. Because the reference is 1Vpk , it shows the peak value of this sine wave as 0dB.



■ Figure 20. Show a 1Vpk sine signal in frequency domain with dB scaling.

Another display format is called Log, or LogMag. The Log display shows the signal scaled logarithmically with the grid values and cursor readings in actual engineering value. The picture below shows the same signal in LogMag.



■ Figure 21. A 1Vpk sine signal in frequency domain with LogMag scaling.

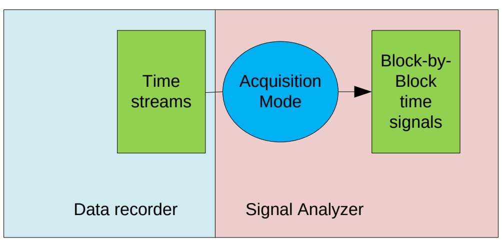
When dB reference is not specified, the dB reference is 1.0 engineering unit. In acoustics application, the dB reference for the sound pressure value is set to 20uPa. The same input signal will result in different dB readings when dB reference is changed.

TRANSIENT CAPTURE AND HAMMER TESTING

Transient Capture

In the previous Chapters of this manual, we have discussed how the acquisition mode can be defined in the CSA Editor and selected on the CoCo device. This chapter will demonstrate how to use CoCo to conduct hammer testing. Hammer testing refers to impact or bump testing that is conducted using an impact hammer to apply an impulsive force excitation to a test article while measuring the response excitation from an accelerometer or other sensor. This type of measurement is a transient event that usually requires triggering, averaging and windowing. First, let's briefly review the Transient Capture function on CoCo.

Transient Capture is one of the most common used functions for dynamic data acquisition. In CoCo the Transient Capture is implemented by setting up the Acquisition Mode. Acquisition Mode defines how to transform the time streams into block by block time signals. It sets the trigger and the overlapping processing. Before the Acquisition Mode stage, the instrument acts as a data recorder while after the Acquisition Mode, it is acts as a signal analyzer.



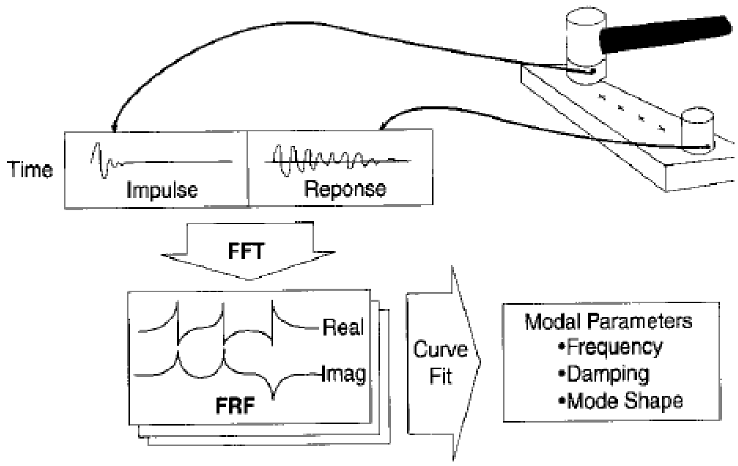
- Figure 22. Transient capture operation on CoCo.

Besides Acquisition Mode, you must first enable at least one time stream as a trigger candidate in the CSA Editor. Trigger candidates are those time streams that can be selected as a trigger source. The names of these trigger candidates will be passed to the CoCo. During runtime, one of the trigger source candidates must be selected as the trigger source.

Impact Hammer Testing

Typically impact hammer testing is conducted with a signal analyzer to measure FRFs of the device under test. The FRFs can be used to determine the modal properties of the device such as the natural frequencies and damping ratios. In addition the data can be exported to third party modal analysis software to compute mode shapes.

An impact hammer test is the most common method of measuring FRFs. The hammer imparts a transient impulsive force excitation to the device. The impact is intended to excite a wide range of frequencies so that the DSA can measure the vibration of the device across this range of frequencies. The bandwidth or frequency content of the excitation input depends on the size and type of impact hammer that is used. The dynamic force signal is recorded by the DSA. After the impact, the device vibrations are measured with one or more accelerometers or other sensor and recorded by the DSA. The DSA then computes the FRF by comparing the force excitation and the response acceleration signals. Impact testing is depicted in Figure 23.

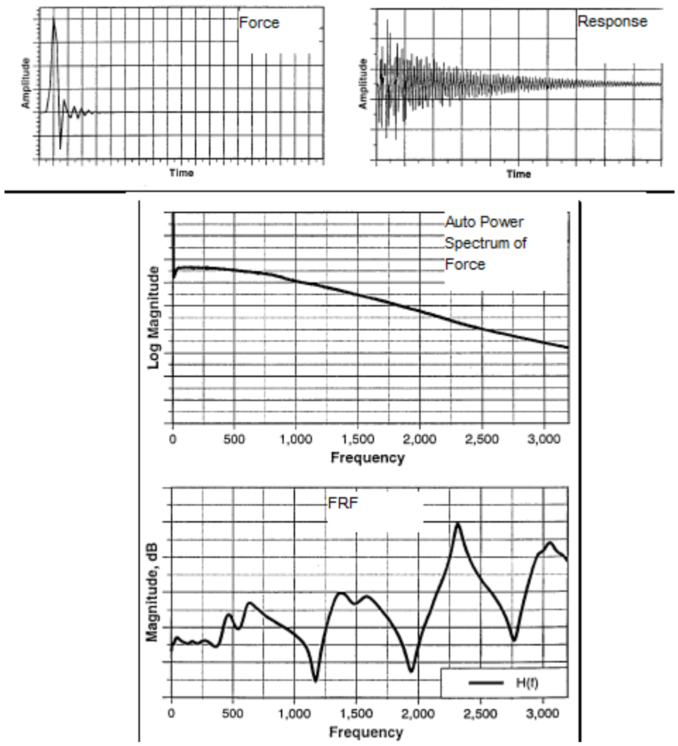


■ Figure 23. Illustration of a typical impact test and signal processing.

The following equipment is required to perform an impact test:

1. An impact hammer to excite the structure. With CoCo we recommend using an impact hammer with IEPE output, which allows the hammer to be connected directly to the analyzer without extra signal conditioning.
2. One or multiple accelerometers that are fixed on the structure. Again, IEPE accelerometers can be used directly with CoCo without additional signal conditioning.
3. Coco Signal Analyzer
4. The CoCo can be used to extract the resonance frequencies and damping factors of the structure. In addition third party software can be used to extract modal shapes and animate the vibration modes.

A wide variety of structures and machines can be impact tested. Of course, different sized hammers are required to provide the appropriate impact force, depending on the size of the structure; small hammers for small structures, large hammers for large structures. Realistic signals from a typical impact test are shown in Figure 10.



■ Figure 24. Typical impact test data. Top left shows excitation force impulse time signal, top right shows response acceleration time signal and bottom shows FRF spectrum.

Impact Test Analyzer Settings

The following settings are used for impact testing.

1. **Trigger Setup** including trigger level and pre-trigger delay are used to capture the transient signal for FRF processing. It is important to capture the entire short transient signal it in the sampling window of the FFT analyzer. To insure that the entire signal is captured, the analyzer must be able to capture the impulse and impulse response signals *prior to the occurrence of the impulse* with the pre-trigger.

2. **Force & Exponential Windows**. Two common time domain windows that are used in impact testing are the force and exponential windows. These windows are applied to the signals after they are sampled, but before the FFT is computed in the analyzer.

   The *force window is used to remove noise* from the impulse (force) signal. Ideally, an impulse signal is non-zero for a small portion of the sampling window, and zero for the remainder of the window time period. Any non-zero data following the impulse signal in the sampling window is assumed to be measurement noise. CoCo has a unique way to implement the force window. This was discussed in the data windowing section in the previous chapter.

   The exponential window is applied to the impulse response signal. The *exponential window is used to reduce leakage* in the spectrum of the response.

3. **Accept/Reject**: Because accurate impact testing results depend on the skill of the operator, FRF measurements should be made with **averaging**, a standard capability in all modern FFT analyzers. FRFs should be measured using at least 4 impacts per measurement. Since one or two of the impacts during the measurement process may be bad hits (too hard causing saturation, too soft causing poor coherence or a double hit causing distortion in the spectrum), an FFT analyzer designed for impact testing should have the ability to accept or reject the result of each impact after inspecting the impact signals. An *accept/reject* capability saves a lot of time during impact testing since you don't have to redo all measurements in the averaging process after one bad hit.

4. **Modal Damping Estimation**. The *width of the resonance peak* is a measure of modal damping. The resonance peak width should also be the same for all FRF measurements, meaning that *modal damping is the same in every FRF measurement*. A good analyzer should provide an accurate damping factor estimate. CoCo uses a curve fitting algorithm to estimate the damping factor. The algorithm reduces the inaccuracy caused by the poor spectrum resolution or noise.

5. **Modal Frequency estimation**. The analyzer must provide capability of estimating the resonance frequencies. CoCo uses an algorithm to identify the resonance frequencies based on the FRF.

References

To understand the topics of this article, we found the following three books to be very helpful:

1. Julius S. Bendat and Allan G. Piersol, **Random Data, Analysis and Measurement Procedures**, 2nd Edition, Wiley-Interscience, New York, 1986.

2. Julius S. Bendat and Allan G. Piesol, **Engineering Applications of Correlation and Spectral Analysis**, 2nd Edition, Wiley-Interscience, New York, 1993.

3. Sanjit K. Mitra and James F. Kaiser, Ed. **Handbook for Digital Signal Processing**, Wiley-Interscience, New York, 1993.

Advanced Dynamic Signal Analysis

James Zhuge, Ph.D.
Crystal Instruments Corporation
4633 Old Ironsides Drive, Suite 304
Santa Clara, CA 95054, USA
www.go-ci.com

(Part of CoCo-80 User's Manual)

Table of Contents

Table of Figures

1. INTRODUCTION

CoCo is a hardware platform that can run in either DSA (Dynamic Signal Analyzer) or VDC (Vibration Data Collector) mode.

This CoCo Advanced DSA Users Manual discusses the theory, EDM software and CoCo operation for the optional advanced dynamic signal analysis features including:

- Swept Sine Analysis
- Acoustic Data Acquisition: Octave Analysis and Sound Level Meter
- Order Tracking
- Sock Response Spectrum Analysis
- Automated Test and Limit Test
- Real Time Digital Filters
- Histogram and Statistics Measures
- Miscellaneous Operations.

Each topic includes a detailed description of the general theory including mathematical formulation application topics, instructions on how to create a CSA file using the EDM software, and detailed instructions on how to setup the CoCo hardware and make a measurement.

This document references the CoCo Basic Users Manual. This separate document gives details on the basic operation of the CoCo hardware and details on basic frequency spectrum measurements including theory, EDM software setup and CoCo operations. We strongly recommend that you read the CoCo Basic Users Manual first before proceeding to this document.

2. SWEPT SINE MEASUREMENTS

This section describes the swept sine measurement capabilities of the CoCo. It includes both theoretical background and application information. The Swept Sine Testing option of CoCo-80 has several unique advantages over similar products in the market, including:

- The measurement channels with very high dynamic range ensure continuous test over high dynamic range UUT (Unit Under Test). It is common to achieve 130~150dB dynamic range with CoCo-80.
- Special tracking filters realized based on TVDFT (Time Variant Discrete Fourier Transform) provide excellent spectrum estimation.
- Special algorithm enables test in wide frequency range. The result of both low and high frequency testing is excellent.
- Time domain signals are always available for viewing and recording.
- Log, Linear sweep modes are available.
- Auto-gain adjustment with closed-loop control capability to prevent input range overloading.

Sine Signal Used for Testing

Broadband random, sine, step or transient signals are widely used as excitation signals in test and measurement applications. Figure 1 illustrates that an excitation signal x, can be applied to a UUT (Unit Under Test) and generate one or multiple responses denoted by y. The relationship between the input and output is known as the transfer function or frequency response function and represented by H(y,x). In general a transfer function is a complex function that modifies the input signal magnitude and phase as the excitation frequency changes.

■ Figure 1 Left: a UUT with one response; Right: a UUT with two responses.

With swept sine excitation, the characteristics of the UUT system can be measured experimentally. These characteristics include:

- Frequency Response Function (FRF), which is described by:
- Gain as a function of frequency
- Phase as a function of frequency
- Resonant Frequencies
- Damping factors
- Total Harmonic Distortion
- Non-linearity
- Others

Frequency response can be measured using the FFT, cross power spectral method with broadband random excitation. Broadband excitation can be a true random noise signal with Gaussian distribution, or a pseudo-random signal of which the amplitude distribution can be defined by the user. The term "***broadband***" may be misleading, as a well implemented random excitation signal should be frequency band-limited and controlled by the upper limit of the analysis frequency range. That is, the excitation need not excite frequencies above that which can be measured by the instrument. The CoCo random generator will only generate random signals up to the analysis frequency range. This will also concentrate the excitation energy on the useful frequency range.

The advantage of using broadband random excitation is that it can excite the whole frequency range in a short period of time so the total testing time is less. The drawback of broadband excitation is that its frequency content is spread over a wide range within a short duration. The energy contribution of the excitation at each frequency point will be much less than the total signal energy (roughly, it is -30~ -50dB less than the total). Even with a large number of averaging in the FRF estimation, the broadband signal will not effectively measure the extreme dynamic characteristics of the UUT.

Swept sine measurements, on the other hand, can optimize the measurement at *each frequency point.* Since the excitation is a sine wave, all of its energy is concentrated at a single frequency, eliminating the dynamic range penalty in a broadband excitation. In addition, if the frequency response magnitude drops, the tracking filter of the response can help to pick up extremely small sine signals. Simply optimizing the input range at each frequency can extend the dynamic range of the measurement to beyond 150 dB.

Introducing Sweeping Sine

A sine signal with a fixed frequency f_0 can be expressed as:

$$\mathrm{x(t)} = \sin(2\pi f_0 t)$$

where t represents time. A sweeping sine signal has a changing frequency that is usually bound by two limits. The frequency change can be either in the linear scale or logarithmic scale based on different user requirements. The swept sine signal can be defined by the following parameters:

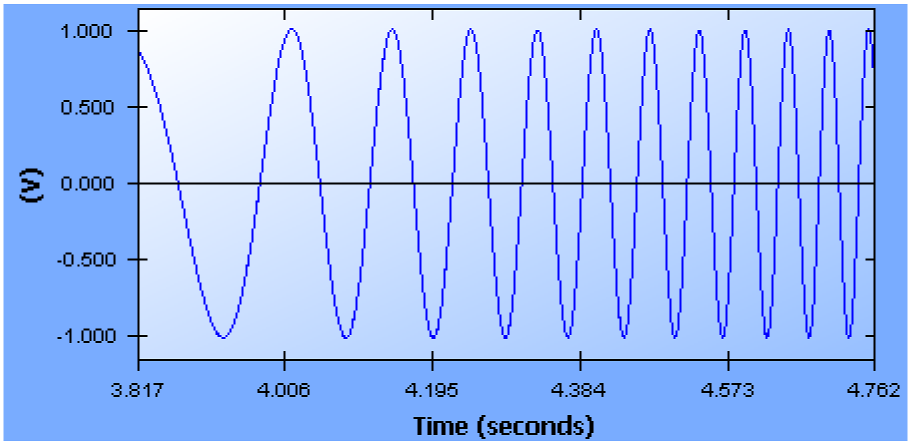
- The low frequency boundary, which is simply called Low Frequency or f_{Low}
- The high frequency boundary, which is simply called High Frequency or f_{High}
- The sweeping mode, either logarithmic or linear
- The sweeping speed, in either octave/min if the sweep mode is logarithmic, or in Hz/Sec if the sweeping mode is linear
- The amplitude of the sine signal, A(f, t), which can be a constant or a variable of time and frequency.

$$\mathrm{x(t)} = \mathrm{A(f,t)} * \sin(2\pi\left(f\left(f_{Low}, f_{High}, Speed\right)\right)t)$$

The instantaneous frequency $f\left(f_{Low}, f_{High}, Speed\right)$ represents the current frequency of the sweeping sine. It is a changing variable and usually displayed on the screen as ***Sweeping Frequency***.

The sweeping frequency can also be manually controlled during the test with the ***Hold, Resume, Jump or Pause*** **controls**.

Unlike some DSA products which use swept sine test with ***multiple discrete stepped sine tones in a sequence***, the CI swept sine test uses a true digital synthesizing technique to generate sine sweeps with extreme analog-like smooth transition from one frequency to another. This ensures that there are no sharp transitions during the test that might “shock” the UUT. The picture below shows a typical swept sine signal with 1.0 Vpk.



■ Figure 2. Typical digitally synthesized swept sine signal.

Sweeping Mode: Logarithmic or Linear

A swept sine can sweep in either linear or logarithmic mode. Linear sweep means the frequency will change at a constant speed, with units of *Hz/sec*. In this case the sweep rate is constant and the same at all frequencies.

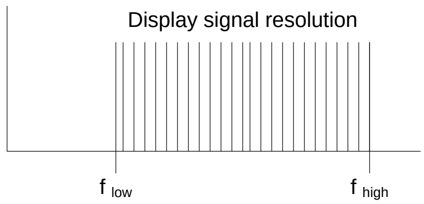
Alternatively, the sweeping mode can be set as logarithmic or Log. In Log mode, the sweeping speed is slower at low frequencies and fast at higher frequencies. In Log mode, the sweeping speed units are in *Octave/Min*. *1.0 Oct/min* means that the frequency will take one minute to double from 1kHz to 2kHz, or from 100Hz to 200Hz, or from 0.5Hz to 1.0Hz.

Most testing specifications ask for logarithmic sweeping for two reasons. The first is due to the fact that it takes longer to measure one or multiple sine cycles at low frequency than at high frequency. The second is that most mechanical and electrical systems exhibit characteristics that are better described in logarithmic frequency scale. This is because dynamics such as resonant frequencies occur over large frequencies spans: some at low frequencies and some at high frequencies. If linear sweeping is adopted, you may find that whatever the speed you choose, it is either too slow in the high frequency end or too fast in the low end. With Log sweeping mode, this problem is solved.

On the CoCo, once the sweeping mode is set to either Linear or Log in a test, the frequency distribution of the display signals will be set to linear or logarithmic accordingly. This will be discussed in the following section about the display resolution. The sweeping speed unit will also be set to either Hz/Sec. or Oct/Min automatically.

Resolution of Display Signals

In the CoCo the sweeping sine signal is point-by-point digitally synthesized. It has “infinitely” fine resolution in frequency transition. It does not jump from one frequency to another. The user may wonder how the sweeping signal is displayed. The user first needs to set the size of the displaying signals, say 1024 or 2048. The CoCo will distribute the frequency bins between f_{Low}, f_{High}. In Linear mode, the frequency spacing between two adjacent lines is represented by the frequency resolution; In Log mode, the frequency spacing between two adjacent lines of the signal will be represented by a ratio.



For example, if a linear sweep is defined with $f_{Low} = 100Hz$; $f_{High=}1000Hz,\ Signal\ Size = 1024$, then the first line of the signal will be allocated to 100Hz, the last to 1000Hz. The frequency bins of the signals will be evenly distributed with frequency resolution of

(1000-100)/(1024-1) = 0.879765Hz

If a logarithmic sweep is defined with **Log Sweep Mode**: $f_{Low} = 100Hz$; $f_{High=}1000Hz,$ $Signal\ Size = 1024$ Then the first line of the signal will be allocated to 100Hz, the last to 1000Hz. The Frequency resolution will be represented by a ratio as:

$$1000\text{ Hz} = 100\text{ Hz} * \text{ratio}^{(1024-1)}$$

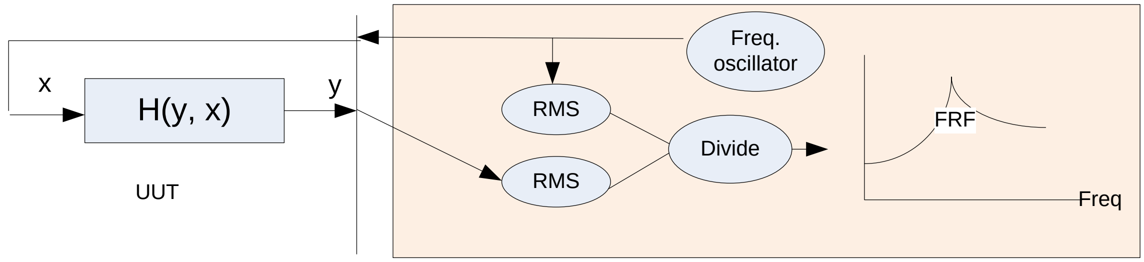
$$ratio = \left(\frac{1000}{100}\right)^{\frac{1}{1024-1}} = 1.00225335$$

This means that if the first line is at frequency 100Hz, the second line will be at 100.225335Hz, the third at 1.00451178Hz and the 1024th line at 1000Hz.
Once allocated, the display signals will keep the history of each calculated result. The CoCo will update the points that are near the instantaneous frequency of the sweep. This is how the display signals are created. With this design, the user should understand that increasing the resolution of the display signals will not increase or decrease the quality of the swept sine.

Tracking Filters

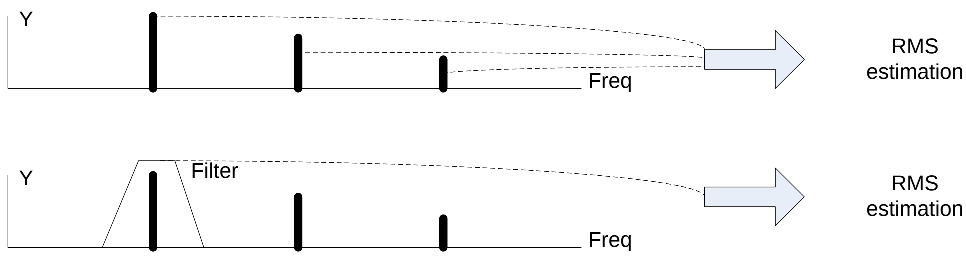
Historically, swept sine tests were originally conducted using analog technology where sine generator and measurement were all implemented in the analog domain. A very simple swept sine testing instrument consists of the following components:

- A sine oscillator of which the frequency can be changed
- An RMS estimator to the output source
- An RMS estimator to the input signal
- A divider that divides the RMS measurements between input and output
- A display or plotter to show the divided results.



■ Figure 3. Analog swept sine implementation.

In many cases the UUT response is not linear. With very pure sine excitation, the response signal may contain strong harmonics. For example with a sine tone excited at 100Hz, the response signal may contain content at 200Hz, 300Hz and so on. A simple RMS estimator will not be able to distinguish the amplitudes at these content therefore the FRF calculation will not be accurate.
To overcome this problem, a tracking filter can be applied that centers at the sweeping frequency, and the RMS estimator can be applied to the output of the tracking filter as shown in Figure 4.



■ Figure 4. Tracking filter implementation.

With a filter in place, the RMS estimator will accurately measure the frequency amplitude at the sweeping frequency. The energy at other frequencies will be filtered out.

The challenge of realizing such a filter in the swept sine test is that the filter has to track the center frequency of the sine frequency. Not only does the center frequency of the filter need to change, but also the bandwidth. To give an example, when the sweeping frequency is at 100Hz, it is reasonable to use a filter bandwidth of 50~100Hz. When the sweeping frequency goes down to 10Hz, the next harmonics will be at 20Hz. A filter with bandwidth of 50~100Hz will be too wide to use. To address the problem, a so called the tracking filter is used. Tracking filter changes both its center frequency and bandwidth according to the sweeping frequency. In the analog-made swept sine equipment, this is realized using mixing frequency technology with expensive electronic components. With digital technology, the digitally synthesize tracking filters are implemented in software at no additional hardware cost.

The bandwidth of the tracking filter is a key control parameter. In CoCo, it is defined as a percentage of the sweeping frequency. The user can select a percentage between 100% and 7%. A percentage of 100% means the equivalent bandwidth of the tracking filter is the same as its sweeping frequency. 50% means its bandwidth is ½ of the sweeping frequency.

CoCo uses a proprietary digital filter that allows very fast response and clean detection of the sine RMS value.

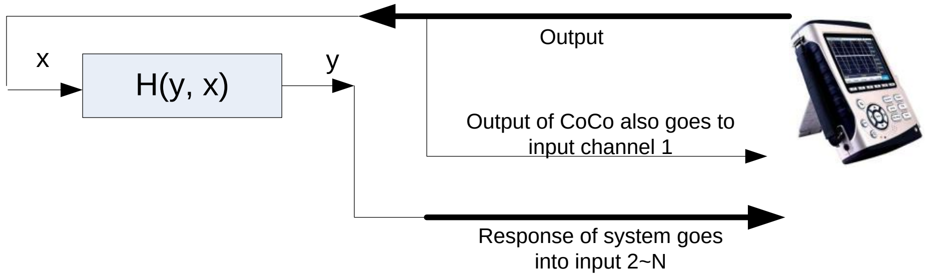
Measurement Quantities

Measurement quantities that can be monitored during the swept sine test include: time stream of each channel (raw data), spectrum of each channel, frequency responses, coherence, and phase between responses to the reference channel.

Time streams: time streams appear the same as any other applications on CoCo. Time streams are always available for viewing and recording. It is a very useful tool to observe whether the input signals are in the valid range. The recorded sine wave can be used for further post-processing. In CoCo, the time streams are often denoted as ch1, ch2 etc.

Spectra: The term spectrum is used to refer to the measurement trace in the frequency domain of each channel. It is represented in 0~Peak. The engineering unit of the spectrum is determined by the sensor used by the input channel. The resolution of spectra does not affect the quality of sine wave. In CoCo, the spectra are often denoted as Spec(ch1), Spec(ch2) etc..

Frequency Response Functions (FRF): FRF of UUT can be measured using input channel 1 as reference channel and other channels as response channels. The connection should as shown in Figure 5.



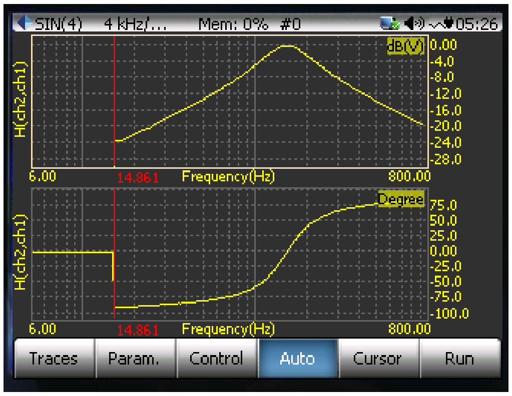
- Figure 5. Frequency response measurement with CoCo.

The CoCo will provide the FRF functions of each response channels to the output channel. FRF signals include both phase and magnitude information. In CoCo, the FRF are often denoted as Hyx

The number of FRF signals that can be monitored depend on the number of input channels on the CoCo hardware. For example, a CoCo-8, with 4 input channels can monitor 3 FRFs: H(ch2,ch1), H(ch3,ch1) and H(ch4,ch1).

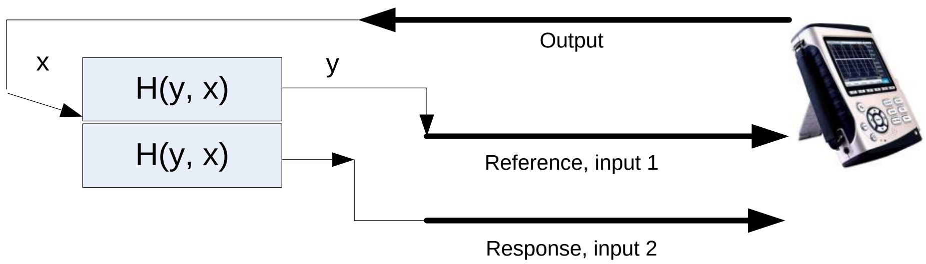
To connect the signal source to the input of UUT and back to input channel 1, you can use a BNC T-connector.

■ Figure 6 BNC T-Connector



■ Figure 7. Frequency response function.

When you measure the ratio between two response channels, it is more accurate to refer to the signals as ***Transmissibility functions*** instead of FRF because the reference signal is not really the excitation to the UUT. Figure 8 shows how to connect and measure the transmissibility between two response channels.



■ Figure 8. Typical transmissibility measurement.

Transmissibility measurements are used in many applications. For example, it can be used in “back-to-back” transducer calibration where an accurate reference transducer is used to calibrate a less accurate one.

Output Control Modes

Recall that the sine tone amplitude can be a variable of time and frequency.

$$\mathrm{x(t)} = \mathrm{A(f,t)} * \sin\left(2\pi\left(f\left(f_{Low}, f_{High}, Speed\right)\right)t\right)$$

We have discussed how the frequency can be changed and controlled by the sweeping mode, sweeping range and sweeping speed. This section discusses how the output amplitude $A(t)$ is controlled.

There are three Output Modes provided in the CoCo:

- Constant Output Level
- Output Level Profile
- Input Profile with Auto Gain Control

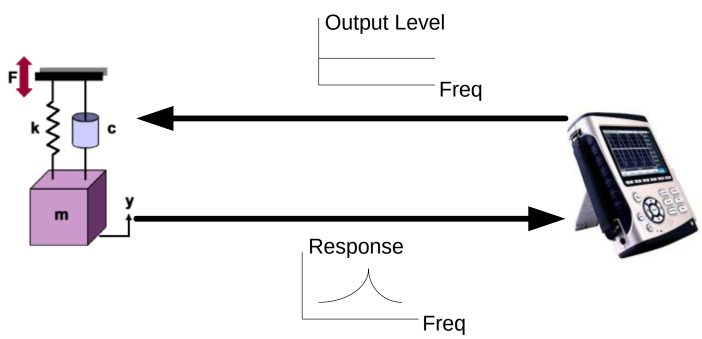
Constant Output Level

$A(f, t) = \text{constant}$

The Constant Output Level is the simplest way to generate the output. It uses a constant output level that is usually defined in the 0~peak volt. For example a 1Vpk means the output swept sine is in 1V in 0~peak.

When the constant output level is used, the response may show peaks or valleys.

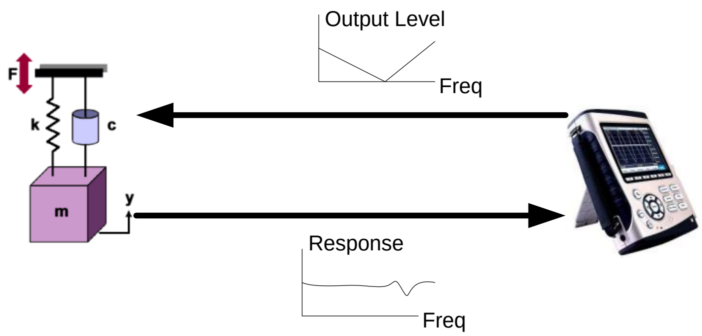
The picture below shows that a constant level sweep is applied to a Single Degree of Freedom (SDOF) device. The voltage output of the CoCo will be converted to force using a mechanical excitation system. We would expect the response measured in either displacement, velocity or acceleration will show a resonant peak.



The drawback of using constant level mode is that sometimes the dynamics of the system vary so extreme that the response signal may exceed the input range. This is very common with systems that have light damping. For example a UUT with 60dB dynamic range, which is quite common, will show the magnitude of the response change 1000 times over the test.

Output Level Profile

With output level profile control the output level $A(f, t) = A(f)$ is defined by the user. To overcome the problems with large range of variation of the response, it is possible to attenuate the excitation signal at certain frequency ranges. In the example above, because the resonance frequency is likely known to the operator, we can set the output to a lower level in that specific sweeping frequency range. We call this frequency dependent output level control the Output Level Profile.



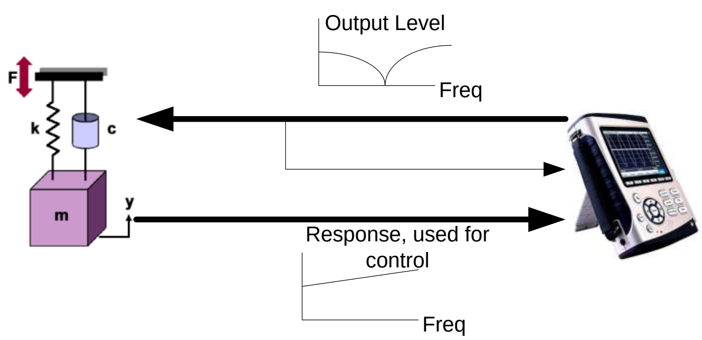
■ Figure 9. Output level profile.

Figure 9 shows that we purposely create a notch in the output level profile so that the response signal is attenuated in the resonance area.

Making the output a frequency dependent signal may help to improve the FRF or transmissibility measurement. It is much better than a constant level output. The drawback of this method is that the UUT dynamics must be known before the test. Another issue is that the output level profile may not be created accurately to match the dynamic characteristics of the UUT. To overcome this difficulty, the CoCo also includes a close-loop control method to allow the auto-gain control.

Auto Gain Control

With auto gain control, $A(f, t)$ is calculated in real time based on the target input and close-loop control gain. This advanced method can be explained in Figure 10.



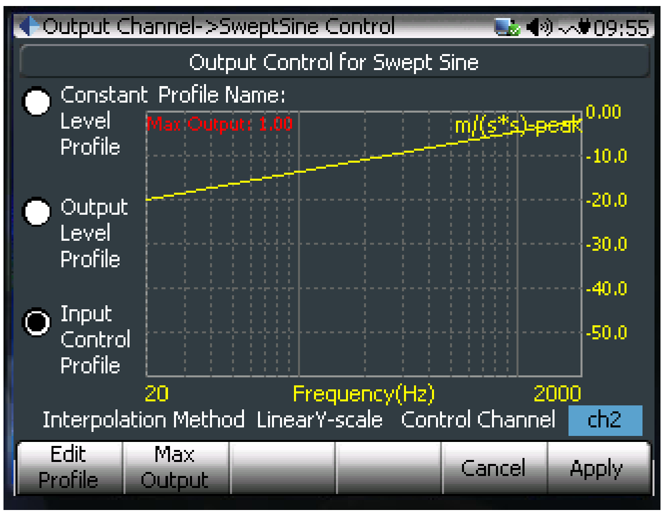
■ Figure 10. Auto gain control mode.

First the user must set up the target profile for one of the response channels (input to the CoCo). The shape of this target profile (Input Profile) does not need to be a straight line. Then during the sweep, the CoCo measures the transfer function between the response and the output. Taking this transfer function into consideration, the CoCo automatically adjusts its output so that the magnitude of the measured input signal matches the input control profile. Because the transfer function changes with frequency, this method requires a close-loop control logarithm.

The input profile with auto-gain control is the most effective way to excite the system. It can maximize the dynamic range of the input channels. However, care must be taken so that the output channel does not get too large and the input channel is saturated, or the output channel gets too small and the input channel reduces to the background noise level.

It must be noticed that the Output and Input Profiles have different engineering units. The Output Profile always has the engineering unit of Vpk for the sine wave. The Input Profile will have the engineering unit of whatever is measured by that channel. For example if the response sensor is a displacement sensor, then the Input Profile will have displacement units, in 0~Peak. If it is an accelerometer, then it will have acceleration units, in 0~Peak.

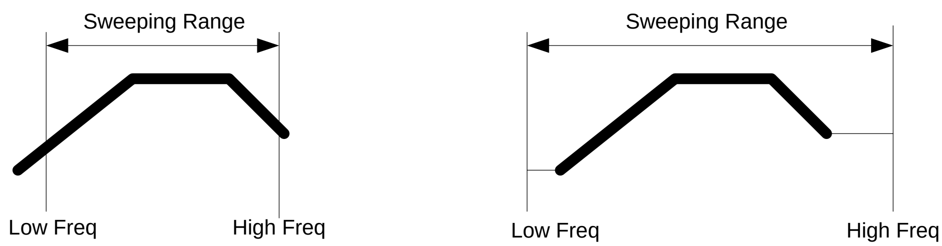
When Input Control Profile is selected, in default we use channel 2 as control channel. You can select any channel other than the reference channel (channel 1) for control.



■ Figure 11. Input auto gain control mode profile.

Sweeping Range and Profile

The sweeping range is controlled by the two boundaries of the frequency range. If the profile setting is not at the same range of the sweeping range then the CoCo will automatically adjust the range.



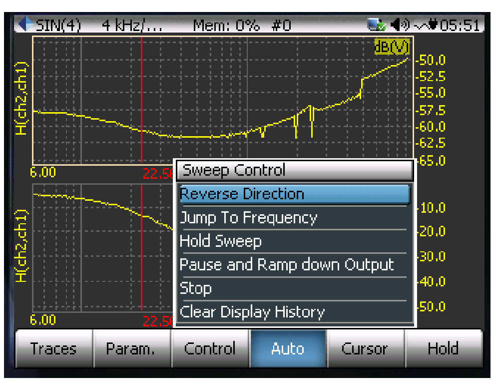
■ Figure 12. Sweeping range and profile.

In Figure 12, the thick line represents the profile. If the Low Freq or High Freq limits do not match the Profile, the software extends the ends to the left and right so there are always valid profile value points when the output sweeps.

Sweep Control

The swept sine output is controlled by ***sweeps***. One sweep indicates that the output will generate the sine frequency from the Low Frequency to the High Frequency, or high to low. In addition the user can control the sweep with manual controls including:

- Start Output
- Stop Output: this action will abort the test
- Reverse Direction
- Jump to Frequency
- Hold Sweep: this action will not ramp down the output voltage amplitude. The frequency will be fixed
- Resume Sweep
- Clear Display History



■ Figure 13. Sweep control options.

To avoid shocking the UUT, a sine output will never start or stop abruptly. Instead, the sine wave amplitude slowly ramps up from zero to the desired level. The ramping rate is defined as dB/sec. a 40dB/sec means the sine wave will ramp up or ramp down for 100 times in magnitude in a second. This is a user-defined advanced value.

3. ACOUSTIC DATA ACQUISITION: OCTAVE ANALYSIS

The Acoustics Data Acquisition software option includes Fractional Octave Filter Analysis, Sound Level Meters and Microphone Calibration functions.

The Fractional Octave Filter Analysis applies a bank of real-time octave filters to the input time streams and generates two types of signals at the same time: fractional frequency band signals, i.e., *octave spectra*, and the *RMS time history* of each filter band. The output of each real-time filter bank is in fact a 3D waterfall signal that is arranged in the x-axis as logarithmic frequency and z-axis as time. In the frequency direction, frequency weighting is applied. In the time axis, the time-weighting is applied.

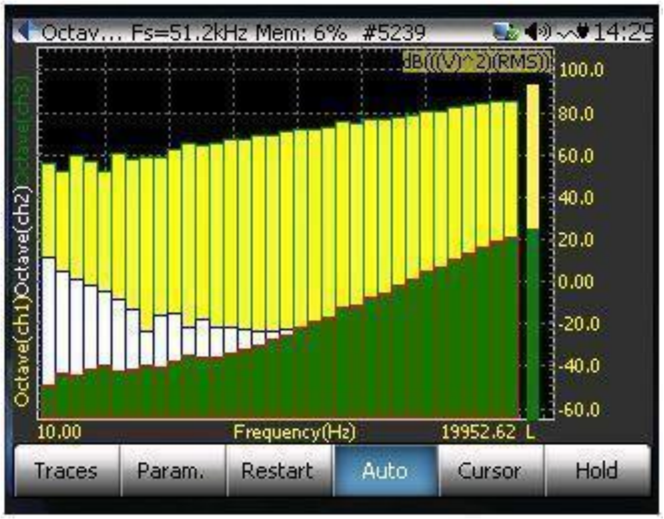
The Sound Level Meter (SLM) is a similar application to octave filters in the acoustic data acquisition. This application is also referred to as an Overall Level Meter. The SLM applies ONE frequency weighting filter to the input signal and time weighting to the output. Various measures are then extracted from both the input and output signals of this frequency weighting filter.

Fractional Octave Filter Analysis

Acoustics Analysis provides 1/N octave analysis using true real-time digital filters that conform to Conforms to ANSI std. S1.11:2004, Order 3 Type 1-D and IEC 61260-1995 specifications. A, B and C weighting filters can be applied to the input data. Output results are weighted or un-weighted RMS values. The output can be normalized with a calibration value. The results can be plotted on log or linear axes and exact or preferred frequency values are supported.

The Acoustics Analysis provides 1/N octave analysis using true real-time digital filters that conform to ANSI S1.11 and IEC 61260 specifications. Each band filter is designed in accordance to ANSI S1.11 and IEC 61260 specifications by transforming the original analog transfer function to the digital domain by means of the bilinear transform. The filter order can be specified and the frequency ratio can be calculated using the binary or decimal system.

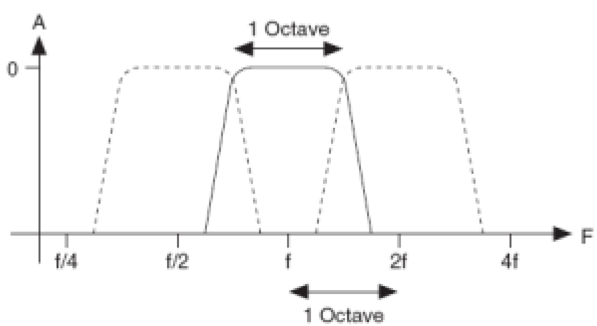
The RMS reading of each octave filter is usually represented by a “bar” in the spectrum plot. Keep in mind that the octave filters have “skirts” on both sides. They are not as straight as the bars depicts. The adjacent filters always overlap. Due to this reason, a sine tone at 1kHz will not only excite the filter with center frequency at 1kHz, but also all other filters. Figure 14 shows how the energy in each band is displayed on the octave spectrum plot using bars.



- Figure 14. 1/3 octave filter banks.

Full Octave Filters

An octave is a doubling of frequency. For example, frequencies of 250 Hz and 500 Hz are one octave apart, as are frequencies of 1 kHz and 2 kHz.



- Figure 15. Full octave filter shape.

Full octave analysis, i.e., 1/1 octave, displays the frequency characteristics of a signal by passing the signal through a bank of band-pass filters where the center frequency of each filter is one octave apart. If the lower and upper cutoff frequencies of a band-pass filter are f_L and f_H, then the center frequency, f_c can be determined with:

$$f_c = \sqrt{(f_L * f_H)}$$

The nominal frequency ratio G is determined by:

$$G = f_H / f_L$$

Two systems are used in the industry, Base-two or Base-ten systems. For base-two systems, G = 2. For base-ten systems, G = $10^{3/10}$. Base-ten system is used in CoCo.

The proportional bandwidth property will divide the frequency information uniformly over a log scale. It is very useful in analyzing a variety of natural systems. For example, the human response to noise and vibration is very non-linear and many mechanical systems have a behavior that is best characterized by proportional bandwidth analysis.

Fractional Octave Filters

To gain finer frequency resolution, the frequency range can be divided into proportional bandwidths that are a fraction of an octave. For example, with 1/3 octave analysis, there are 3 band-pass filters per octave where each center frequency is $10^{1/10}$ the previous center frequency.

In general, for 1/N octave analysis, there are N band pass filters per octave such that:

$$\frac{f_H}{f_L} = \left(10^{3/10}\right)^{1/N}$$

$$f_{c\,j+1} = f_{c\,j} * \left(10^{3/10}\right)^{1/N}$$

where 1/N is called the fractional bandwidth resolution.

For CoCo the equation and table below define the center frequency of each fractional filter.

$$f_c = 10^{3X/10N}$$

For example for 1/1 Octave (N =1) the first center frequency (index X = 1) is computed as

$$f_c = 10^{\frac{3*(-3)}{10*1}} = 0.125 \text{ Hz}$$

- Table 1. Octave center frequencies.

		1/1-Octave	1/3-Octave	1/6-Octave	1/12-Octave
Standard		IEC 225-1966 DIN 45651 ANSI S1.11-2004 Order 7 Type 1-D	IEC 225-1966 DIN 45651 ANSI S1.11-2004 Order 3 Type 1-D	N/A	N/A
	X (index)	-3 ~ 14	-10 ~ 43	-20 ~ 86	-40 ~ 172
	Total number of Filters	18	54	107	213
	f_c (Hz)	0.125 – 16k	0.1 – 20k	0.1 – 20k	0.1 – 20k

Nominal center frequencies (midband frequencies)
Nominal center frequencies are "round" numbers that were inherited from the old analog octave filters. They are rounded midband frequencies for the designation of band pass filters. The nominal midband frequencies for 1/1-octave and 1/3-octave are listed in the ANSI S1.11-2004 Annex A. The standard also describes how to decide the nominal midband frequencies for other fractional octave bands.

The exact center frequency of the filter band is usually not the same as that of nominal frequency. For example, in a 1/3 octave, the exact center frequencies 794.33Hz, 1000Hz and 1258.9Hz are used to correspond to the filters with nominal frequencies 800Hz, 1000Hz and 1250Hz.

Band Edge Frequencies of Fractional Filters
The low and high edge frequencies of a filter can be calculated based on the frequency ratio, G and the fractional octave resolution N (=1, 3, 6, 12…)

$$Lower\ Edge\ Frequency \quad f_L = f_c * \left(10^{3/10}\right)^{-1/2N}$$
$$Upper\ Edge\ Frequency \quad f_H = f_c * \left(10^{3/10}\right)^{1/2N}$$

The bandwidth of the filter is: BW = $f_H - f_L$

When starting or resetting the filtering operation of the fractional-octave filters, a certain time is required before the measurements are valid. This time is called the *settling time* and is related to the bandwidth of any particular filter. The lowest frequency band has the smallest bandwidth and defines the settling time required before you can consider the complete fractional-octave measurement valid. A good rule of thumb is that the settling time is approximately five divided by the bandwidth.

$$Settling\ time = \frac{5}{BW} = \frac{5}{f_H - f_L}$$

Note the settling time depends on the bandwidth which changes with center frequency. A narrower filter and a lower frequency band requires a longer settling time.

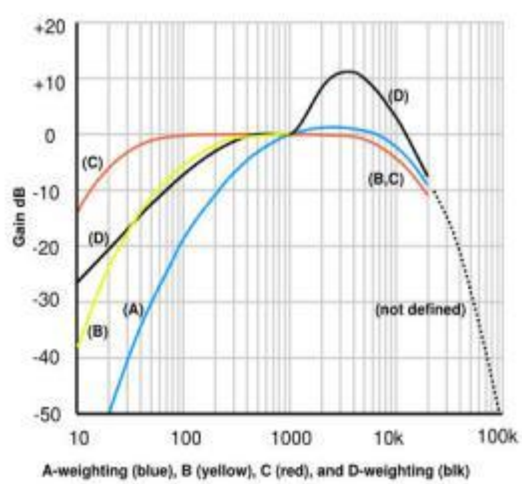
Analysis Frequency Range

In CoCo, the user can decide the analysis range by changing the lowest and highest f_c as the Analysis Parameters:

Analysis Range	1/1 Octave	1/3 Octave	1/6 Octave	1/12 Octave
Lowest f_c (Hz)	0.125 1 8	0.1 1 10 100	0.1 1 10 100	0.1 1 10 100
Highest f_c (Hz)	1000 4000 16000	1000 2000 5000 10000 20000	1000 2000 5000 10000 20000	1000 2000 5000 10000 20000

Frequency Weighting

The human hearing system is more sensitive to some frequencies than others, and its frequency response varies with level. In general, low frequency and high frequency sounds appear to be less loud than mid-frequency sounds, and the effect is more pronounced at low pressure levels, with a flattening of response at high levels. Octave analysis and sound level meters therefore incorporate weighting filters, which reduce the contribution of low and high frequencies to produce a measurement that corresponds approximately to what we hear.



- Figure 16. Frequency weighting filter shapes.

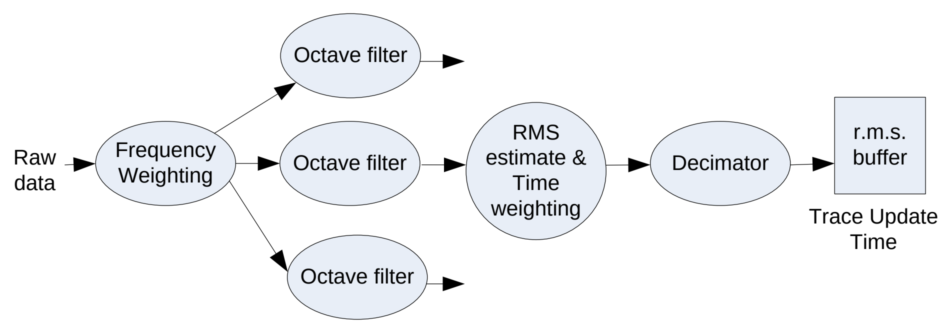
CoCo provides A, C, Z weightings conforming to IEC 61672-1 2002 and B weighting conforming to IEC 60651 in both of Octave analysis and Sound Level Meter. The Frequency weighting in the octave filters will affect the results of all filter bands.

Time or RPM based RMS Trace of the Octave Filters

The ANSI and IEC standard do not require storing the time history of the band pass filter output. However the user may be interested in viewing this information. On the CoCo the RMS history of all the band pass filters are stored, in the RMS quantity. Below is the description about how the RMSs. history is calculated.

The RMS history can be stored against one of two variables: Time or RPM.

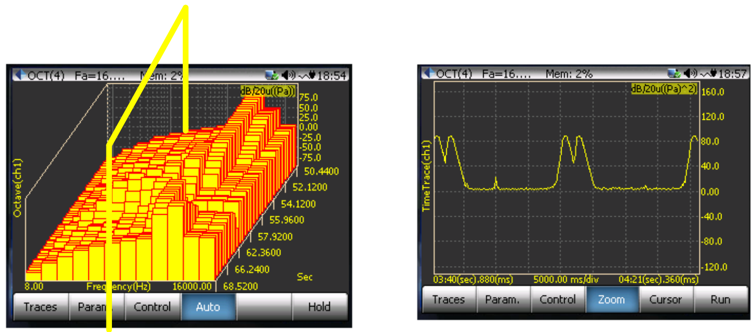
Both the input and output of a digital filter are a series of data points. While it requires excessive memory to keep all the time data of all the filters, it is useful to keep the so called RMS history of each filter output. The RMS time history is computed after the time weighting averaging operation as shown in Figure 17.



- Figure 17. RMS time history calculation.

The *Decimator* is used so the user can choose the length of time to save the RMS data. For example, given a buffer length of 1024, a *Trace Update Time* of 5 ms will keep about 5 seconds of RMS history; if this update time is set to 5 seconds, it will record 1.4 hours of RMS history.

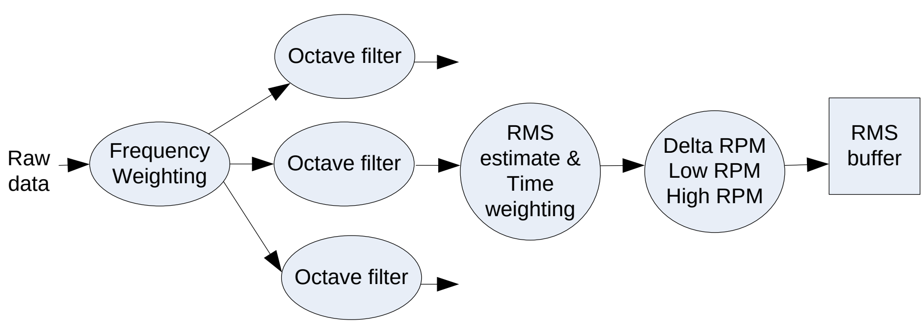
Figure 18 shows the 3D waterfall display of a 1/1 octave filter output. If *a cut* is made in the Z axis direction, the result will be an octave spectrum. If *a cut* is made in the X-axis, the result will be called a *Time Trace*.



■ Figure 18. A cut of 3D Waterfall of octave filter output (left) maps to an RMS time trace (right)

The *Time Trace* stores the history of the RMS of each filter output. The spacing between two points of the *Time Trace* is called *Trace Update Time, in seconds*. On CoCo, one *Time Trace* is allocated for each channel for display. Keep in mind that this buffer of Time Trace is the output of a specific filter, the user can change the center frequency of the filter for the Time Trace during run time. In the other words, this time trace display buffer will change its content completely when the user switches the *Time Trace Frequency* from one to another.

Alternatively the RMS trace can be stored using RPM as a variable. This method is particularly useful in the automotive NVH applications. The picture below shows how one of the outputs of filters can be stored in RPM trace.



■ Figure 19. Store RPM based RMS traces.

Exponential and Linear Averaging

Linear average: The Linear average method uses a fixed time period to sum up the historical power value of each filter and then takes the square-root to calculate the averaged RMS value. In Linear average, the RMS trace update time is governed by the time period of the averaging. For each time period of averaging, one RMS value per frequency bin is produced.
Exponential average: Exponential average applies an exponential time constant to the historical power values of each filter and takes the square-root of the averaged power value. A time constant of 0.125 seconds is equivalent to “Fast” averaging and 1.0 second is equivalent to “Slow”

averaging in the acoustics. In exponential average, the RMS trace update time is independent of the time constant.
Peak Hold: Peak Hold retains the maximum value in each frequency bin over the period of time since last “start” or “restart”.

As we discussed previously, each filter may have different settling time.

Measurements available to CoCo in Octave Analysis mode

The following measurement quantities are available on the CoCo in the octave measurement mode.

Time streams of input channels
In CoCo, time domain data is always available in the form of long time history. The user can view and record the time signals. The limitation is that the sampling rate of the time signals cannot be arbitrarily changed. It is always set internally by the system based on analysis frequency range, i.e., the highest center frequency of the filter bands.

Octave Spectra
Each input channel will have an octave spectrum.

RMS Trace
Each input channel will have one RMS trace to display the RMS history. This RMS is the output of the filter for a specific band. The RMS trace is defined as the Time-RMS trace or RPM-RMS trace at the CSA Editor level. You cannot change between Time and RPM based for a specific CSA.

4. ACOUSTIC DATA ACQUISITION: SOUND LEVEL METER

An analog sound level meter measures the sound pressure level. The standard sound level meter is more correctly called an *exponentially averaging sound level meter* as the AC signal from the microphone is converted to DC by a RMS circuit and thus it must have a time-constant of integration. This is referred to as time-weighting. Three of these time-weightings have been standardized, 'S' (1s) originally called Slow, 'F' (125 ms) originally called Fast and 'I' (35 ms) originally called Impulse. The output of the RMS circuit is linear in voltage and is passed through a logarithmic circuit to give a readout in decibels (dB). This is 20 times the base 10 logarithm of the ratio of a given root-mean-square sound pressure to the reference sound pressure. Root-mean-square (RMS) sound pressure is obtained with a standard frequency weighting and standard time weighting.

With the advent of digital technology and increasing accuracy of the electronic circuits, the sound level meter functions are more recently calculated in the digital domain. One of the important factors of such implementation is that the instrument must provide very high dynamic range so that both weak and strong signals can be calculated and observed. CoCo provides 130dB dynamic range. High dynamic range is one of the most important measures of the quality of an acoustic analyzer.

A traditional sound level meter only includes the 1/1 and 1/3 octave filters. In the CoCo system octave analysis is provided in addition to the other analysis functions providing more flexibility and computation power than a traditional sound level meter.

You should use Octave Analysis as the template to create a CSA projects when fractional octave analysis is needed. In both the Octave Analysis and Sound Level Meter templates the user can see the frequency weighted readings (such as dBA). But the reading results may be slightly different when comparing Octave Analysis and Sound Level Meter results. This is because the data processing flow in octave filter analysis and sound level meter is computed differently. In the octave analysis, the dBA, i.e., the A-weighted sound level is computed by applying the frequency weighting function to the output of each individual filter bank; while in SLM, the A-weighted sound level is created by applying the A-weight filter in the entire time domain. The SLM template should be used to obtain the dBA or similar overall readings for most sound studies that might be compared to results taken with a traditional sound level meter because the computation is more similar to that of a traditional sound level meter.

Terms and Definitions

In this section we will define the terminology used in the SLM software options.

Reference sound pressure It is conventionally chosen as 20 µPa. This is the threshold of hearing of the average person and is used to compute the sound pressure level in the dB scale.

Sound pressure level (in dB)

Sound pressure level (dB) is defined as twenty times the logarithm to the base ten of the ratio of the RMS of a given sound pressure to the reference sound pressure. Sound pressure level is expressed in decibels (dB); symbol L_p.

Peak sound pressure
The peak sound pressure is the greatest absolute instantaneous sound pressure during a stated time interval.

Peak sound level (in dB)
The peak sound level (dB) is defined as twenty times the logarithm to the base ten of the ration of a peak sound pressure to the reference sound pressure, peak sound pressure being obtained with a standard frequency weighting. (example letter symbols are L_{peak}, L_{Cpeak})

Frequency weighting
Frequency weighting is the difference between the level (dB) of the signal indicated on the display device and the corresponding level of a constant-amplitude steady-state sinusoidal input signal, specified in the IEC or ISO standards as a function of frequency. It accounts for A, B and Z frequency weighting discussed in the previous section.

Time weighting
Time weighting is an exponential function of time, of a specified time constant, that weights the square of the instantaneous sound pressure. This is the same as exponential averaging in the time domain to the instantaneous sound pressure.

It is a continuous averaging process that applies to the output of a frequency weighting filter or one of the fractional filters. The amount of weight given to past data as compared to current data depends on the exponential time constant. In exponential averaging, the averaging process continues indefinitely.

In a sound level meter the time weighting exponential averaging mode supports the following time constants:

Slow uses a time constant of 1,000 ms. Slow averaging is useful for tracking the sound pressure levels of signals with sound pressure levels that vary slowly.
Fast uses a time constant of 125 ms. Fast averaging is useful for tracking the sound pressure of signals with sound pressure levels that vary quickly.
Impulse uses a time constant of 35 ms if the signal is rising and 1,500 ms if the signal is falling. Impulse averaging is useful for tracking sudden increases in the sound pressure level and recording the increases so that you have a record of the changes.
User Defined allows you to specify a time constant suitable for your particular application.

Time-weighted sound level (in dB)
This is twenty times the logarithm to the base ten of the ratio of a given RMS sound pressure to the reference sound pressure, RMS sound pressure being obtained with a standard frequency weighting and standard time weighting. (example letter symbols are L_{AF}, L_{AS}, L_{CF}, L_{CS})

Maximum and minimum time-weighted sound level (in dB)
This is the greatest and lowest time-weighted sound level within a stated time interval. (example letter symbols are L_{AFmax}, L_{ASmax}, L_{CFmax}, L_{CSmax}, L_{AFmin}, L_{ASmin}, L_{CFmin}, L_{CSmin})

Time-average sound level (equivalent continuous sound level) (in dB)

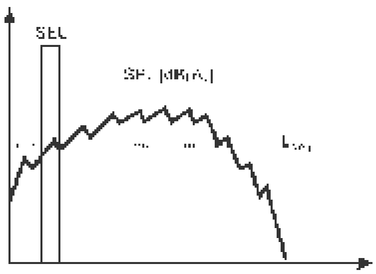
This is twenty times the logarithm to the base ten of the ratio of a RMS sound pressure during a stated time interval to the reference sound pressure, sound pressure being obtained with a standard frequency weighting. (example letter symbols are L_{Aeq}, L_{Ceq})

Sound exposure

This is the time integral of the square of sound pressure over a stated time interval or event. Sound exposure is used to measure high-level, short duration noises and to study their effects on humans.

Sound exposure level (in dB)

Sound exposure level is the total sound energy of a single sound event and takes into account both its intensity and duration. Sound exposure level is the sound level you would experience if all of the sound energy of a sound event occurred in one second. This normalization to duration of one second allows the direct comparison of sounds of different durations.



- Figure 20. Sound exposure level illustration.

Figure 20 shows the relationship between the Sound Exposure Level (SEL), the Sound Pressure Level (SPL), and the Leq. The Leq is the constant level needed to produce the same amount of energy as the actual varying sound (the SPL).

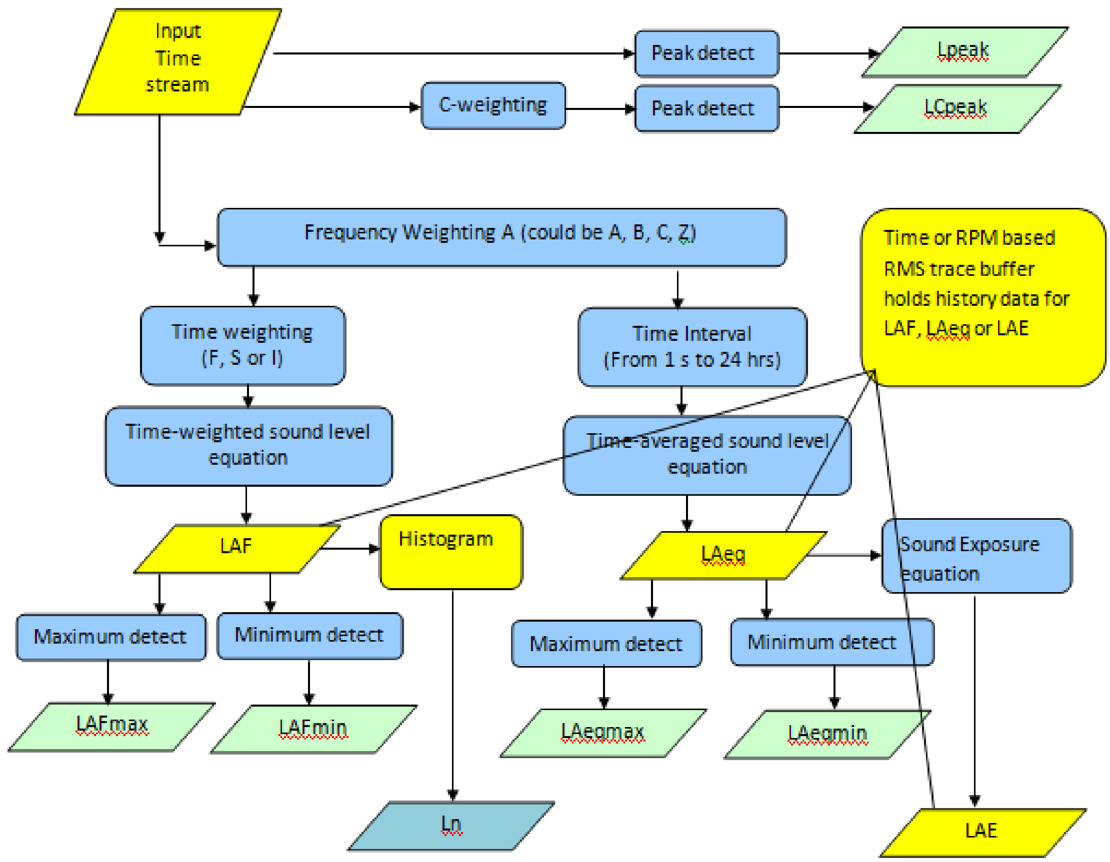
The SEL is the Leq normalized to 1 second. It is what the Leq would be if the event occurred over a one second duration.

Statistical Level (L_N)

LN is defined as the sound pressure level which is exceeded N% of the time over the duration of a measuring time interval. L0 is the maximum level over the duration of the measurement. L100 is the minimum.

Data Processing Diagram

Figure 21 shows the data processing diagram for ONE input channel for all the SLM measurements when A-weight is applied.



■ Figure 21. Sound level meter computation diagram.

In the SLM measurement, after the digitized data comes in, it is split into three paths: one goes to frequency weighting A, B, C or Z and one goes to C weighting or no weighting. The peak detection is computed from the output of C weighting or no weighting. The output of frequency weighting (A, B, C or Z) is further split into two paths. The first will go to a *time weighting function* which is more or less equivalent to an exponential averaging mode to calculate LAF; the second path goes to a time averaging function, which is equivalent to a linear averaging mode to calculate Leq.

With A-weighted applied as shown in the example, the list of symbol to be used by this instrument is:

Symbol of Measured Values	Description
LAF	A-weighted, F time-weighted sound level
LAFmax	Maximum A-weighted, F time-weighted sound level
LAFmin	Minimum A-weighted, F time-weighted sound level
LCpeak	Peak C sound level, greatest absolute instantaneous C-weighted sound pressure level
Lpeak	Peak sound level, greatest absolute instantaneous sound pressure level
LAeq	A-weighted, time-average sound level (equivalent continuous sound level)
LAeqmax	Maximum A-weighted, time-average sound level (equivalent continuous sound level)
LAeqmin	Minimum A-weighted, time-average sound level (equivalent continuous sound level)
LAE	A-weighted sound exposure level
L_N (N = any integer between 0~100)	Statistical Level general term
L1, L5, L50, L95....	Statistical Levels with specific N values. The sound level exceeds this level 1, 5, 50 or 95 percent of the time for the duration of the measurement.

Measurements available to CoCo in SLM mode

There are two ways to view sound level measurements: instantaneous SLM measures and RMS history. Instantaneous SLM measures represent the most current value of the measurement. RMS history not only shows the most current value, but also a record of historical values against time or RPM. Some of the measures allow only instantaneous values others allow both.

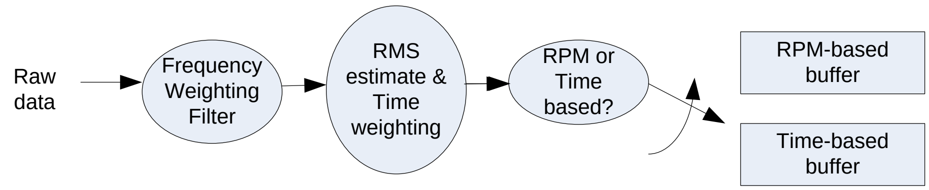
The following measurement quantities are available to CoCo in the measurement.

SLM Measures

The following SLM measures are available for real-time reading and can be saved as a data structure for future review.

Time Weighted Sound Levels

In CoCo, time weighted sound level is the output of frequency-weighting and then time weighting filters. Time weighting serves an exponential averaging operator. The computation is illustrated in Figure 22.



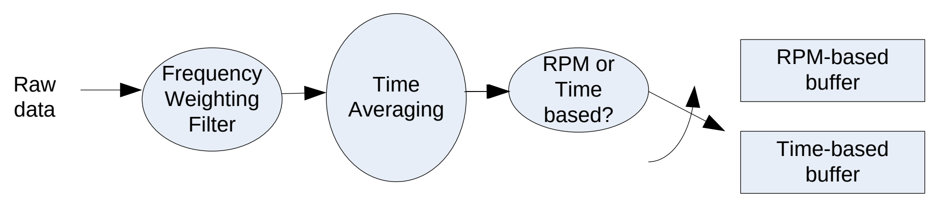
■ Figure 22. Time weighting sound level computation and storage against RPM or Time

The table below shows the symbols for the time-weighted sound level.

Symbol used for time weighted value		**Frequency Weighting**			
		Z	**A**	**B**	**C**
Time Weighting	**F(Fast)**	L_{ZF}	L_{AF}	L_{BF}	L_{CF}
	S(Slow)	L_{ZS}	L_{AS}	L_{BS}	L_{CS}
	I(Impulse)	L_{ZI}	L_{AI}	L_{BI}	L_{CI}
	Custom	L_{ZC}	L_{AC}	L_{BC}	L_{CC}

Time Averaged Sound Levels

In CoCo, time averaged sound level is the output of frequency-weighting and then time average operation. Time average serves a linear averaging operator. Figure 23 illustrates the computation.



■ Figure 23. Time averaged sound level computation.

The table below shows the symbols for time-average sound level. In the time averaging sound level measurement, Frequency weighting can be selected as A, B, C or Z. The time interval for time averaging can be set to any value between 1 second and 24 hours.

Frequency Weighting	Z	A	B	C
Symbol	L_{eq}	L_{Aeq}	L_{Beq}	L_{Ceq}

Peak sound level
Only C-weighted and un-weighted are available for peak sound level. This is required by the standards.

Symbol	L_{peak}	L_{Cpeak}

Sound exposure level
Sound exposure level and time-average sound level have the same frequency weighting and same time interval.

Frequency Weighting	Z	A	B	C
Symbol	L_{E}	L_{AE}	L_{BE}	L_{CE}

Statistical level: value reading
Any statistical level L_N is the sound level which is exceeded for N% of the defined measurement duration.

Symbols for L_N, N = 1, 5, 50, 95	L1	L5	L50	L95

Input Channel Time Streams

In CoCo, time domain data is always available in the form of long time history. The user can view and record the time signals. The limitation is that the sampling rate of the time signals cannot be arbitrarily changed. It is always set internally by the system based on analysis frequency range.

RMS trace of weighted level, time averaged level or sound exposure

CoCo records an RMS trace of the sound level. The user must choose between the time weighted level L_{AF}, the equivalent time averaged level L_{AEQ} or sound exposure level L_{AE}. Only one can be recorded at a time.

The RMS trace must be selected using one of Time or RMS as variable at the CSA Editor stage.

Histogram of Time Weighting

CoCo also records a signal containing a histogram of the dB values of the time weighted signal. This signal is used to compute the Ln data.

5. ORDER TRACKING

Order Tracking is a general term describing a collection of software functions used for analyzing the mechanical dynamic behavior of rotating or reciprocating machinery for which the rotational speed can change over time. Unlike the power spectrum and other frequency-domain analysis standards where the changing variable is the frequency, Order Tracking functions present the data related to the variable rotating speed, i.e., RPM (revolution per minute).

The most useful measurements are order spectra and order tracks. An order spectrum gives the amplitude of the signal as a function of harmonic order of the rotation frequency. This means that a harmonic or sub-harmonic order component remains in the same analysis line independent of the speed of the machine.

The technique that observes the changes of any quantity vs. RPM is called ***tracking***, as the rotation frequency is being tracked and used for analysis. Most of the dynamic forces exciting a machine are related to the rotation frequency so interpretation and diagnosis can thus be greatly simplified by use of order analysis.

Order tracks are simply the observations to the amplitude of the components with fundamental frequency or harmonics. It is one typical type of *tracking*. There are other types of tracking. For example, the user can track the FFT-based PSD spectra, a fixed band or an octave band etc.; all these can be called tracking.

With the CI Order Tracking package, the instrument can:

- Process a tachometer signal and give a high fidelity RPM measurement
- Measure the order spectra
- Measure the order tracks
- Measure the RPM FFT spectrum
- Measure the energy in fixed bands vs. RPM
- Measure the amplitude and phase of an order relative to the tachometer.

There are several different applications for order tracking. A discussion of some is given below.

The first application, often referred to Run Up/Run Down, is used to evaluate the noise or vibration dynamic response when RPM is used as a changing variable. In this case, the RPM range can be very large, from a few RPM to 10,000 RPM. Typical application tests are used in the automotive or aircraft engines. The measurements can be any physical quantities such as sound, displacement, velocity, acceleration, torque, etc. The analysis measure can be the amplitude or the power of an order, the energy over a fixed frequency band, a bin of octave filter, etc. The phase information of the responses to tacho is less important in this type of application. In fact the rotating element might be hidden inside of mechanical system. The primary result for this type of measurement is the magnitude of the responses vs. RPM.

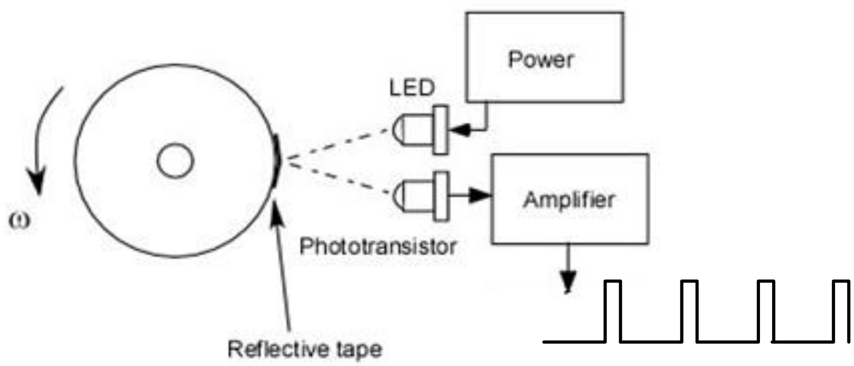
The second application is rotating machine analysis that focuses on the measurement of displacement or velocity of the rotors while it is rotating. The instrument measures the amplitudes of specific orders and their relative phase to a reference signal. The phase is calculated relative to the tachometer input or a separate reference input. This application is common for machine diagnosis and balancing. In this case, the RPM is stable or quasi-stable. Order tracking technology is useful to increase the accuracy of the estimation of orders.

Order signals with phase are useful in the test of rotating machine in the Run Up/Run Down process. This is often presented as a “Bode Plot”. The Bode Plot is a borrowed concept from control theory; it is a collection of Amplitude and Phase data over a changing speed range (i.e. Run Up or Coast Down). Some of the setup information depends on the rate of change of the RPM. The Run Up or Coast Down could take anywhere from a few minutes to a few hours (such as for a cold startup on a turbine). Other displays such as the orbit plot is useful as well.

The CoCo includes the ability to measure RPM based octave analysis and sound levels. This feature is similar to order tracks except that spectra are recorded in octave bands with A, B, C or Z frequency weighting. This feature is included in the Acoustic Analysis and Sound Level Meter CSA Templates instead of the Order Tracking Template. Refer to these sections for more details.

Tachometer Signal Processing and RPM Measurement

A **tachometer (tacho)** converts the angular velocity of a rotating shaft into an electrical signal, typically a voltage. It is common for calibrated instruments to provide a measurement of the shaft in units of revolutions per minute (RPM) or revolutions per second (RPS). Many modern rotating machines (electric motors, generators, pumps, turbines, IC engines, etc.) have integrated tachometers that can measure shaft angular velocity. An example of an optical tachometer is shown in the figure below.



- Figure 24. Optical tachometer setup.

The goal of tacho signal processing is to get a clean and stable RPM reading. The tacho signal must be carefully processed to provide a base of tracking. Any order tracking results can only be thought of as being as accurate as the tachometer signals that were used to estimate the instantaneous frequency of the order in the analysis process. If the quality of the tachometer channel is poor, the results from all other channels will be poor or even unreliable.

In old analog methods, tacho channels were conditioned with a tracking ratio tuner with a phase lock loop. The disadvantage of this method is the limited slew rate and the use of complex hardware. To overcome these limitations various digital tacho processing methods have been developed.

From hardware design point, there are two ways to implement a tacho input channel: use a dedicated tacho channel with a digital counter, or use an analog input channel.

Dedicated tacho channel using counter
Using a dedicated tacho channel, usually without A/D converter, has been popular. This hardware approach contains its own tacho clock which runs at a much higher speed; typically in MHz. This tacho hardware also contains special counters which maintain a continuous counter reading to avoid skipping any triggered cycles of the tacho signal. There is also an option to allow these counters to "average" several tacho periods for cases when the input tacho frequency is very high.

Using Analog Input Channel
Alternatively, some systems use an analog input data channel as a tacho channel. In this case, the tacho clock is actually the sample rate of the data channels. This sample rate usually limits the tacho frequency range since the tacho range is now set by the input data frequency range requirement. In addition, due to the "frame processing" nature of some not-so-well designed input sampling processes, some instruments may be limited to how they acquire the tacho signal. This restriction usually means they get several tacho cycles in every data frame. The result is often an "averaged" value which is okay unless the tacho signal is changing frequency during the data frame event, which is often the case.

With the advances of electronics and lower cost of electronic components cost is less of a concern. The approach of dedicated tacho channel with a digital counter, without A/D, may or may not be the best choice.

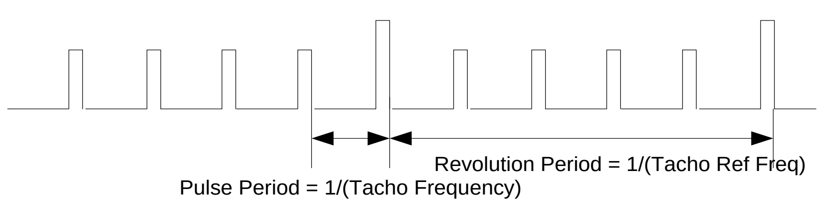
The CoCo-80/90 can use any of the data channel as tacho channel. For simple interface design, usually channel 1 is used for the tacho. While the data input channel is used as a tacho measurement, the special hardware circuitry allows this data channel to sample at the highest possible sampling rate. In the other words, the accuracy of tacho speed measurement is depending on the current range of the analysis frequency. This technique has several obvious advantages:

- The time domain signal of the tacho input is transformed by A/D converter into a digital signal. The user can observe the pulse trains of the tacho signal and set threshold arbitrarily.
- Accurate phase information can be obtained relative to each data channel because the tacho channel, which is fed by high frequency sampling counter, is synchronized with data channels.
- The RPM estimation is not influenced by the current data sampling rate.

High Pulse per Rev

Pulse per Rev is defined as the number of pulses per revolution. Pulse per Rev. must be defined by the user so the instrument can calculate the *reference frequency* of tacho using *tacho frequency*. The relationship is:

Tacho Reference freq = Tacho freq / Pulses per Rev



In most rotor tests, especially in balancing, the *Pulses per Rev* is simply 1. However, in other cases, such as in flywheel or geared data measurement, the *Pulses per Rev* can be as high as hundreds. To deal with this situation, a dedicated tacho channel with high speed counter might work better.

In the CoCo-90, in addition to using any data channel as tacho input, a dedicated tacho channel is installed to measure a high speed RPM, or deal with high Pulses/Rev or digital TTL trigger. The counter speed is about 25MHz. This second choice provides a more versatile solution to the user for their applications.

The special tacho hardware design on the CoCo system with the Order Tracking package offers the most accurate possible approach for performing a wide range of Real-Time machinery-related vibration and noise analysis.

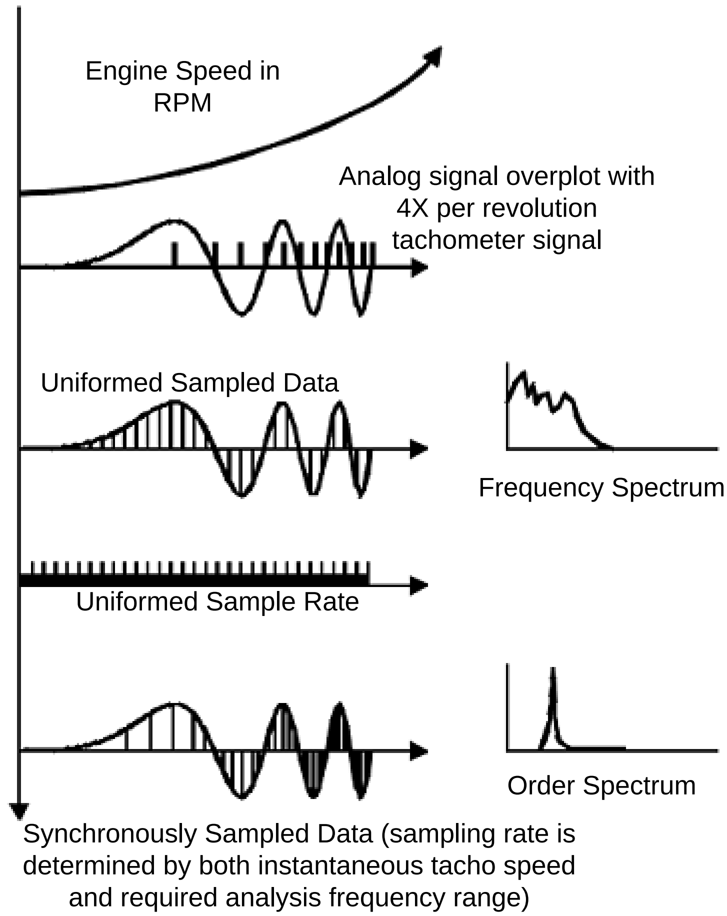
Pulse Detection
A good tacho processing device should allow the user to see the tacho signal in its original time waveform visually, and set the *Pulse per Rev*., the threshold of pulse detection. This will help setup the tachometer and diagnose any problems quickly. In the CoCo hardware, a special display window is created so the user can switch between the RPM trace and the tacho original time waveform displays conveniently. The pulse detection threshold can easily be controlled by using Up/Down buttons.

Order Tracks and Order Spectrum

Knowledge of the rotating speed allows presentation of measurement results in the angle and order domains, corresponding to the time and frequency domains. *An order is a frequency normalized with some reference frequency, e.g. the shaft frequency*. This means that the order of a vibration component in the order spectrum indicates the number of vibration cycles per shaft revolution. The magnitude, which can be measured using EU_{pk}, EU_{rms} or EU_{rms}^2 , of an order is the measurement extracted through a tracking filter with the center frequency located at this frequency. Multiple measurements of a range of orders will construct a so called *Order Spectrum*. An order power spectrum measurement gives a quantitative description of the amplitude, or power, of the orders in a signal. It provides a good view of all order components of a signal. This can help you find significant orders and compare the level of different order components

There are two methods to perform rotationally coherent sampling, phase-locked frequency multipliers and digital re-sampling. Phase-locked frequency multipliers were mainly used in early work. They generate sampling pulses based on a rotational reference signal. These sampling pulses control the sampling process. Note that the sampling frequency will depend on the rotational speed, and thus an adjustable anti-aliasing filter is needed. This complicates the method considerably. In the digital re-sampling technique, the time signal is conventionally sampled together with some rotational reference signal. The time signal is then digitally re-sampled to the angle domain by interpolation techniques. The rotational reference signal can be acquired with a tachometer or an incremental pulse encoder.

The following picture shows conceptually how angle data re-sampling can be used to analyze vibrations from an engine during start up. Once the signal has been transformed into its angle domain, the FFT can be applied to analyze the order spectrum of the vibrations.

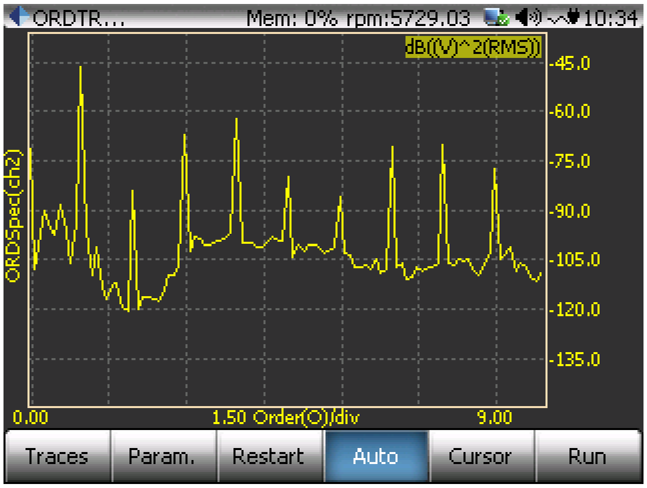


- Figure 25 Angular Data Re-sampling of a Chip Signal

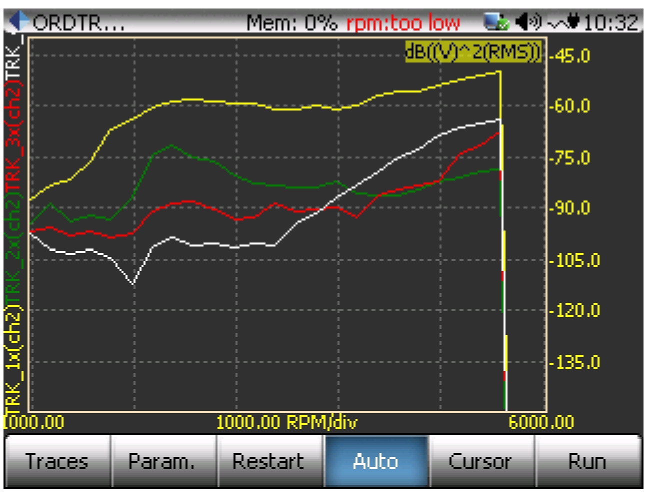
The last plot in the picture shows that the sampling rate will be determined by both instantaneous tacho speed and required analysis frequency range.

In the CoCo system, the order tracks and order spectrum are computed with a proprietary technology that combines digital re-sampling, data decimation, and interpolation, DFT and FFT calculations.

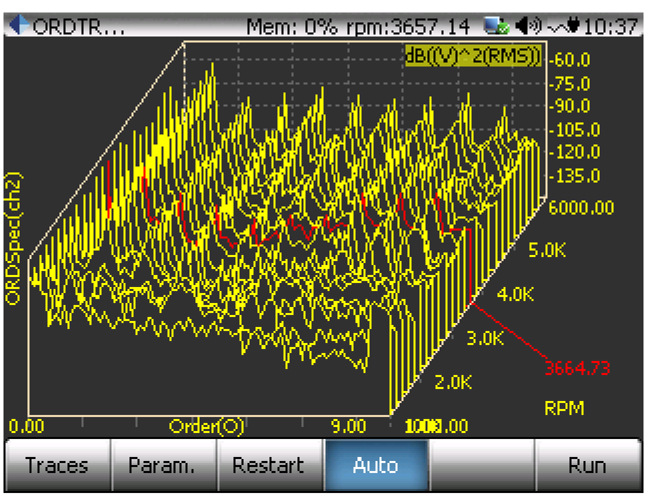
Three measurements can be generated from order tracking computation. The 3D RPM Order Spectrum is simply the a 3-dimensional view of the other two types of measurement.. Another way to visualize these types of plots is that the order spectrum is a cross section of the 3D plot along a fixed RPM value while the order track is a cross section along a fixed order number. The relationship of them is:



■ Figure 26. Typical order spectrum.



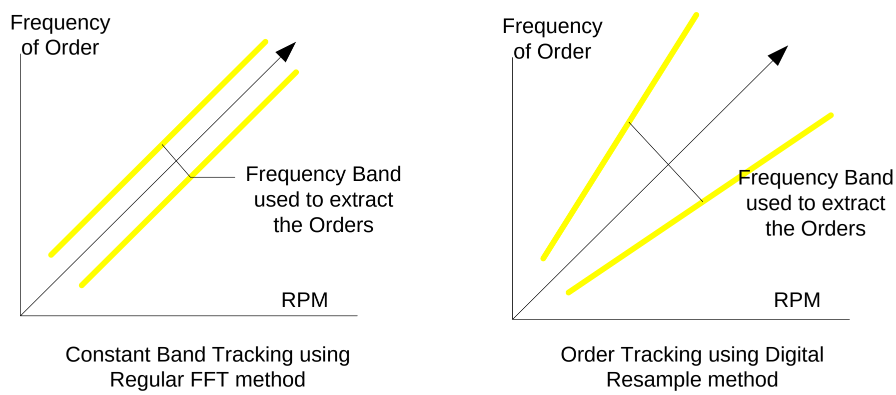
■ Figure 27. Typical order tracks.



■ Figure 28. Typical 3D order track waterfall plot.

An important concept that must be introduced now is called delta order, *ΔOrder*. In the FFT based frequency spectrum analysis, the frequency span and frequency resolution are fixed. The capability of discriminating frequency components is equal in both low and high frequency. In rotating machine analysis, we want to have better analysis resolution in the low frequency than that in high frequency. For example, if the rotating speed is at 60 RPM, we definitely care if the instrument can tell the difference between 1Hz (order 1) and 2Hz (order 2); on the contrary, if the rotating speed is at 6000 RPM, the user probably won't care if the instrument can discriminate the measurement between 100Hz (order 1) and 101Hz.

With the digital re-sampling technique, the order tracks and order spectrum are extracted based on a filter with equal *ΔOrder* instead of equal *ΔFrequency.* The concept is illustrated in the following figure:

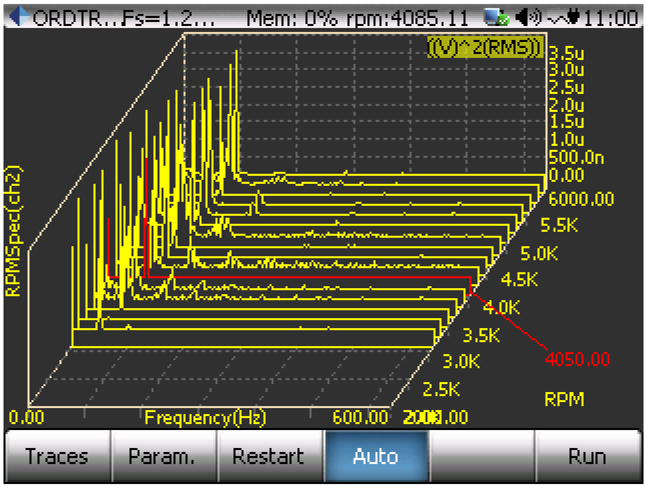


- Figure 29. Comparison of constant band tracking and digital re-sampling method.

In this figure, the left side shows when the order tracks are extracted using conventional FFT method with fixed resolution, the *ΔFrequency* of tracking filter will be fixed; the right side shows that if the order tracks are extracted using digital re-sampling, the *ΔFrequency* tracking filter will be increased proportionally with the RPM. Obviously, the method of digital re-sampling is more desirable in extracting the measurement of orders.

RPM Frequency Spectrum

While the order tracks and order spectra are developed to analyze the characteristics of the system on the order space, the measures of fixed bands are also helpful for analysis. Similar to the RMS time trace for a given frequency band with time as variable, the RMS trace can be extracted for a given frequency band with RPM as the independent variable. This is simply called an RPM Spectrum. An RPM Spectrum can be described as a 3D waterfall as shown below:

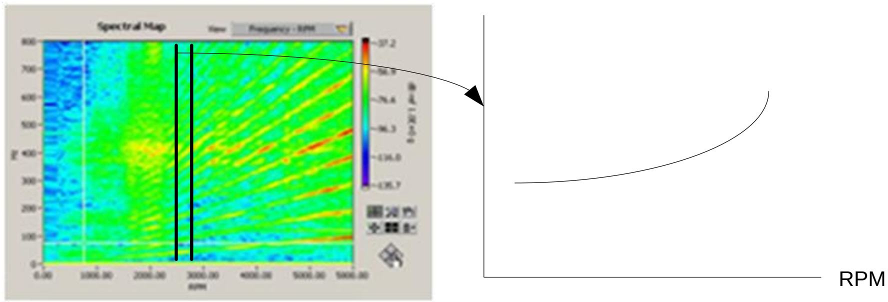


■ Figure 30 RPM spectrum.

The horizontal axis of the 3D RPM Spectrum is frequency. The Z axis is RPM and the measurement unit is usually EU_{rms}^2 or EU_{rms}. A color map can also be used to describe the magnitude of the whole range as shown below.

■ Figure 31 Color map of an RPM spectrum.

With a 3D RPM frequency spectrum, the instrument can extract the total energy over a fixed frequency band, and plot it with RPM as the independent variable. This is called *Fixed Band RPM Spectrum* as shown below.



Fixed Band RPM Spectrum

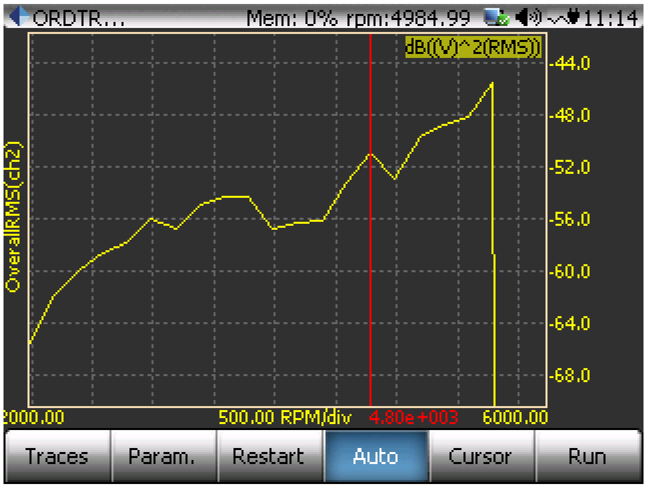
■ Figure 32 Fixed band RPM spectrum.

The measurement engineering unit of Fixed Band RPM Spectrum is EU_{rms}^2 or EU_{rms} representing the total power in a fixed band measured versus rotating speed change. This data is particularly useful to watch the total magnitude in a resonance area when the rotating speed of the shaft is changing. You can define the frequency band around the resonant frequency and perform a run up/down test. Both order tracking and order spectrum cannot extract the response magnitude of the resonance as accurately as a fixed band RPM spectrum because the bandwidth of the tracking filter of order tracking is not explicitly controlled.

Overall Level Measurement

In order tracking, about it is important to monitor the overall RMS level or power level of the measurement versus RPM. The overall level is a good reference for comparing with other signals such as order tracks or fixed band RPM spectrum.

Overall level can be in unit of RMS (EU_{rms}) or power (EU_{rms}^2). The horizontal axis is RPM. Below is a typical overall level display.



■ Figure 33.Overall RMS level plot.

Raw Data Time Streams

In many other order tracking software products, the user can either conduct real-time order tracking analysis, or record the data with other tools and then post process the order tracks, but not both at the same time. The CoCo is a high performance data recorder in addition to a real time analyzer and can do both at the same time. Continuous time streams of each input channels are always available even while order track data is computed..

Order Tracks with Phase

The Phase in Rotating Machine Analysis

Many mechanical faults are associated with certain orders, analyzing order magnitude and phase can help you detect mechanical faults directly. For example, a strong first order magnitude indicates imbalance in most cases. Analyzing the first order magnitude can help you identify the imbalance. Moreover, the magnitude and phase of the first order can help you correct the imbalance by adding weights on the appropriate rotor positions. However to fix such an imbalance problem requires phase information of order tracks. A list of the sources of vibration in the rotating machine are:

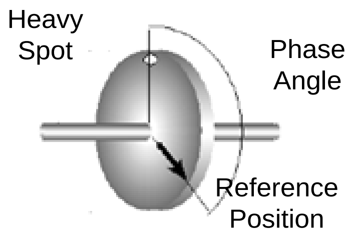
Order	Source of Problem
0.05X~0.35X	Diffuser Stall
0.43X~0.49X	Instability
0.5X	Rubbing
0.65X~0.95X	Impeller Stall
1X	Imbalance
1X+2X	Misalignment
(#Vane)X	Vane/Volute gap
(#Blades)X	Blade/Diffuser Gap

As previously discussed, an order track is the measurement taken for an order, i.e., normalized frequency, versus RPM. In most of the applications of engine related test, the phase information of order tracks are not important. In rotating machine analysis, the phase of the signal is vitally important.

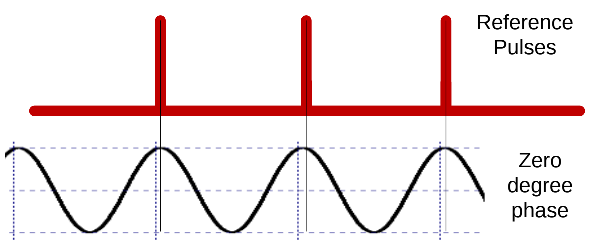
Phase is a relative measurement quantity and can only be measured with a pair of signals. It indicates the time delay at certain frequency between two signals. The phase value can be translated into the difference of relative angle, relative position or propagation time if additional information is given. When we refer to the phase information of one signal, we imply its phase is relative to a reference signal that was mentioned in context.

In rotating machine analysis, the phase of the first order of the rotor can be directly mapped to an angular difference between a signal and the reference signal. The reference signal can be another channel of measurement, or the tachometer signal. The phase difference between two waveforms is often called a phase shift or phase delay. A phase shift of 360 degrees is a time delay of one cycle, or one period of the wave, which actually amounts to no phase shift at all. A phase shift of 90 degrees is a shift of 1/4 of the period of the wave, etc. Phase shift may be considered positive or negative, i.e., one waveform may be delayed relative to another one, or one waveform may be advanced relative to another one. These conditions are called phase lag and phase lead respectively.

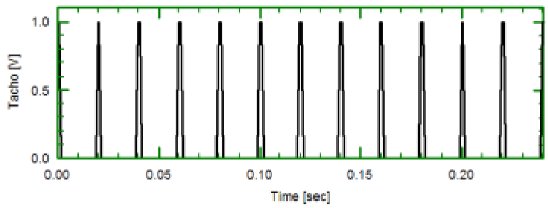
An example of this is the phase of an imbalance component in a rotor with reference to a fixed point on the rotor, such as a key way. To measure this phase, a trigger-pulse must be generated from a certain reference point on the shaft. This trigger can be generated by a tachometer or some type of optical or magnetic probe that senses a discontinuity on the rotor, and is sometimes called a "tach" pulse.

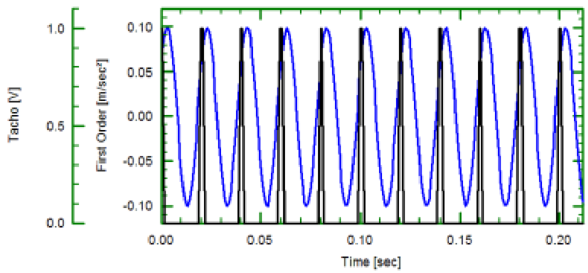


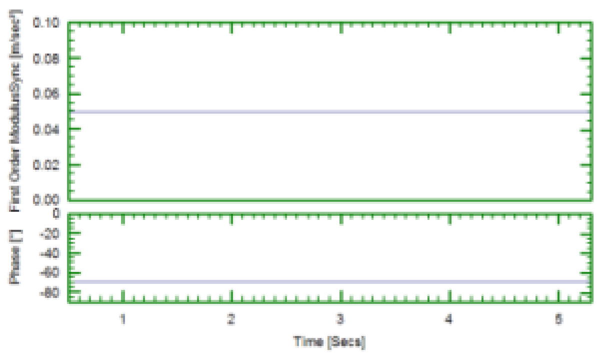
A zero degree phase delay at a frequency can be depicted as a series of pulses overlaid with a sine wave where the pulse edge is exactly located in peak position of the sine wave.



In Figure X a section of the tacho signal is shown on its own and then overlaid on the vibration signal. The tacho signal in this example crosses the vibration signal at exactly the same point on each cycle. If the phase of the vibration signal were to change then its position relative to the tacho pulse would also change. Extracting the first order modulus and phase, as before, gives the curves shown in Figure Y. The phase is now constant near -60^o as it should be for such a signal. Because the rotating period of the signal is about 20ms, -60^o corresponds to a 20*60/360=3.3ms delay.







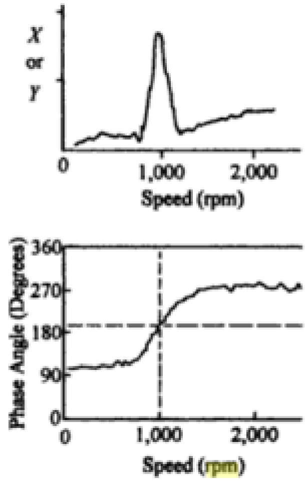
The phase measurement at higher orders will have the similar physical interpretation although they are difficult to comprehend intuitively.

It must be noted that the order tracks with phase, or Complex Order Tracks by name, are not regular complex signals as frequency response or cross spectrum. They are really auto-spectra with assigned phase. These synthesized signals can certainly be viewed as a complex signal using tools including Bode Plot, polar and orbit diagram. However the user must keep in mind that the magnitude and phase of a complex order track are calculated separately.

In the following sections, we will present how the order tracks with phase can be presented graphically with the Bode, polar and orbit plot.

Bode Plot

The term Bode Plot is borrowed from the field of control theory, referring to a plot of magnitude and phase angle between the input and output verses frequency of a control system. Many in the rotating machine vibration industry have adopted this term to describe the steady-state vibration response amplitude and phase angle versus rotational speed (RPM). It turns out that the Bode Plot is the best way to describe order tracks with phase. You typically use Bode plots for transient analysis in both start-up and coast-down conditions. A Bode plot can help to identify the resonance speed of a rotor or examine the rotor dynamics on an order basis. A typical Bode Plot for an order track is shown below:

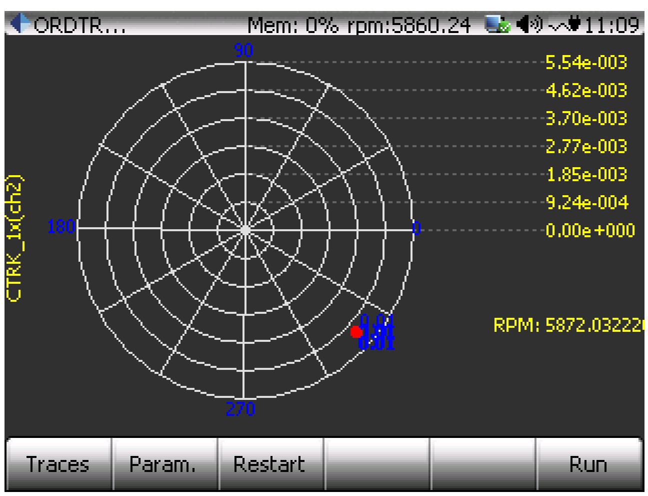


- Figure 34 This picture must be replaced

In the CoCo system, after the order tracks are acquired together with their phase information then the Bode Plot can show one or multiple tracks.

Polar Plot

The Polar Plot is another useful tool to view the order tracks for both amplitude and phase information. A polar plot draws the amplitude and phase on a polar coordinate. A typical polar plot is shown below.



- Figure 35. Polar plot shows magnitude and phase on a polar axis.

In the polar plot, the dot shows the current order track value. The distance between the dot and the center indicates the magnitude of the order track while the angle corresponds to the phase measurement. The polar plot only shows the instantaneous measurement. It cannot keep the history versus RPM.

The Polar plot is often used to visually indicate the imbalance of the rotor. Polar plots and Bode plots often are combined to describe the rotating speed vector signal locus during speed changes. A Bode plot provides excellent change visibility with respect to speed, while the polar plot shows improved phase variation resolution.

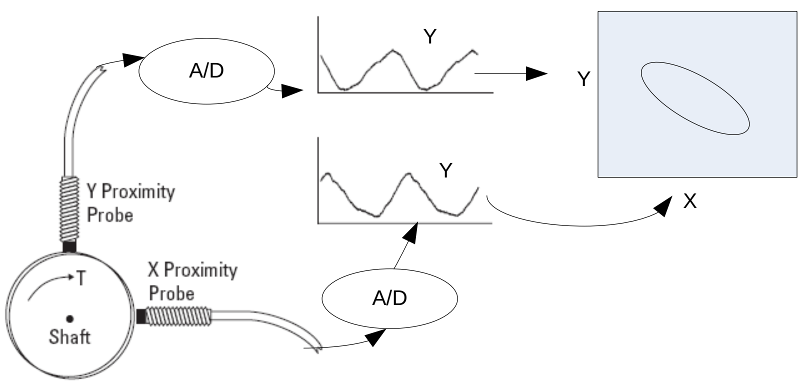
Orbit Display

The Orbit display uses two data channels in the time domain. The signals from two channels are drawn on an X and Y plane to display the shaft position change versus angle of rotation. Orbit displays give a two dimensional visual picture of the motion of a rotating shaft.

A well balanced shaft with no movement in any direction and would produce a dot in the middle of the plot. The shaft movement can give an indication of the vibration source e.g. if there is a lot of up/down movement it may be that the machine feet are not bolted down tightly enough.

To create an orbit plot you need to take a dual channel simultaneous measurement to capture data at the horizontal and vertical axes at the same time. The displacement or acceleration sensors must be placed 90° apart from each other.

Orbit display uses a pair of measurement in time domain. It does not need the technique of *order tracking*.



The orbit display is similar to the Polar display in that it only displays the instantaneous status at the current RPM. In theory, the orbit display does not need a tachometer or another reference signal because X and Y reference to each other.

Summary

With the Bode, polar and orbit plots, the order tracks with phase can be presented visually. These are effective tools used for rotating machine analysis. Bode plot is mostly used in the Run Up/Coast Down process. Polar plot and orbit, which only show the instantaneous status of an order at current RPM, are adequate for applications at steady or quasi-steady rotating speed such as dynamic balancing.

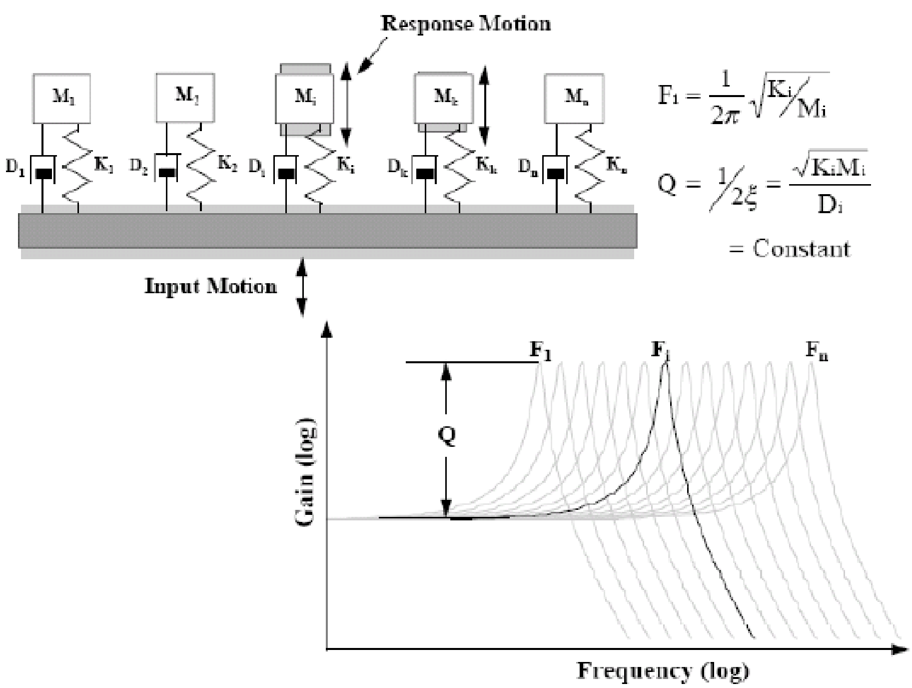
6. SHOCK RESPONSE SPECTRUM ANALYSIS

A **Shock** Response Spectrum (SRS) is a graphical representation of an arbitrary transient acceleration input, such as shock in terms of how a Single Degree Of Freedom (SDOF) system (like a mass on a spring) responds to that input. Actually, it shows the peak acceleration response of an infinite number of SDOFs, each of which have different natural frequencies. Acceleration response amplitude is represented on the vertical axis, and natural frequency of any given SDOF is shown on the horizontal axis.

A SRS is generated from a shock waveform using the following process:

1. Pick a damping ratio for the SRS
2. Assume a hypothetical Single Degree of Freedom System (SDOF), with a damped natural frequency of x Hz
3. Calculate (by time base simulation or something more subtle) the maximum instantaneous absolute acceleration experienced by the mass element of your SDOFs at any time during (or after) exposure to the shock in question. Plot this in g's (g's are standard, but pick any unit of acceleration you want) against the frequency (x) of the hypothetical system.
4. Repeat steps 2 and 3 for other values of X, for example, logarithmically up to 1000x.

The resulting plot of peak acceleration vs. test system frequency is called a Shock Response Spectrum, or SRS. This process can be depicted in the following picture:



■ Figure 36. Illustration of multi-degree of freedom system model used to compute SRS.

A SDOF mechanical system consists of the following components:

- Mass, whose value is represented with the variable, M
- Spring, whose stiffness is represented with the variable, K
- Damper, whose damping coefficient is represented with the variable C.

The resonance frequency, Fi, and the critical damping factor, ζ, characterize a SDOF system, where:

$$Fi = \frac{1}{2\pi}\sqrt{\frac{K}{M}}$$

$$\zeta = \frac{\mathrm{c}}{2\sqrt{\mathrm{KM}}}$$

For light damping ratio where ζ is less than or equal to 0.05, the peak value of the frequency response occurs in the immediate vicinity of Fi and is given by the following equation, where Q is the quality factor:

$$Q = \frac{1}{2\zeta}$$

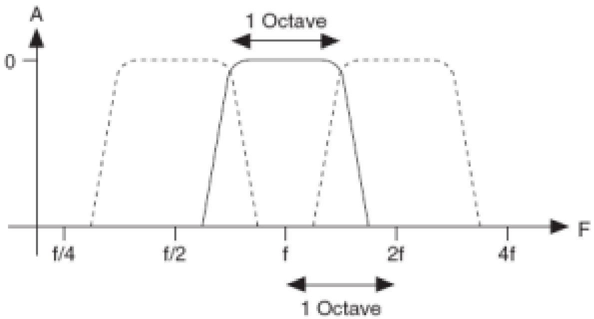
Any transient waveform can be presented as an SRS, but the relationship is not unique; many different transient waveforms can produce the same SRS (something one can take advantage of through a process called "Shock Synthesis"). The SRS does not contain all the information about the transient waveform from which it was created because it only tracks the peak instantaneous accelerations.

Different damping ratios produce different SRSs for the same shock waveform. Zero damping will produce a maximum response. Very high damping produces a very flat SRS. The level of damping is demonstrated by the "quality factor", Q which can also be thought of transmissibility in sinusoidal vibration case. A damping ratio of 5% results in a Q of 10. An SRS plot is incomplete if it doesn't specify the assumed Q or damping ratio value.

Frequency Spacing of SRS Bins

Usually the SRS spectrum consists of multiple bins distributed evenly in the logarithmic frequency scale. The frequency distribution can be defined by two numbers: a reference frequency and the fractional octave number, such as 1/1, 1/3 or 1/6.

An octave is a doubling of frequency. For example, frequencies of 250 Hz and 500 Hz are one octave apart, as are frequencies of 1 kHz and 2 kHz.



■ Figure 37. Full octave filter shape.

The proportional bandwidth property will divide the frequency information uniformly over a log scale. It is very useful in analyzing a variety of natural systems. For example, the human response to noise and vibration is very non-linear and many mechanical systems have a behavior that is best characterized by proportional bandwidth analysis.

To gain finer frequency resolution, the frequency range can be divided into proportional bandwidths that are a fraction of an octave. For example, with 1/3 octave spacing, there are 3 SDOF filters per octave. In general, for 1/N fraction octave, there are N band pass filters per octave such that:

$$f_{c\,j+1} = f_{c\,j} * 2^{1/N}$$

where 1/N is called the fractional octave number. f_r, the ***reference frequency***, is simply any of the frequencies $f_{c\,j}$. All other center frequencies of SDOF filters reference to this frequency. When the reference frequency and the fractional octave number are fixed, the frequency distribution over the whole frequency range is determined.

SRS Measurement Quantities

Measurement quantities available to the CoCo SRS test are: time stream of each channel (raw data), block captured time signals and three SRS of each channel.

Time streams: this is the same as any other applications on the CoCo. Time streams are always available for viewing and recording. It is a very useful tool to observe whether the input signals are in the valid range. The recorded sine wave can be used for further post-processing. In CoCo, the time streams are often denoted as ch1, ch2 etc.

Block time signals: These are the block captured signals that are used for SRS analysis. Acquisition Mode will control how the block time signals are acquired.

SRS: Shock Response Spectra will be calculated for each block time signals. The engineering unit of the spectrum is determined by the sensor used by the input channel. In CoCo, the spectra are often denoted as three types: Maximum Positive spectrum; Maximum Positive spectrum and Maximum-Maximum spectrum.

Maximum Positive Spectrum: This is the largest positive response due to the transient input, without reference to the duration of the input.

Maximum Negative Spectrum: This is the largest negative response due to the transient input, without reference to the duration of the input.

Maximax Spectrum: this is the enveloge of the absolute values of the positive and negative spectra. It is the most often used SRS type. The log-log Maximax is the universally accepted format for SRS presentation.

Other common SRS measures include the so called Primiary SRS, Residue SRS and Composite SRS. The CoCo only calculates the Composite SRS.

7. AUTOMATED TEST AND LIMIT CHECK

Automated limit testing allows engineers and technicians to set up a pass/fail measurement on Any measured signal. This feature automates the process of determining whether an acquired signal meets, or is within a given set of criteria.

A limit test typically consists of comparing a waveform to upper and lower boundaries which the measured waveform must not cross. These boundaries are typically defined by the user to specify a tolerance band around a waveform. If any part of the waveform falls outside the limit, the software returns a failure message and the location of the failure on the waveform.

Application Examples

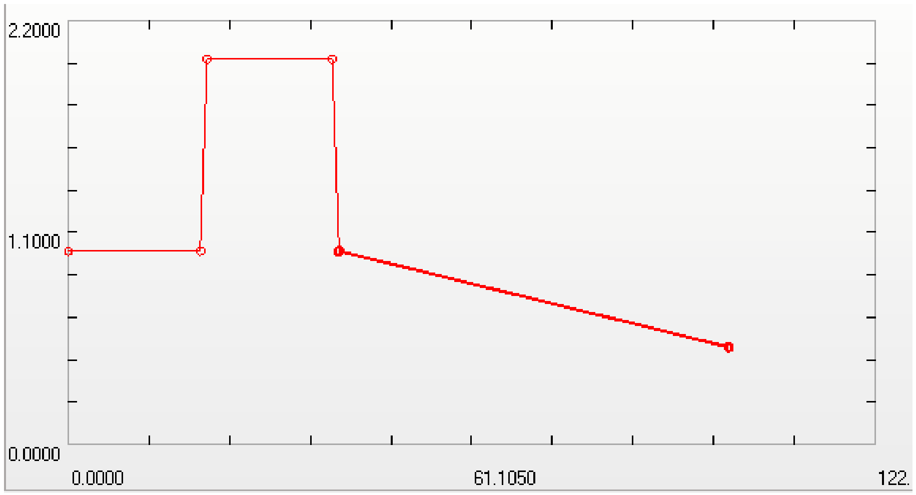
A common example for automated testing is related to structural testing. When excited, a structure will resonate at its natural frequencies. Structures can be excited through impact or by other means. Structural defects can result in a shift in resonant peaks. Therefore, in structural tests, frequency 'alarms' are used to monitor the frequency response in areas of spectral interest.

Another example of automated testing is related to rotating machinery. Rotating or moving assemblies produce vibration and noise patterns that can be examined to identify the fingerprint of 'good quality.' Product defects will cause additional spectral peaks, or changes in peak levels. Therefore, in 'self-excited' product testing, 'level alarms', which can be set to trigger on peak or RMS values, are placed around the areas of spectral interest. Unwanted signals, from background noise, are therefore ignored.

Testing Limit Signals and Testing Schedule

Automated testing can be performed on a wide variety of signals including a time domain capture, an auto-power spectrum, an octave spectrum, an order track signal or a frequency response. The CoCo instrument compares the limits to the live measured signal in real-time, after every single frame of measurement. If the limits are exceeded, the CoCo takes the appropriate actions based on the user setup.

An upper and or lower limit can be applied to a signal to be tested. Limit signals are constructed by defining breakpoints. A breakpoint is controlled by a pair of X/Y values. Figure 38 shows a typical automated test limit signal with 4 breakpoints.



■ Figure 38. Typical automated test limit signal with 4 breakpoints.

To automatically control the limit checking test, a ***testing schedule*** is developed for CoCo. The testing schedule defines the various operations to automate the process. For example the testing schedule can tell the instrument when the limit checking will be turned on, when it will be turned off and for how long the test will be conducted.

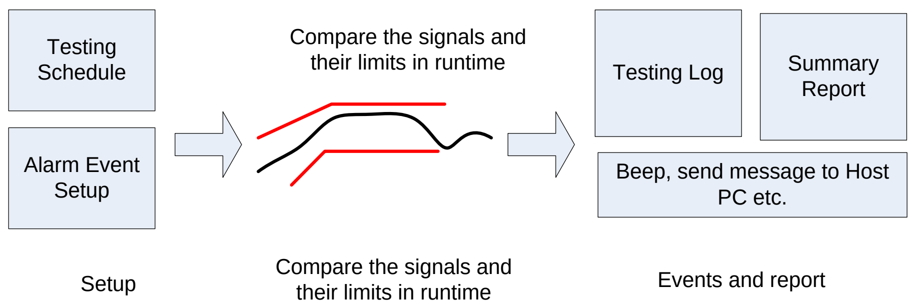
To record the events of the test a Testing Log and a summary report are needed. The Testing Log records the important events, including whether the limits are exceeded, in chronicle order. The Summary Report provides the status of limiting check since the last time when the test was started.

When the limit signal is exceeded then a user defined limit alarm event will be triggered. This can include an audible beep, Save Signals, send messages and so on.

To summarize, an automated limit checking test requires the following building blocks:

- At least one test signal
- At least one limit signal applied to the measured signals
- A testing schedule
- A testing log and summary report
- A setup for the limit alarm events

Figure 39 illustrates the automated testing process



■ Figure 39. Illustration of automatic testing process.

Test Signals: Any block signal can be used for testing. Typically the test signals are time captured blocks, auto power spectra, frequency response, octave spectra or order tracks. Time streams are not used for limiting test. **Limit signals**: Limit signals including upper and/or lower limits are defined in the CSA Editor. Limit signals are applied to testing signals. Up to a maximum of 64 segments can be defined for each limit signal. The maximum number of limit signals is 64.

Testing Schedule: The testing schedule automatically controls the test using an event driven process. Multiple testing schedules can be developed and one is executed at a time. Testing schedule event entries include: Loop/End-Loop, Set Sampling Rate, Set All Input Mode, Run Duration, Hold, Limit Check on, Limit Check off, Start Recording, Stop Recording, Save Signals, Turn Signal Source On and Turn Signal Source Off.

Testing Log: A log file is automatically created for each run of the schedule to record major events.

Limit Check Alarm Event Setup: Events include an audible beep from the CoCo, CoCo screen flashing, entry into the Testing Log, send message to host PC via EDM software and Save Signals.

When a limit is exceeded, the predefined events are triggered. For example, the CoCo may beep, flash the screen, save the signals to the storage device, or send the message to the host PC.

Networked CoCo used for Automated Test

CoCo has an Ethernet network interface that provides a unique advantage that multiple CoCo units can be connected remotely using an Ethernet network. This is particularly useful for vibration monitoring or production test that requires long distance access.

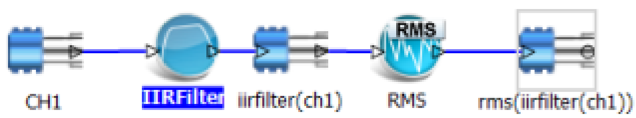
■ Figure 40. Remote operation for multiple CoCo units with automated testing.

When multiple CoCo units are managed through the EDM software, the host computer can record or react to the alarm events from each remote CoCo unit.

8. REAL TIME DIGITAL FILTERS

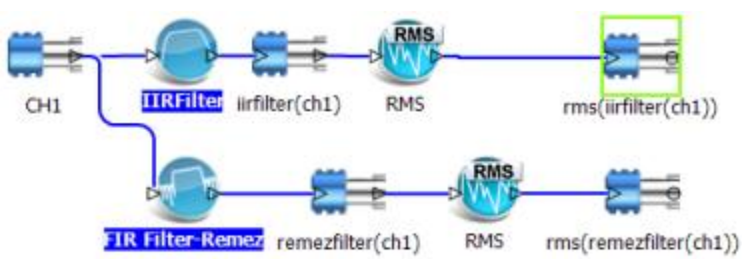
Real Time Digital Filters is a powerful analysis tool that can be used to filter a measured signal in real time and then apply the FFT and time based analysis built into the CoCo. You can precisely define the filter characteristics to meet your specific application. The filter definition is performed in the EDM software and the filter is included in the CSA file that is downloaded to the CoCo. This capability in a small portable unit makes the CoCo a very powerful analysis tool.

For example, a user might want to look at the energy distribution over time, for a specific band of frequencies instead of the entire frequency spectrum from zero to the maximum sampling rate. This can be done by creating a band-pass filter then applying an RMS estimator to the output of the filter. Figure 41 shows the graphical representation of this process which is used to define the real time filter in the EDM software. The icon on the left, CH1 represents the native measured time stream. It is connected to an IIR Filter which computes a signal named iirfilter (ch1) which is connected to an RMS estimator. The output of the RMS estimator is a signal named rms(iirfilter(ch1)). The EDM software will be discussed in more detail later.



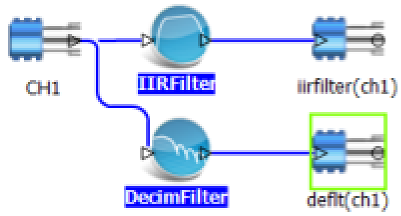
- Figure 41. Example real time digital filter application.

Another example is that the user might want to look at the frequency energy over 100Hz to 200Hz and 1000Hz to 2000Hz separately. This can be done by deriving two output streams from the native channel 1, then applying the band-pass filter to each path as shown in Figure 42.



- Figure 42. Digital Real Time Filter example with two output streams.

In another example, a user might want to look at the very fast time characteristics of a channel at high frequency, and the same channel at a very low sampling rate. This can be done by applying a decimation filter to the native time stream as shown in Figure 43 . The native channel time stream is split into two streams so the signal from the same channel is recorded at both high and lower sampling rates.



■ Figure 43. Example computing high and low sampling rate with a decimation filter.

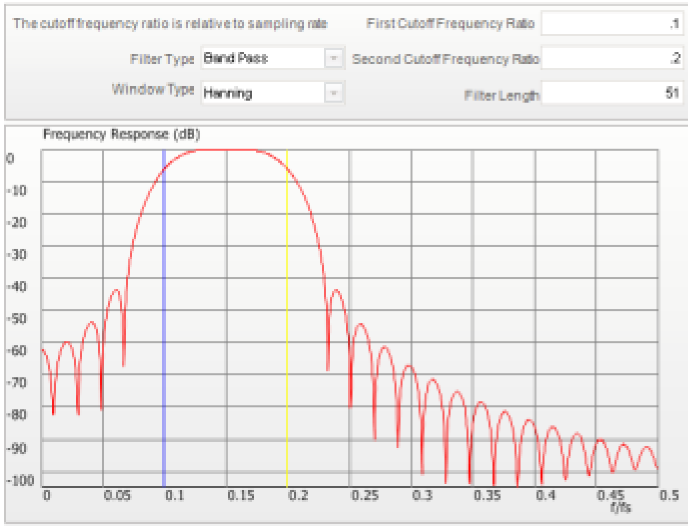
The Real Time Digital Filters option includes three types of digital filters: FIR, IIR and decimation filters. For FIR and IIR filter, you can specify low-pass, high-pass, band-pass or band-stop types with several different methods. This chapter first explains the theory about the filter design, and then introduces the operations within the CSA Editor and CoCo hardware.

Real Time Filter implementation can be divided into two steps that include the filter definition on the EDM software, and secondly download and run the CSA on the CoCo hardware.
You design a filter based on certain criteria such as cut-off frequency, pass band ripple, attenuation level and so on. The EDM software walks you through this process. The outcome of this design process is simply a number of filter coefficients that represent the filter which are included in a CSA. The software will upload the CSA including the filter coefficients to the CoCo hardware.

After filter is defined and the coefficients downloaded to the CoCo hardware you can run the CSA. When the CoCo is running, the filter coefficients created in the filter design process will be used. The time streams will pass through the filters and generate new time stream signals.

The goal of filter design is to calculate a series of filter coefficients, also known as "taps" based on the user specified criteria. The criteria are often described by following variables:

- **Number of filter coefficients**: this is also known as the order of the filter. The filter order defines how many coefficients are required to define the filter. A lower order filter consists of a fewer number of coefficients. A low order filter responds relatively faster than a higher order filter, that is there is less of a time lag in the output of the filter.
- **Cutoff frequencies**: For low pass or high pass filters, only one cutoff frequency is needed. Band pass or band stop filters require two cutoff frequencies to fully define the filter shape. Figure 44 shows a typical band pass filter design with the two cutoff frequencies set to approximately 0.1 and 0.2 Hz as indicated by the blue and yellow vertical lines.
- **Attenuation of stop band in dB**: This defines how much of the input signal is cut out of the output at the rejected frequencies. In theory the higher the attenuation the better. In Figure 44 the stop ban attenuation is > 40 dB as seen from the highest side lobe just below 0.25 Hz.
- **Pass band ripple**: Ripple is an unavoidable characteristic if a digital filter. It refers to the fluctuation in the filter shape at transition frequencies. If a very flat filter is required then it can be specified by choosing a very low ripple. In Figure 44 ripple is seen in the stop band and no ripple is evident in the pass band. Ideally the pass band should be very flat and some ripple is tolerable in the stop band.
- **Width of transition bands**: This refers to the filter shape between a band pass and a band stop region. Ideally this transition band should be very small. However, a very narrow transitional band requires a higher order filter which affects the filter response time and can also affect ripple. In Figure 44 the transition bands are between 0.05 to 0.1 and 0.2 to 0.25.



■ Figure 44. Filter design shows cutoff frequencies, ripple, band stop attenuation.

In most cases filter design includes making tradeoffs between minimizing the filter order, ripple, transition band width and response time. Not all can be satisfied at the same time. Filter design can be an iterative process and experience is helpful.

FIR Real Time Digital Filters

Finite Impulse Response (FIR) filters have the distinctive trait that their impulse response lasts for a finite duration of time as opposed to, an Infinite Impulse Response (IIR) filters whose impulse response is infinite in duration. This trait is due to the fact that there are no feedback paths in the FIR filter. FIR filters offer several advantages over IIR filters:

- Completely constant group delay throughout the frequency spectrum. Group delay refers to the time delay between when a signal goes into the filter and when it comes out. Constant group delay means that an input signal will come out of the filter with all parts delayed the same amount with no distortion.
- Complete stability at all frequencies regardless of the size of the filter.

FIR filters also have some disadvantages as well:

- The frequency response is not as easily defined as it is with IIR filters
- The number of coefficients required to meet a frequency specification may be far larger than that required for IIR filters.

CoCo allows up to 128 taps (orders) for the real-time FIR filter.

A digital filter can be understood by considering the difference equation which defines how the input signal is related to the output signal as

$$y[n] = b_0 x[n] + b_1 x[n-1] + \cdots + b_N x[n-N]$$

where $x[n]$ is the current input signal sample, $x[n-1]$ is the previous signal sample and $x[n-N]$ is the last sample in the series. The series multiplies the most recent N+1 samples with associated the N+1 filter coefficients. , $y[n]$ is the current output signal and b_i are the filter coefficients. The number N is known as the *filter order*; an N^{th}-order filter has $(N + 1)$ terms on the right-hand side and N+1 filter coefficients also referred to as "*taps"*.

This equation illustrates why a higher order filter has a slower response time. It takes more samples and therefore more time for an event to work its way though the series until the output is no longer affected by the event as compared to a lower order filter with fewer coefficients.

The previous equation can also be expressed as a convolution of the filter coefficients and the input signal.

$$y[n] = \sum_{i=0}^{N} b_i x[n-i]$$

The Impulse Response of the filter shows how the historical data affect the current filtered value. The longer the impulse response, the farther the old data will affect the current filtered value. To find the impulse response we set

$$x[n] = \delta[n]$$

where $\delta[n]$ is the Kronecker delta impulse. The equation below shows that the impulse response for an FIR filter is simply the set of coefficients b_n, as follows

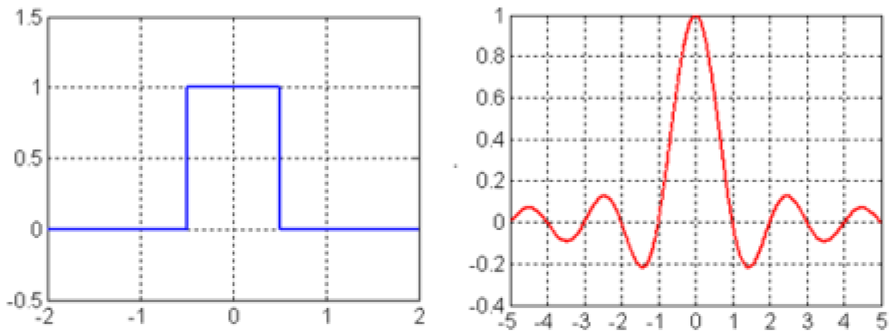
$$h[n] = \sum_{i=0}^{N} b_i \delta[n-i] = b_n \qquad for\ n = 0\ to\ N$$

FIR filters are clearly stable, since the output is a sum of a finite number of finite multiples of the input values, so can be no greater than $\sum_{n=0}^{N} |b_n|$ times the largest value appearing in the input.

Data Windows FIR Filter Design

In the academic world, hundreds of methods are available to design an FIR filter to meet various criteria. The EDM includes the most popular filter design methods: Data Window and Remez. Both methods are discussed below.

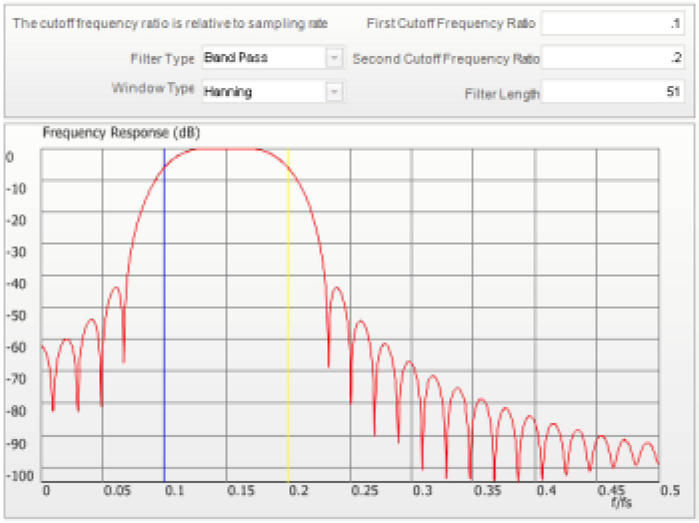
The Data Window FIR Filter Design method is the easiest to understand. The name "Window" comes from the fact that these filters are created by scaling a *sinc* (SIN(X)/X) function with a window such as a Hanning, Flat Top, etc. to produce the desired frequency effect.



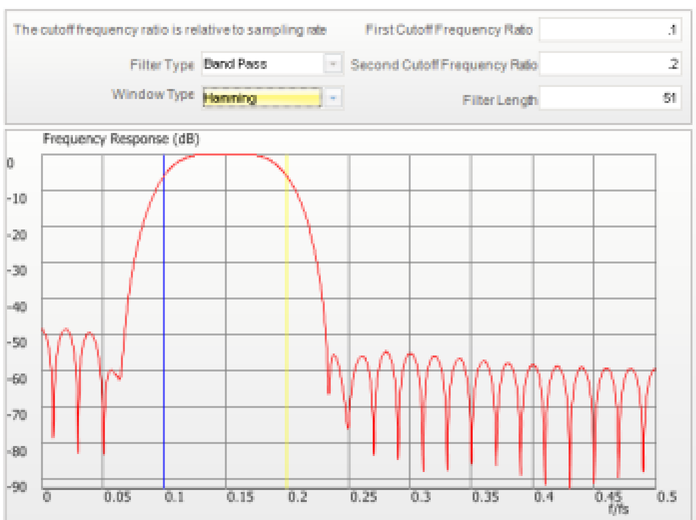
- Figure 45 Sinc function is the Fourier transform of a square shape.

A data window FIR filter is generated by starting with an ideal "brick-wall" shaped filter, that is a filter with vertical edges or zero transition band width as shown on the left in Figure 45. The brick-wall filter is specified by the cutoff frequencies and has a band pass amplitude of 1 and a stop band amplitude of zero. The problem with the ideal brick-wall filter is that the time response oscillates forever and it requires an infinite number of filter coefficients. This ideal filter can be modified by applying a data window to force the time response to decay in a finite time. Of course this degrades the shape of the ideal brick-wall filter performance. It introduces ripple, increases the transition band width and increases the stop band attenuation. However it allows the filter to be defined by a finite number of filter coefficients. The filter performance can be modified by using different data windowing functions and making the tradeoff between filter order and response time. The user must choose these settings during the filter design.

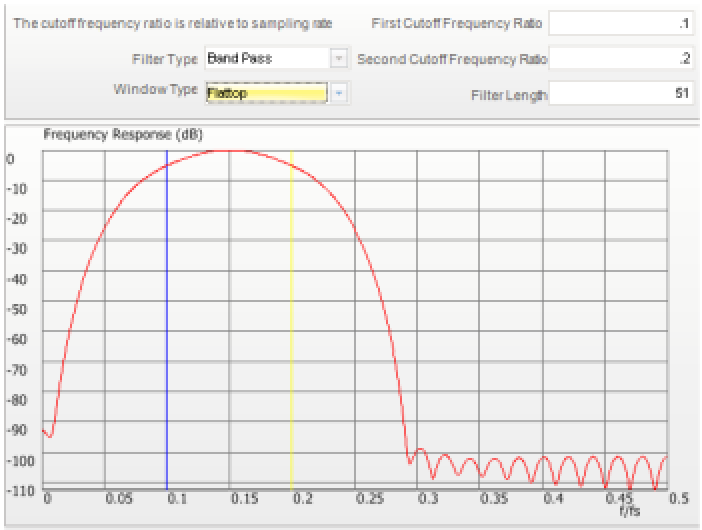
Figure 46 through Figure 51 show a comparison of different data window choices for the same filter settings. In all cases the low and high cutoff frequencies are 0.1 and 0.2 relative to the sampling frequency. The number of filter taps is 51.



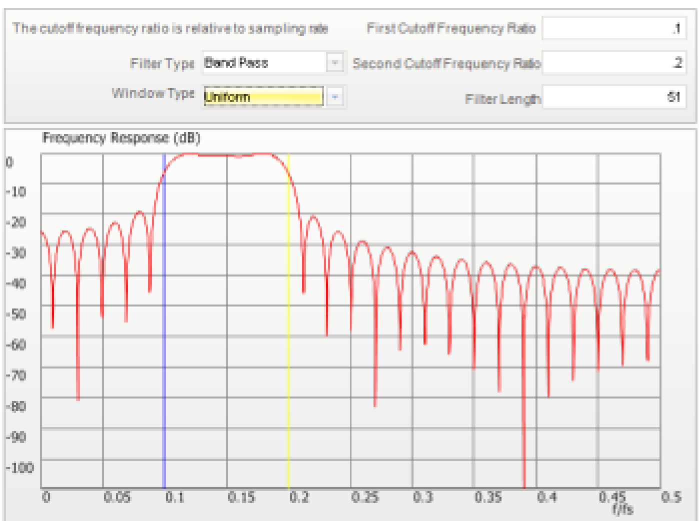
■ Figure 46 Hanning window method.



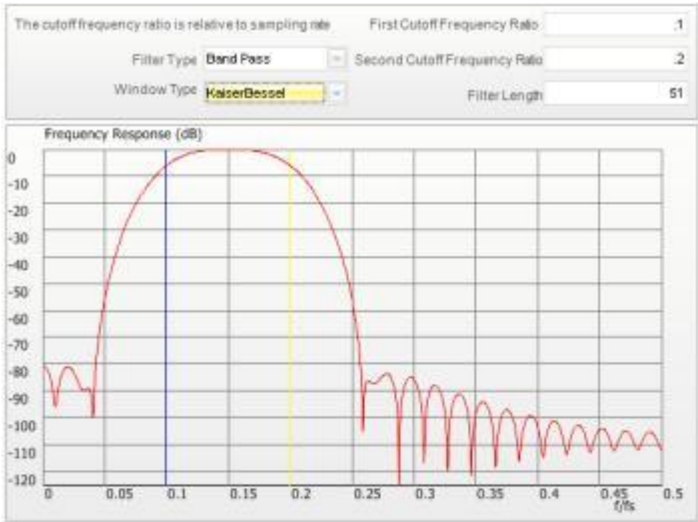
■ Figure 47 Hamming window method.



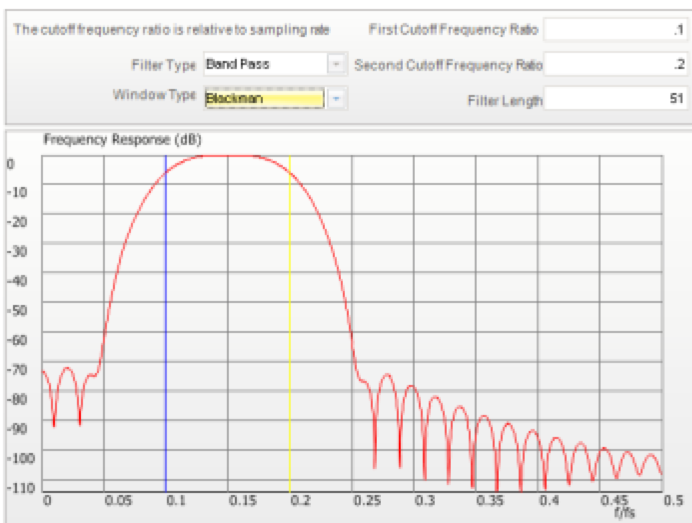
■ Figure 48 Flattop window method.



■ Figure 49 Uniform window method.



■ Figure 50 Kaiser Bessel window method.



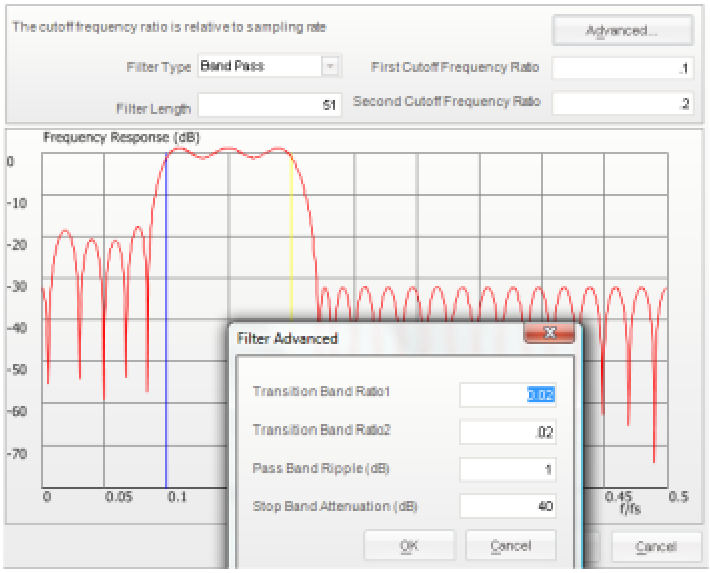
■ Figure 51 Blackman window method.

As shown in the pictures, different window methods produce different filter performance, i.e., different attenuation of the main lobe and side lobes. The best data window choice depends on your specific application. Refer to the Basic Spectral Analysis section for a comparison of windowing functions.

Remez Filter Design

The Remez Filter is a different method for designing an FIR filter. It is more computationally intensive than the data window method. A Remez filter is generated with iterative error-reducing algorithms designed to reduce the pass band error. In addition to allowing stop band ratio and frequency definition, the Remez filter allows the "Ripple Ratio" to be defined as a user specified parameter.

Figure 52 shows an example of a filter design using the Remez method in the EDM software. The low and high cutoff frequencies are 0.1 and 0.2 relative to the sampling frequency. The number of filter taps is 51.



- Figure 52. Ramez FIR Filter design dialog.

The software is intelligent enough to automatically calculate the total FIR filter length based on these criteria. For example if the user asks for very high attention, very small ripple or very sharp transition band, the filter length will go very high. The user must make tradeoffs between these parameters so that appropriate filter length can be generated and used.

IIR Real Time Digital Filters

Infinite impulse response (IIR) filters have the trait that their impulse response decays very slowly but theoretically lasts forever. This is due to the fact that the filter input includes the measured signal and also the filter output creating a feedback path which results in the infinite impulse duration. This is in contrast to finite impulse response filters (FIR) which have fixed-duration impulse responses.

The design procedures for IIR filters is somewhat more complicated than FIR filter design because there is no direct design method like the data window method for FIR filters. Instead IIR filters are typically designed by starting with an ideal analog filter in terms of the frequency response characteristics such as the Chebyshev, Butterworth, or Bessel filter. Then the analog filter is converted into a digital filter using a method known as the Bilinear transformation or impulse invariance method.

An IIR digital filter can be understood by considering the difference equation which defines how the input signal is related to the output signal as

$$y[n] = b_0 x[n] + b_1 x[n-1] + \cdots + b_p x[n-P] - a_1 y[n-1] - \cdots - a_q y[n-Q]$$

where P is the feed-forward filter order, b_i are the feed-forward filter coefficients, Q is the feedback filter order, a_i are the feedback filter coefficients, x[n] is the input signal and y[n] is the output signal.

The previous equation can also be expressed as a convolution of the filter coefficients and the input signal.

$$y[n] = \sum_{i=0}^{P} b_i x[n-i] - \sum_{j=0}^{Q} a_j y[n-j]$$

which, when rearranged, becomes:

$$\sum_{j=0}^{Q} a_j y[n-j] = \sum_{i=0}^{P} b_i x[n-i] \quad if\ we\ let\ a_0 = 1$$

To find the transfer function of the filter, we first take the Z-transform of each side of the above equation, where we use the time-shift property to obtain:

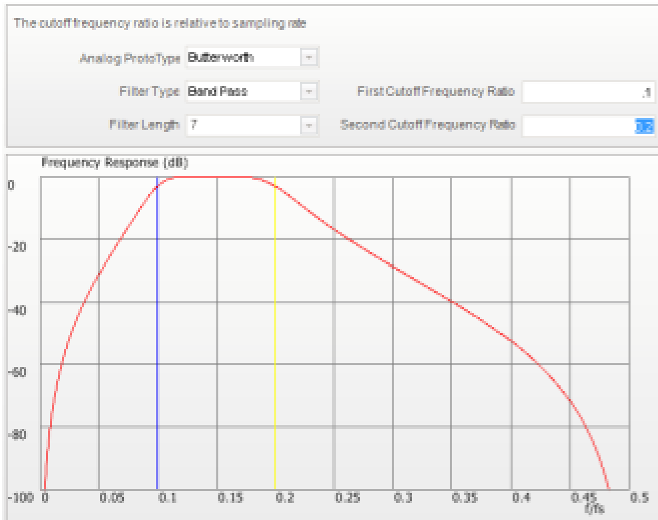
$$\sum_{j=0}^{Q} a_j z^{-j} Y(z) = \sum_{i=0}^{P} b_i z^{-i} X(z)$$

We define the transfer function to be:

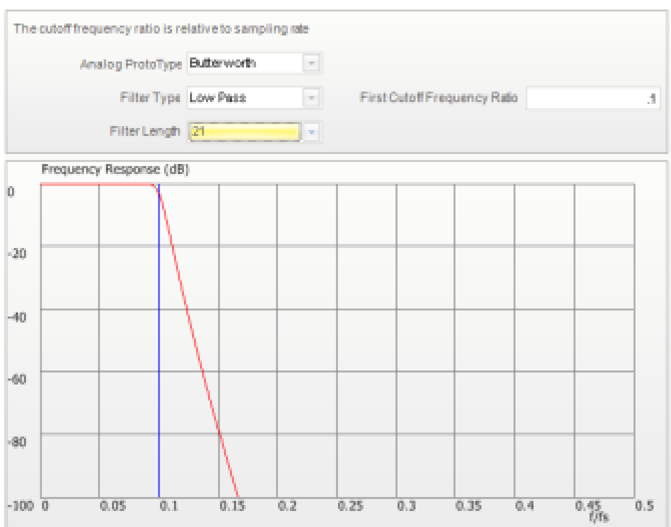
$$H(z) = \frac{Y(z)}{X(z)} = \frac{\sum_{i=0}^{P} b_i z^{-i}}{\sum_{j=0}^{Q} a_j z^{-j}}$$

The transfer function gives the frequency response that relates the input to the output magnitude and phase relationship.

Various analog filter types can be used as the basis for the IIR filter. The Butterworth Filter is the filter type that results in the flattest pass band and contains a moderate group delay. Below are examples of Butterworth low pass and band pass filters.

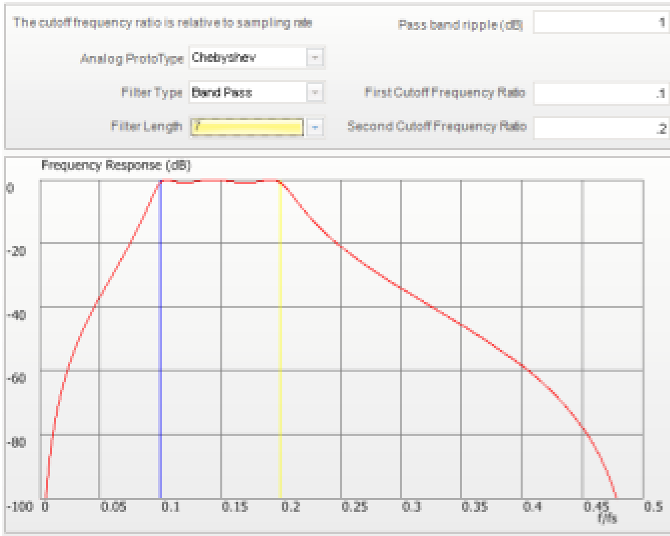


■ Figure 53. Butterworth band pass filter.

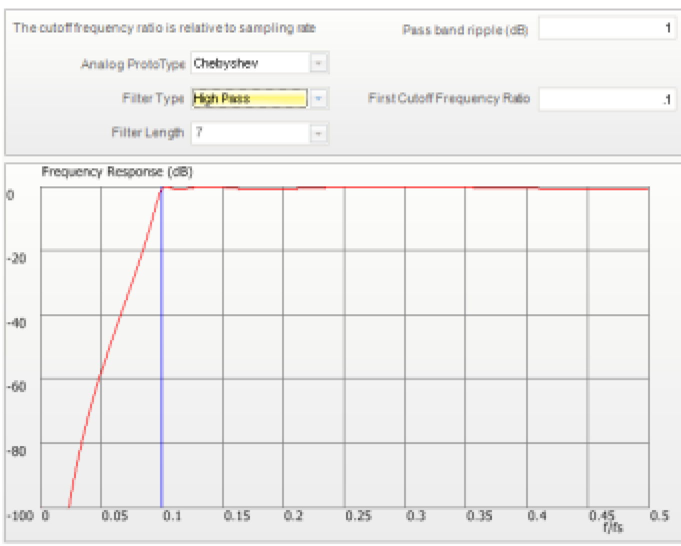


■ Figure 54. Butterworth low pass filter.

The Chebyshev Type I Filter results in the sharpest pass band cut off and contains the largest group delay. The most notable feature of this filter is the significant ripple in the pass band magnitude. A standard Chebyshev Type I Filter's pass band attenuation is defined to be the same value as the pass band ripple amplitude. Below are examples of Chebyshev Type I band pass and high pass filters.

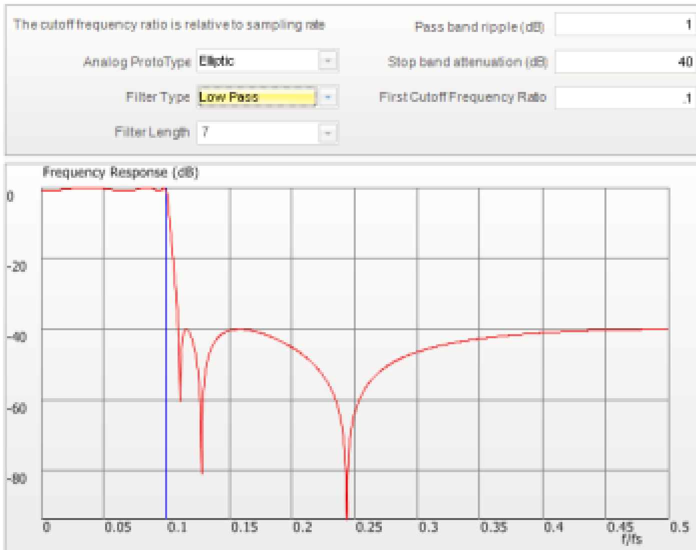


■ Figure 55. Chebyshev type I band pass filter.

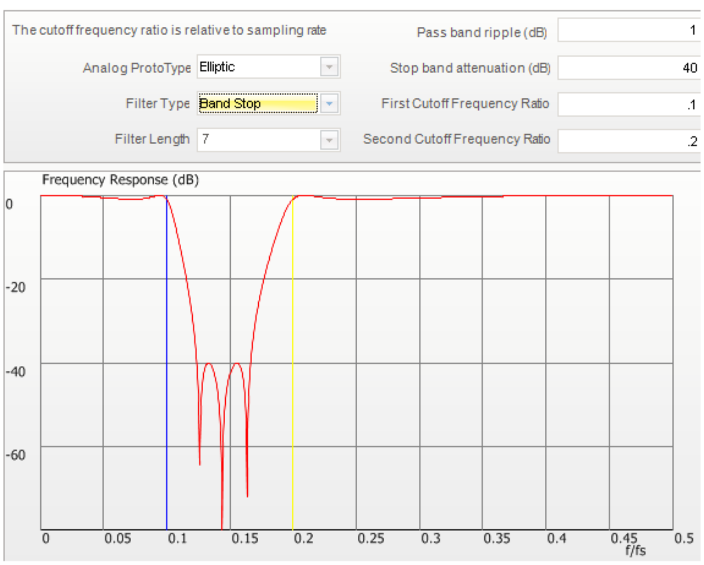


■ Figure 56. Chebyshev type 1 high pass filter.

The Elliptic Filter contains a Chebyshev Type I style equi-ripple pass band, an equipped stop band, a sharp cutoff, high group delay, and the greatest possible stop band attenuation. Below are examples of 7th order Elliptic low pass, band stop filters.



- Figure 57. Elliptical low pass filter.



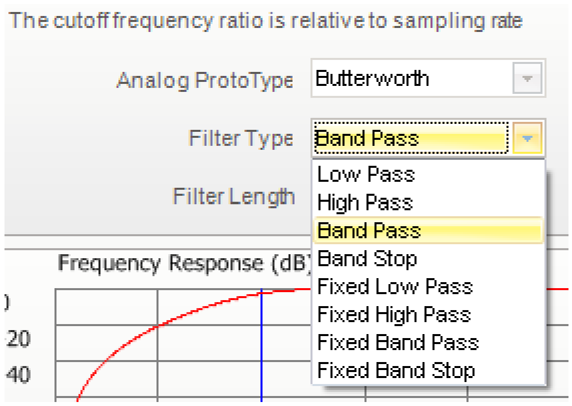
- Figure 58. Elliptical band stop filter.

Filter Design Using Fixed instead of Relative Frequency

Filter design can be accomplished using either fixed or relative frequency methods. In the relative frequency method the cutoff frequencies are defined relative to the maximum sampling rate. For example if the sampling rate is 1000 Hz and a low pass filter is defined with a cutoff frequency of 0.5 with the relative frequency method then the cutoff frequency is 500 Hz. Note that if the sampling rate is changed to 500 Hz and the cutoff frequency is no changed from 0.5, then the cutoff frequency will change to 250 Hz. The relative frequency method is the preferable method because the filter performance such as ripple or transition band width will not change when the sampling rate is changed.

The alternative is the fixed frequency method where the cutoff frequency is defined by a fixed frequency. With this method the cutoff frequencies do not need to be changed when the sampling rate is changed. While this method is more user friendly than the relative frequency method, it is not the recommended method because the filter performance, such as ripple or transition band width can vary when the sampling rate is changed.

To change between fixed or relative frequencies, go to Filter Type menu and select a filter type with Fixed. The filter types without Fixed are relative by default. :

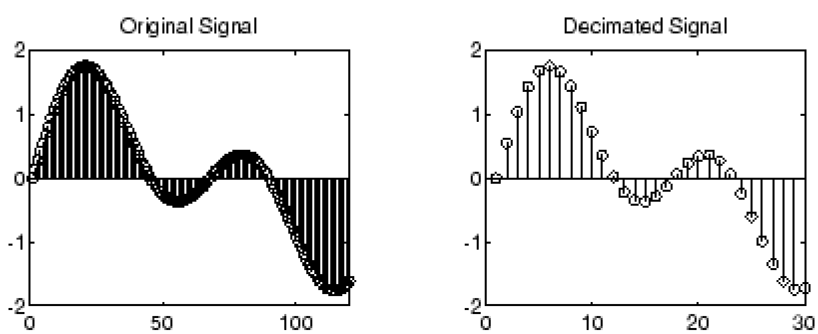


- Figure 59. Fixed or relative frequency setting.

Decimation Filters

The decimation filter is a special filter available on the CoCo. Decimation reduces the original sampling rate for a sequence to a lower rate. The decimation process filters the input data with a low-pass filter and then re-samples the resulting smoothed signal at a lower rate.

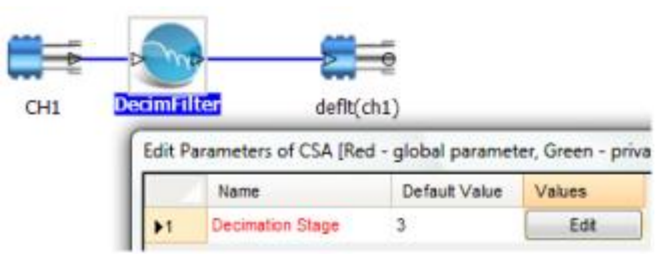
Figure 60. Illustration of a decimation filter. shows how a decimation filter reduce the number of sampled points from 150 to 30 while the signal shape which is dominated by the low frequency components is still retained.



- Figure 60. Illustration of a decimation filter.

Low pass filtering is important in the decimation process to ensure no aliasing occurs. Aliasing refers to the effect of under sampling a high frequency signal and misrepresenting the high frequency behavior by a lower frequency. When aliasing occurs there is no way to distinguish the erroneous aliased signal from the actual signal. In the CoCo hardware the decimation filter uses a fixed proprietary low pass FIR filter with excellent ripple performance in the pass-band and very high attenuation in the stop band.

In the CoCo hardware, the decimation filter module contains multiple stages of decimation filters. In each stage the data is decimated by a factor of two. After N stages of decimation, the data will be reduced to its $1/2^N$. In the example below, since the decimation stage is set to 3, the data will be reduced to $1/2^3$ = 1/8 of its input points after this decimation module.

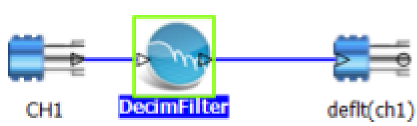


■ Figure 61. Decimation filter in the EDM software.

The decimation filter is widely used to view, analyze and record low frequency signals. For example, a system may be used to acquire the vibration and pressure or temperature data simultaneously. While the signals are all sampled at the high data rate, the pressure channel and the temperate channel should be viewed and recorded at a much lower rate because these types of signals typically do not change dynamically (at a high frequency). In this case, we can simply apply decimation filters to the channels that measure and record pressure or temperature.

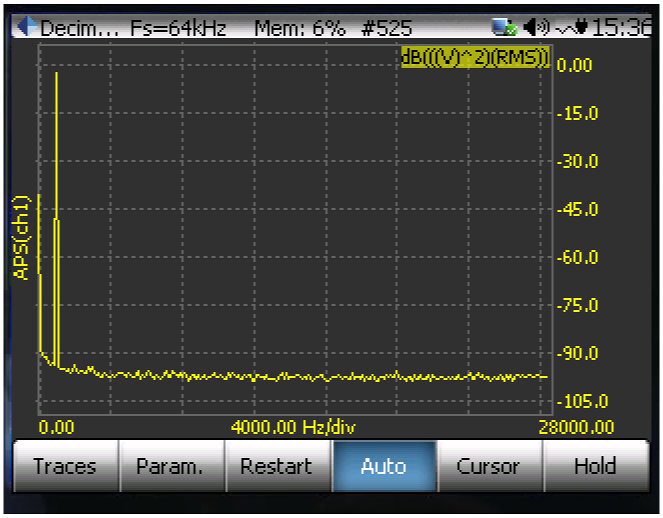
Integrating decimation filter with other filter techniques allows the user to analyze high frequency and lower frequency signals with different frequency resolution simultaneously. This capability is unique to the CoCo system.

For example, in a CSA file you can apply a 3-stage decimation filter to channel 1 as shown in Figure 62.



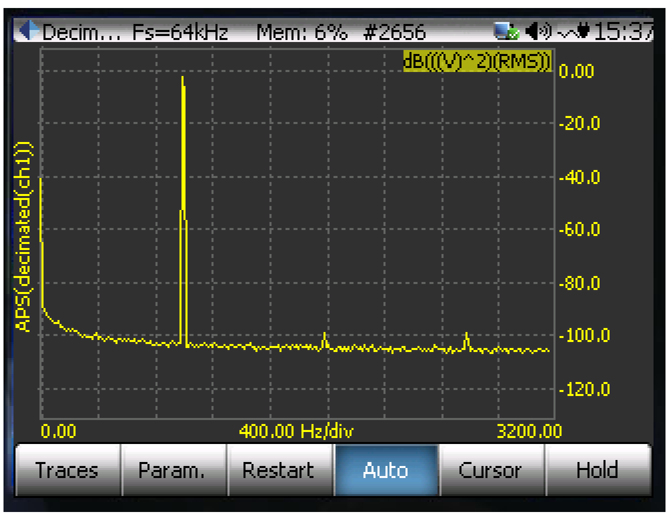
■ Figure 62. Decimation filter example.

If you connect a signal source with 1 kHz sine output to channel 1 and set the sampling rate to 64 kHz, you can see the broad spectrum up to 28 kHz as shown in Figure 63.



- Figure 63 The auto spectrum of an 1kHz sine wave when sampled at 64kHz

However you can also **simultaneously** show the spectrum of the decimated signal, *decimated(ch1)*, with 8 times frequency resolution as shown in Figure 64. Note that in this example the two spectra are of the exact same time signal not of two samples acquired at different times.



- Figure 64 The same sine wave at 1kHz, after decimated 8:1, shows at different location on the spectrum

9. HISTOGRAM AND STATISTIC MEASURES

A histogram is a graphical display that shows the number (or frequency) of events that fall into each of several or many specified categories. Figure 65 shows a typical histogram.

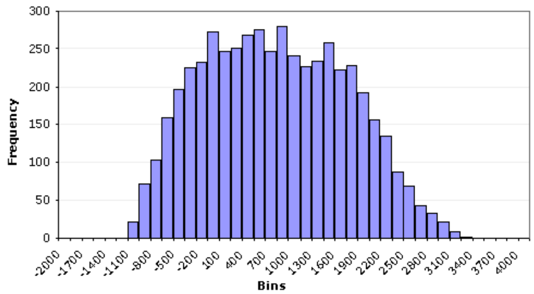


Figure 65. Typical Histogram.

Mathematically, a histogram is a mapping m_i that counts the number of observations that fall into various disjoint categories (known as *bins*). Thus, if we let n be the total number of observations and k be the total number of bins, the total number of events can be found by adding the frequency in all the bins as

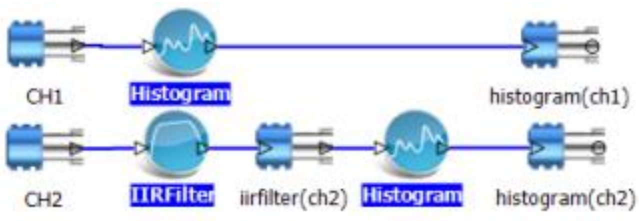
$$n = \sum_{i=1}^{k} m_i$$

Cumulative Histogram

A cumulative histogram is a mapping that counts the cumulative number of observations in all of the bins up to the specified bin. That is, the cumulative histogram M_i of a histogram m_i is defined as:

$$M_i = \sum_{j=1}^{i} m_j$$

In CoCo implementation, the Histogram and Statistics function is a single CSA module that can be applied to *any* time stream. The output of the Histogram and Statistics module is a histogram signal and the associated statistics results. You can change the display format on CoCo.



■ Figure 66. Histogram example.

CoCo provides the following measurement parameters for a histogram: bin number for the bar chart and amplitude ranges. It provides the following display formats for the histogram graph: normalized-linear, normalized -logarithmic, un-normalized and cumulative. While the histogram is measured, it also provides the following statistics values: *mean, max, min, RMS, variance, skewness, crest factor* and *kurtosis*. The definitions of these statistics measures for N samples are:

$$Mean = \mu_x = \bar{x} = E(x) = \left(\sum_{i=1}^{N} x_i\right)/N$$

$$Variance(x) = \left(\sum_{i=1}^{N} (x_i - \bar{x})^2\right)/N$$

$$standard\ deviation\ \sigma = \left[\left(\sum_{i=1}^{N} (x_i - \bar{x})^2\right)/N\right]^{\frac{1}{2}}$$

$$rms(x) = \left[\left(\sum_{i=1}^{N} (x_i)^2\right)/N\right]^{\frac{1}{2}}$$

The *rms(x)* is equal to the standard deviation when the mean is 0.

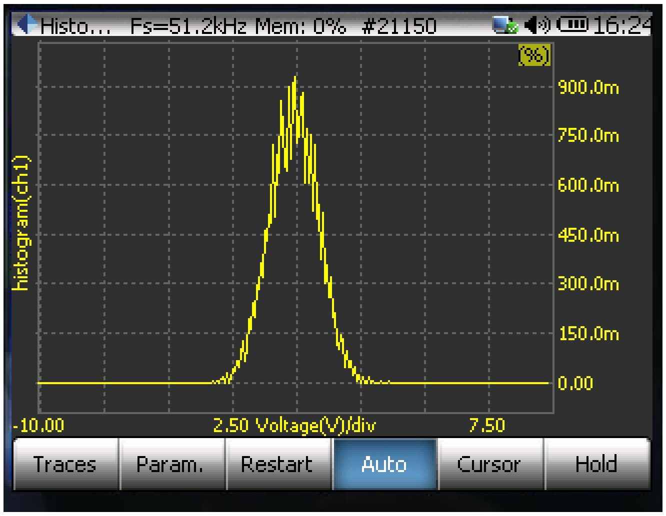
Skewness is a measure of the asymmetry of the data around the sample mean. If the skewness is negative, the data are spread out more to the left of the mean than to the right. If the skewness is positive, the data are spread out more to the right. The skewness of the normal distribution (or a perfectly symmetric distribution) is zero.

$$Skewness(x) = \left(\sum_{i=1}^{N} (x_i - \bar{x})^3\right)/N\sigma^3$$

Kurtosis is a measure of how outlier-prone a distribution is. The kurtosis of the normal Gaussian distribution is 3. Distributions that are more outlier-prone than the normal distribution have kurtosis greater than 3; distributions that are less outlier-prone have kurtosis less than 3.

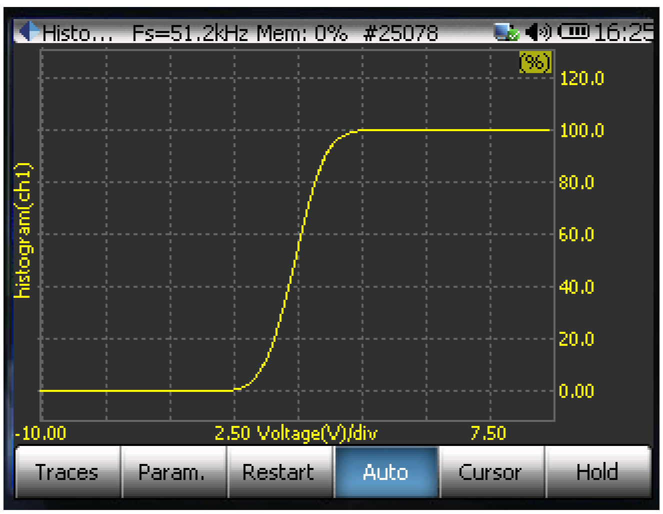
$$Kurtosis(x) = \left(\sum_{i=1}^{N} (x_i - \bar{x})^4\right)/N\sigma^4$$

As an example, a histogram of a Gaussian random noise is displayed in Figure 67.



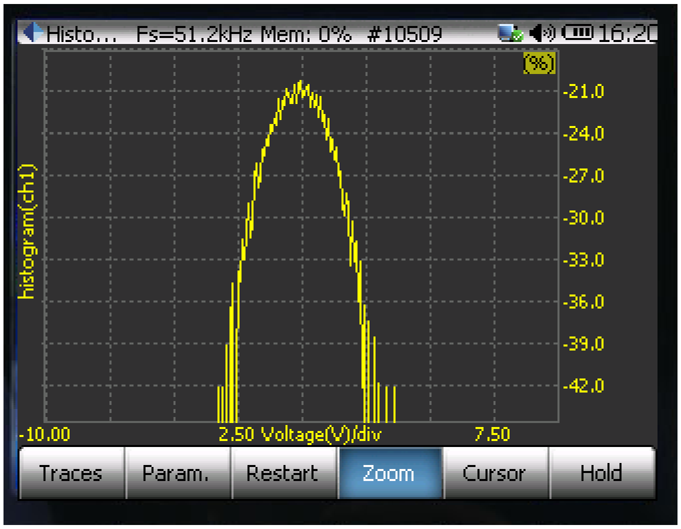
- Figure 67. Histogram of a Gaussian distributed random signal.

Its cumulative histogram is displayed in Figure 68.



- Figure 68. Cumulative Histogram of random Gaussian signal.

The display format can be changed to analyze the histogram. For example, a signal distribution with high Kurtosis can be observed using a logarithmic vertical scale with dB units, in the unit of dB, as shown in Figure 69. Here the low frequency outliers can more easily be seen.



■ Figure 69. Histogram with dB scale.

------------------------ END OF DOCUMENT -----------------------------

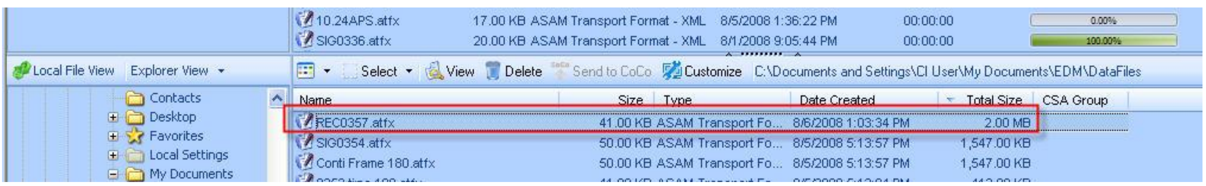
EDM Post Processing Step by Step Tutorial

This step by step tutorial demonstrates how to use the EDM software to post process time stream data that was recorded with the CoCo hardware. The first part demonstrates the procedure for selecting a data record, applying analysis functions and processing the data. The second part demonstrates additional features such as exporting processed data, changing the time range for processing, saving the processed data, and exporting the analysis profile.

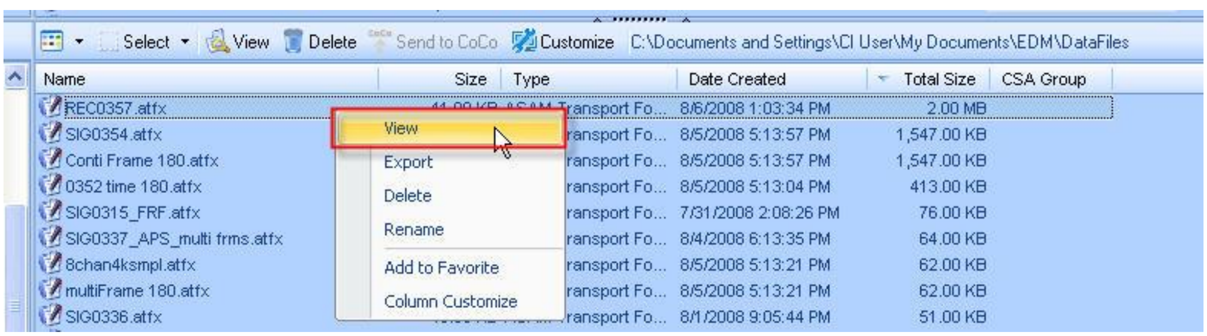
The user needs to purchase the post processing functions in EDM. License Key has to be entered during the installation process to enable the post processing functions.

Post Process Time Streams

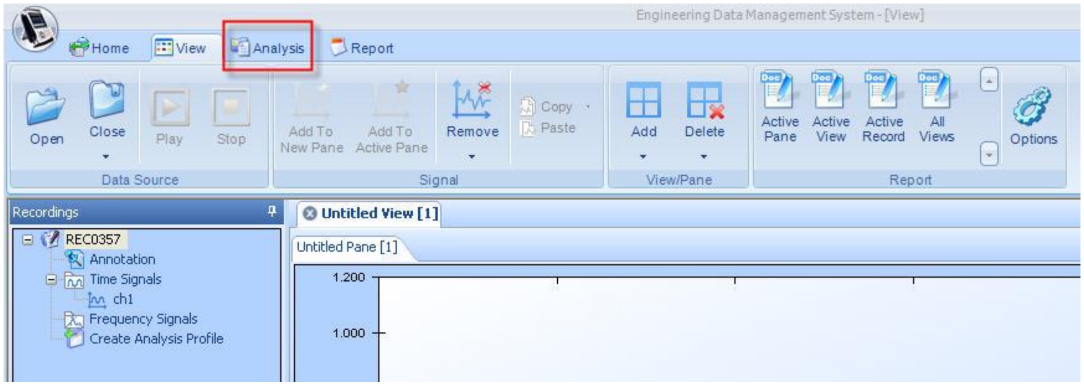
1. The first step is to download a time stream signal from your CoCo or SD card to the EDM software on the PC



2. After the data file is downloaded, right click on it and select view from the pop-up menu.

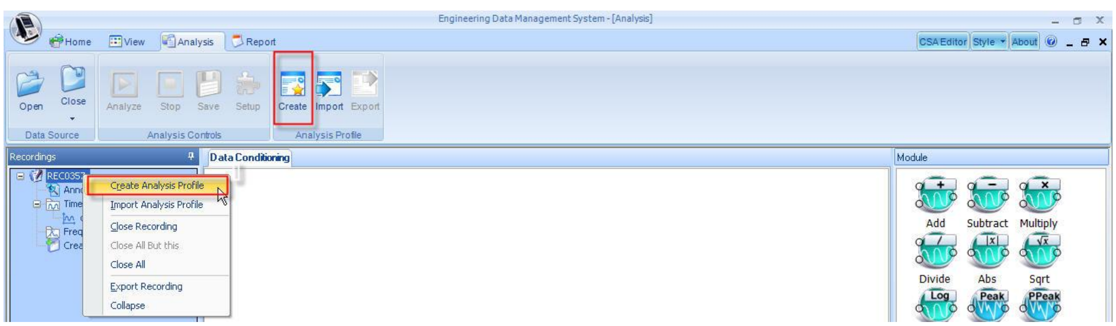


3. Next click the **Analysis Tab**. All the post processing features are located on the Analysis Tab.

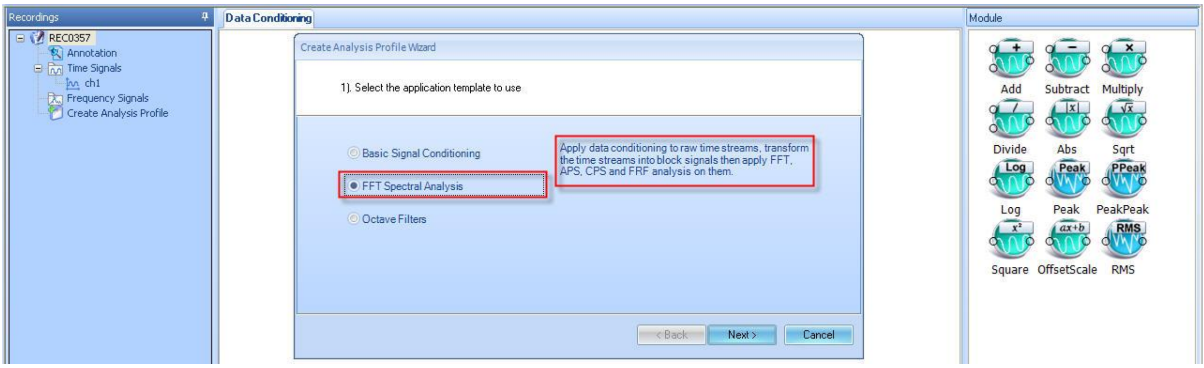


4. To start the analysis, we should create an analysis profile first. The profile saves all the analysis settings. This is useful if we want to perform the same analysis over and over again or use it at a later date. We can save the profile it in two ways. One way is to right click on the data file and select "**Create Analysis Profile**", the other is to press the "**Create**" button on the toolbar.

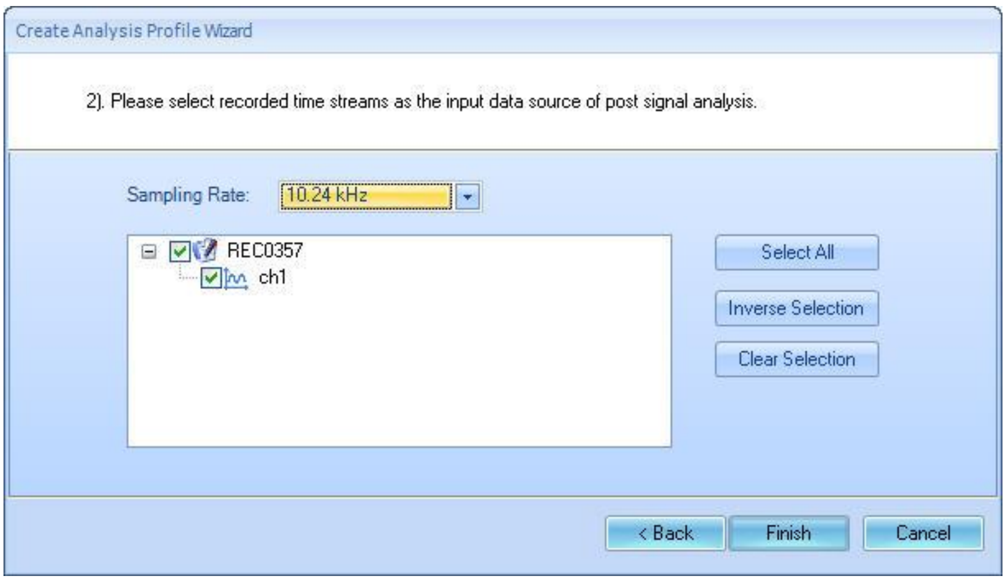
   An Analysis Profile defines the required output signals and the analysis parameters for post processing,.



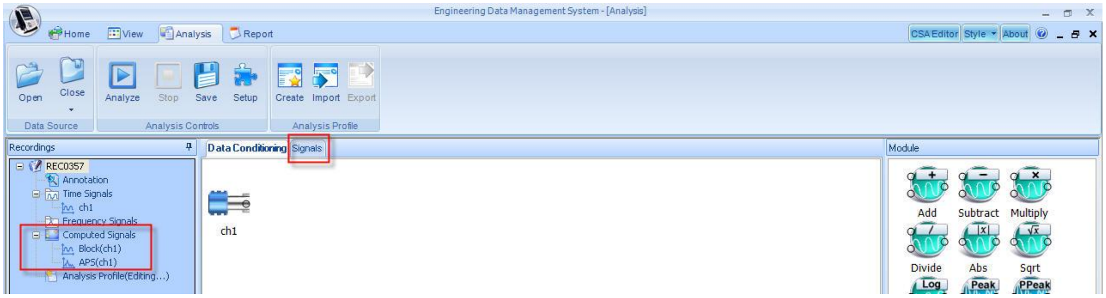
5. The wizard launches. In this demonstration we will use the **FFT Spectral Analysis template**. Select this template and press the **Next Button**.



6. Next, **select the recoded time streams** as the input data source for the post processed signal analysis. Click the **Finish Button**.

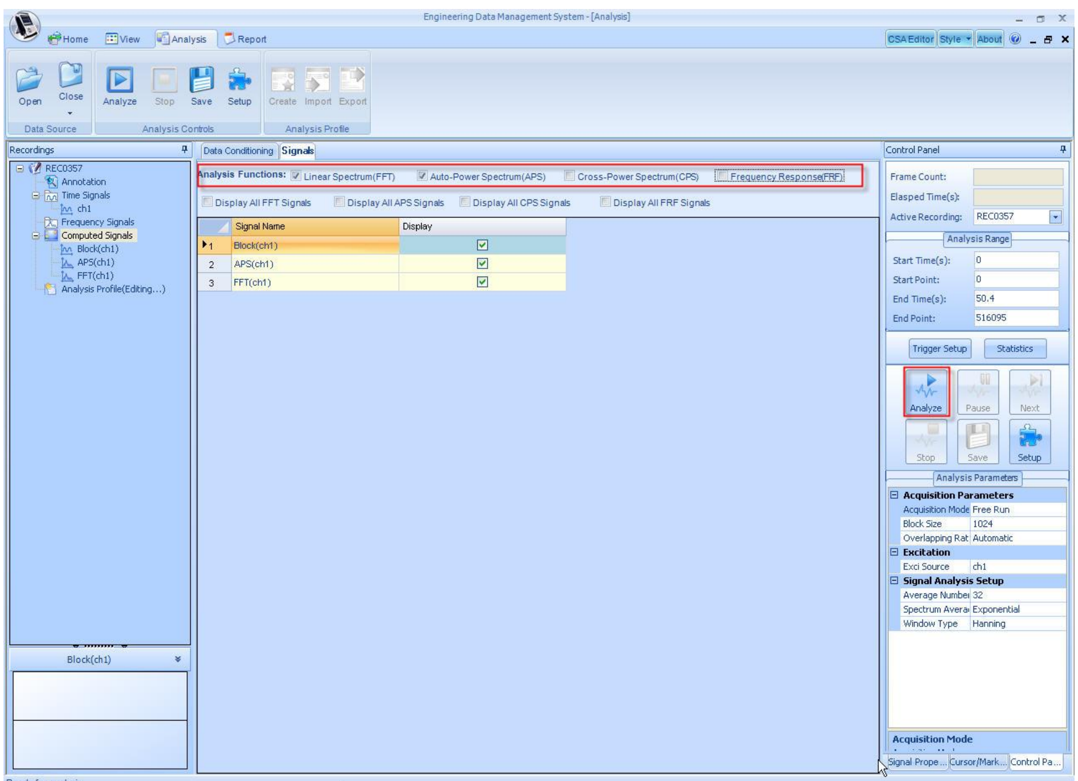


7. After the wizard is finished. two computed signals show up in the left panel and the Data Conditioning and the Signals Tabs are shown. Right now the FFT and APS compute signals shown in the Recording pan are empty because the data has not been processed yet.

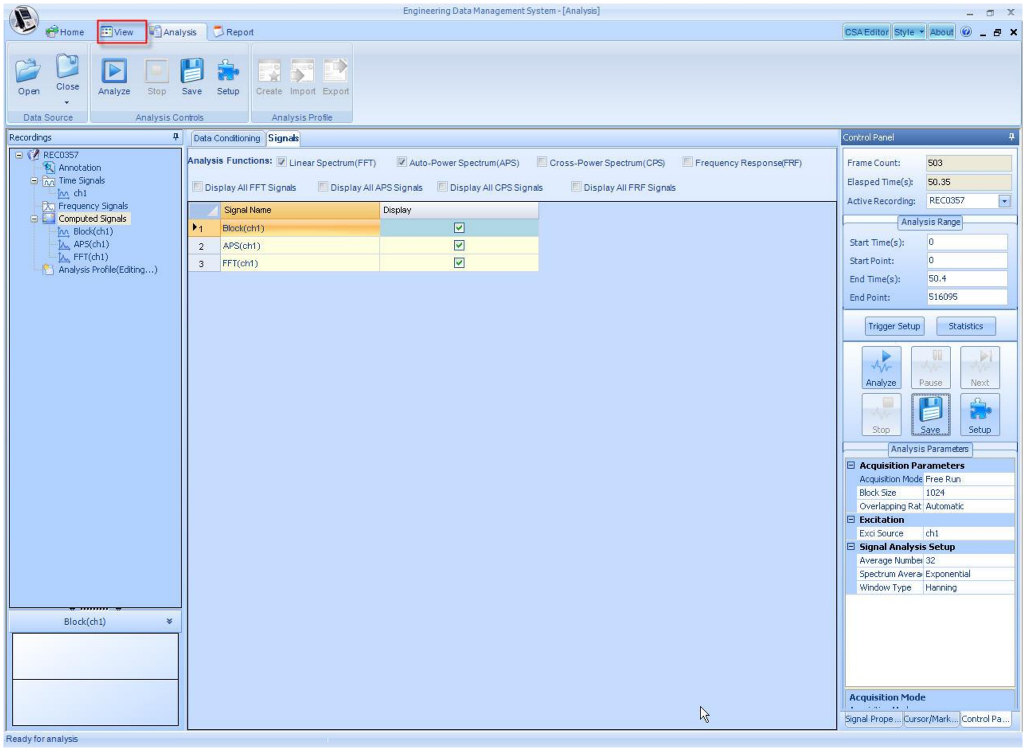


8. Select the **Signals Tab**. Since there is only 1 time signal in the file, we can only analyze it with an FFT and an APS. If we had more than two time signals in the file, we could do a CPS and FRF analysis. The FFT(ch1) appears in the left panel when we enable the FFT analysis function.

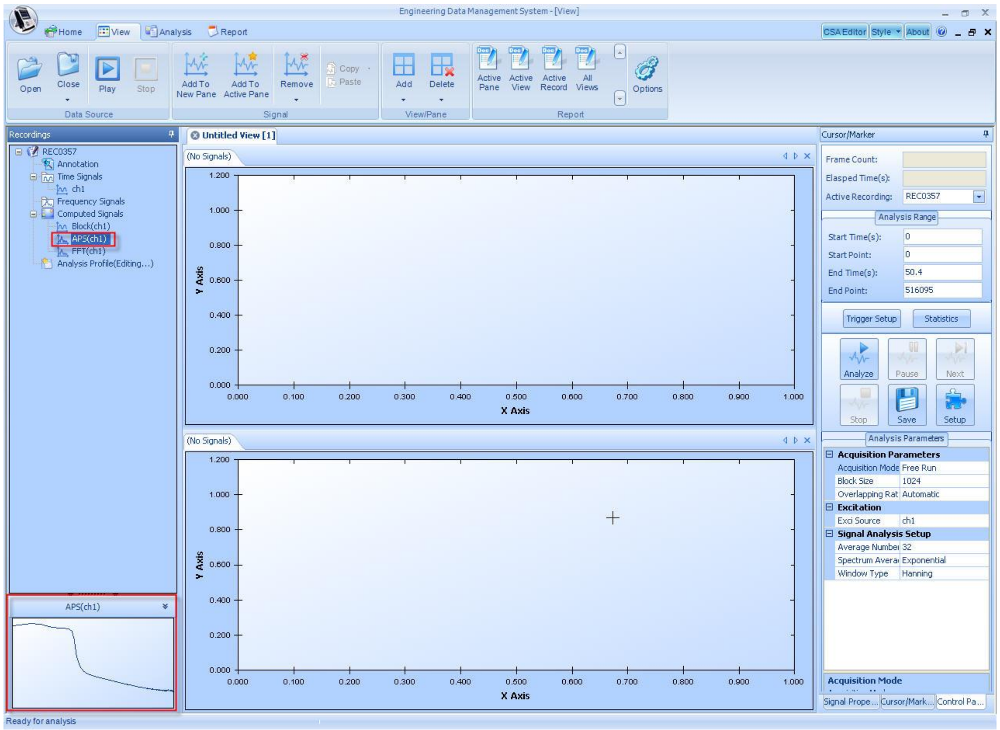
   After we enable the desired analysis functions, press the **Analyze Button** on the control panel.



9. Now, the Block(ch1), APS(ch1), and FFT(ch1) are computed. Click on the **View Tab**.

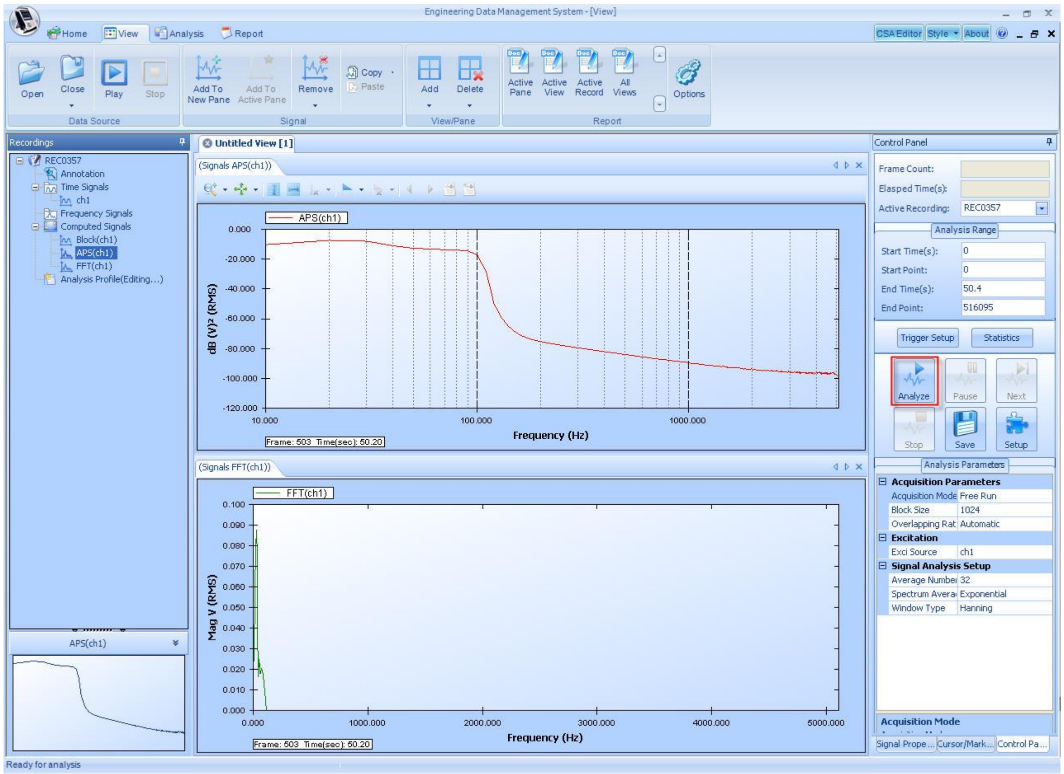


10. **Select APS(ch1) in the Recordings pane**. We can see the preview of ASP(ch1) on the bottom left pane.



11. Next, **drag the APS(ch1) and FFT(ch1) signals into the blank panes** . Press the **Analyze Button** on the control panel and we can see the APS(ch1) and FFT(ch1) signals are being analyzed frame by frame.

12. The recorded data is post processed using the settings in the analysis profile. You can change the settings and then reanalyze the data. Click on **Spectrum Averaging** and change Linear to Peak/Hold. Then click on the **Analyze Button** again to reprocess the data with the different averaging setting.

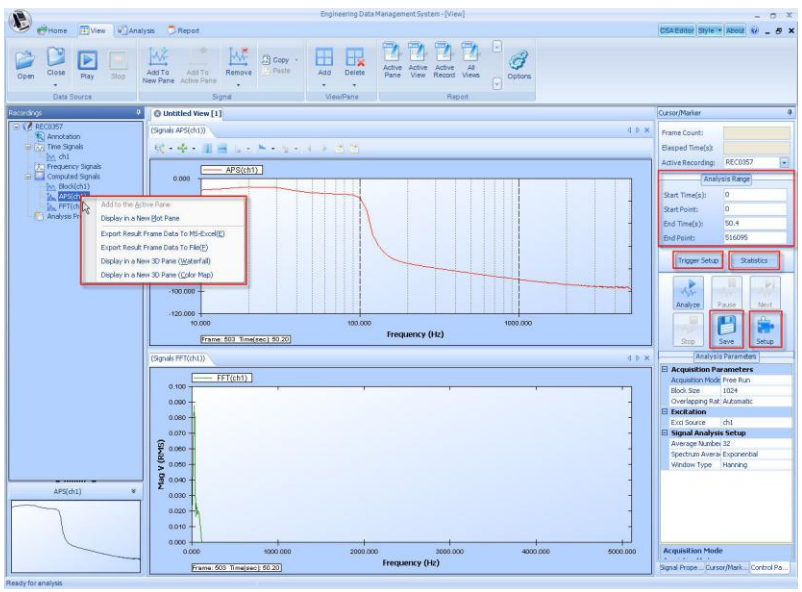


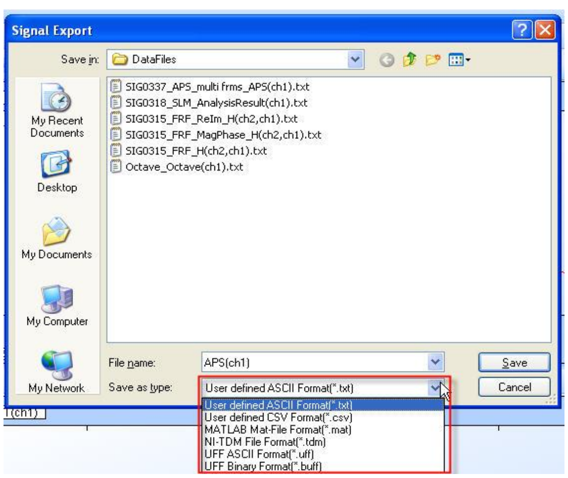
This is a good time to go back and practice changing other analysis parameters and analyzing the data before moving on to the next section.

Additional Analysis Features

The previous section gave a quick demonstration of how to access the analysis features, select a data recording and process the data. The following sections introduce more details of the EDM Analysis functions including exporting the processed data, changing the time range for processing saving the processed data and exporting the analysis profile.

13. **Right click on one of the calculate signals**. We can export the frame data to an Excel file or a file in other formats supported by EDM, such as ASCII, CSV, Matlab, TDM, UFF, and BUFF.



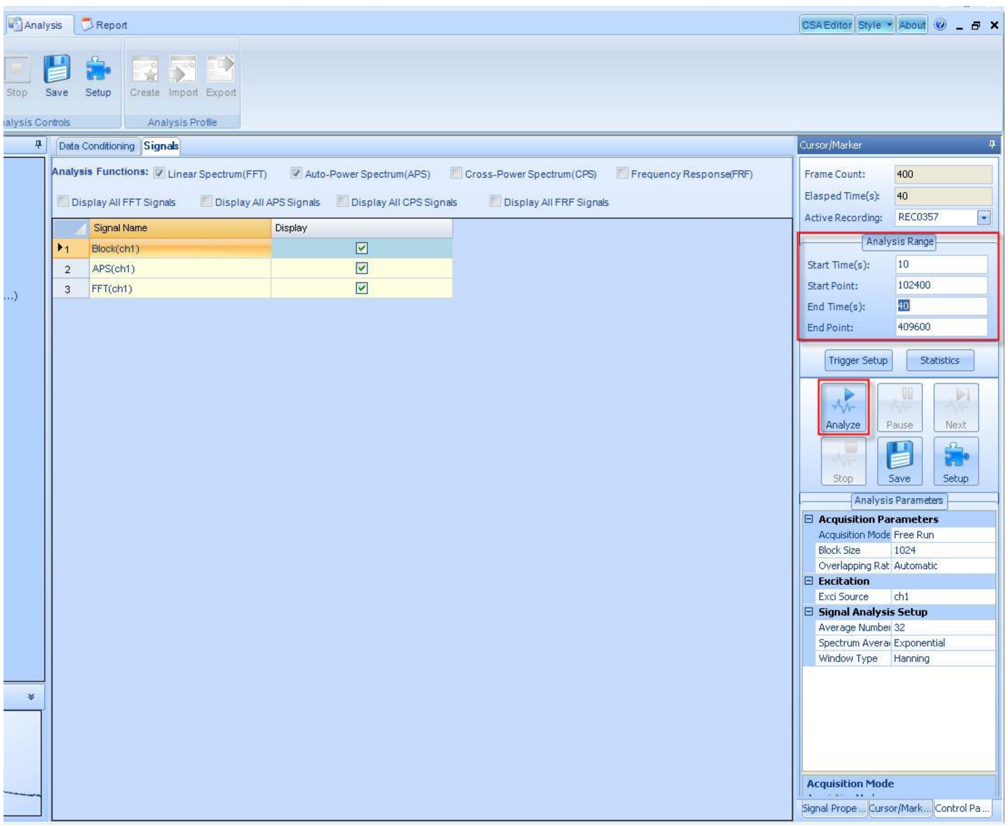


14. We can also change the analysis range.

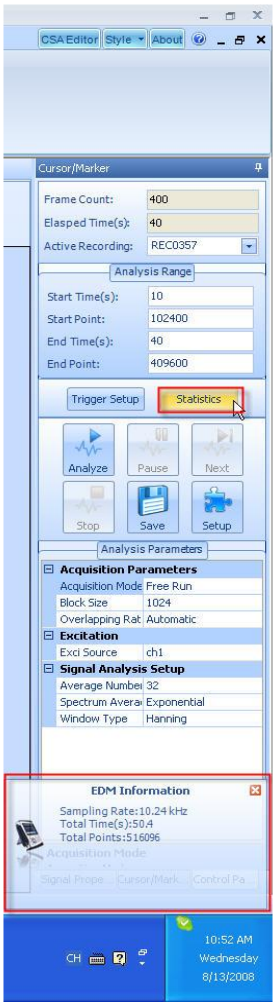
    Click on the **Analysis Tab** and then the **Signals Tab** below.

    Change the **Analysis Range** on the control panel. We can specify either the **Start Time or Start Point** (the other one will be calculated automatically). Also enter the desired **End Time or End Point.**

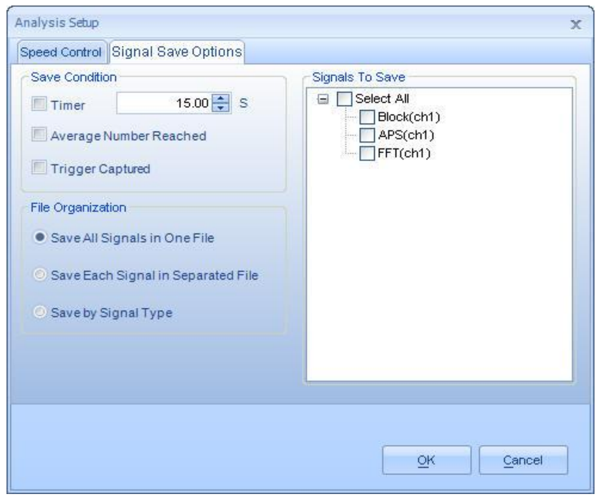
    Press the **Analyze Button** on the control panel and the APS(ch1), FFT(ch1), and Block(ch1) will be calculated from the original time stream signal in this range.



15. Press the **Statistics Button** on the Control Panel and EDM will show some information of the original time stream signal such as sampling rate, duration and number of data points.

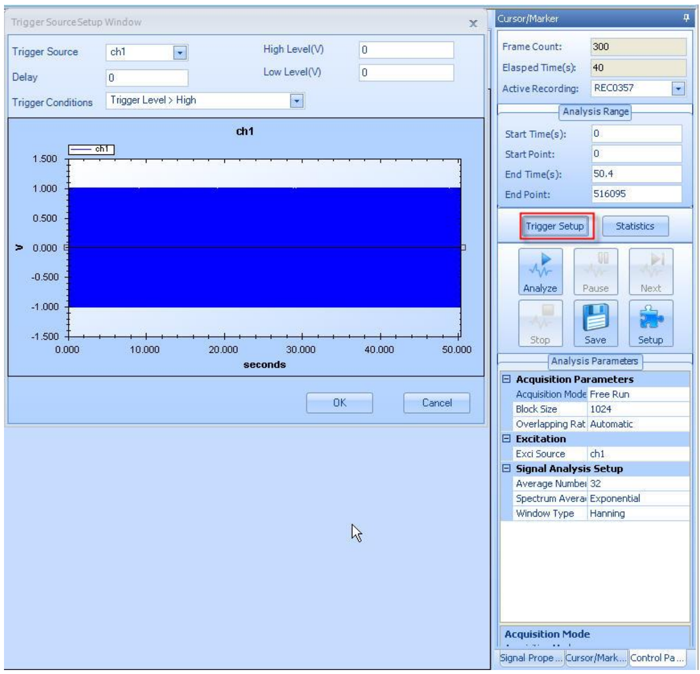


16. Next press **Setup** on the ribbon**.** There are two tabs: one is for analysis speed control and the other is for Signal Save options. After signals are selected to save, we can press **Save** on the control panel to save the signal.



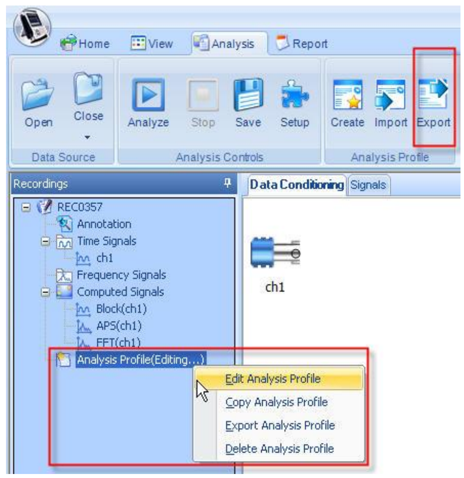
17. We can save signals when a triggered event is captured when the **Trigger Captured** is enabled in **Save Signal Options**. We need to setup the trigger conditions.

    Click **Trigger Setup** on the control panel. The trigger source setup window will pop up for trigger settings. Select the **Trigger Source**, conditions and levels and click the **OK Button**.



18. The analysis profile can be used for other data files by exporting it. Right click on the **analysis profile** to select E**xport Analysis Profile** or press **Export** to export it to a file so that you can reuse it anther time.

    When we would like to reuse the same analysis profile, just select another time stream signal on the Recordings panel and press **Import** on the toolbar.



This concludes the EDM Analysis Step by Step Tutorial. After completing this tutorial you should be able to select a recorded data file, post process it, change the processing settings, view and save the processed data and save the analysis profile.

Engineering Data Management Software

User's Guide

- **CoCo DSA Mode**
- **Spider Real-Time Mode**
- **Post Analysis Mode**
- **(CoCo VDC Mode not included)**

Version 1.00

Crystal Instruments Corporation

4699 Old Ironsides Drive, Suite 100

Santa Clara, CA 95054, USA

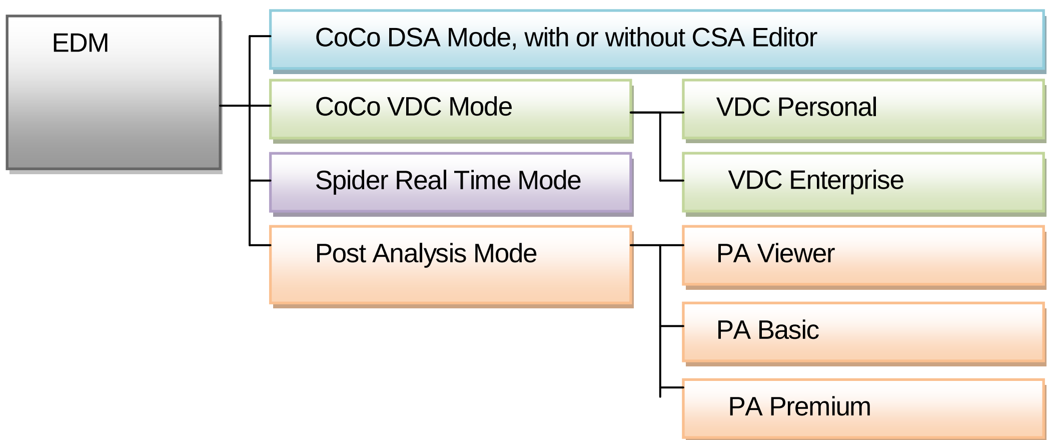
Table of Contents

Table of Figures

1. Introduction

The Engineering Data Management (EDM) is a PC software used for data management, post signal processing, viewing, report and the connection between the Crystal Instruments hardware, the PC and the data storage system. EDM provides connectivity to one or more CoCo or Spider devices. It provides data management tools that allow you to search through many tests, records and view file properties or waveform characteristics. The analysis tools allow you to display data in a wide variety of formats and configurations and let you identify important signal characteristics using cursors. The report tool allows you to document the hardware configuration or data analysis results in a user formatted document.

The basic structure of EDM software is:



EDM has four working modes:

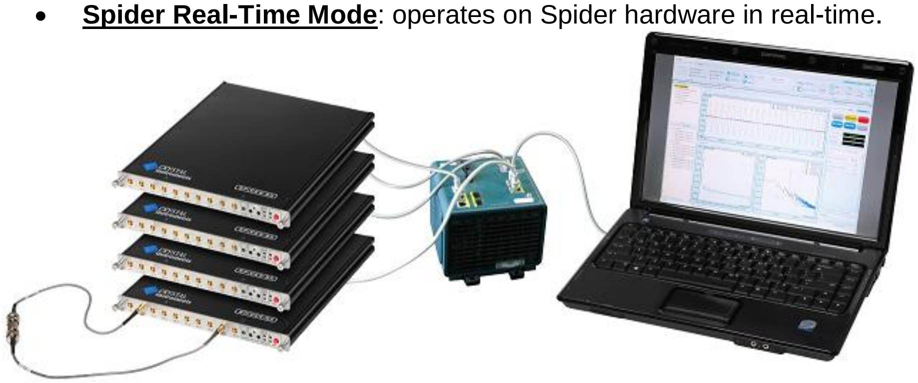
- **<u>CoCo-80 DSA mode</u>**: accesses CoCo-80 in its DSA mode, download files and view data files. CSA Editor, a tool of editing CoCo testing projects, will be included in this mode.

- **<u>CoCo-80 VDC mode</u>**: creates route data collection database, upload settings to CoCo, download data to PC, trending and alarm analysis. There are two versions of

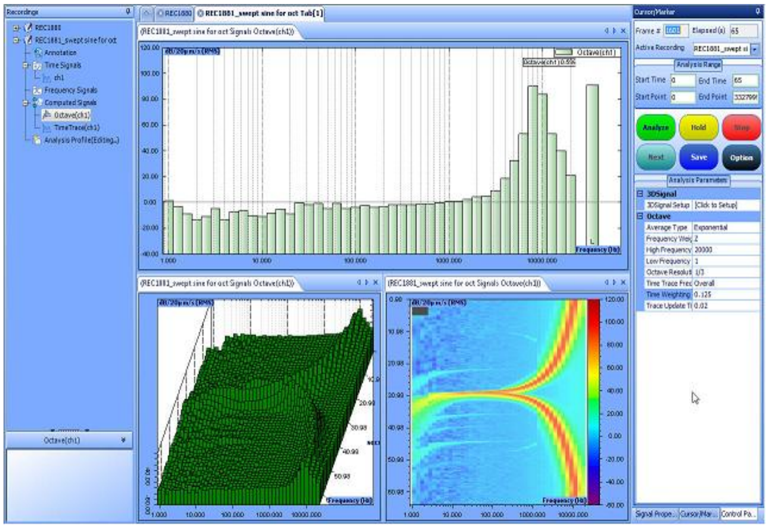
VDC modes: personal version allows the user access the database on his local PC. Enterprise version allows multiple user access the database on the LAN.

- Figure 1: EDM in VDC Mode, Signal Analysis Display

- **Spider Real-Time Mode**: operates on Spider hardware in real-time.



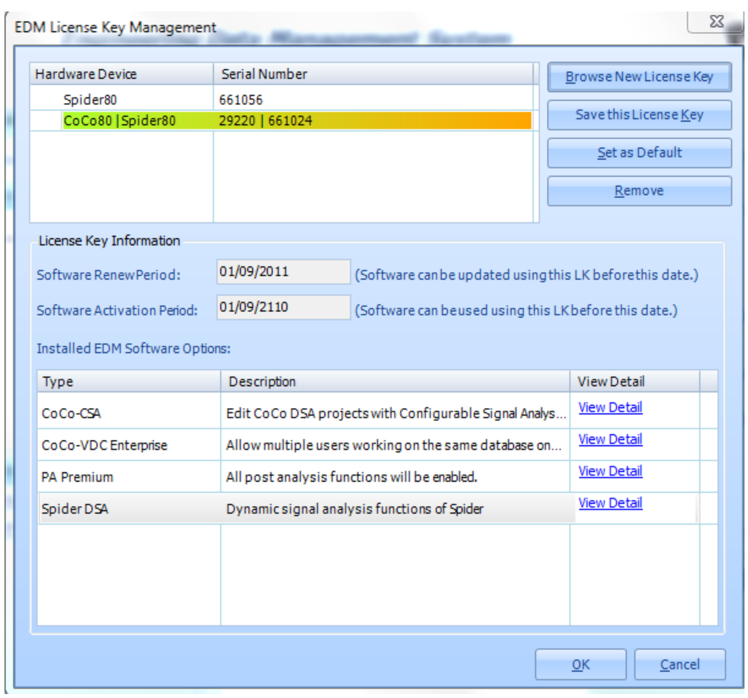
- **Post Analysis Mode**: analyzes the data files on PC using various algorithms. PA has three versions: PA Viewer allows the user to view the data and create report; PA Basic has FFT spectral analysis and 3D signal display functions; PA Premium has all post analysis functions.



This manual describes the basic operation of CoCo DSA Mode, Spider Real-Time Mode and Post Analysis Mode. The description about CoCo VDC Mode is not included in this manual. For CoCo VDC Mode, please refer to the CoCo VDC manual.

EDM software is registered to a CoCo or a Spider device. To activate the EDM software, the user must have a License Key. EDM software uses a License Key file to enable or disable certain functions. License Key is also used to control the Activation Period and Software Subscription Renew period. Multiple License Keys can be installed in one EDM installation. This allows an instance of EDM runs multiple hardware devices.

A typical management page for license keys is shown below:



Software Renew Period: this is the time period that this EDM installation can be upgraded using the current installed License Key. When the time expires, the EDM software will still be functional but cannot be updated.
Software Activation Period: this is the time period that this EDM installation can be used using this License Key.

EDM Post Analysis (PA) Functions

EDM PA is one of the working modes. There are three versions of Post Analysis functions, PA Viewer, PA Basic and PA Premium. The function comparison is listed below:

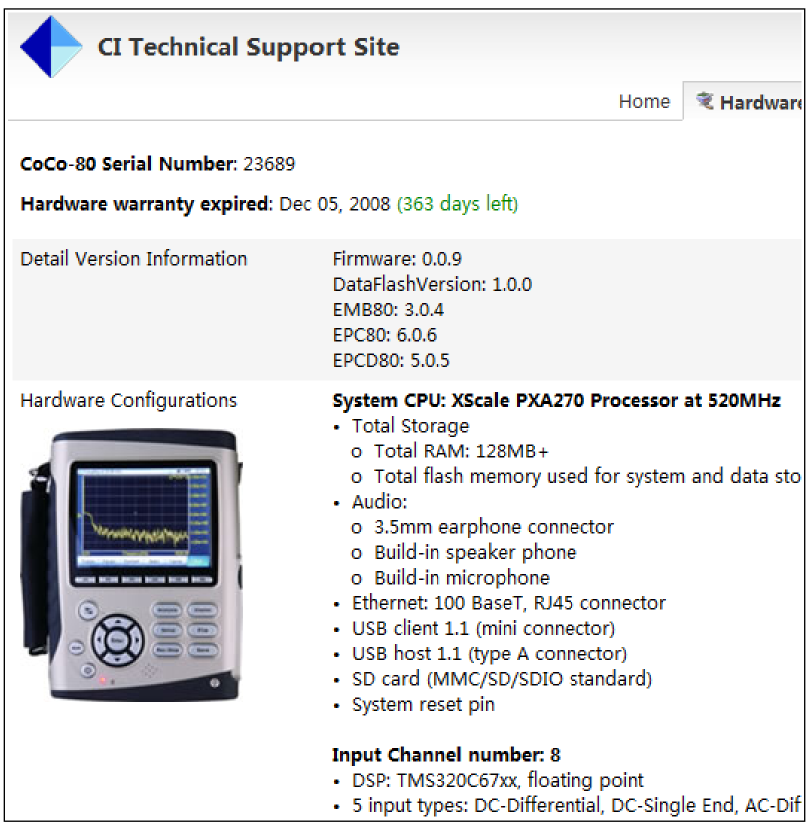
Functions	PA Viewer	PA Basic	PA Premium
Browse, display and edit long waveform files	√	√	√
Signal display with different spectrum unit and X-Y scale	√	√	√
Signal annotation, cursor, play sound, calculate RMS, THD, ZOOM-in, ZOOM-out, auto scaling	√	√	√
Create template-based report in HTML, Excel, Word or PDF	√	√	√
Engineering unit conversion, dB reference	√	√	√
Export to standard formats including ASAM-ODS, UFF, BUFF, MATLAB, TDM, user-defined ASCII and wave files	√	√	√
Import user-defined ASCII file, wave file, Pacific Instrument file		√	√
Acceleration, velocity and displacement conversion		√	√
FFT Spectral analysis: FFT, auto power spectra, cross power spectra, frequency response function		√	√
3D display: waterfall, color map		√	√
Math Functions: abs, +, -, *, /, square, square-root, log, integration, differentiation, RMS, peak, offset and scale,			√
Digital Filters: IIR, FIR, Low-pass, High-pass, Band-pass			√
User defined data conditioning modules			√
Fractional octave filters: 1/1, 1/3, 1/6, 1/12			√
Order Tracking: RPM spectra, order spectra			√
Polynomial Curve Fit			√
Data file batch processing			√

On-Line Support

To access product information about your Spider, CoCo and the EDM software, please go to the product page of the CI website at: http://www.go-ci.com/support.asp, log in with the serial number of the CoCo or Spider and the password included in the automated email message sent to you when products are shipped. After you log-in, you will be able to review and download the latest information which is restricted to CoCo or Spider users, including:

- Product Information
- New CSA Files
- User's Manual
- Shipping and Repair History
- User Forum
- Technical Support
- Software Updates
- Technical Issues

A typical CoCo page of CI Technical Support website is shown below:



■ Figure 2. Crystal Instruments CoCo-80 Support Website.

The latest CoCo-80 application software, device drivers or CSA Files can be downloaded while the CoCo-80 subscription is maintained.

2. Installation Procedure for CoCo Users

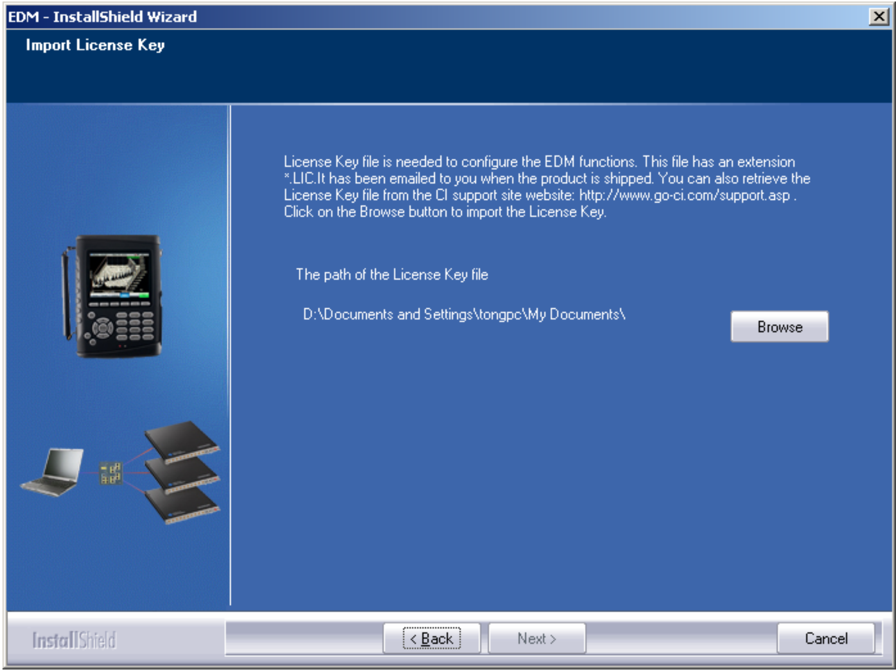
This section describes the EDM software installation procedures for CoCo users.

EDM Software

To install the EDM software, place the installation CD in the CD drive on your PC. The Welcome Screen will automatically open as shown in Figure 3. If the Welcome Screen does not automatically open then run the setup.exe file on the root level of the CD.

■ Figure 3. Welcome Screen for EDM Installation CD.

The EDM software uses Microsoft .NET resources. This software is native in Windows Vista but must be installed for Windows XP. If the installation detects that .Net is not installed then it is automatically installed from the CD.



■ Figure 4: EDM Software Installation Wizard

Where is My License Key?

There are two ways to obtain your EDM software License Key:

When your EDM Software and Spider are shipped from Crystal Instruments, The CI IT system will send out an automated email message providing shipping information, your License Key and the Serial Number of your instrument. The License Key is a file that contains the extension *.LIC.

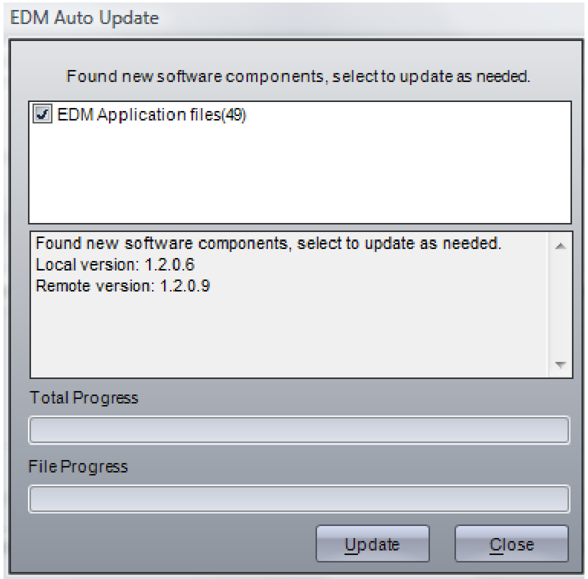
If you have not received the automated email message, or do not have your License Key, log into the CI Technical Support Site: http://www.go-ci.com/support.asp using the Spider or CoCo serial number and the password provided in the automated message mentioned above. You can then retrieve the License Key from the technical support site.

> It is a good idea to store the License Key, Serial Number of your instruments and Password somewhere safe where you can always find it. You will need this information to log on to the CI Technical Support Site for assistance and updates. If they are lost, please call CI Technical Support at 1-408-986-8880.

EDM Update

There are several ways to get EDM updated:

(1). Press the Update button of EDM. The software will automatically detect the new version and update your computer



(2). Log into the Technical Support Site and download the EDM SETUP.EXE file, and install it manually.

(3). Request an installation CD from CI.

USB Device Driver

After the installation is complete, the CoCo-80 can be connected to the PC using any of the connection methods described below. If a USB cable is used as the connection then the USB driver must first be installed. . This requires the following steps:

1. Install the EDM software on the PC.

2. Install the RNDIS USB driver on the PC.

3. Connect CoCo-80 to the PC through the provided USB cable. This cable has a mini-client port connecting to the CoCo-80 and a flat USB port connecting to the PC.

For detailed instructions on setting up the USB driver click on **How To Setup the CoCo-80 USB Driver** on the Welcome Screen. For complete hardware installation instructions refer to the CoCo-80 Users Manual.

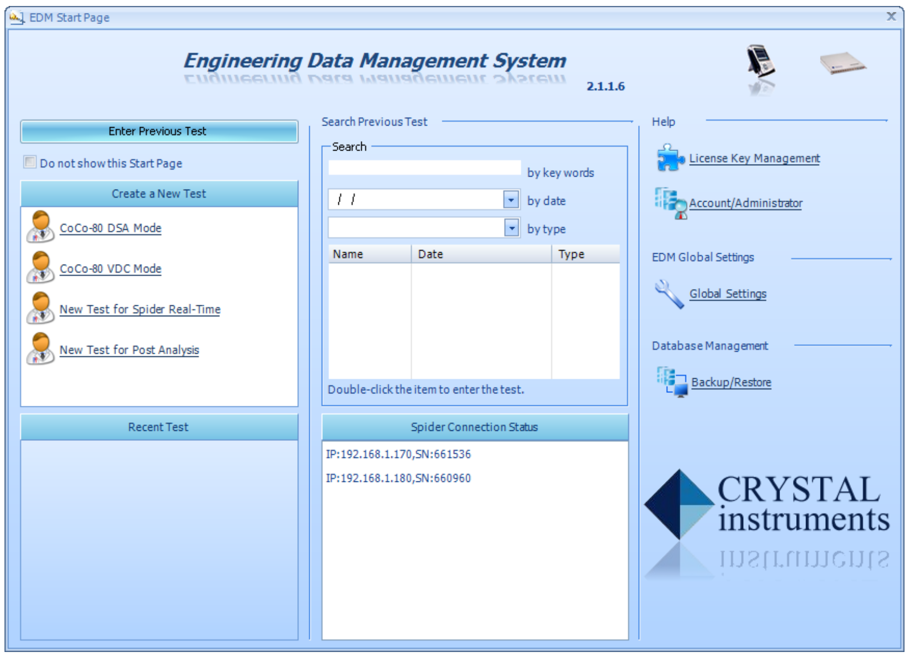
Microsoft Word

The automatic Report feature requires Microsoft Word 2000 or later. Word should be installed before using the Report feature.

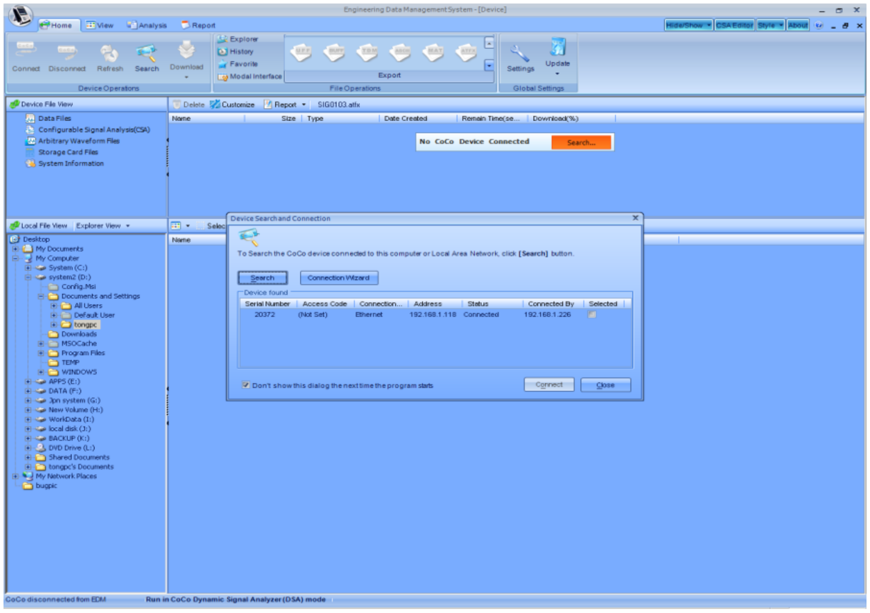
Getting Started and Getting Connected

This section gives a brief introduction to get you up and running in a short time. For more details about the use of each component of EDM refer to the sections following this section.

Launch EDM from the shortcut on the desktop. When following dialog box shows up, select CoCo-80 DSA Mode.



The first step is to establish a connection between the PC and the CoCo-80. The Device Search and Connection window is automatically displayed as shown in Figure 5. Click the Search Button to search the PC for a CoCo-80 Device and display it in the Device Found List. If a CoCo-80 Device is found then highlight it in the Device List and click on the Connect Button. If this is the first time EDM has been run or if no device is found then click the Connection Wizard Button to set up the connection.



■ Figure 5. Device Search and Connection window.

The Connection Wizard will ask you to select which connection method you would like to use. You can choose one of following four connections:

- Connect CoCo-80 to a PC directly using a USB cable
- Connect CoCo-80 to a PC directly using Ethernet via cross-over cable
- Connect CoCo-80 to a local network using Ethernet where a host PC resides on the local network
- Connect CoCo-80 to a local network using a wireless SD card

The table below summarized the configuration for these connections.

■ Table 1. PC to CoCo-80 Configuration Summary.

Connection Method	**CoCo-80 Configuration**	**Host PC Configuration**
Connect CoCo-80 to a PC directly using **USB**	No special configuration required	Install the EDM host PC software Install the CoCo-80 USB RNDIS Driver
Connect CoCo-80 to a PC **directly** using **Ethernet** via cross-over cable	CoCo-80 must be configured with a fixed static IP	Host PC IP must be configured with fixed static IP at the same subnet mask as that of CoCo-80
Connect CoCo-80 to a local network using **Ethernet** where a host PC resides on the local network	If DHCP server is installed on the local network, CoCo-80 can obtain an IP address automatically. If DHCP server is not	If DHCP server is installed on the local network, host PC can obtain an IP address automatically. If DHCP server is not installed on the local network, fixed static IP

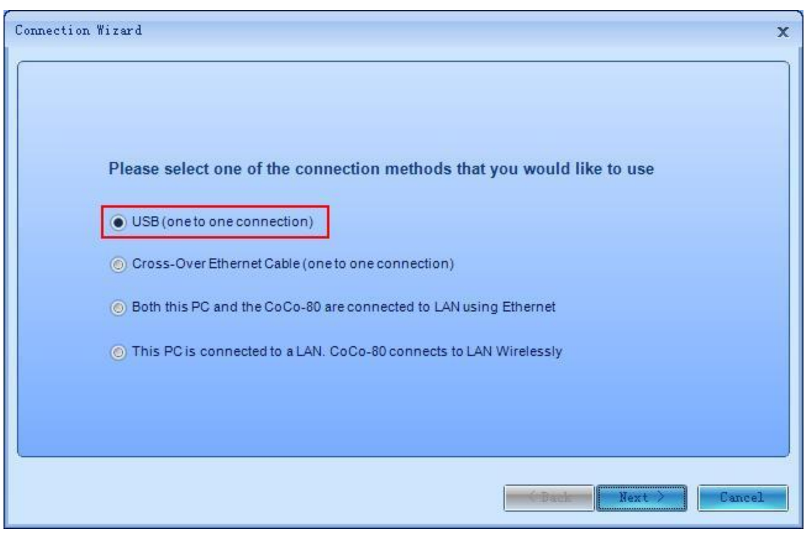
	installed on the local network, fixed static IP address must be configured on CoCo-80.	address must be configured on the host PC. Same subnet mask must be used.
Connect CoCo-80 to a local network using **wireless SD card**	If DHCP server is installed on the local network, CoCo-80 can obtain an IP address automatically If DHCP server is not installed on the local network, a fixed static IP address must be configured on CoCo-80	If DHCP server is installed on the local network, host PC can obtain an IP address automatically" If DHCP server is not installed on the local network, fixed static IP address must be configured on the host PC. The same subnet mask must be used.

In this table, *DHCP (dynamic host configuration protocol) server* refers to software installed on the local area network, either wired or wireless, that supports the "Obtain an IP address automatically" function on any networked device. DHCP is commonly used in most office networks.

For more detailed information on establishing a CoCo-80 to PC connection including a comparison of the different connection methods and the hardware requirements refer to the CoCo-80 Users Manual.

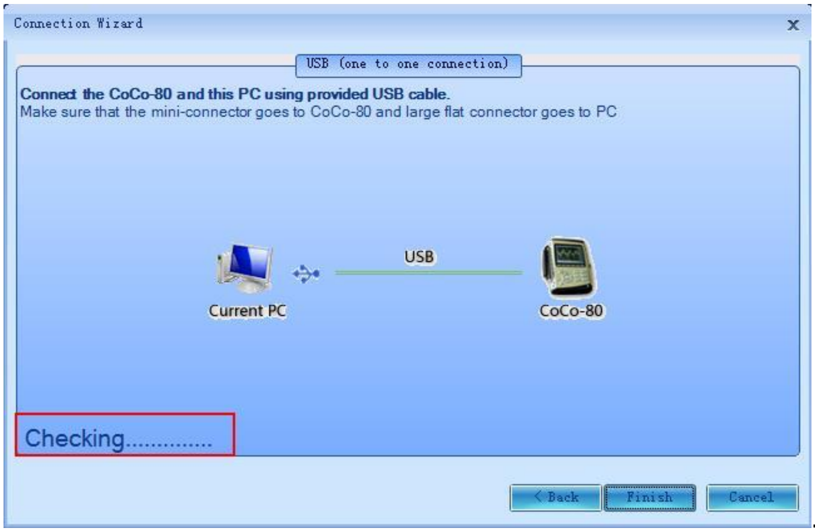
USB Connection

The USB connection is the easiest method to establish between the CoCo-80 and the PC. For this connection select ***USB (one to one connection)*** shown in Figure 6 and click the Next Button.



- Figure 6. Connection Wizard

EDM then checks for the connection. When the connection is found then click the Finish Button.

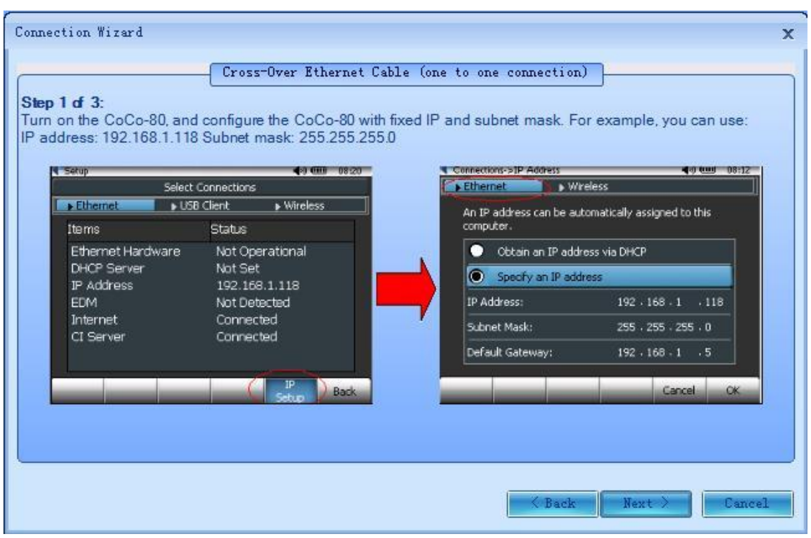


■ Figure 7. USB Connection Wizard

Cross-Over Ethernet Cable Connection

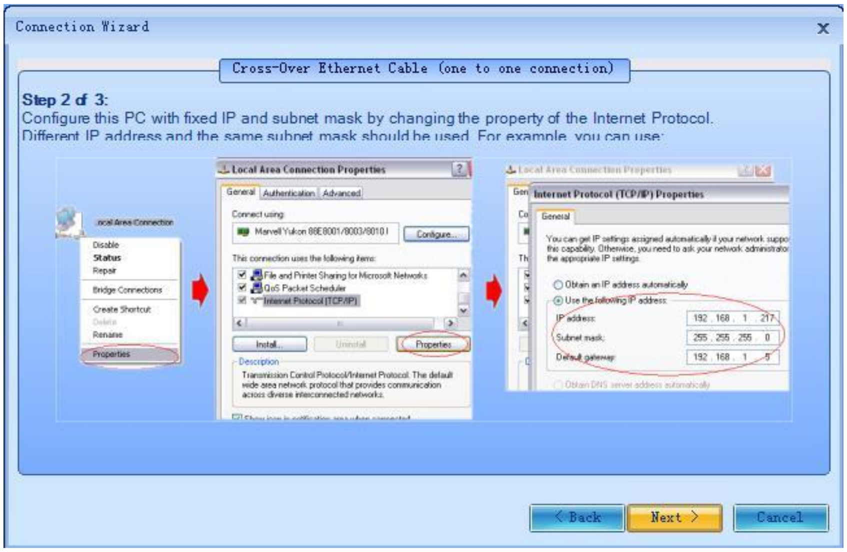
If the CoCo-80 is connected directly to the PC network card with a cross-over Ethernet cable then choose *Cross-Over Ethernet Cable (one to one connection)* and click on the Next Button.

For a cross-over Ethernet cable connection you must specify the IP and subnet mask to agree between the PC and the CoCo-80 as shown in Figure 8. Set the IP address and subnet mask on the CoCo-80 following the instructions and then click the Next Button.



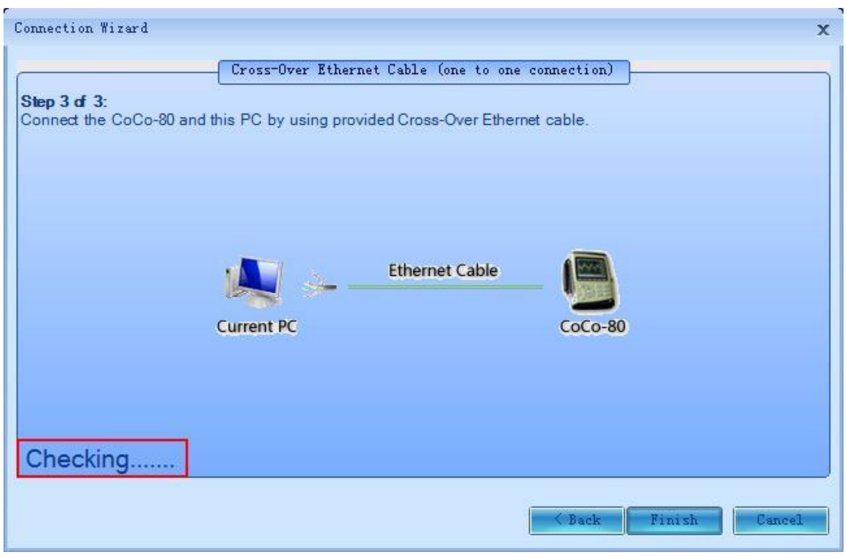
■ Figure 8. Cross-Over Ethernet Cable Connection Step 1.

Next set the IP address and subnet mask on the PC following the instructions shown in Figure 9 then click the Next Button.



- Figure 9. Cross-Over Ethernet Cable Connection Step 2.

EDM then checks for the connection. When the Wizard shows that the CoCo-80 has been found click the Finish Button.

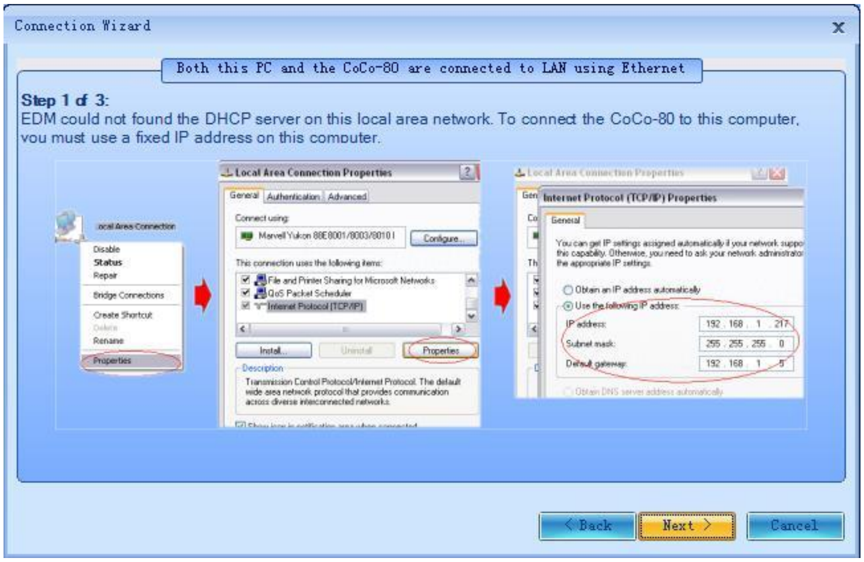


- Figure 10. Cross-Over Ethernet Cable Connection Step 3.

Wired Local Area Network Connection

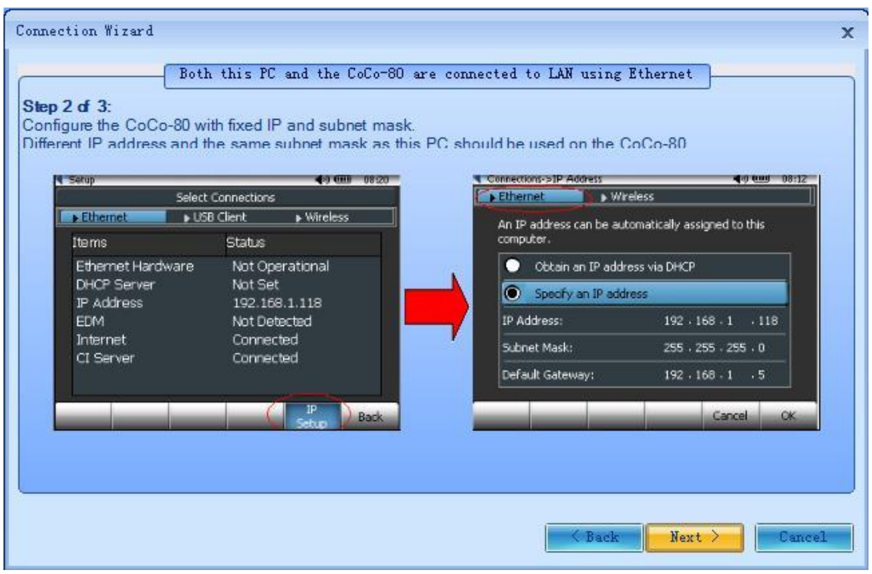
If both the PC and the CoCo-80 are directly connected to a local area network (LAN) then choose ***Both this PC and the CoCo-80 are connected to LAN using Ethernet*** then click the Next Button.

Follow the instructions to set the IP address or use DHCP on the PC.



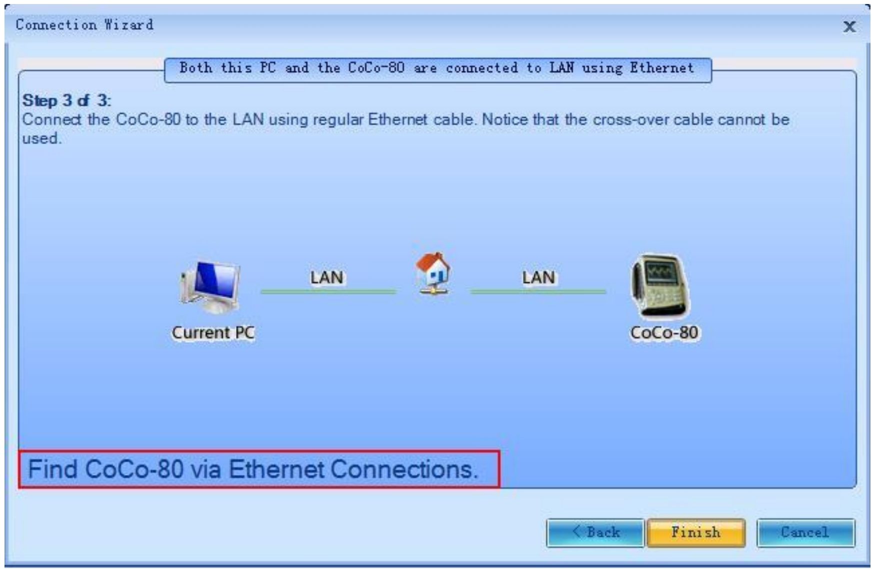
■ Figure 11. Wired LAN Connection Wizard Step 1.

Set the IP and subnet mask on the CoCo-80.



■ Figure 12. Wired LAN Connection Wizard Step 2.

Next EDM will check the connection. When the CoCo-80 Device has been found click the Finish Button.

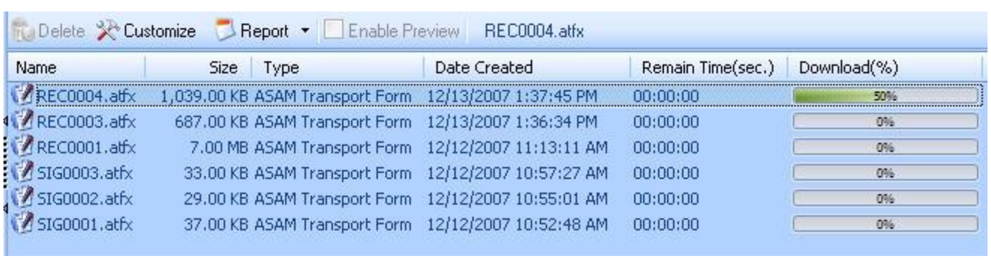


■ Figure 13. Wired LAN Connection Wizard Step 3.

EDM checks for the connection. When the connection is found click the Finish Button.

Download Files from the CoCo-80 Device

After a connection is established the next step is to download data from the CoCo-80 to the PC. A list of files on the CoCo-80 is shown in the Device File View Pane. Right click on one file name or select Download All. You can also use Ctrl + click to select several files or Shift + click to select several files in a row, then right click and select Download to download the selected files. The files will then be downloaded and can be viewed, analyzed or sent to a report. Refer the following sections or more details on these operations.



■ Figure 14. Device File View

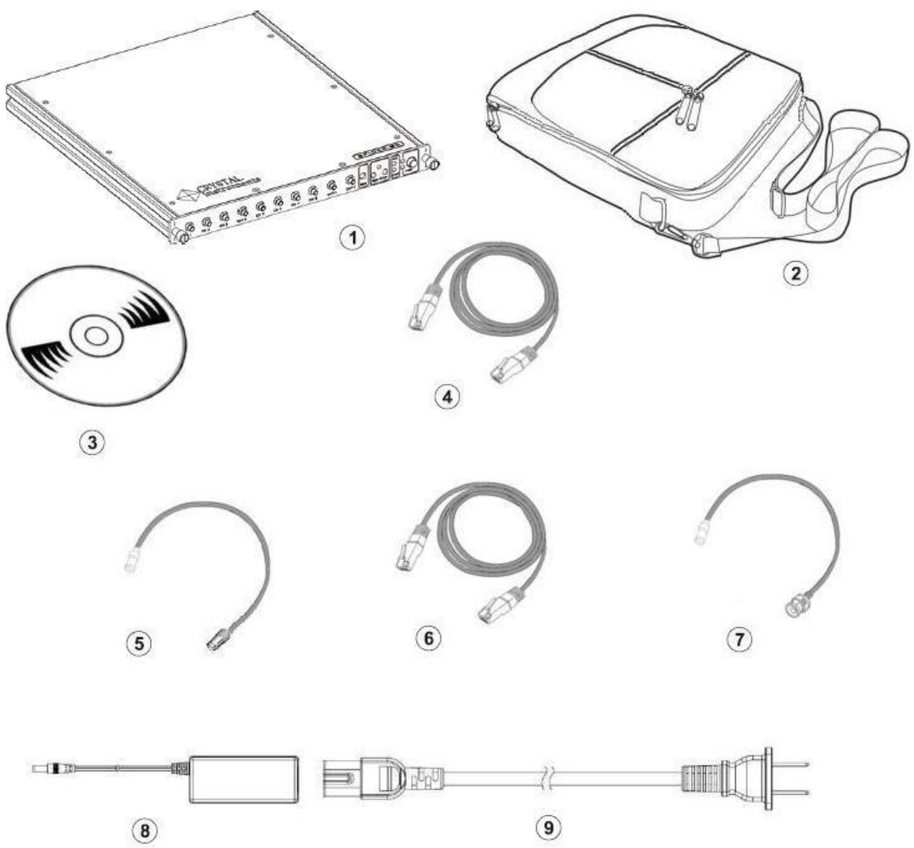
The files with extension .ATFX stand for are native ASAM-ODS files that the EDM supports.

To open/view the file, highlight the file or multiple files in your local folder, right click, and select the View command.

3. EDM Installation for Spider Users

Open the Shipping Box

Packing Item Description:



#	Description	Quantity
1	Spider-80 Dynamic Measurement System (Default IP Address: 192.168.1.153, Subnet Mask: 255.255.255.0)	1
2	Carrying bag	1
3	CD for EDM, the PC software. User's Manual in PDF	1
4	Regular Ethernet cable	1
5	SMB-BNC(Female) cables for input channels	8
6	Cross-Over Ethernet cable	1

7	SMB-BNC(Male) cable for output channels	2
8	AC/DC power adapter	1
9	Universal power plug	1
10	Calibration certificate (not shown in the drawing)	1

Summary of Installation and Configuration Procedures

The installation and configuration procedures for Spider include five steps:

- Install the MySQL database server (keep the password)
- Install the Engineering Data Management (EDM) software
- Connect the Spider to the computer or Local Area Network (LAN)
- Set Spider IP to match with that of your computer
- Run EDM

Installation of Engineering Data Management Software

Software Included on Installation CD

The Installation CD includes Crystal Instruments Engineering Data Management Software (EDM) plus several other necessary software files required for proper operation. This is the software and user interface used to connect the Spider hardware to the computer and manage your data. EDM uses Microsoft Word for generating reports and it should be installed on your computer prior to the installation of EDM.

MYSQL Server Software: MySQL Server is required to manage the database system that stores route and measurement data on the PC and Spider. It should be installed before the EDM software.

Microsoft .NET Framework: Microsoft.NET will be installed automatically if it is not already installed on your computer.

EDM Software CD

To install the EDM software and related software systems included on the CD, place the installation CD in the CD drive on your PC. If the Welcome Screen does not automatically open you can run the Setup.exe file on the root level of the CD. Select the physical drive with the CD, open the EDM folder, and double click on Setup.exe.

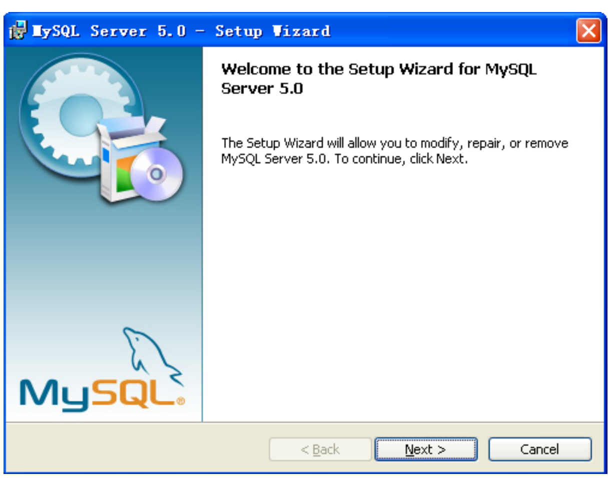
■ Figure 15: Welcome Screen for EDM Installation CD.

Prior to installing EDM, the MySQL Database Server needs to be installed on your system. Select the MySQL button to initiate.

MySQL Database Server Installation

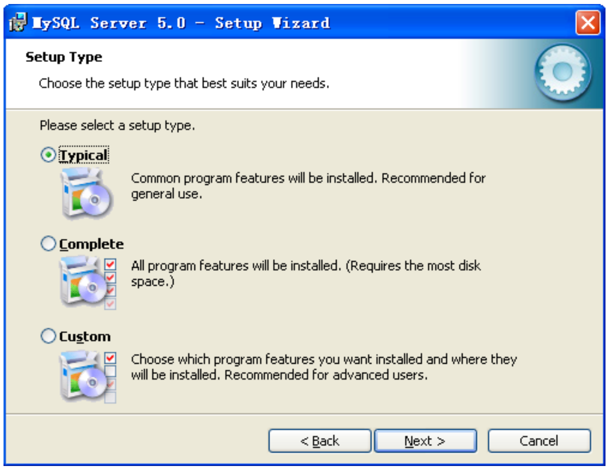
The following instructions describe how to install the MySQL Server Software. MySQL Server is separate software licensed for use with EDM that manages the database where route and measurement data is stored on the PC and Spider.

Click MySQL Server on the EDM Software Installation screen. The MySQL Setup Wizard will start:



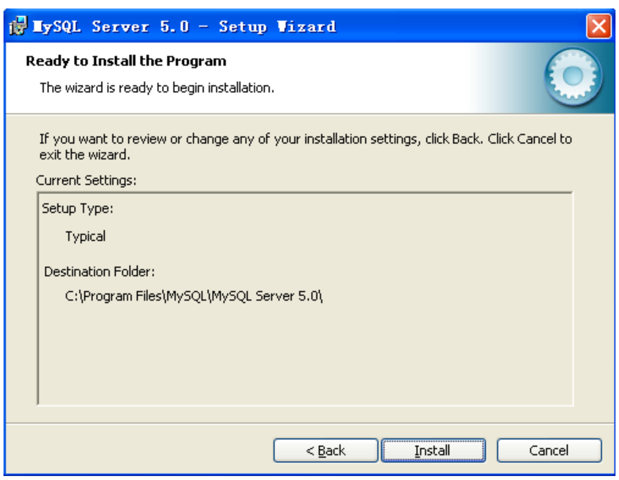
■ Figure 16: MySQL Setup Wizard

Click on *Next* button. You can then choose the program folder, or use the default folder. Then select *Typical.* It is not necessary to install all components of MySQL Server:



■ Figure 17: Typical installation of MySQL Server

Click on *Next* button. Then Click on the *Install* button:



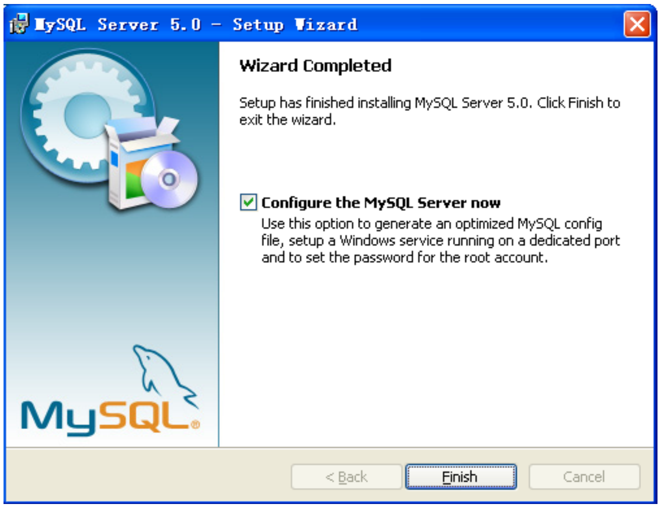
■ Figure 18: MySQL Server installation destination

Wait for a moment while the system sets up the installation process.

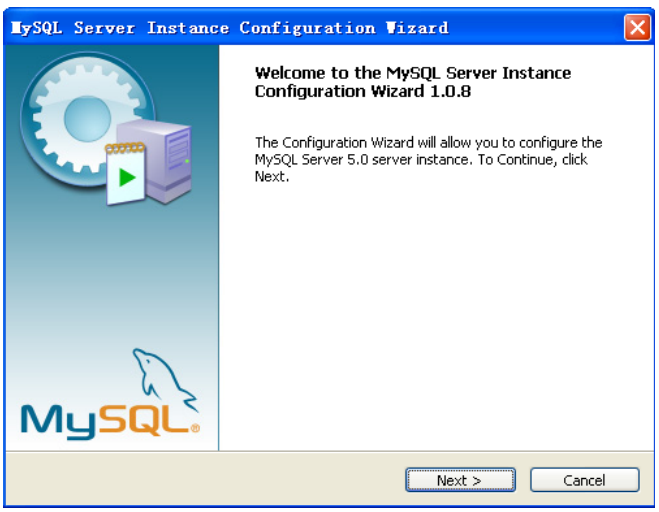
Click on the *Next* button.

Click on the *Next* button again and the Setup Wizard will complete its initial operation.

You are now ready to configure MySQL Server. Select the *Configure the MySQL Server now*, and click the *Finish* button.

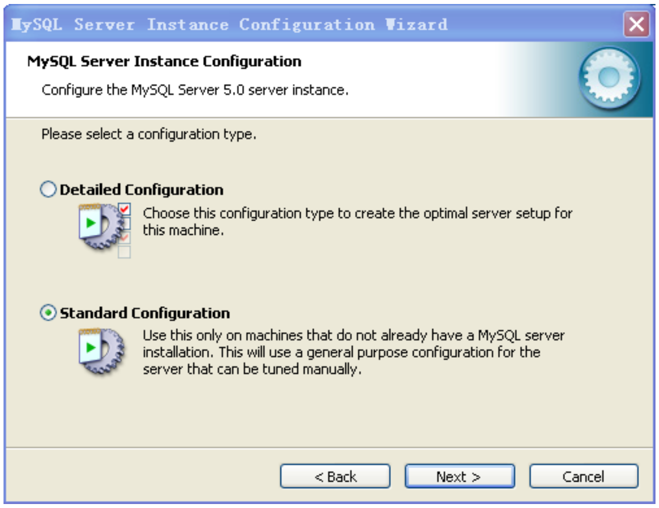


■ Figure 19: Configure MySQL Server



■ Figure 20: Configure MySQL Server Instance

Click on the *Next* button. When the following screen appears, select *Standard Configuration*, click on the *Next* button.

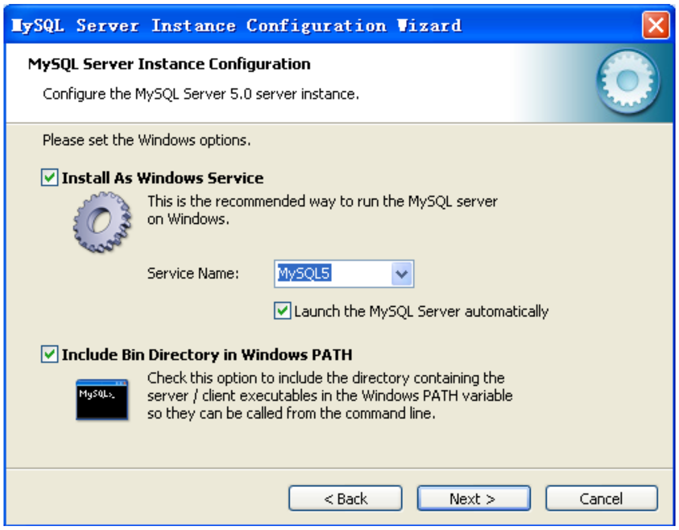


■ Figure 21: Select the Standard Configuration of MySQL Server

To properly configure MySQL for use with EDM, the following configuration choices should be selected:

1. Select ***Install As Windows Server***.
2. Enter a ***Service Name***, or simply select ***MySQL5***. The name you create must not have any spaces in it.
3. Check the ***Include Bin Directory in Windows PATH***.
4. Select ***Launch the MySQL Server automatically***.

> Caution: Be sure that all these selections have been made. EDM runs on MySQL and will not operate properly if these options have not been selected during MySQL installation.



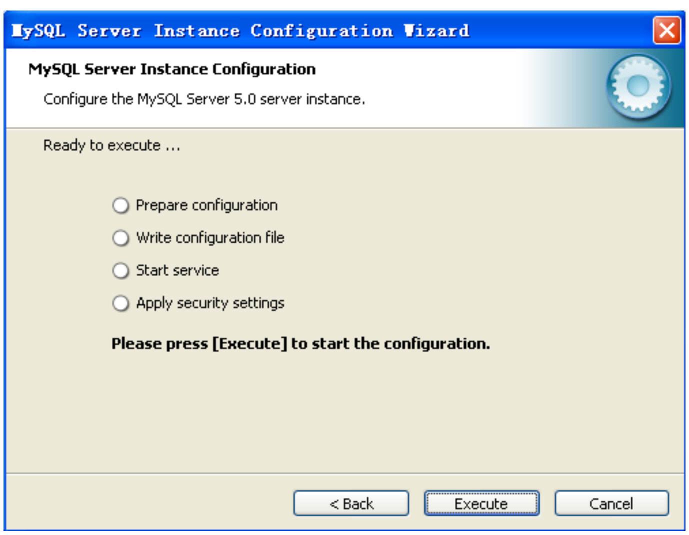
■ Figure 22: Configuration choices for MySQL Server

Next, enter a password for the root super administrator. If you want other computers to access the database on this computer, check ***Enable root access from remote machines***.

> Important! You should make a record of the password and keep it in a safe location. This password is required whenever you access the database. This would also be a good time to make a record of the License Key for EDM, in case there should be a time when you need to look it up.

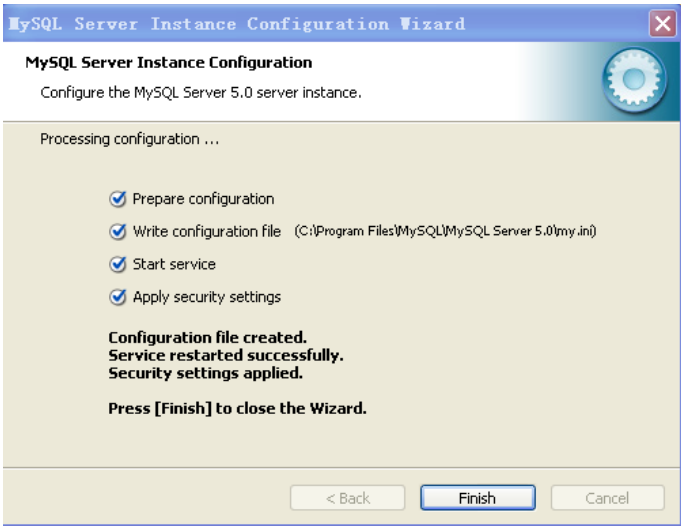
Click on the *Next* button.

Click on the *Execute* button.



■ Figure 23: Press *Execute* to start the configuration

Click on the *Finish* button. The installation is now complete.

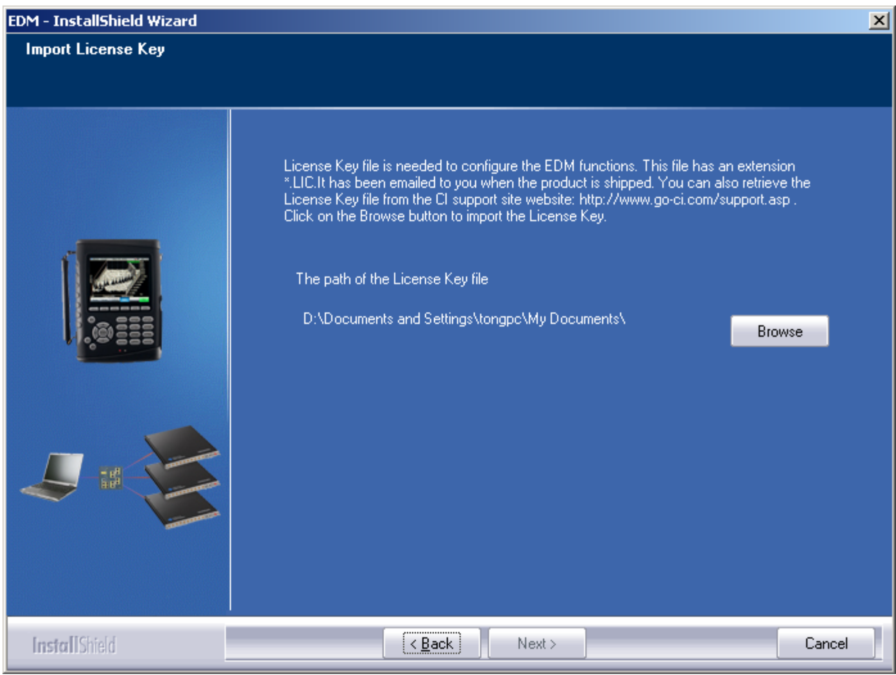


■ Figure 24: Completed MySQL Server Installation Configuration

EDM Software Installation Wizard

The next step is to install the EDM application software.

To install the EDM Software please click on Install EDM Software and follow the instructions. The EDM software uses Microsoft .NET resources. The .NET resource software is native in Windows Vista/Windows-7 but must be installed for Windows XP. If the installation detects that .Net is not installed then it is automatically installed from the CD.



■ Figure 25: EDM Software Installation Wizard

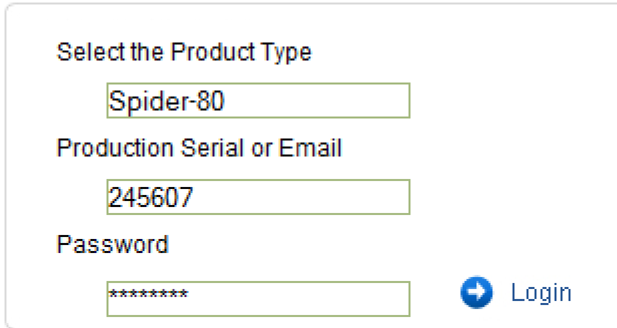
Where is My License Key?

There are two ways to obtain your EDM software License Key:

When your EDM Software and Spider are shipped from Crystal Instruments, The CI IT system will send out an automated email message providing shipping information, your License Key and the Serial Number of your instrument. The License Key is a file that contains the extension *.LIC.

If you have not received the automated email message, or do not have your License Key, log into the CI Technical Support Site: http://www.go-ci.com/support.asp using the Spider serial number and the password provided in the automated message mentioned above. You can then retrieve the License Key from the technical support site.

> It is a good idea to store the License Key, Serial Number of your instruments and Password somewhere safe where you can always find it. You will need this information to log on to the CI Technical Support Site for assistance and updates. If they are lost, please call CI Technical Support at 1-408-986-8880.



- Figure 26: Product Account Login at www.Go-CI.com

Connect Spider and Turn It On

Connect Spider to PC

The Ethernet CAT5 jack on Spider is on the rear of the instrument. There are two ways to connect the Spider module to a computer, either through a LAN (Local Area Network) or using a crossover Ethernet connection to connect directly to the computer.

If your Spider system consists of more than one Spider module, you will need to configure multiple Spider modules with a network switch or router and the PC as shown below.

- Figure 27: Computer private LAN with router and multiple Spider modules

This is called LAN (Local Area Network). It is highly recommended that the Spider system uses a private LAN instead of sharing your office LAN for general use.

Figure 28: Computer directly connected to Spider

Set Master or Slave Mode

The M/S switch on the back is used to control whether the Spider module is configured as Master or Slave mode. “M” stands for master and “S” for slave mode. In a Spider system, only one module can be configured into Master mode.

If you are using only one Spider module, the system will internally configure it as master mode and ignore the user setting. If additional modules are added at a later date, it will be necessary to configure the first module as the Master and each of the additional modules as Slave mode.

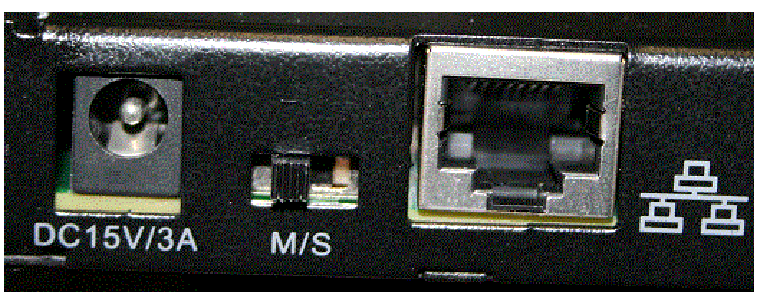


Figure 29: Switch for setting Master/Slave mode

Connect the Power Adapter

The power jack of the Spider module is on the rear side, as shown in Figure 15 above..

Now connect the power adapter to Spider and AC power source. The power adapter is an AC to DC converter that accepts the 100 VAC to 240 VAC and output at 15VDC. The total power consumption of a Spider module is less than 10 watts during full operation mode.

Turn Spider Power On

After the Ethernet cable and power adapter are connected to the Spider, press the red Power button on the front side.

- Figure 30: Front display panel of Spider module

The PWR LED indicates the power status.

The SYNC LED indicates the time synchronization status when multiple Spider modules are networked.

The LED above Run and Stop button indicates whether the test is running.

The MEM LED indicates whether the Spider is accessing its internal memory. When it records the signals, this LED will be lit.

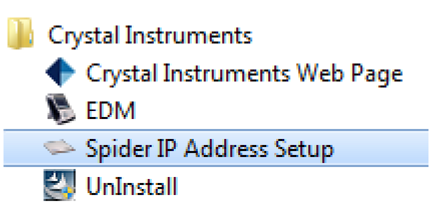
Set Spider IP Address

When each Spider is shipped, unless the user has requested otherwise, it is set to following default IP address and subnet mask:

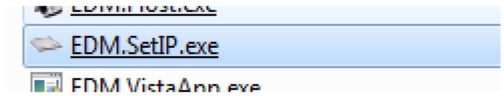
IP Address: 192.168.1.153
Subnet Mask: 255.255.255.0

If the user asks to set the IP address and subnet mask to match their computer settings, the production department will set them per user request.
The Spider IP address can be set by using a simple tool, ***Spider IP Address Setup***, or using EDM when the EDM is connected to the Spider.

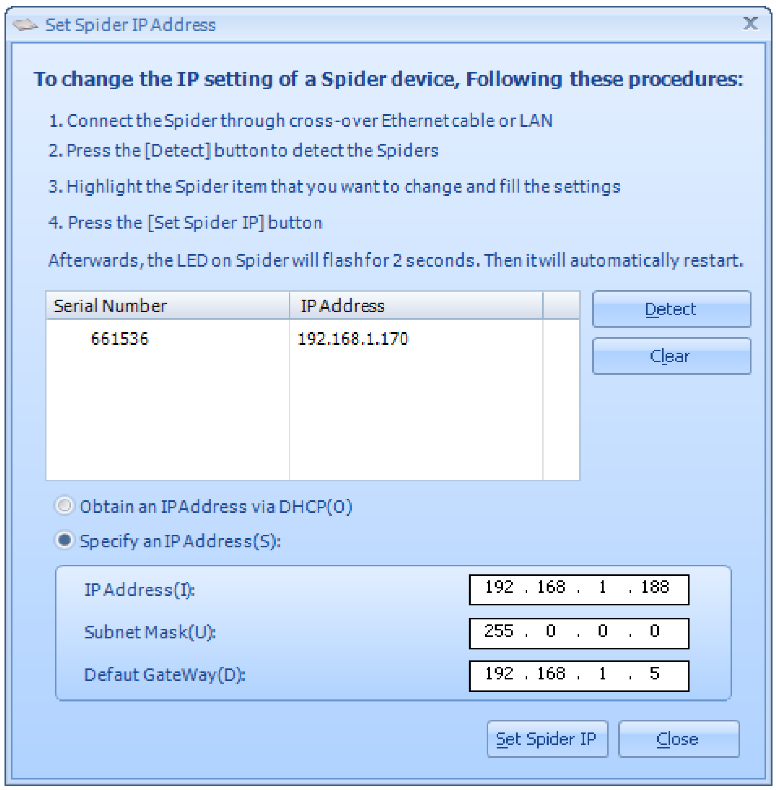
The ***Spider IP Address Setup*** tool can be found in the installed folder:



Or the executable file, EDM.SetIP.EXE, can be directly found in the installation folder:



After the tool is launched, you will see following user interface.



■ Figure 31: Spider IP Address Setup Tool (EDM.setIP.exe)

Select the module in detected Spider module list, and set the IP address configuration as you intended, and click the button ***Set Spider IP***. Afterwards, the LED on Spider will flash for about 2 seconds. It indicates that the new IP has been assigned. Make a note of the IP address you have assigned to each module and where the module is being physically located. This will be helpful later when you are creating a network of Spider modules.

Reset Spider to Factory Default IP Address (Optional)

If the user lost the information about the current IP setting of the Spider module and was unable to connect to the Spider, he can always reset to the default IP address, 192.168.1.153, by following these steps:

1. Turn the Spider power on
2. Switch the Master/Slave button to the Slave position
3. Press the Stop button for 4 seconds.

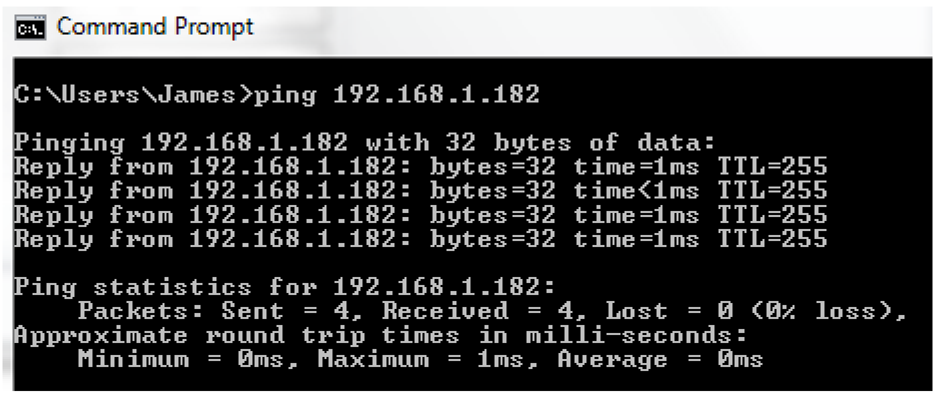
This process will reset the IP address to the known factory default IP address.

Computer Network Settings

The computer has to have the same subnet Mask to match with that of Spider in order to make the first connection.

To verify the network connection, use the PING command with the PC Command Prompt tool to test it. In Windows, click Start>Run to get to the Command Prompt. Then type ***ping 192.168.1.182*** and ***enter***.

If the computer can communicate with the Spider, the computer should receive the Ping messages without loss as shown:

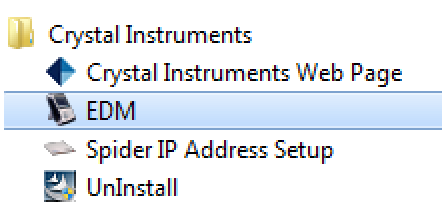


■ Figure 32:Command prompt and ping to test communication with the Spider module on the network

Run the EDM Software

Launch EDM

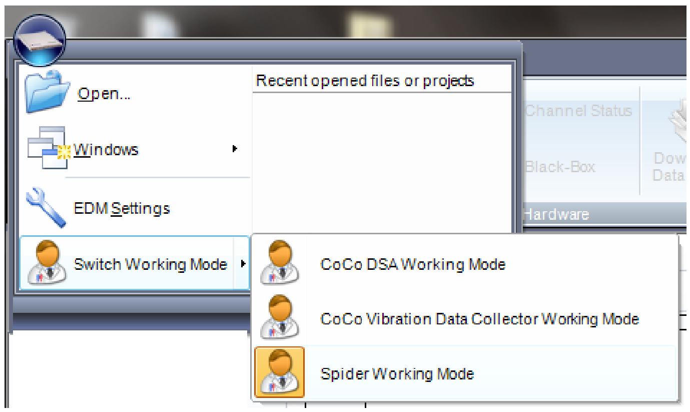
To continue the installation of the Spider module, you will need to launch the EDM software. Select the EDM icon from your Crystal Instruments folder or from the folder where you have installed the software.



■ Figure 33: Select EDM from the Crystal Instruments folder

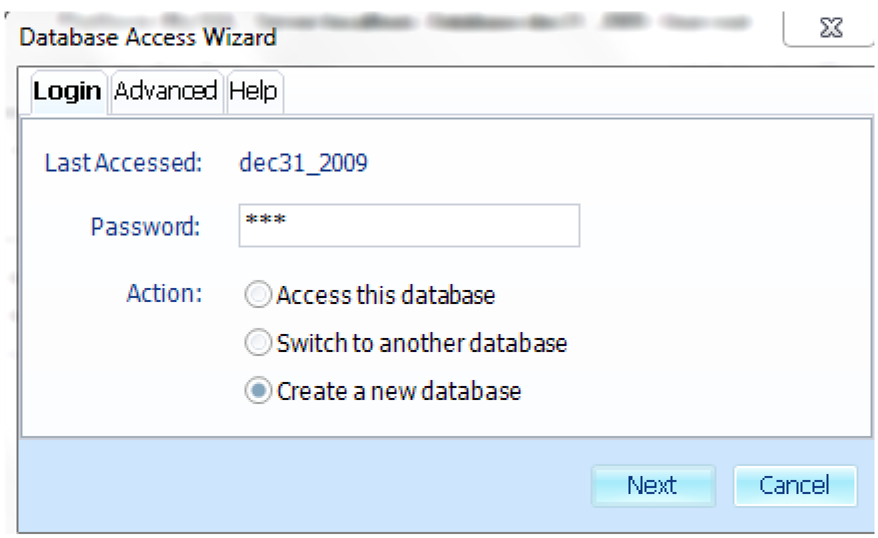
Select Spider Real-Time Working Mode

Make sure that the Spider Real Time working mode is selected. Click on the top left EDM button. Under the Switch Working Mode, select Spider Working Mode.



Create a Database

Once EDM is running it is necessary to create a database to store and access the measurements that will be performed by the Spider module. For complete instructions, please refer to the EDM Manual. This section will assume that you are familiar with EDM.

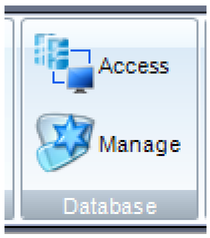


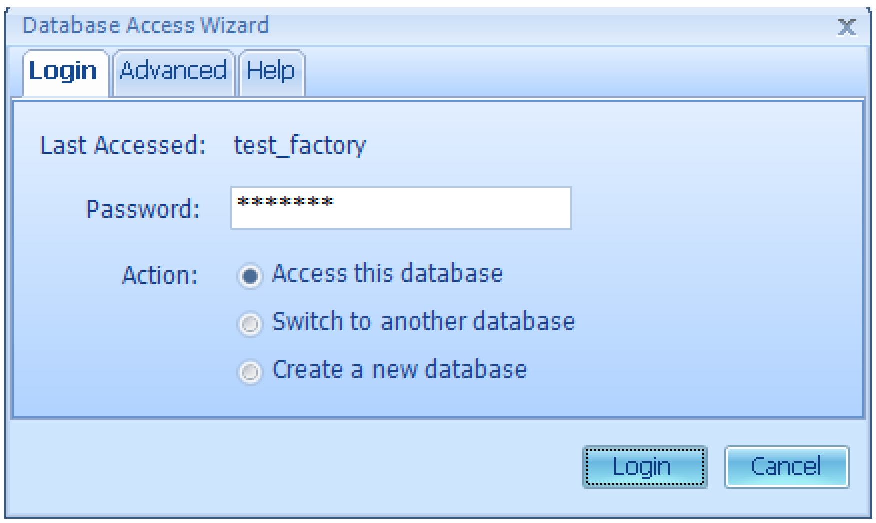
■ Figure 34: Create a database using the Database Access Wizard

Login to EDM and select ***Create a New Database*** to create the new database for the Spider module.

Database Toolbar

After you have logged in to EDM, the Database Toolbar includes commands related to accessing and managing the database where routes and data are stored. Select ***Access*** on the toolbar ribbon to switch to an existing database.

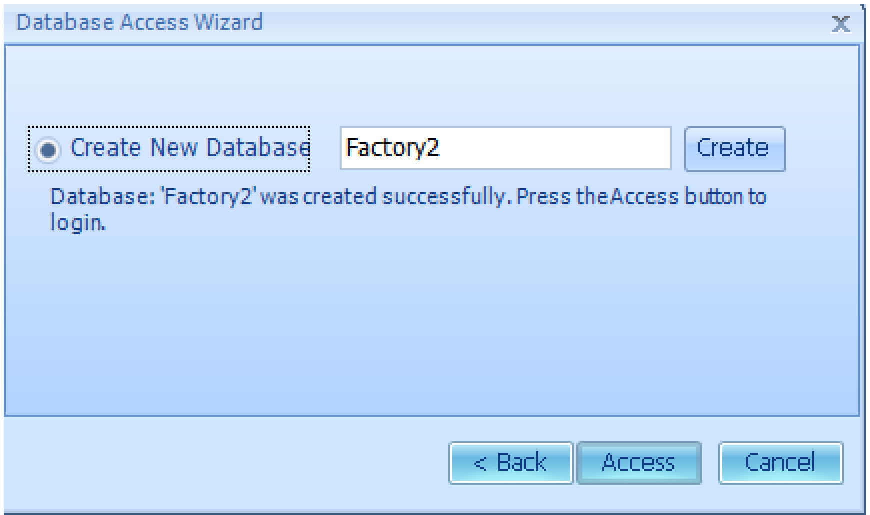




■ Figure 35: Database Access Wizard, switch database to test factory

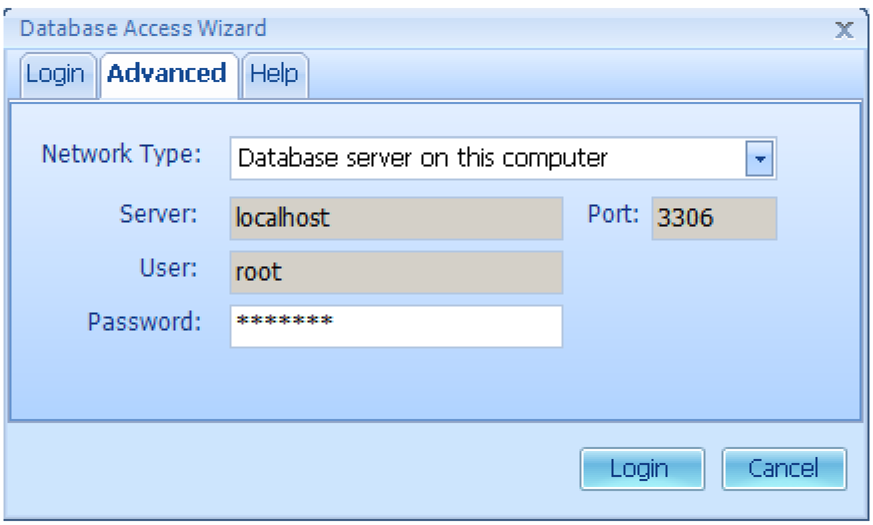
- **Access** – is used to connect to a database.
- **Login** – the login tab allows you to select which database to access or to create a new database.
- **Password** – enter the password that was defined when the current database was created.

If no database has been previously created, you will only be able to select **Create a new database**. Click on the ***Next*** button and it will open the Database Access Wizard. Enter the new database name, click the ***Create*** button and wait while the software sets up the new database. Click the ***Access*** button after the database has been created.



■ Figure 36: Database Access Wizard, Create a new database

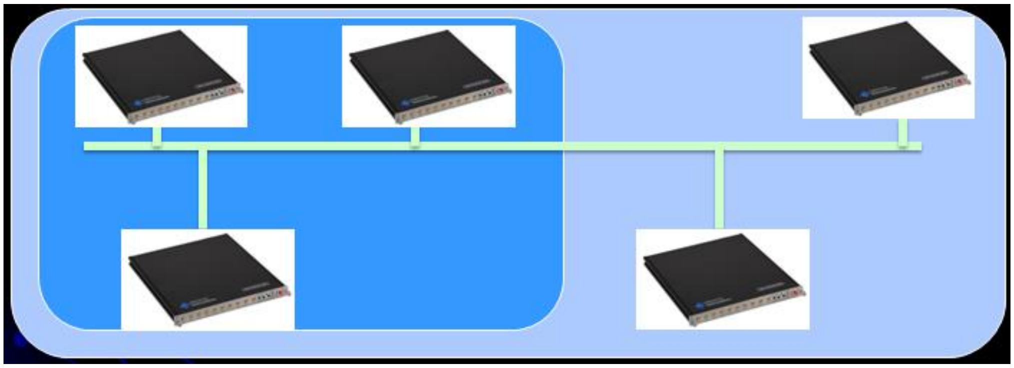
Advanced Tab – This tab is where you specify the Network Type: Database server on this computer or on a network, the Server name, User, Port and password.
Enter the MySQL Server password that was created when MySQL Server software was installed. Refer to the EDM Installation Chapter for more information.



■ Figure 37: Advanced tab for the Database Access Wizard

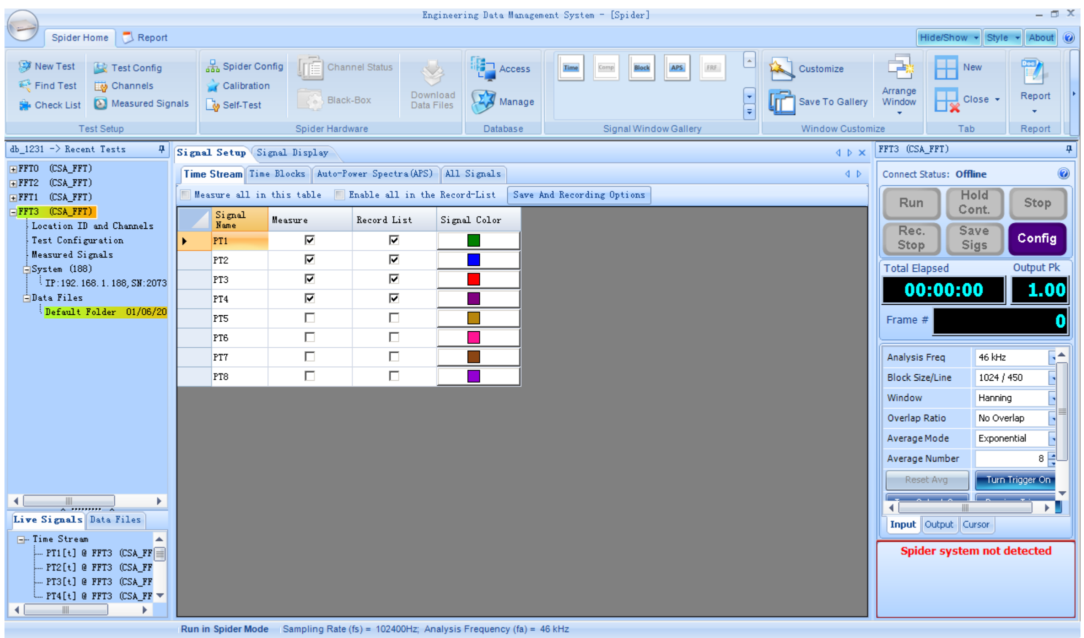
Configure a Spider System

A Spider system can be configured using any combination of Spider modules that are available on the LAN. EDM software will store multiple configurations for the network and can recall any one of them.



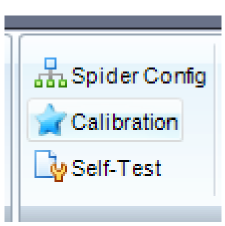
■ Figure 38: Spider modules on a LAN

For example, the user can configure a group of Spiders on LAN as one system, and configure a second system with another combination of Spider modules on the same LAN.



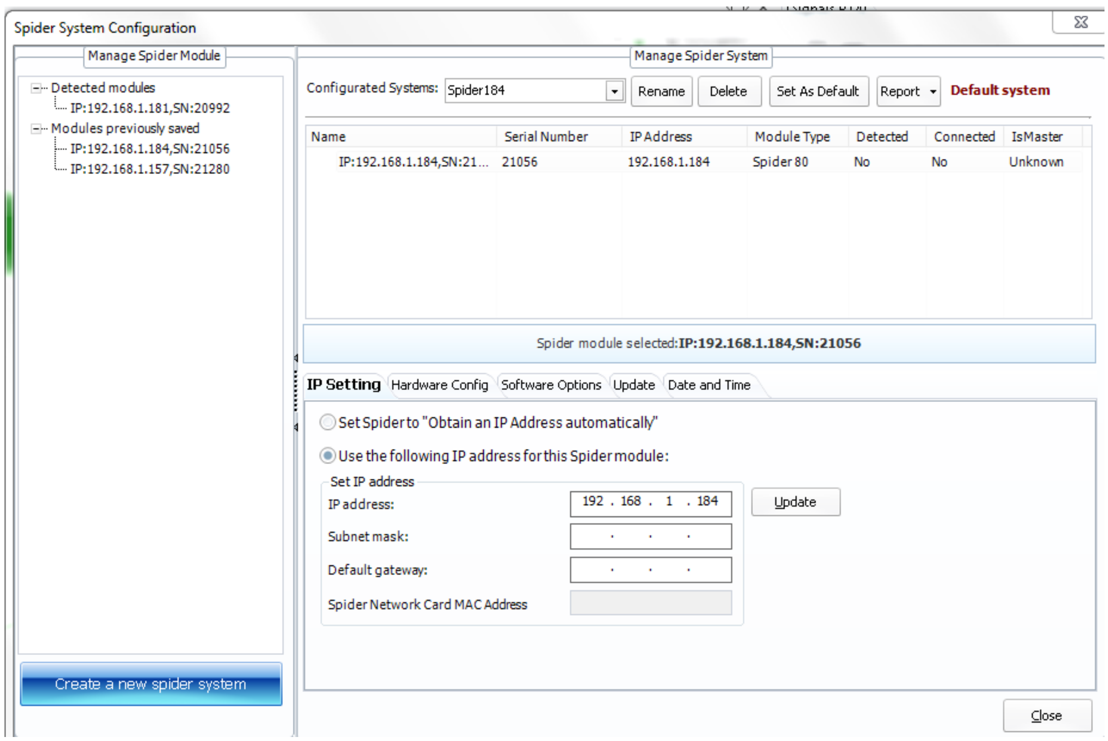
■ Figure 39: EDM in Spider Mode

When all the Spider modules have been connected to the LAN, click on ***Spider Config*** on the ribbon. This will start the Spider Configuration process.



■ Figure 40: Spider Configuration button on the Ribbon

The Spider System Configuration dialog box below will appear. It shows all the Spider modules connected to the system. The IP addresses that you have assigned and the serial number of each Spider module will be shown.

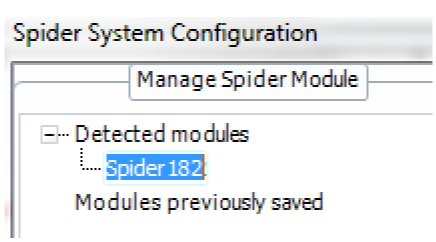


■ Figure 41: Spider System Configuration dialog box

The EDM will detect all currently Spider modules that are being connected. Click on the ***Create a new Spider system*** button on the left side, and check those modules that are to be included in a system, and assign a Spider system name.

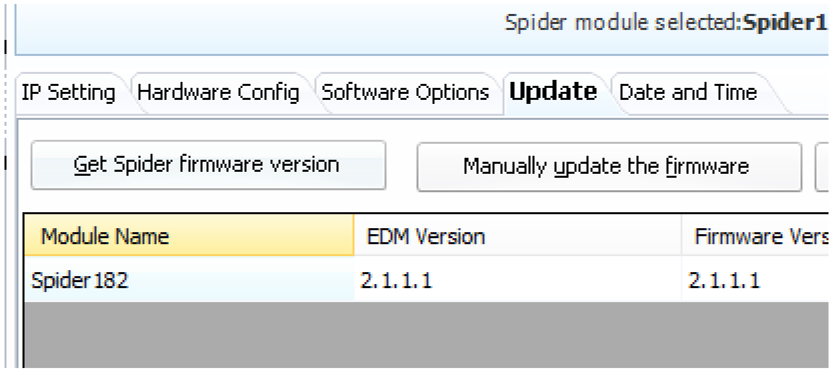
Manually Update the Firmware (Optional)

The version of EDM and the Firmware version must be the same throughout the system to make it functional. The EDM software will automatically detect the firmware version and update it when it is needed. However, the user can also choose to manually update the firmware on any Spider module. To do so, first make sure that the Spider module is on the Detected Modules list. Then highlight it:



■ Figure 42: Select Spider module to update

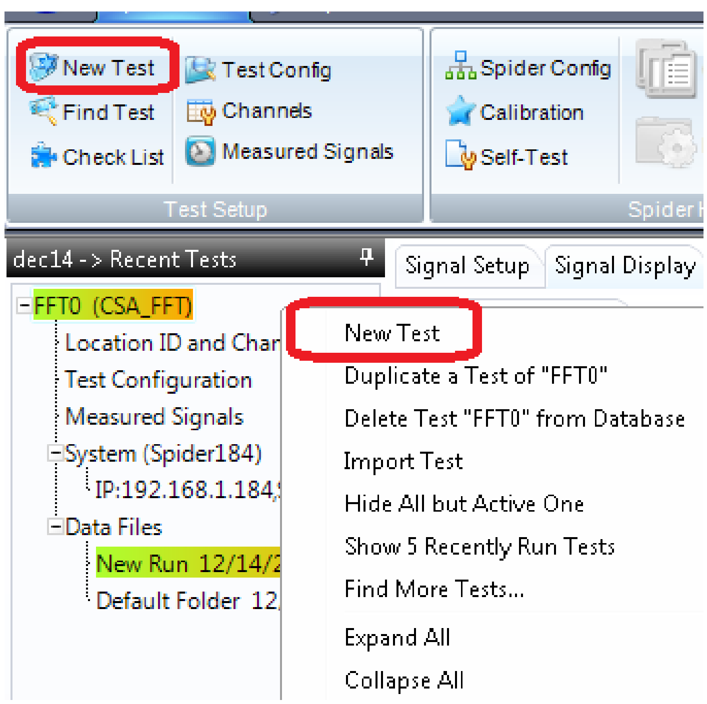
Click on the ***Update*** tab, then the button ***Manually Update Firmware.***



■ Figure 43: Display of Spider information

Create a New Test

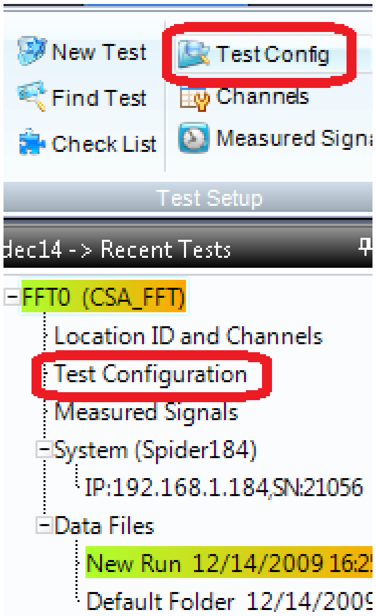
Once the Spider module has been connected and configured on the LAN, it is necessary to program it to capture the signals and time stream. There are two ways to create a New Test. First, click on the ***Test Setup*** tab and then click on ***New Test***.



■ Figure 44: Test Setup, creating a New Test

Set up the Test

There are two ways to access the Test Configuration dialog box:



■ Figure 45: Test Setup, configure a new test

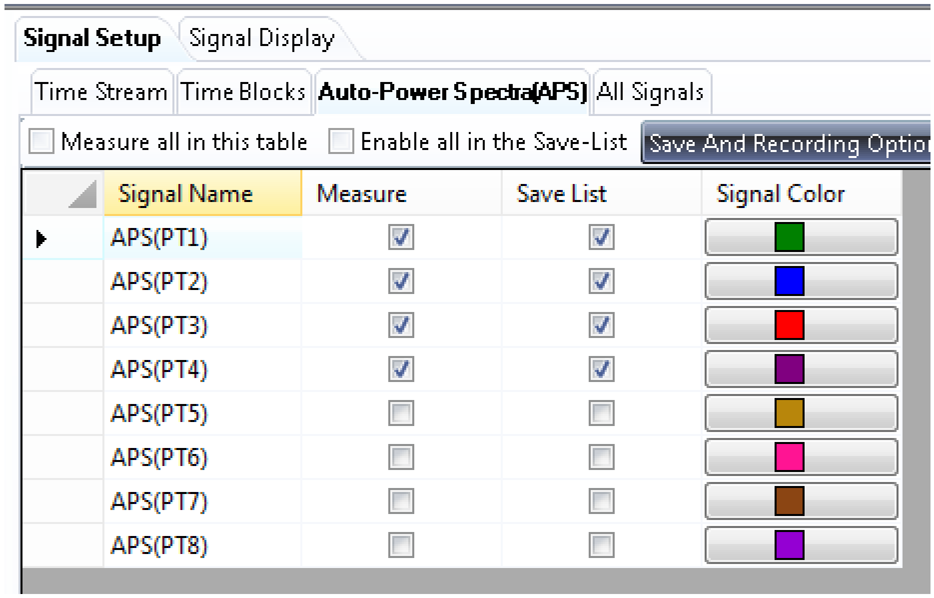
The test configuration is a multiple -tab dialog box that allows the user to set up the analysis parameters, schedule, event-action rules etc.. Some of these parameters can also be set directly on the control panel while the Spider is running.

Execute the Test

■ Figure 46: Executing the test

The test is controlled by the push buttons on the control panel.

Save Signals and Record the Time Streams



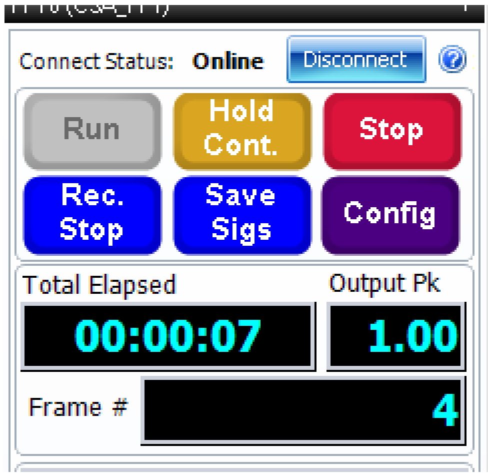
■ Figure 47: Signal Setup, showing Auto-Power Spectra

Based on the different possible applications of the Spider system, several ways are available to save the signals that are being measured. The storage media may be internal flash memory of the Spider module, a network hard drive or the hard drive of the local computer where EDM is installed.

Save Long Time Waveform Signal: Time streams can be saved either automatically by a preset schedule or manually.

Save Block Signals: The transient capture time signals, frequency signals or other block signals can be saved automatically or manually.

Record Long Time Waveform Signals



■ Figure 48: The Spider dialog box allows you to control the Spider module manually from the computer

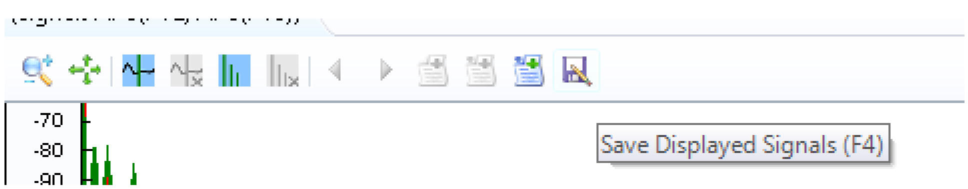
The ***Rec./Stop*** Button is used to manually control the capture of time stream data to memory. After a test project is selected, pressing the ***Rec./Stop*** button will start the display and also start recording the time stream to memory. The red flashing ***Rec*** icon at the top of the screen indicates that the data is recording.

To stop the recording press the ***Rec./Stop*** button again. The red flashing ***Rec*** icon will not be displayed indicating that the recording has stopped.

Save Block Signals

There are several ways to save the block signals:

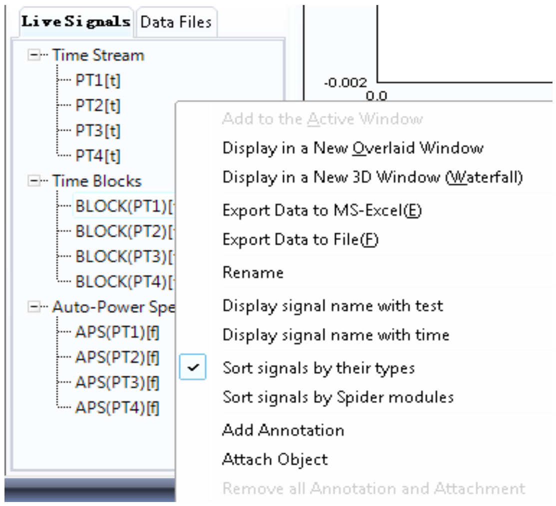
- Method 1: Press the ***Save Sigs*** or ***F5*** button in the control panel to save the signals in the pre-defined list.
- Method 2: Press ***F4*** button to save signals being displayed, or click on the small ***Disk*** icon below:



- Method 3: Use the ***Run Schedule*** to save the signals in the pre-defined list.

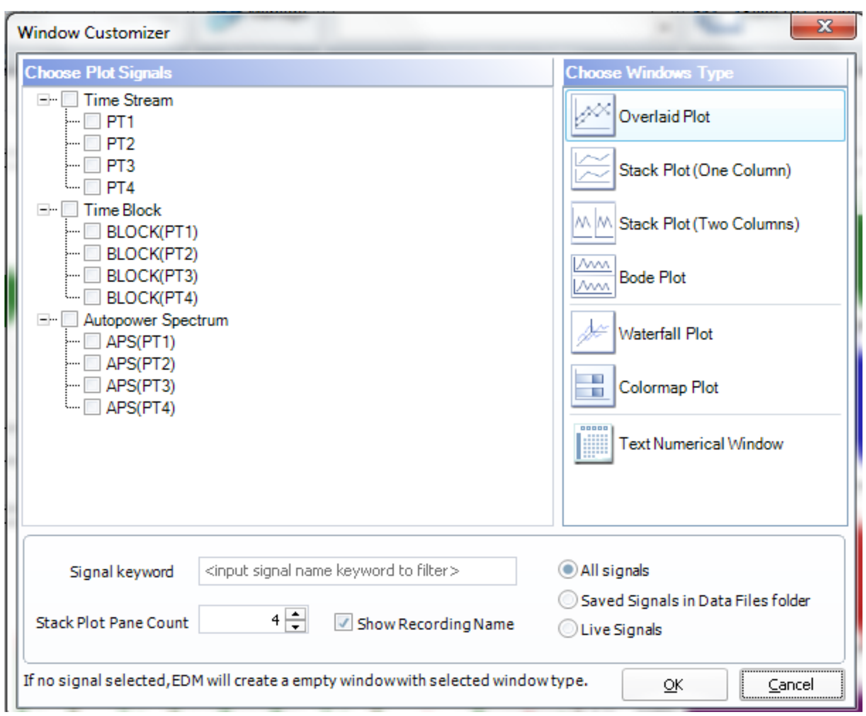
View Live Signals

To view live signals in real time, right click on the signal item in the Live Signals pane. This will bring up a dialog box where you can select the type of display.



■ Figure 49:Live Signals dialog box

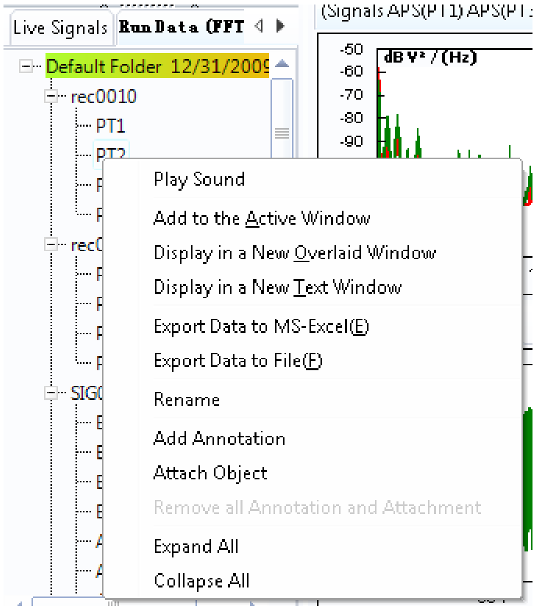
You can also click on the Customize icon on the toolbar ribbon. This brings up a Window Customizer dialog box where you can select the Plot Signals and the type of display you would like to see.



■ Figure 50: Window Customizer dialog box

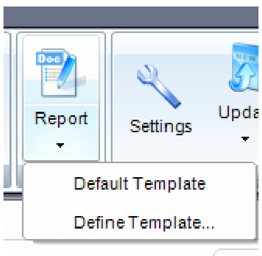
View Saved Data

When you are in the Run Data pane, you can right-click on any item to display the signals.



■ Figure 51: Display Data dialog box

Create Report



Click on the ***Define Template*** command under Report to define a template. Then Click on any templates that were previously defined to generate the report.

Run Spider in Black Box Mode

The Spider-80 modules can be run in “black box mode” where a preprogrammed schedule is uploaded to the unit from a computer. The Spider module may then be started manually or it can be based on an external event trigger. The ability to use any module in black box mode or in a network distributed system means that you can place your modules close to the measurement object.

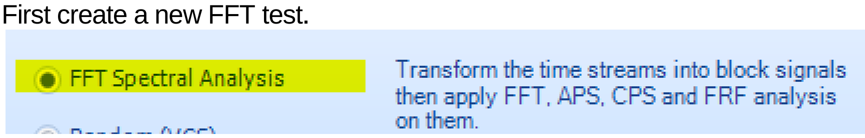
Each Spider module has its own mass storage media that keeps the operating software and stores the measurement data. This truly distributed structure guarantees signal recording at full speed and is not subject to any network speed limitation or fluctuation due to traffic on the network.

To run the Spider in Black Box mode, please follow the procedures in the following example.

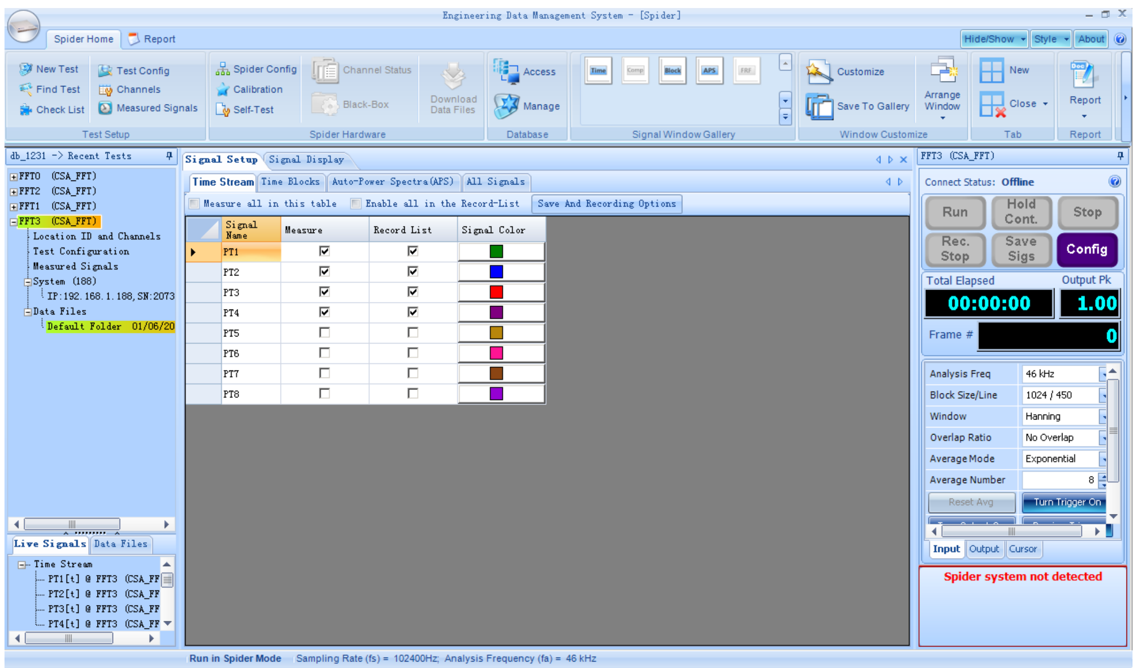
Create a Test in Black Box Mode

All the tests that can run in PC operating mode can be pushed to the Spider module and run in Black Box Mode. Let's start to show this process by an example.

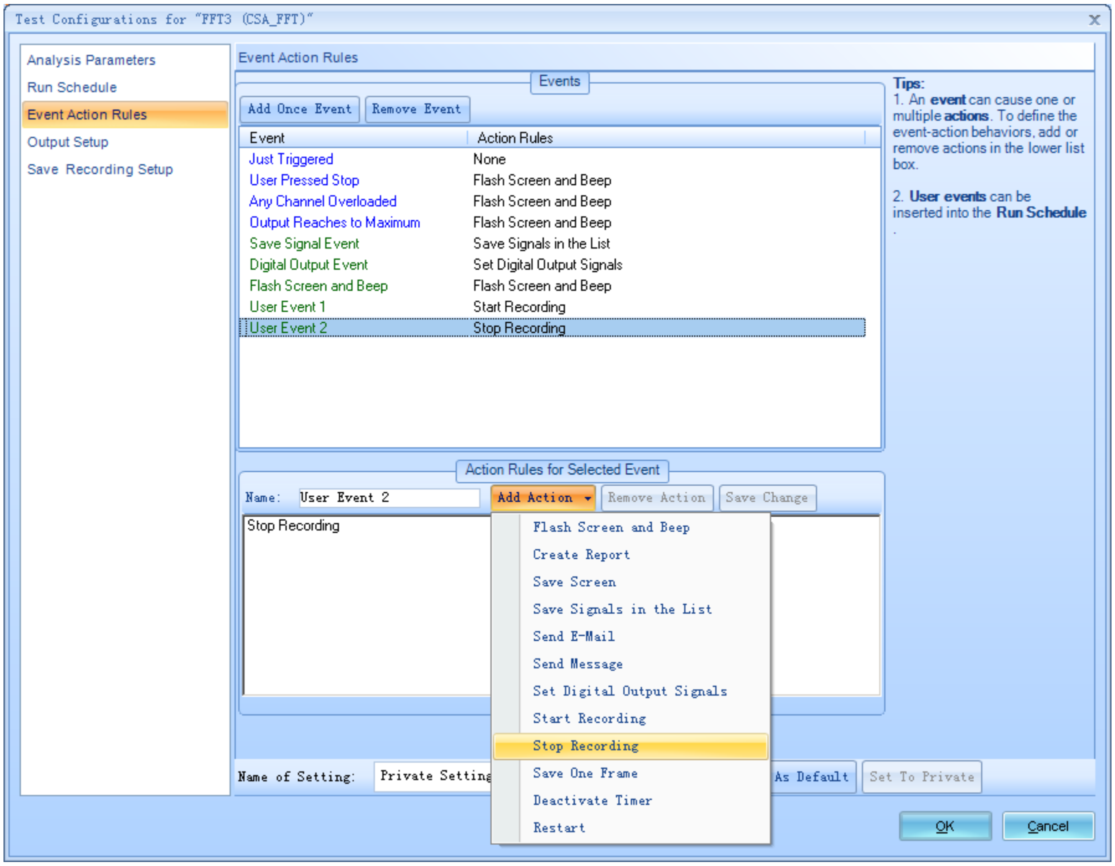
First create a new FFT test.



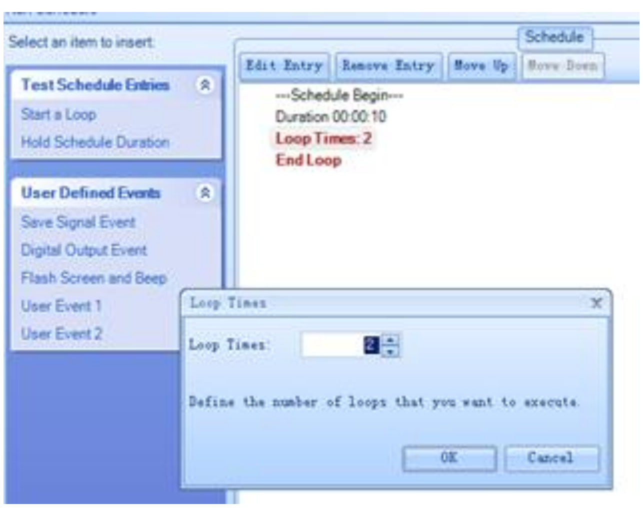
Make selections to the recording signals:



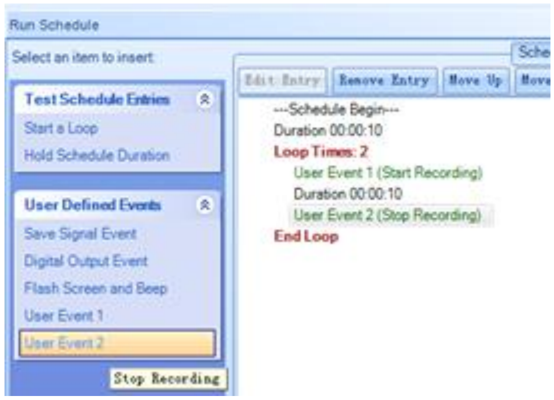
Create Schedule Event: Start Recording Event and Stop Recording Event



Make a Looped Schedule



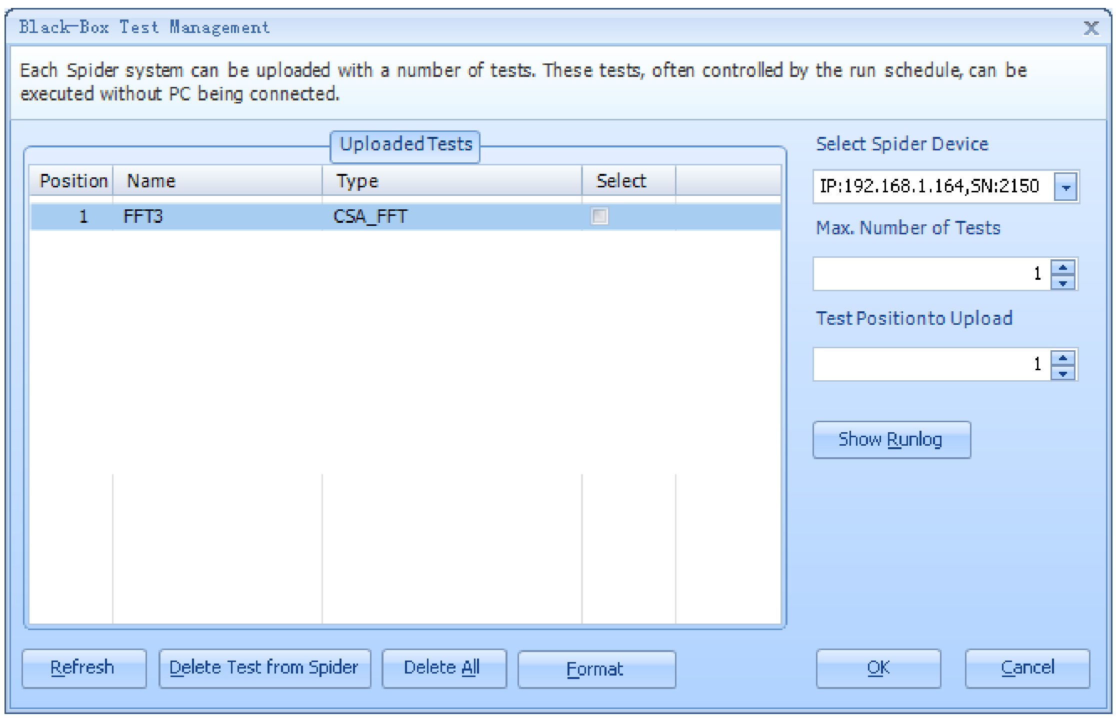
Insert the Event



Connect the Spider, Run the Test from PC once.

Open Black Box dialog box and press Refresh button.

It is shown that the testing project has been uploaded to the Spider.

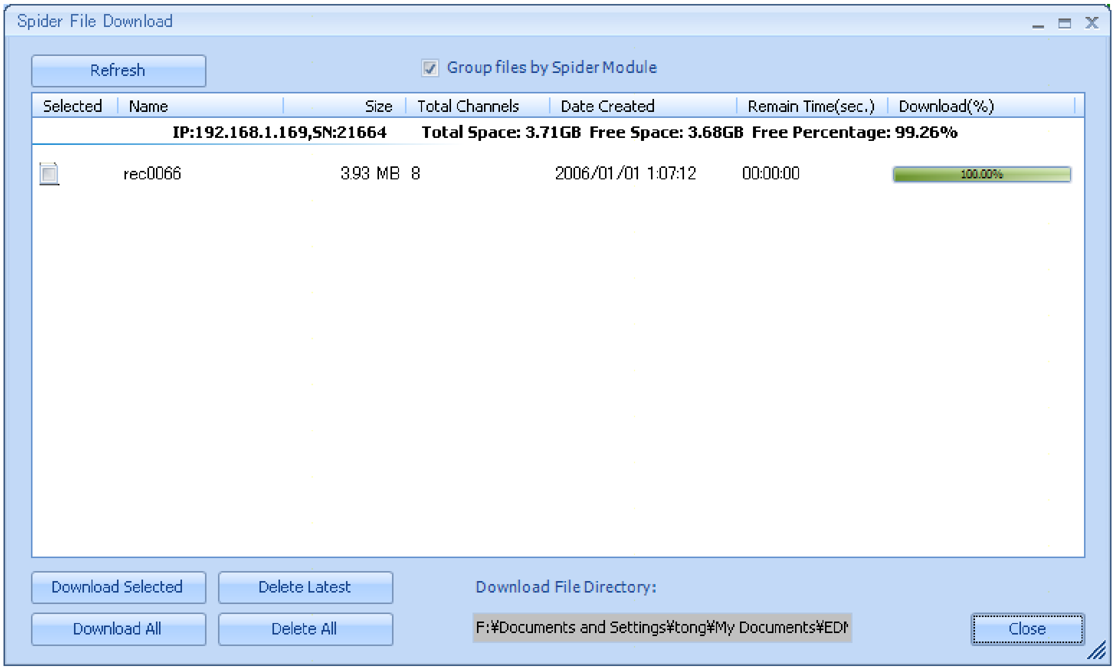


Click Disconnect button on the Spider. Exit the EDM.

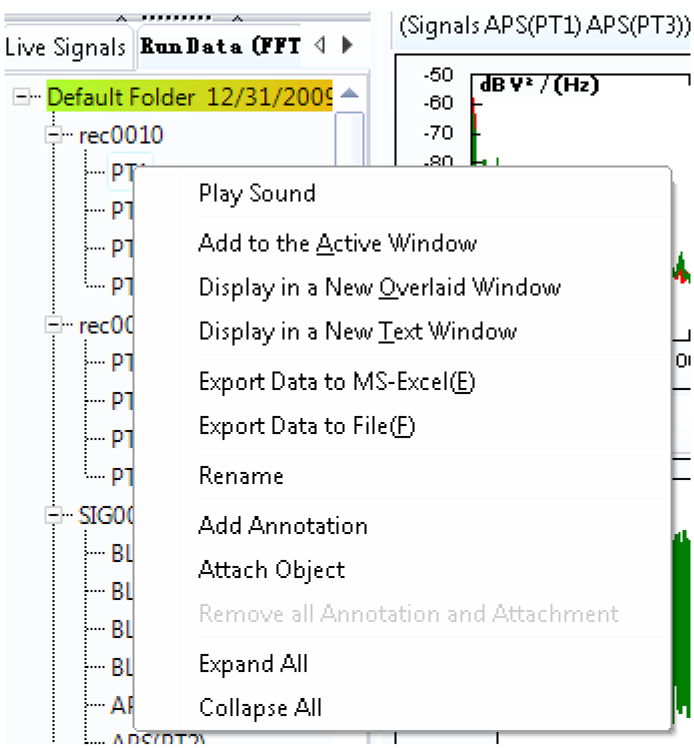
To run the Spider in Black Box mode, press the ***Start*** hard button on Spider, watch the MEM LED. When it is lit, it indicates that the recording is on-going.

Press the ***Stop*** hard button to stop the test.

Reconnect the EDM, click ***Download*** button. Download the files from the Spider to PC.



Right click on the signal files in the low left corner the Data Folder pane. The downloaded signals can be viewed.



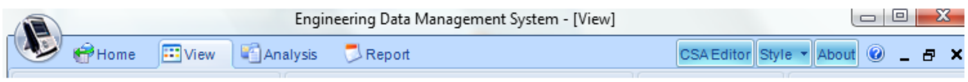
This concludes the description of how a test can be executed in the Black Box mode.

4. EDM User Interface

The EDM user interface has a unique design that emphasizes the use of graphical tools to perform most operations. The interface includes several components designed to allow you to navigate between displays quickly and easily. The following section gives a brief explanation of these tools. More detailed explanation of each page is given in the following sections.

Page Views

EDM is divided into several different Pages including: Home, View, Analysis, Report, CSA Editor, Style and About Tab. You can change from one page to another at any time by clicking on any of the page tabs shown in Figure 52. Each Page View is described in more detail in the following sections.



■ Figure 52. Page tabs can be used to change between the Home, View, Analysis, Report, CSA Editor, Style and About Pages.

5. Home Page

This section describes the Home Page in detail. The Home Page is used to view and transfer files between the PC and the CoCo-80 Device. It includes the following components.

Start Button – includes commands such as open data files, manipulate windows, settings and remote display.

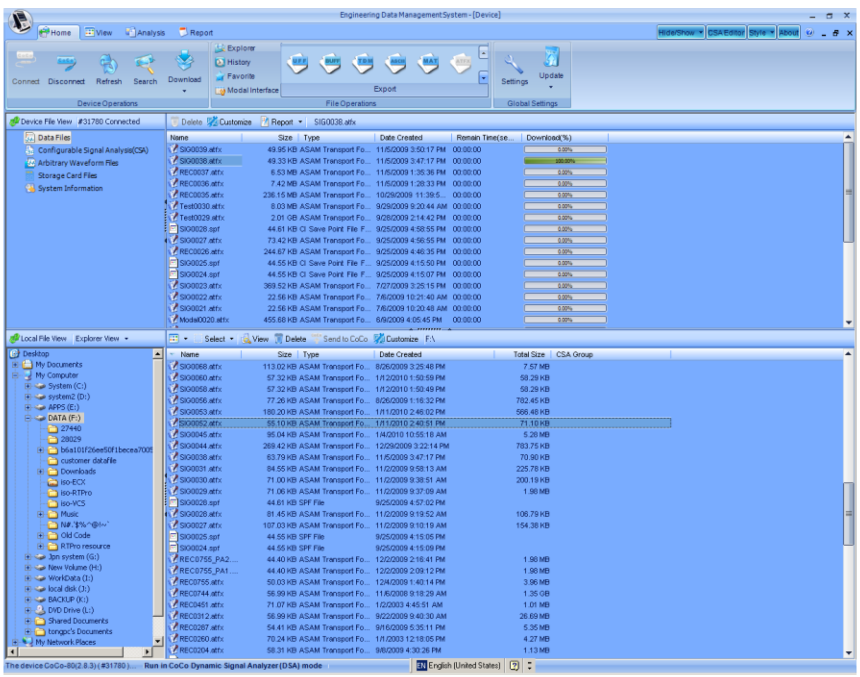
Home Page Ribbon – includes commands that are commonly used for each page.

Device File View – shows files located on the connected CoCo-80 Devices.

Local File View – shows files located on the PC.

Status Bar – displays status messages.

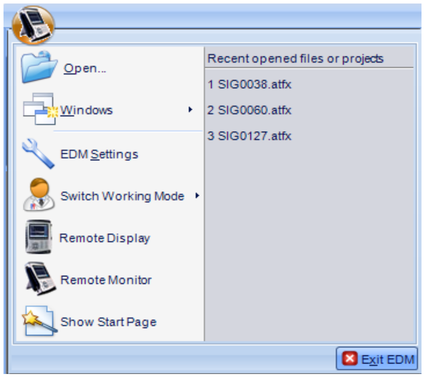
Pop-Up Messages – displays help and status messages.



■ Figure 53. EDM Home Page Display.

Start Button

The Start Button is located in the upper left corner and appears as a graphical image of the CoCo-80. It includes the following commands.



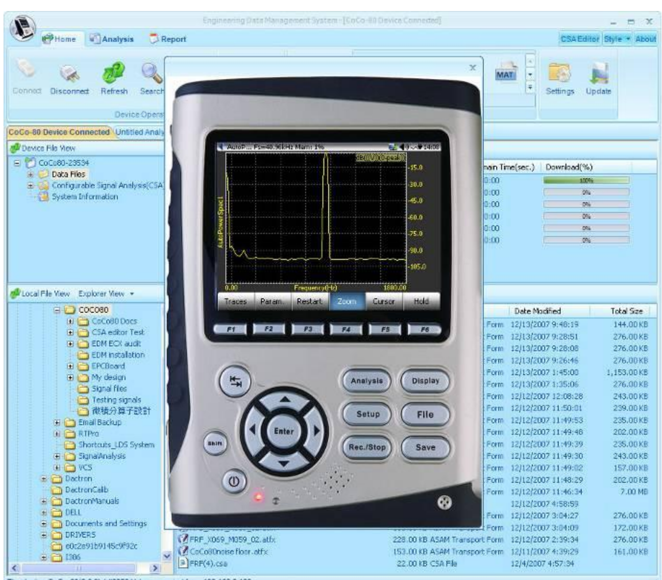
■ Figure 54. Start Button.

Open – presents a dialog to open a data file. This is one of many methods for opening data files.

Windows – commands to manipulate windows such as cascade, tile, minimize, restore, etc. You can also change from the Home to the Analysis or Report Page.

EDM Settings – presents the EDM Settings Dialog. Refer to the Home Page Ribbon section below for more information.

Remote Display – opens a virtual display of the CoCo-80 Device. You can click on the buttons on the PC display to control the remote CoCo-80 Device and a live display is shown on the EDM screen that is identical to the device screen. The live display update rate can be slow depending on the speed of the network connection between the PC and the CoCo-80 Device. This feature is ideal for remote operation of the device from the PC.



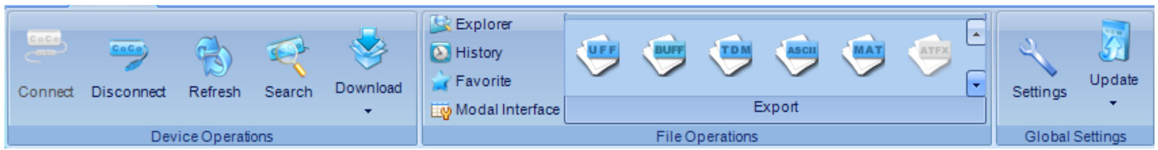
■ Figure 55. Remote Display

Recent Opened Recording Files – lists recently opened files. You can select from this list to open these files again.

Exit EDM – closes the EDM software.

Home Page Ribbon

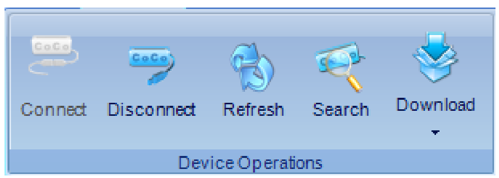
The Home Page Ribbon is displayed when you view the Home Page. It includes commands related to connecting to the CoCo-80, managing files and settings. When you change to the View or Analysis Page the ribbon changes from the Home Page to the View or Analysis Page Ribbon. A detailed description of each item is given below.



- Figure 56. Home Page Ribbon.

Device Operations

Device Operations Ribbon includes commands to connect the PC to CoCo-80 Devices.



- Figure 57. Device Operations Ribbon.

Connect – establishes a connection between the PC and the CoCo-80 Device.

Disconnect – closes the connection between the PC and the CoCo-80 Device.

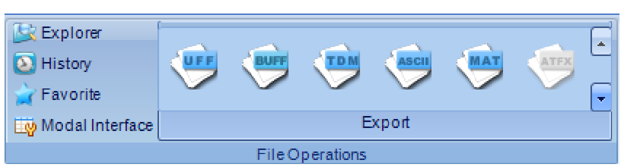
Refresh – updates the list of connected devices and the Device File View. This can be used after a device is connected to force it to appear on the EDM or to update the file view after files have been deleted from the CoCo-80 Device interface.

Search – scans the network for any CoCo-80 Devices that are available and adds the available devices to the Device Found List.

Download – starts the download process for the files on the CoCo-80 Device. This will transfer recorded data and CSA Files from the CoCo-80 to the PC.

File Operations

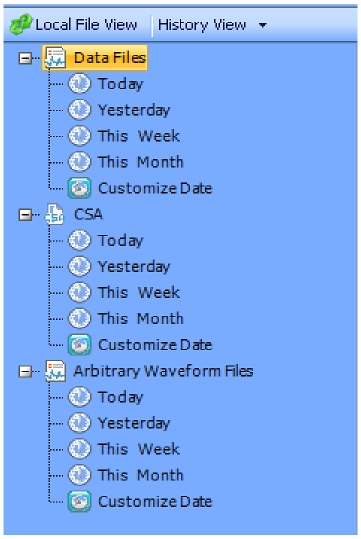
The File Operations Ribbon includes items related to viewing and exporting data files.



- Figure 58. File Operations Ribbon.

Explorer – changes the Local File View to the Explorer Format. This format is similar to Microsoft Windows Explorer format.

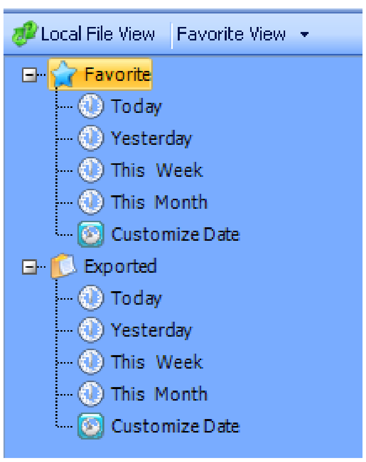
History – changes the Local File View to the History format. This format is organized by time so that you can easily find files based on the time of creation. The display separates the files into Data Files and CSA Files.



- Figure 59. Local File View History View.

Favorite – changes the Local File View to the Favorite Format. Files can be added to the favorite list by right clicking and selecting Add to Favorite on the pop-up menu. Use this feature to easily access files that you use often.

Exported - lists all files that have been exported based on date.



- Figure 60. Local File View - Favorite View.

Export - includes commands that initiate the exportation of files from EDM into one of the export data formats. To export one or more files, highlight the files in the Local File View window and click the Export Format Button. Then specify the location of the files. For more information on data export formats refer to Section 11 Data Export Formats. Data can be exported in the following formats:

UFF – Universal File Format

BUFF – Binary Universal File Format

TDM – National Instruments TDM Format

ASCII – User defined ASCII Text Format

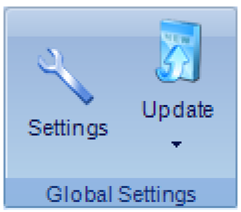
MAT – MATLAB 5 Format

CSV – Excel Format

WAV – Wave File Format

Settings and Update

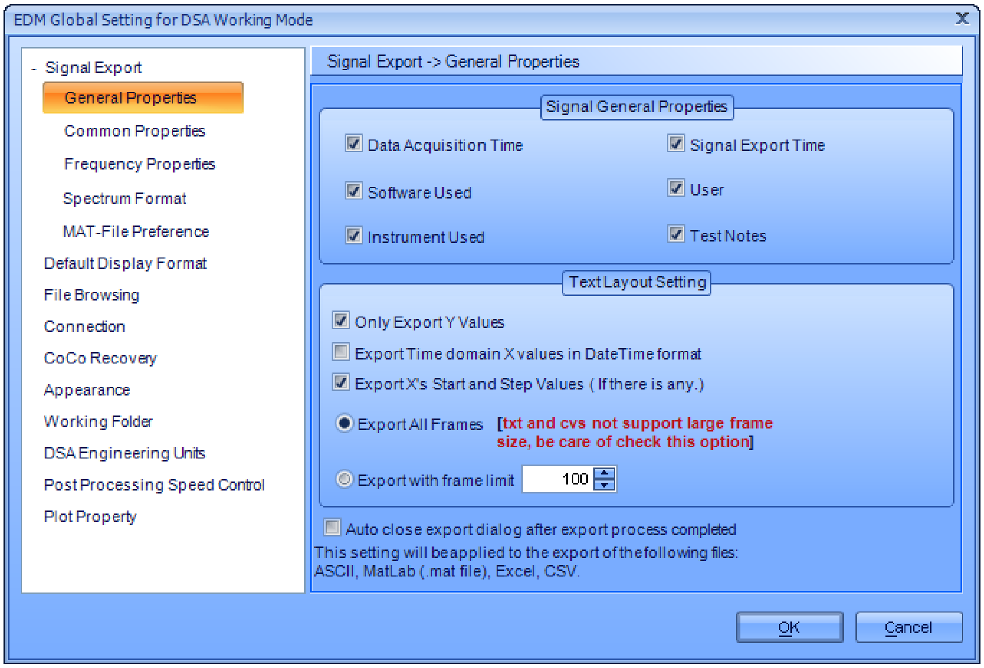
The Settings Button opens the settings dialog from which all the EDM settings can be configured. The Update Button checks the Crystal Instruments Sever over the Internet connection for updates to the EDM software. A detailed description of each is given below.



- Figure 61. Settings and Update Ribbon.

EDM Settings

The EDM Settings dialog allows you to configure various settings for the EDM software. To change the settings click on the item on the left and the display on the right will show the settings. Settings can be turned on and off by adding or removing the check mark by clicking on the box next to each setting.



- Figure 62. EDM Settings Dialog.

Signal Export – includes four sub sections for defining the signal export settings.

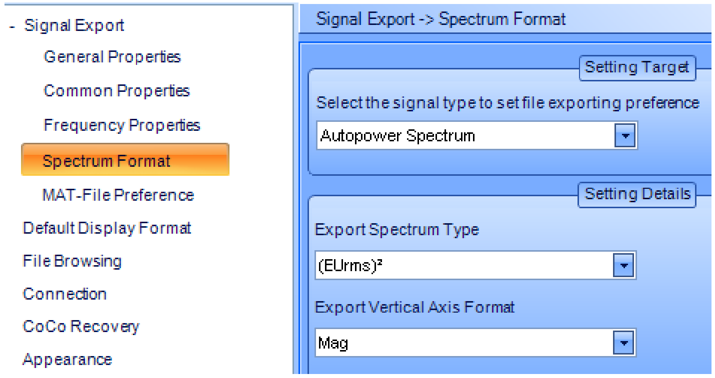
General Properties – defines the signal attributes to be exported including: spectrum format, window type, window correction mode, energy factor, amplitude factor, acquisition/calculation method, amplitude scaling, average mode, lin/exp averaging time constant, number of averages. By default all of these attributes are exported with the data. These settings only apply to ASCII, Mat lab and Excel CSV data export formats. The other formats have a predefined list of attributes that cannot be modified.

The text layout settings are also defined here. The choices are: only export y values or export X's start and step values if any.

Common Properties – defines additional signal attributes to be exported including: signal name, sampling rate, block size, X unit, Y unit and NVH signal type. These settings only apply to ASCII, Mat lab and Excel CSV data export formats.

Frequency Properties – defines additional signal attributes to be exported related only to frequency based signals including: spectrum format, window type, window correction mode, energy factor, amplitude factor, acquisition/calculation method, amplitude scaling, average mode, lin/exp averaging time constant and number of averages. These settings only apply to ASCII, Mat lab and Excel CSV data export formats.

Spectrum Format – defines the default spectrum format when auto-power spectrum are exported:

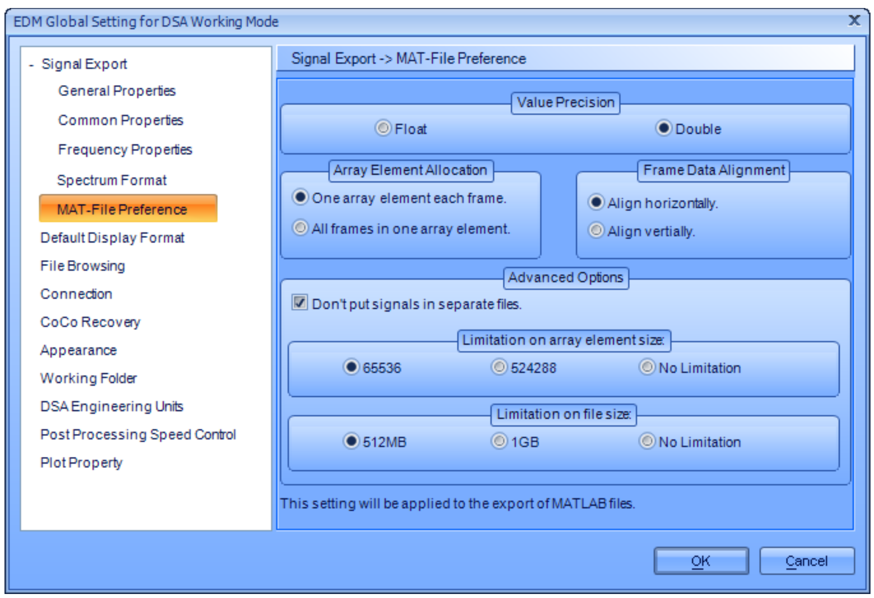


Signal Type: Select one of following signal types to setup the Spectrum Type and Vertical Axis Format for data exporting: Auto Power Spectrum, coherence, complex spectra, cross power spectrum and frequency response spectrum.

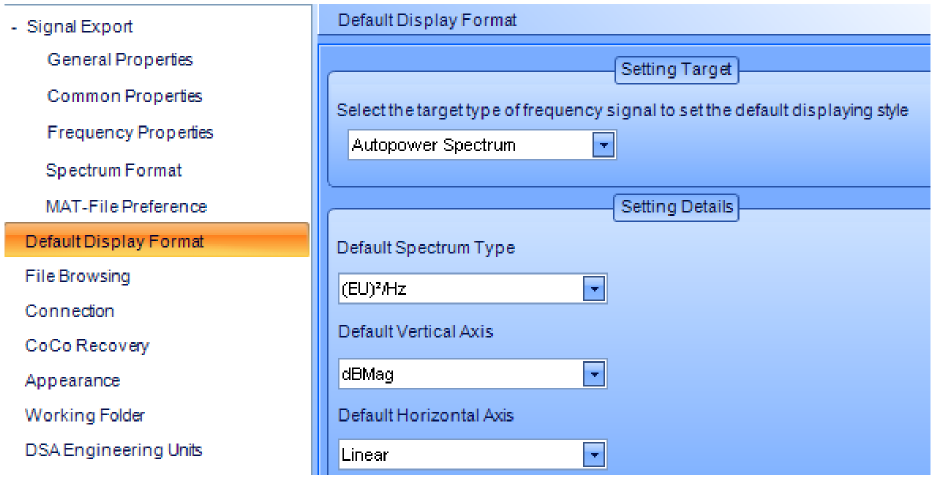
Spectrum Type: $(EU)^2$/Hz, $(EU)^2$s/Hz, $(EUrms)^2$, EUpeak, EUrms

Export Vertical Axis Format: Mag or dBMag

MAT-File Preference– defines the default Matlab file format to export:



Default Display Format – defines the default display format of the frequency spectrum. These settings will be used when a new pane is created in the EDM.



Only frequency signal has an issue of display format. The vertical and horizontal axis scaling vary. For auto-power spectrum, there is also an issue of Spectrum Type. To make appropriate setup, first please select the type of signal, then the select each item in the Setting Details category.

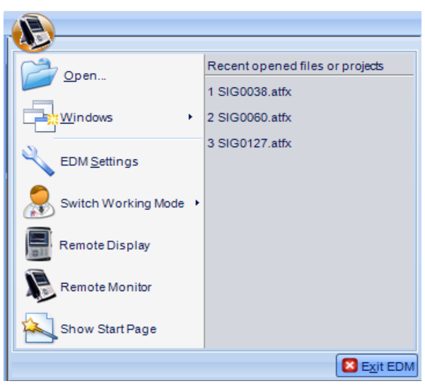
File Browsing – defines the settings for file filtering.

Only show EDM files - hides all files not associated with the EDM software.

Show all files – shows all files in the folder.

Show custom file types - allows you to specify a file suffix.

Max recent recording files to display over start menu - limits the number of files shown in the start menu shown below in Figure 63.



- Figure 63. Recently opened files listed in the Start Menu

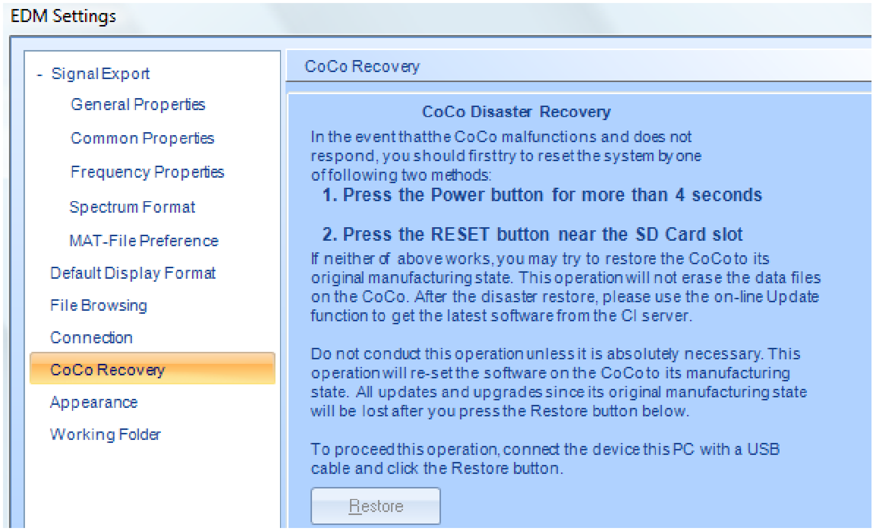
Connection – defines how EDM manages the connection to CoCo-80 Devices. The following options can be turned on or off by checking the box.

Show the dialog box for devices search when application starts up

Always connect EDM to the CoCo-80 Device when it is detected.

Coco-80 Recovery – In the event that the CoCo-80 malfunctions and does not respond, you should first try to reset the system with one of following two methods:

1. Press the Power Button for more than 4 seconds
2. Press the RESET Button near the SD Card slot



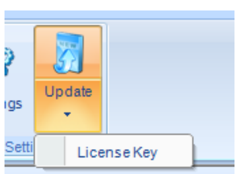
■ Figure 64. CoCo-80 Recovery.

If neither of above works, you may try to restore the CoCo-80 to its original manufacturing state by connecting the device to the PC with a USB cable and clicking on the Restore Button. This operation will not erase the data files on the CoCo-80. After the disaster restore process is complete, please use the on-line Update function to get the latest software from the CI server.

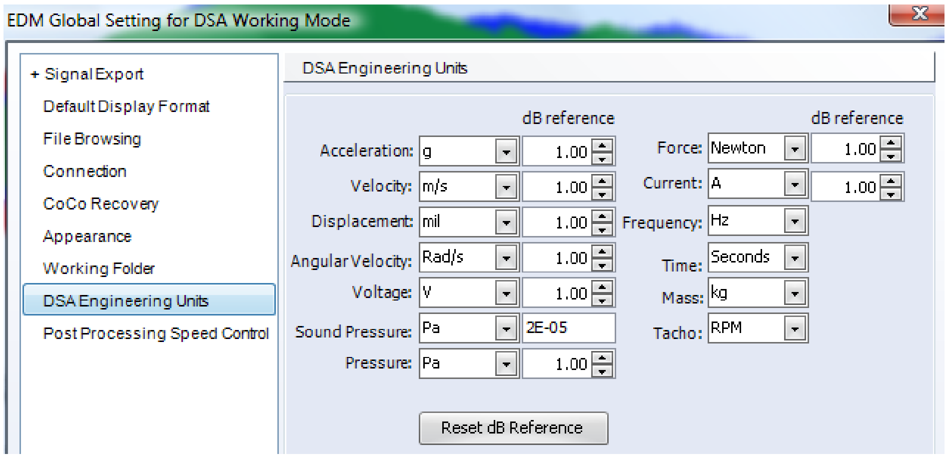
Do not conduct this operation unless it is absolutely necessary. This operation will re-set the software on the CoCo-80 to its manufacturing state. All updates and upgrades since its original manufacturing state will be lost after you press the Restore Button.

Appearance – changes the color scheme of the EDM software.

Working Folder – sets the default working folders for program and data storage.

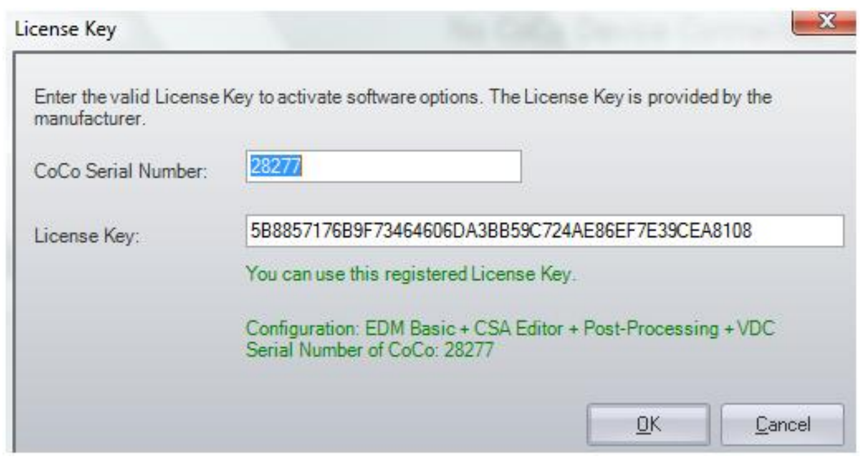


DSA Engineering Units – sets the engineering units. These are global settings that are applied in the scope of this EDM software. The engineering unit display setting does not affect the actual values of the signals. They can also be set differently to the settings when the data is acquired. For example an acceleration signal can be acquired in “g” on CoCo while displayed in “m/s*s “ here.

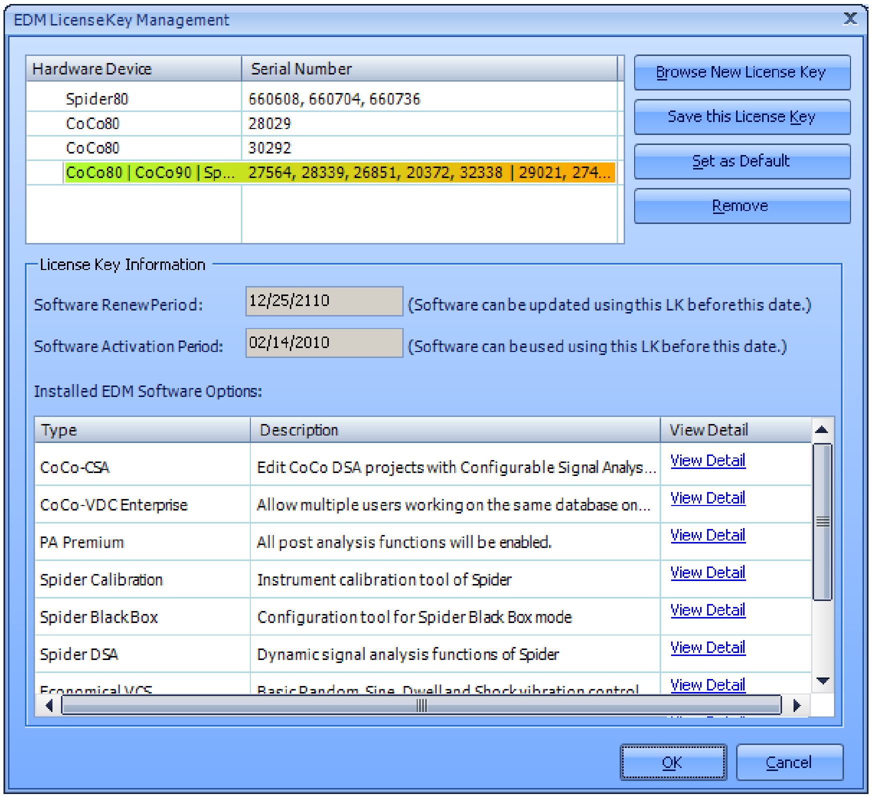


Update Button

The Update function connects EDM to the Crystal Instruments sever via the Internet connection on the PC and checks if there is an update to the EDM available. If an update is available and if the software license is current then you are given the choice to download and install the new files. This feature can be used to maintain the software with the latest version. Contact the manufacturer if the software license is not current.

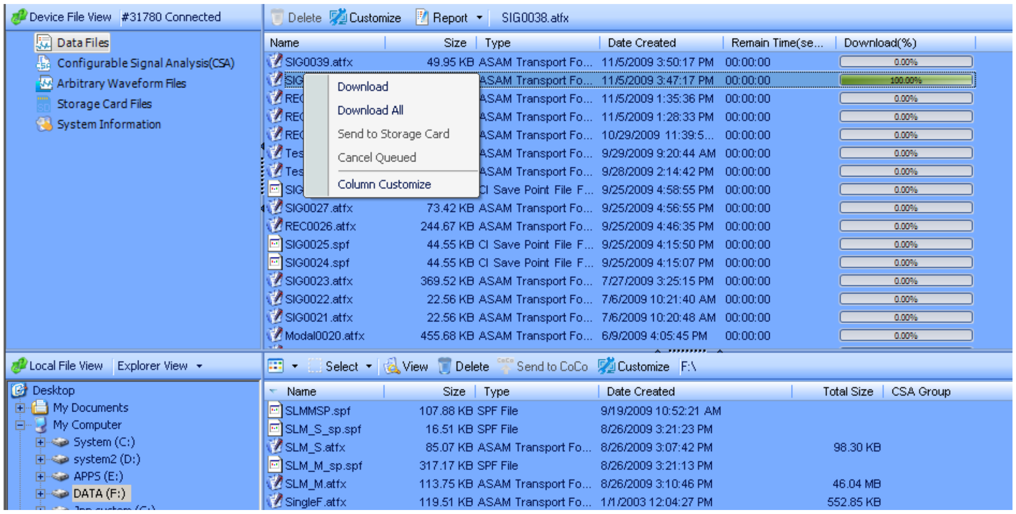


You can also enter the new License Key under the Update button to enable more functions. License Key is provided by the manufacturer.



Device File View

The Device File View displays files located on the CoCo-80 Device. The View is divided into two parts. The left Pane shows the device name and folders that contain the data files, CSA Files and system files. The right Pane displays the contents of each folder when you select one of the data folders on the left. The data can be sorted by any column by clicking on the column heading. Data files can be downloaded by selecting one or more files and then dragging them from the Device File View to the Local File View. When a file is downloaded the Download % shows the download status.



■ Figure 65. Device File View with popup menu.

From the Device File View you can right click on any file to open a popup menu that includes the following commands.

Download – copies the selected file from the CoCo-80 Device to the PC.

Download All – copies all files from the CoCo-80 Device to the PC.

Column Customize – allows you to modify the format of the Device File View List by adding, removing or rearranging the columns.

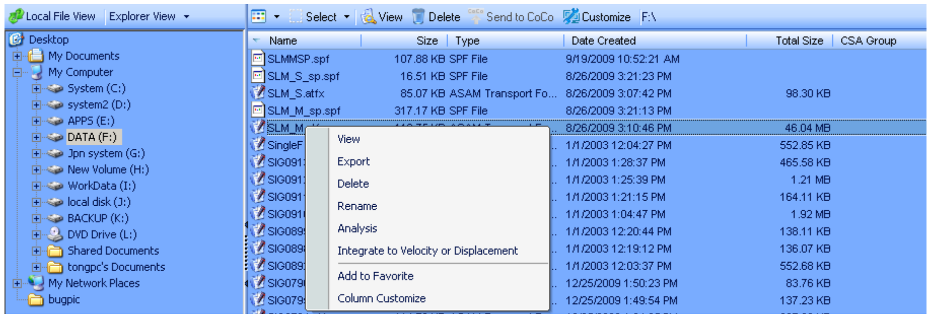
The selected signal name is displayed on the toolbar. The Delete and Customize Buttons perform the same functions as the popup menu. The Device File View includes the following buttons.

■ Figure 66. Device File View Toolbar.

Report – allows you to generate a Device Report that includes information about the configuration of the CoCo-80 Device. **Options** opens the Word Report Options dialog to configure the contents of all reports. For more information on the Report function refer to Section 8 Report Page.

Local File View

The Local File View displays files located on the local PC hard drive. The View is divided into two parts. The left Pane shows the local PC storage devices. The right Pane displays the contents of the current folder. The data can be sorted by any column by clicking on the column heading. Data can be managed by right clicking to open the popup menu or using the toolbar buttons.



■ Figure 67. Local File View with popup menu.

The Local File View popup menu can be opened by right clicking in the right Pane. This menu includes the following commands.

View – opens the selected file in the Analysis Page. Refer to Section 6 for more information on the Analysis Page.

Export – saves the selected file or files in any of the export data formats.

Delete – removes the selected file(s) from the storage device.

Rename –changes the name of the file.

Add to Favorites – adds an attribute to the file so that when the View is changed to the Favorite View the file will be visible. This feature can be used to quickly find files that are used often.

Column Customize – adds, removes and rearranges the columns in the Local File View Pane.

The Local File View can also be operated with the toolbar buttons which include the following commands. The current folder path is displayed at the right of the toolbar.

■ Figure 68. Local File View Toolbar.

Explorer/History/Favorites View – changes the organization of the local File View. Refer to Device File View for a description.

Views - changes the display format of the local files from Details, List or Large Icon View.

Select –includes options to select all, inverse selection, or clear selection. These options can be used to select a specific set of files from a large list.

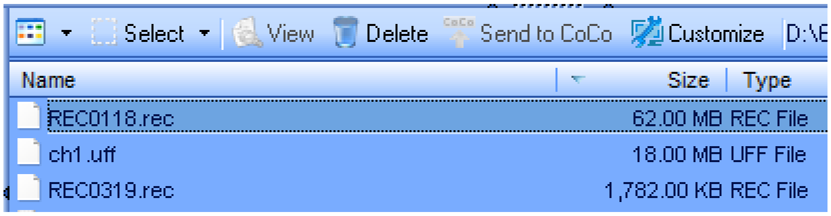
The View, Delete and Customize Buttons perform the same operations as the commands in the popup menu.

Reading the Raw Data from SD Card

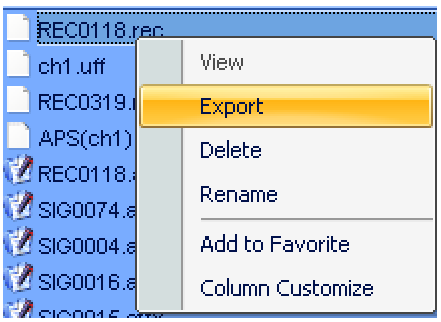
If the data files are recorded into the internal flash memory, the EDM software will read them and save to PC in ASAM-ODS data format. If the data files are recorded to the SD memory card, they are still in the raw data format. The user can use the EDM software to convert the files on the SD card into the ASAM-ODS format. You will need a SD card reader on your PC to access the SD memory card. The steps are:

Optionally, first copy the REC raw data files from the SD memory card into your local folder.

Select the source REC file within EDM:

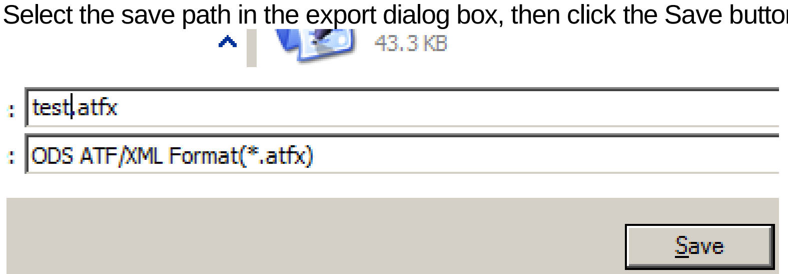


Click the context menu or ribbon button to start the export.

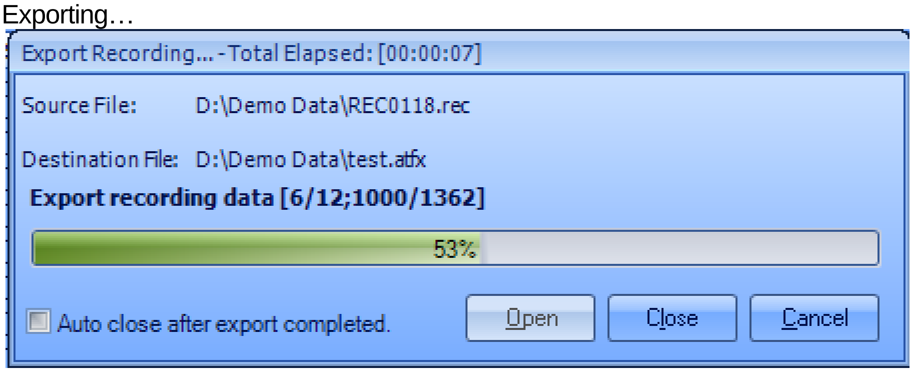


or

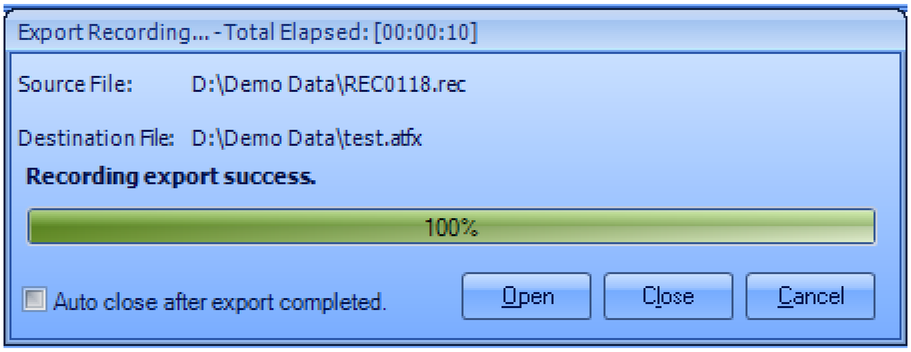

Select the save path in the export dialog box, then click the Save button to start the export.



Exporting…



Export Completed.


Export Recording... - Total Elapsed: [00:00:10]
Source File: D:\Demo Data\REC0118.rec
Destination File: D:\Demo Data\test.atfx
Recording export success.
100%
Auto close after export completed.
Open
Close
Cancel

6. View Page

This section describes the View Page in detail. The View Page is used for displaying the recorded time streams or saved signal files, or displaying the post-processing data. It is also the main page being displayed when post-processing is in progress.

The page includes the following components.

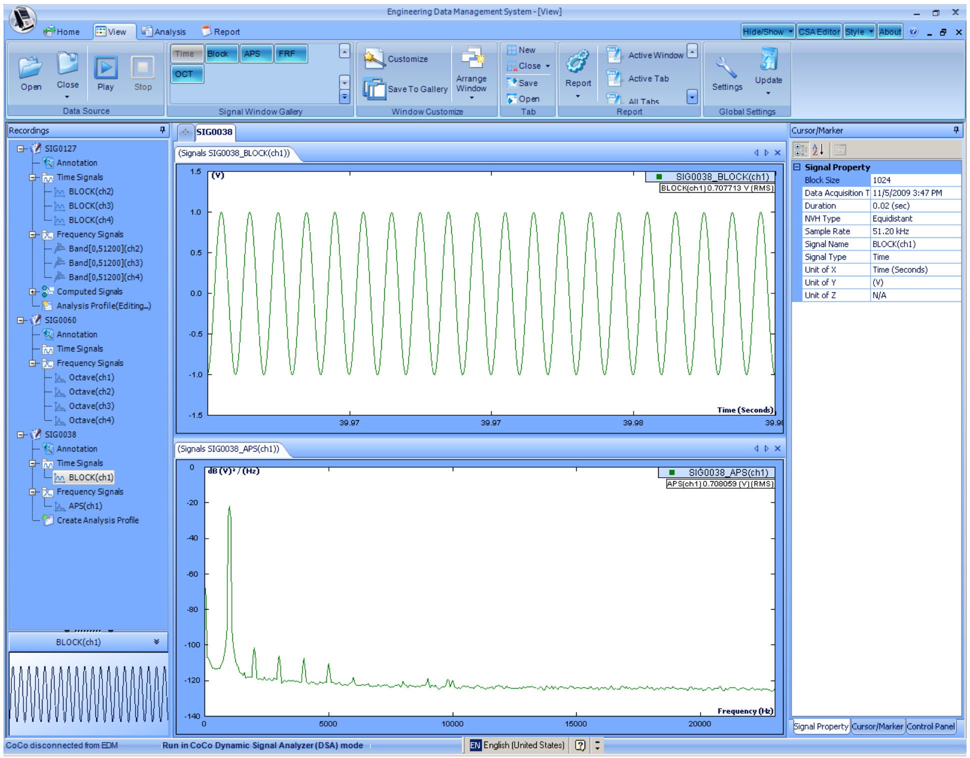
View Ribbon – includes commands that are performed on the View Page.

Recordings List – displays the source files that have been opened from the Home Page.

Views – The View area is where source files are displayed. There are several types of Views including two vertical Panes, two horizontal Panes, 2x2 Panes or Table View. Refer to Views below for a complete description. You can define any number of Views in the Analysis Page. Only one View can be displayed at a time.

Panes – A Pane is an axis that includes one or more graphs. One, two or four Panes can be displayed in a View.

Signal Property – displays the attributes included in the signal file. The properties can be viewed, and sorted by category or alphabetically. They cannot be edited from this display.

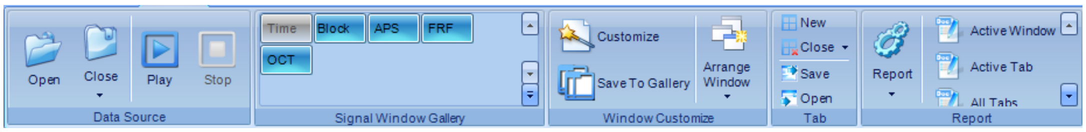


■ Figure 69. Analysis Page.

Cursor/Marker– displays the values of cursors or markers
Control Panel – Control panel for post processing

View Page Ribbon

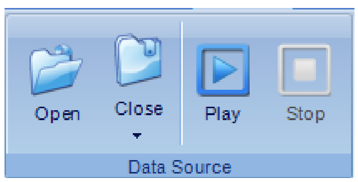
The View Page Ribbon is displayed when you view the View Page. It includes commands related to displaying formatting and analyzing recorded data. When you change from the Home Page to the Analysis Page then the ribbon changes from the Home Page to the View Page Ribbon. A detailed description of each item is given below.



■ Figure 70. View Page Ribbon.

Data Source Ribbon

Data Source Ribbon includes commands to control access to data sources.



■ Figure 71. Data Source Ribbon.

Open – presents a dialog to select a data source file from the PC. After a file is selected it is added to the Recordings List and can then be added to a Pane.

Close – closes a data source file and removes it from the Recordings List. You can close the selected or all files at once.

Play – plays a time recording in oscilloscope mode. This display shows the time waveform as it would appear in an oscilloscope with the frame rate updated as fast as possible.

Stop – ends the playback of the time recording. If the Run Button is pressed again then the playback starts from the beginning of the file.

Word Report – creates a report for the active Pane or for the active View.

Recordings Pane

The Recordings Pane shows the signals that have been selected for viewing in the Home Page. To add a new signal to the Recordings Pane select the signal in the Home Page and right click to open the popup menu and select View or click on the View Button on the toolbar. Any number of signals can be added to the Recordings Pane. Signals must be added to the Recordings Pane before they can be added to a Pane.

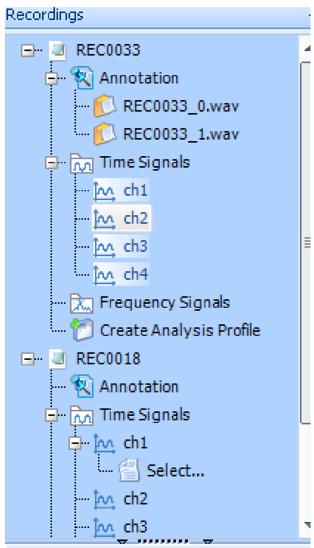
The Recordings Pane shows all signals available to view. Each signal includes the following:

Signal Name – the name associated with the signal file.

General Information – file attribute information such as software version generated the data, format, user name, test note, etc.

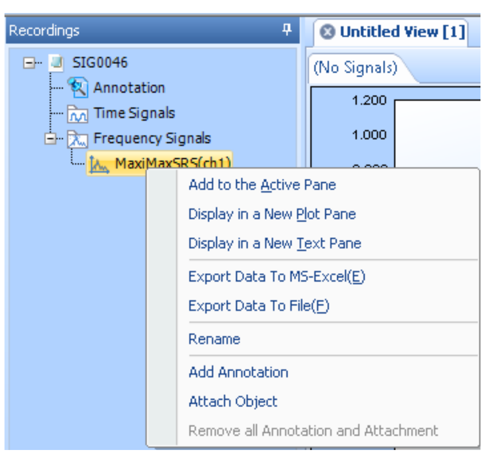
Signals List – includes an item for both time and frequency data even when only one of the two data types is included in the signal. An item is listed in either time or frequency that represents the data values contained in the signal file. Recordings can contain one or more signals of similar or mixed data type. A preview of the signal can be viewed at the bottom of the Recordings Pane by clicking on the signal name.

Items in the Recordings Pane can be shown by clicking the "+" icon to expand the item and view the subsections or hidden by clicking on the "–" icon. These tools can be used to view all details of recordings or to reduce the Recordings Pane to view many recordings at one time.



■ Figure 72. Recording Pane.

Signals can be added to a Pane by right clicking on the signal name to open the popup menu.



■ Figure 73. Recordings Pane Pop-Up Menu.

Add To Active Pane – adds the selected signal to the active Pane. Note that only signals of the same type can be added to the same Pane. For example a time signal cannot be added to a Pane that already contains frequency spectra.

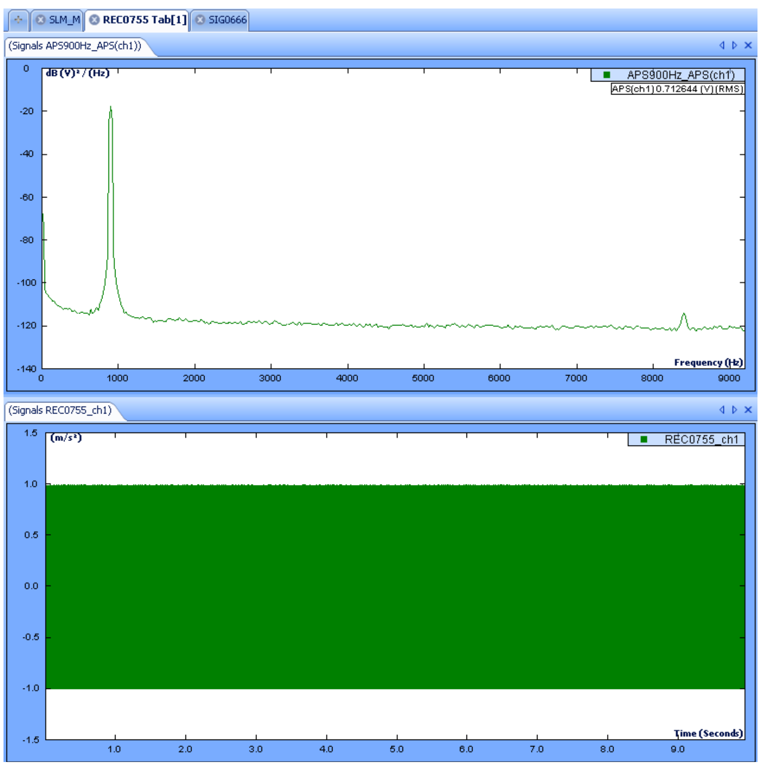
Add to New Pane – creates a new Pane in the active View and adds the selected signal.

Report to Excel – creates an Excel report in the Report Page showing the contents of the signal file.

Signals can also be added to Panes by selecting the signal in the Recordings Pane and dragging them into an existing Pane.

Working with Views

A View is a display that contains one or more Panes. A Pane is an axis where the signals are graphed. You can define any number of Views in the View Page. You can only display one View at a time. Multiple Views lets you organize a large number of Panes on one screen and then easily switch from one to another. Figure 74 shows the Analysis Page with three Views created. The active View contains a Horizontal Pane with a frequency spectrum in the top Pane and a time wave form in the bottom Pane.

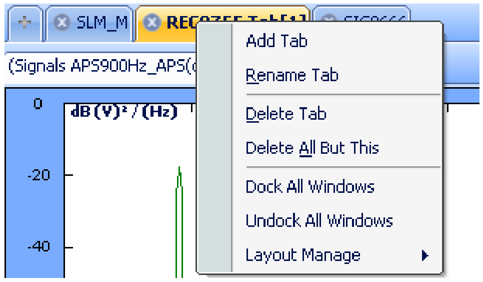


■ Figure 74. Analysis Page with three Views and a Horizontal Pane in the active View.

To change from one View to another, click on the View Name Tab at the top of the screen.

Tab Popup Menu

The View Popup Menu can be opened by right clicking on the View names tab.



■ Figure 75. View Popup Menu.

Add Tab – adds a new View.

Rename Tab – changes the View Name from the default. This can be used to customize the View to give them more meaning and organize the display.

Delete Tab – removes the active View from the display. A View can also be deleted by clicking on the "X" icon at the top of the View.

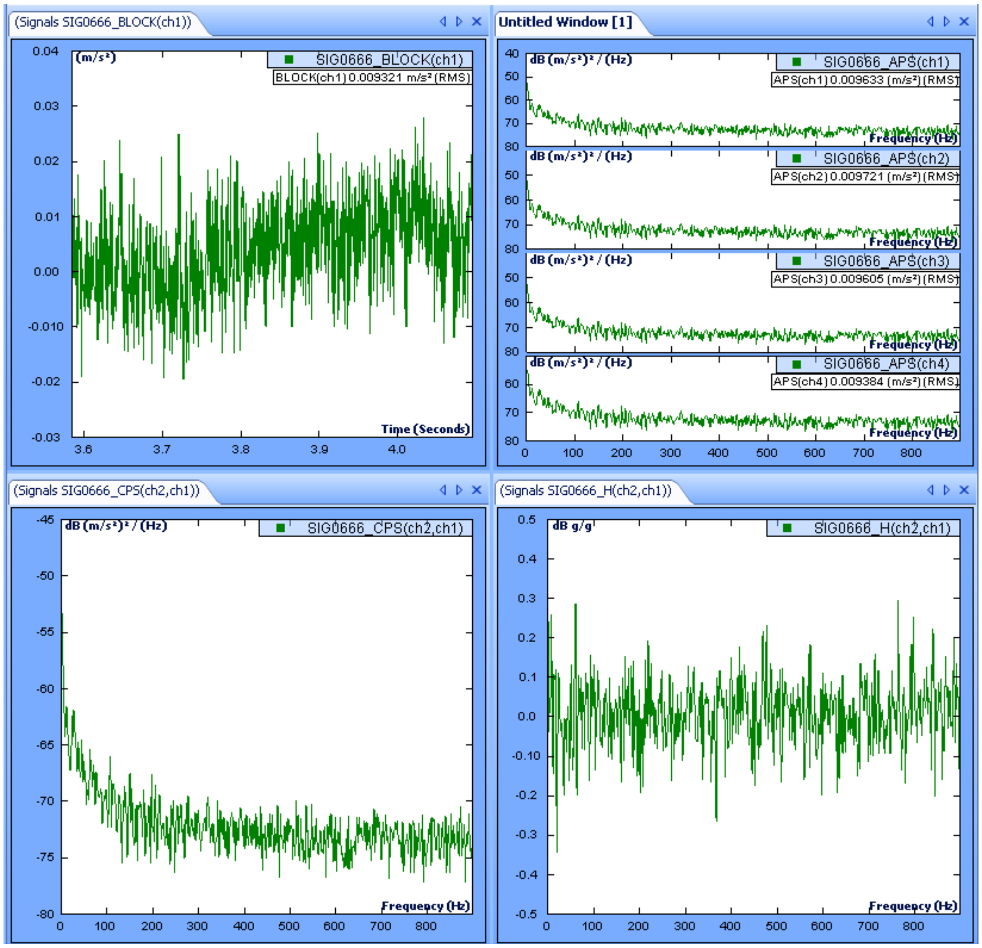
Delete All But This – removes all Views except the active View.

Working with Windows

A Window is an axis where the signals are graphed. Windows are contained in Views. A View can contain several combinations of Windows including a single Window, two Horizontal or Vertical, 2X2, or a Table of data. Refer to the View Ribbon section above for a description of each.

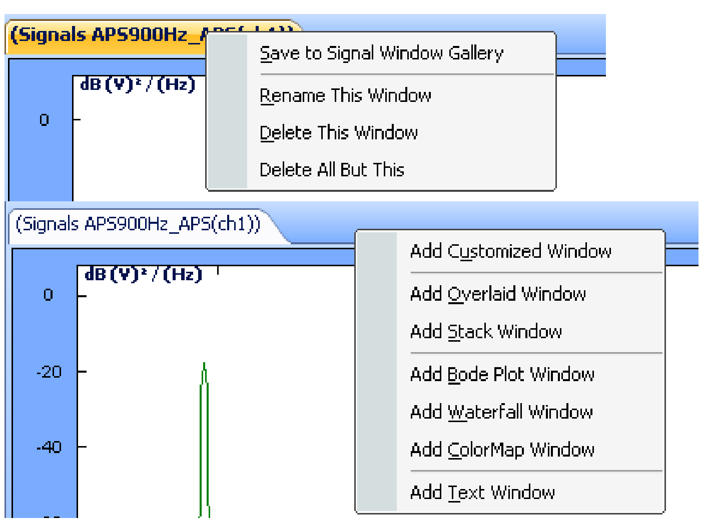
An empty Pane can be added to a View using the Add Analysis Ribbon Button, or by right clicking on the View Name Tab and clicking on Add Pane from the popup menu.

A View can contain any number of Panes. Additional Panes are tiled and stacked on the View as shown in Figure 76. To display each Pane, click on the Pane Name Tab. If several Panes are added to a View then all the Name Tabs may not be visible but you can use the arrow buttons in the upper right corner of the View to move from one Pane to the next. Use the 'X' icon to delete the active Pane.



■ Figure 76. A View with several Panes stacked together.

The Window Popup Menu can be opened by right clicking on any of the Pane Name Tabs. Another Window Popup Menu can be opened by right clicking on any of tabs but Pane Name Tabs.



■ Figure 77. Pane Popup Menu.

Save to Signal Window Gallery –.

Rename This Window – changes the name of the active Pane. This feature helps to customize the display.

Delete This Window – removes the active Pane from the View.

Delete All But This – removes all Panes from the active View except the active Pane.

Add Customized Window –.

Add Overlaid Window –.

Add Stack Window –.

Add Bode Plot Window –.

Add Waterfall Window –.

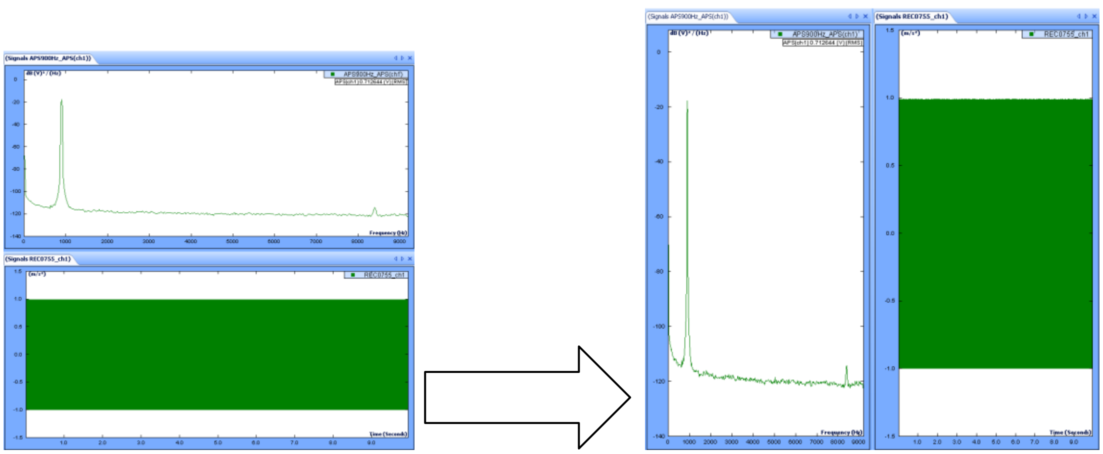
Add ColorMap Window –.

Add Text Window –.

You can rearrange the layout of the different Views by clicking on the Name Tab of any View and dragging it to a new location. The View location icon is displayed. Then move the cursor over one of the icon squares.

For example you can change a two Horizontal View into a Vertical View or change two stacked Panes into two Horizontal Panes.

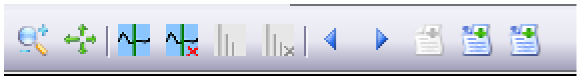
Figure 78 shows the result of clicking on the bottom View Name Tab and dragging it to the right corner of the View location icon.



- Figure 78. Panes can be rearranged by clicking and dragging a View Name Tab to a new location.

Pane Quick Access Toolbar

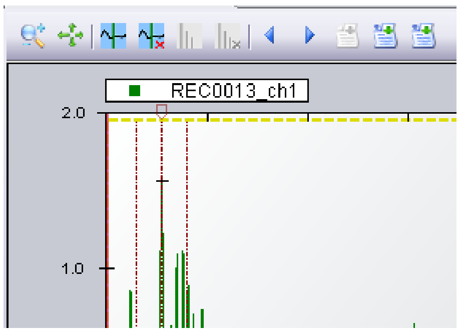
When a pane is set focus, the quick access toolbar will appear. All the commands shown in the Quick Access Toolbar can be accessed through Pane Popup Menu (see below). But with Quick Access Toolbar, the most often used actions can be taken by one-click.



The Cursors icons let you add cursors to the active Pane. Cursors are the best method to measure features of a waveform such as the peak value or the time between two events.

Vertical Cursor – adds a vertical cursor to the active Pane.

Search for Peak– when any vertical cursor is enabled, press the upper arrow key on your PC, it will search for the peak nearby quickly.



Remove – removes all horizontal, vertical or all cursors from the active Pane.

A cursor can be moved by clicking and dragging it with the mouse. The arrow keys on the PC keyboard move the cursors together.

When a cursor is added to a Pane the Cursor View is displayed. It shows the X and Y values and units for all cursors.

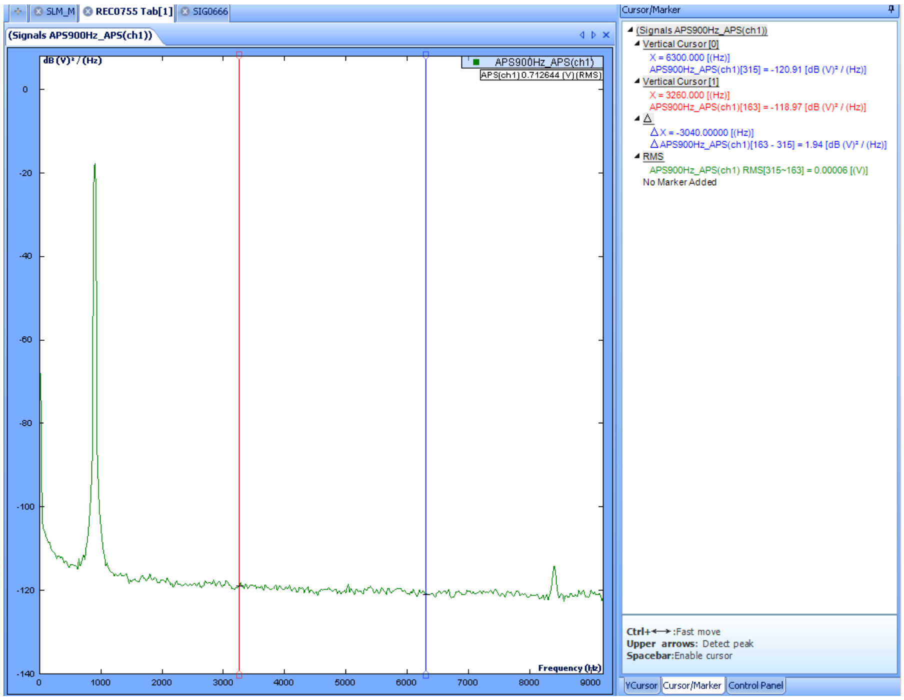
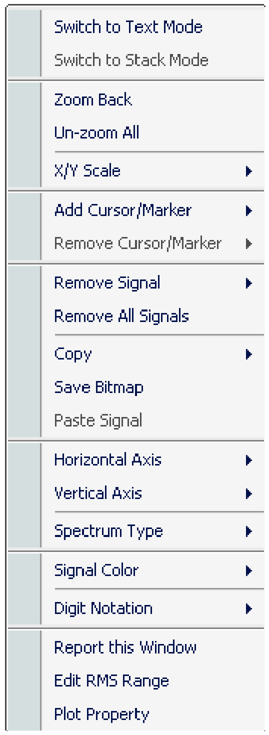


Figure 79. Pane with two vertical cursors

Pane Popup Menu

The Pane Popup Menu can be accessed by right clicking on a Pane. This menu includes commands to change the contents and format of a Pane. Many of these menu items can also be accessed on the Analysis Ribbon.



■ Figure 80. Pane Popup Menu. It's menu items are dynamically shown according to the signal type.

Zoom and Scale

Zoom Back – zoom out to the previous zoom ratio.

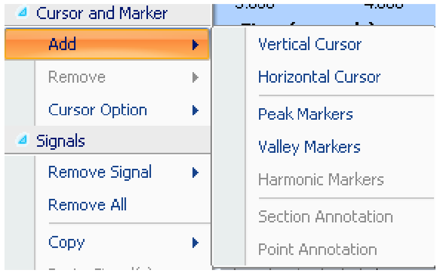
Un-Zoom All – zoom out to the full scale.

Scale

Auto Range – automatically sets the scale for the X, Y or both X and Y axis.

Restore Scale -

Cursor and Marker



Add – adds a vertical or horizontal cursor to the active Pane.

Remove – removes all horizontal, vertical or both cursors from the active Pane.

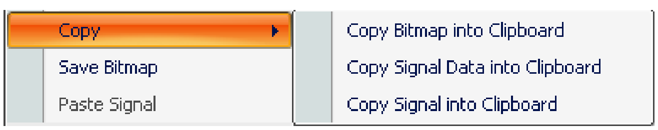
Signals

Remove Signal – removes signals from the active Pane.

Remove All – removes all signals from the active Pane.

Copy Signal(s) – copies the signal(s) from the active Pane into the clipboard. You can then use Paste Signal(s) to paste the signals into a different Pane.

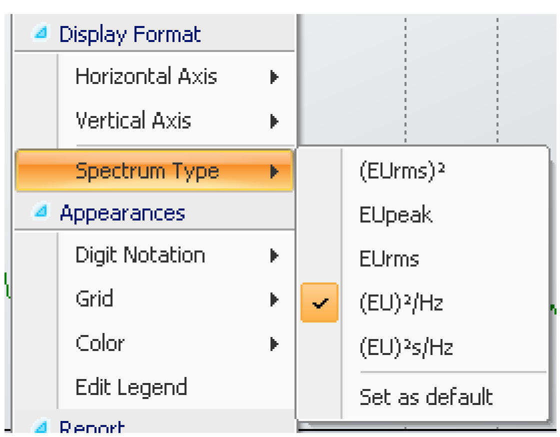
Paste Signal(s) – pastes copied signals into a new Pane.



Display Format

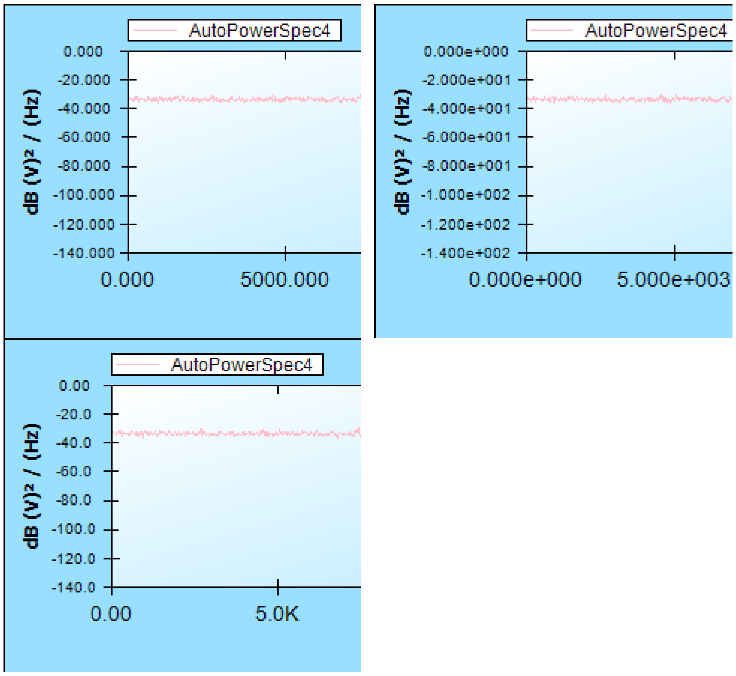
Vertical Axis – defines the vertical axis as dB or Mag scaling. This item is only available for frequency based signals.

Spectrum Type – defines the scaling for spectrum signals as $(EU)^2$/Hz, $(EU)^2$s/Hz, $(EUrms)^2$, EUpeak, or EUrms. The Spectrum Type selection only applies to the auto-spectrum signals. For time domain or any other frequency domain signals, there is no such a notion of Spectrum Type.



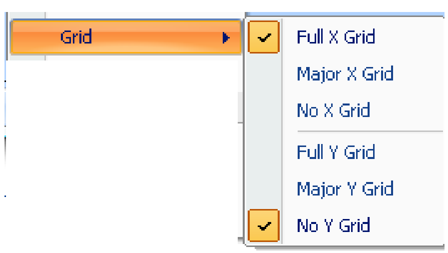
Appearances

Digit Notation – specifies the format for axis labels as floating point, scientific or engineering format.



■ Figure 81. Digit Notation with floating point (left), scientific (middle) and engineering format (right).

Grid – controls the display of a grid in the active Pane. A grid can include a vertical, horizontal grid or both or be turned off.



Signal Color – defines the color scheme or signal color used in the active Pane. A Pane can either be a solid color a fade from one color to another.

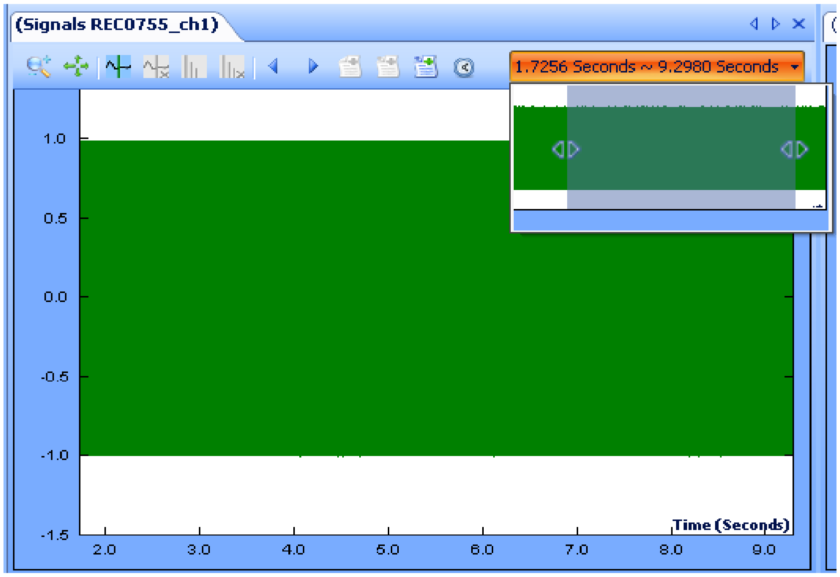
Report to Word

Report the Active Pane –generates a report including the contents of the active Pane. The report will include only one Pane.

Report the Active View – generates a report including the contents of the active View. The report will include all the Panes in the active View.

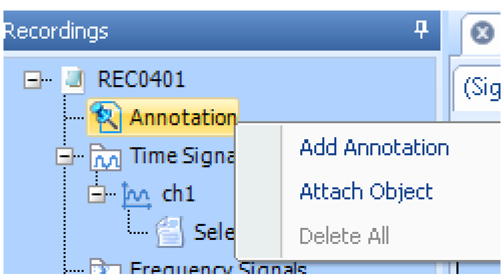
Global View

When a time waveform is added to a pane the Global View is added to the upper right as shown in Figure 82. This view lets you control the time scale by clicking and dragging the arrows on the left and right sides of the gray bar that highlights the portion of the waveform that will be displayed in the main pane. This feature allows you to efficiently view long waveforms. The Global View can be hidden or revealed by clicking on the up or down arrow in the upper right corner of the view.

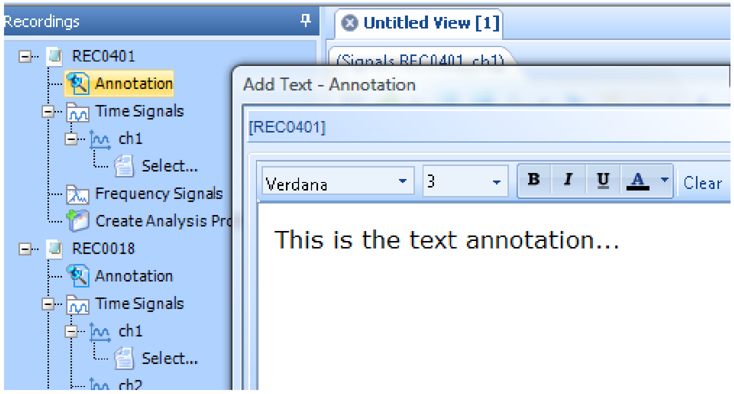


■ Figure 82. Global View for Time Waveforms.

Add Annotation or Attach Objects

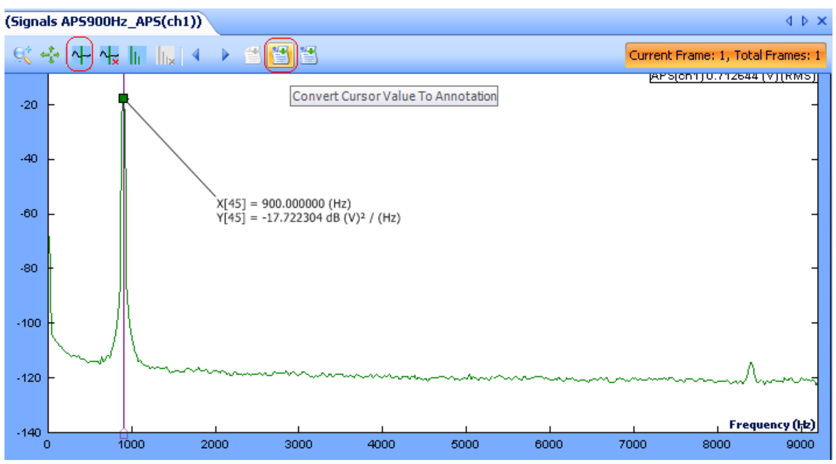


To add annotation or attach objects to a signal, right click on the Annotation icon below the signal name. Then select appropriate menu item:



Annotation or objects attached to the signals can be opened, edited or removed.

Convert Cursor Value to Annotation

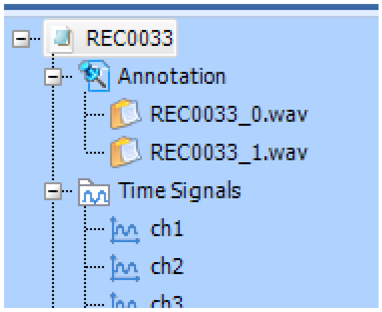


To convert the cursor values into text annotation:

1. Enable a cursor and drag it to the interested point
2. Click on the Convert Cursor Value To Annotation icon
3. The Cursor values will be pasted into the annotation box

Annotation (Voice)

Voice annotation recorded by the CoCo will be associated to the recorded file as wav files.

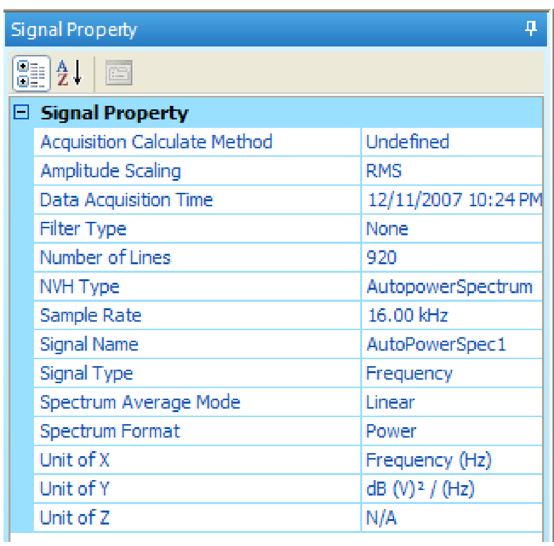


These wave files can be played back in EDM. You can simply click on them and the Windows Media Player will be prompted to play these files.

Signal Property View

The Signal Property View shows all file attribute information associated with the signal selected in the Recordings List. To view the signal properties for a different signal, select the signal in the Recordings List. The attributes are for display only and cannot be modified. A file attribute is defined within the CSA and on the CoCo-80 Device when the signal is recorded. The signal display properties can be changed within a Pane but the properties of the signal file are not affected within the signal file. Note that when a signal is exported the Export Settings are used to specify the signal attributes so that you can modify the attributes by changing the Export Settings and exporting the signal.

The Signal Property View can be arranged by category or alphabetically by clicking on the icons at the top of the View. The View can be hidden or revealed by clicking on the pushpin icon in the upper right corner.



■ Figure 83. Signal Property Display.

Curve Fit and Curve Synthesis

Very often the frequency response measurements made in the FFT and Swept Sine measurement groups need to be compared to a theoretical model of the device's behavior. This comparison can be achieved by so called Curve Fitting and Synthesis. In curve fitting, the analyzer extracts the best fit parameters of a linear frequency response function from a measured frequency response function. In the Curve Synthesis, the EDM transforms a set of frequency response parameters into a frequency response measurement which can be compared with the measured data.

Both curve fitting and curve synthesis use the Curve Tables. The curve tables allow entry and editing of frequency response parameters in one of two formats: polynomial and pole-zero. Once parameters have been entered into the curve tables the corresponding frequency response function can be synthesized into a trace for comparison with measured data.

Polynomial

In this format, the curve table represents a frequency response function as the ratio of two polynomials in the frequency domain.

$$H(jw)=\frac{a_0+a_1(jw)+a_2(jw)^2...+a_M(jw)^M}{b_0+b_1(jw)+b_2(jw)^2...+b_{N-1}(jw)^{N-1}+(jw)^N}=\frac{\sum_{k=0}^{M}a_k(jw)^k}{\sum_{k=0}^{N-1}b_k(jw)^k+(jw)^N}=\frac{C(jw)}{D(jw)}$$

The curve tables allow entry of both the numerator and denominator coefficients as well as the order of the numerator and denominator polynomials. The curve table also contains a constant gain factor which multiplies the polynomials.

Pole-Zero

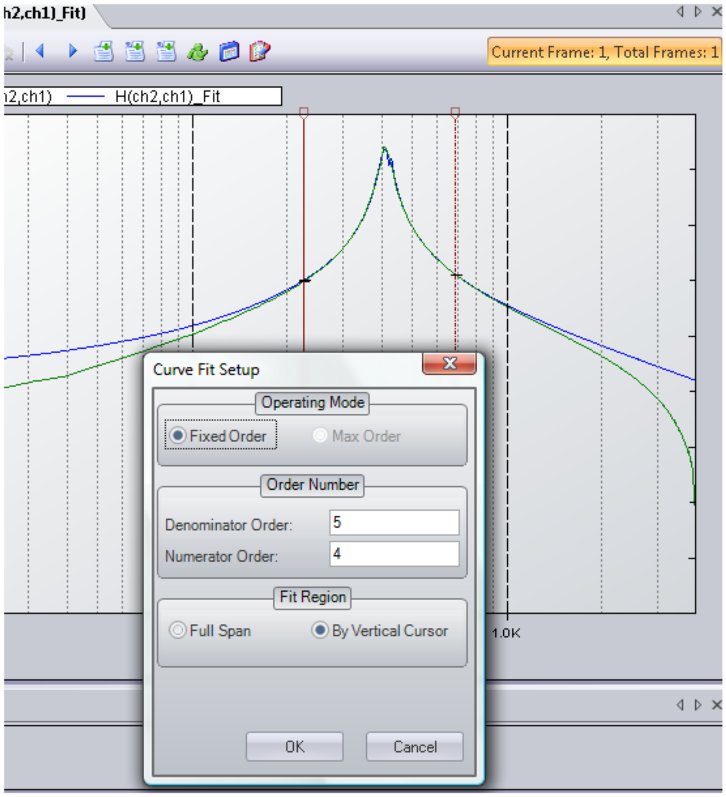
In the pole-zero format, the numerator and denominator polynomials are factored so that the frequency response curve is described by the ratio of the products of the poles and zeros: To ensure a real impulse response, all complex poles and zeros only occur in complex conjugate pairs.

$$H(s)=g\frac{(s-z_0)(s-z_1)...(s-z_M)}{(s-p_0)(s-p_1)...(s-p_N)}$$

The g in the formula is called system gain.

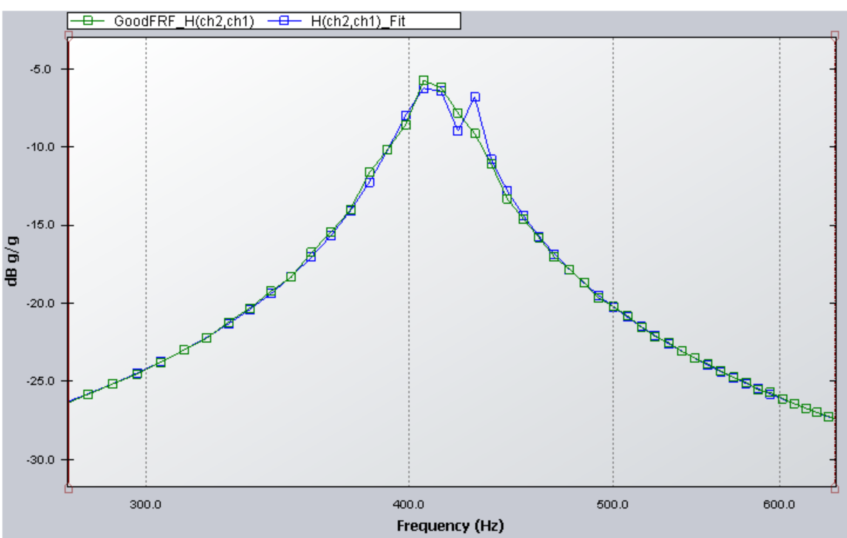
Curve Fit Process

The curve fit is conducted in an active pane that loads with FRF (frequency response function) signal. Press the Curve Fit Setup icon and set the following parameters:

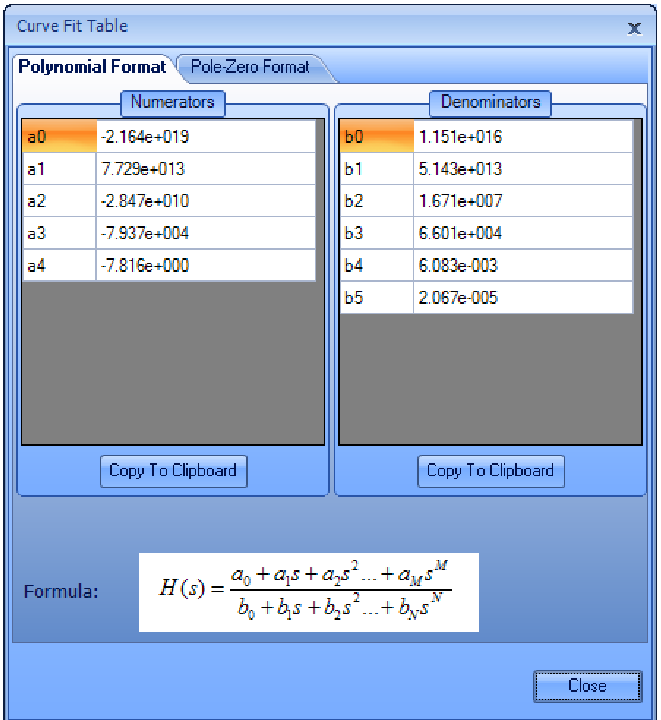


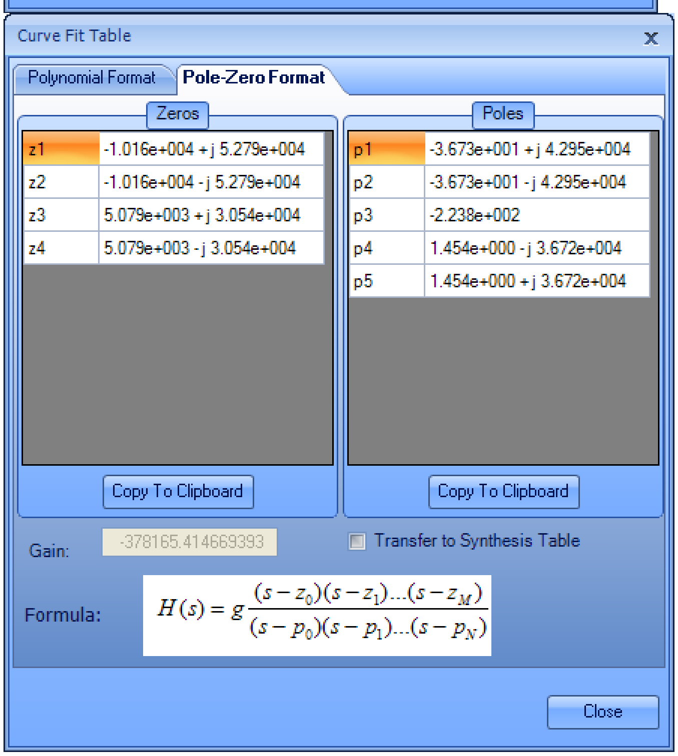
<u>Fixed or Max Order</u>: set a fixed order number for the polynomials or all the system selects the maximum orders.
<u>Order Number</u>: when fixed order is select, please set the order number for numerator or denominator.
<u>Fit Region</u>: select the method to define the frequency band. User either two vertical cursor or select the Full Span.

Press the ***Perform the Curve Fit*** icon to conduct the fit. The measured FRF and fitted data will be shown.



Press the ***Curve Fit Table*** icon to view or copy the curve fit coefficients. Notice that two forms can be selected to view these coefficients.



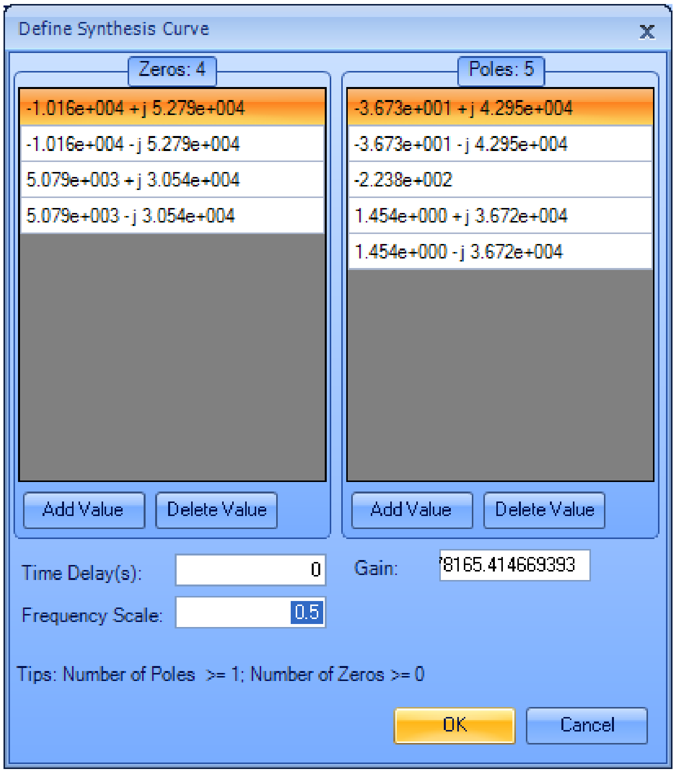


If the box ***Transfer to Synthesis Table*** is checked, this table will be automatically passed to the synthesis table for future synthesis or editing.

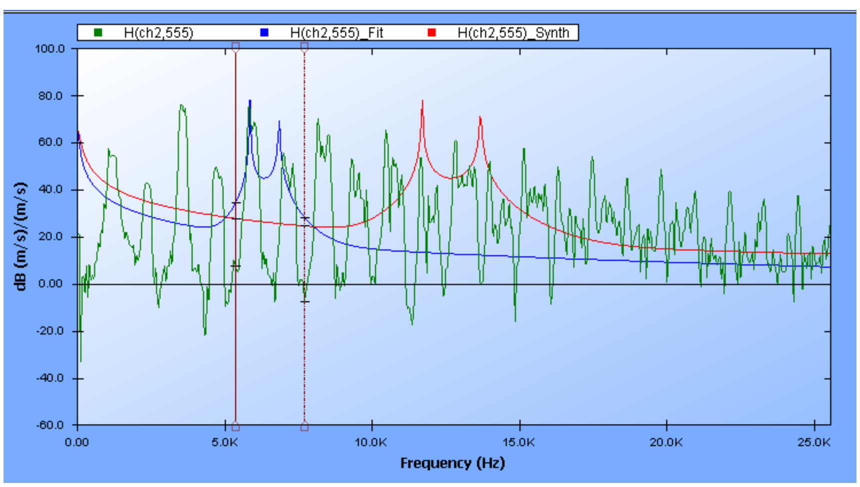
Curve Synthesis Process

The same to that of curve fit, the curve synthesis process is conducted in the pane loaded with an FRF signal.

Press the **Define Synthesis Curve** icon and define the coefficients. You can add and edit zeros, poles, time delay and gain value.

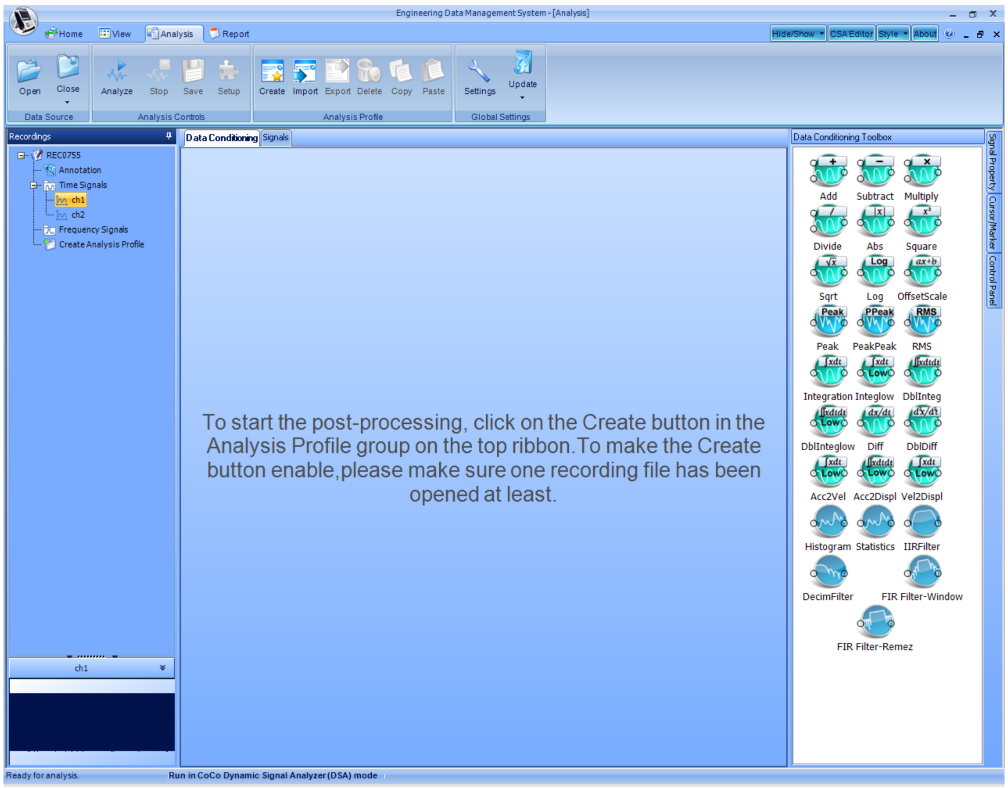


Click OK, then the synthesis curve will be added to the pane:

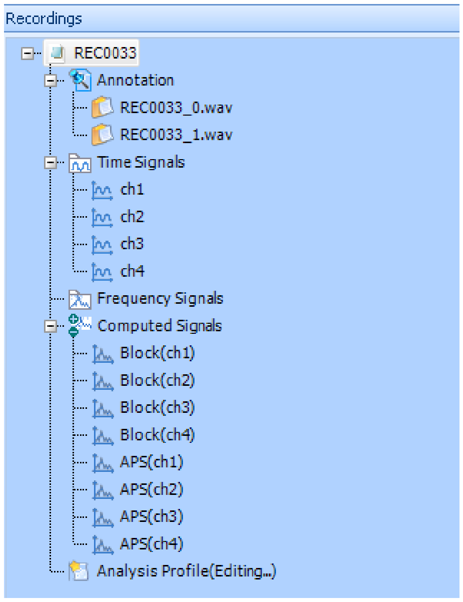


7. Analysis Page

Analysis Page is designed for setting up the post-processing analysis. An Analysis Profile can be created for file that contains time stream signals. The computed signals are the computation results while time streams go through the Analysis Profile.



To start the Analysis, click on the ***Create*** button in the Analysis Profile group on the ribbon, or click on the ***Create Analysis Profile*** item in the Recording view.

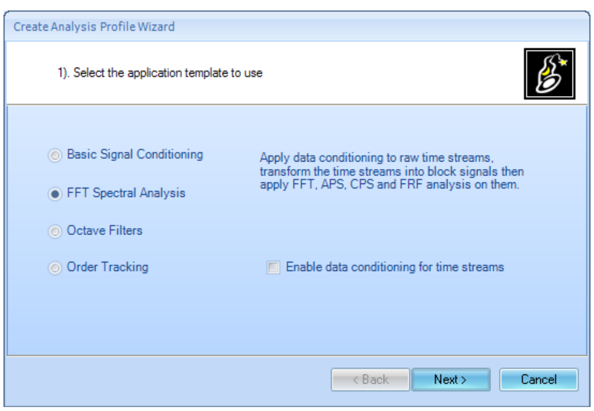


As it can be seen from the picture above, the recording file includes following components:

1. Annotation: text annotation, voice annotation or attached objects
2. Time signals: these are the time streams that can be used for the input of analysis
3. Computed signals: these are the output signals of the Analysis Profile
4. Analysis Profile: this is a collection of settings for post processing

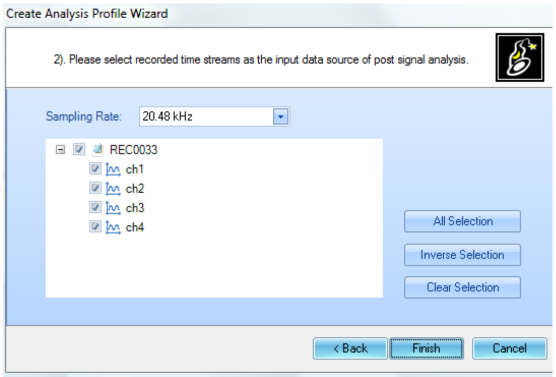
Select Application Template

The Analysis Profile is built based on the Analysis Template:

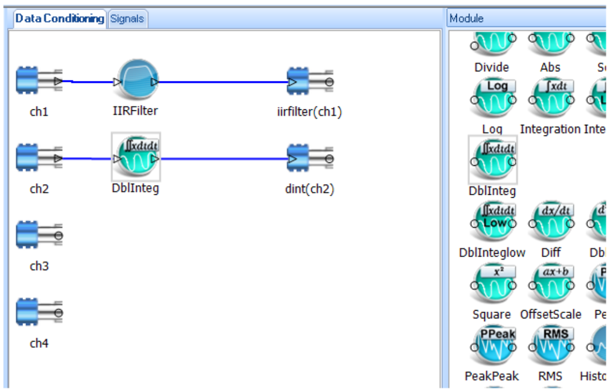


Select Data Source

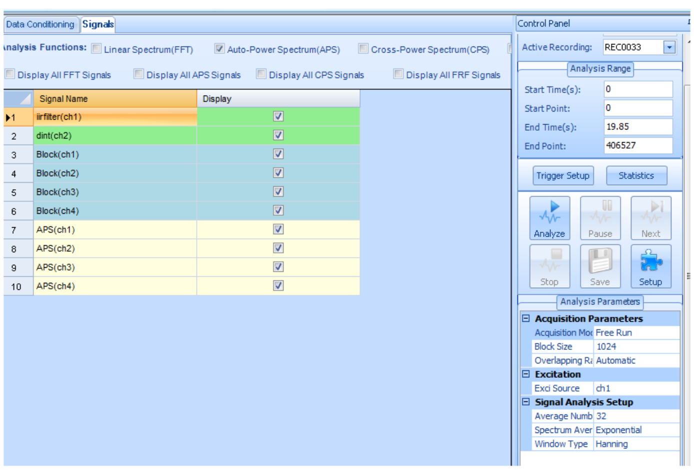
Enable or disable the time streams recorded before for post processing:



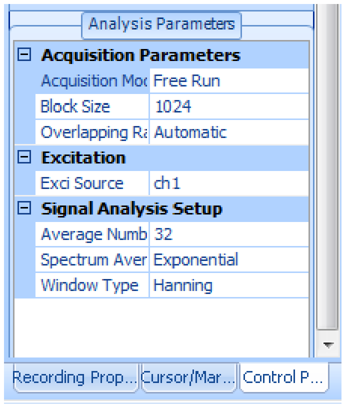
Apply Data Conditioning (Optional)



Enable or Display the Displayed Signals



Setting up the Analysis Parameters



Analysis Parameter settings depend on the application template. The picture above shows the analysis parameters for the template of the basic spectral analysis.

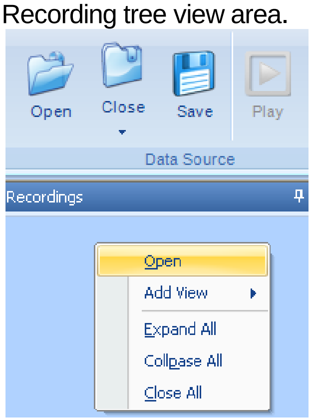
Post Processing Control

When the user push the Analysis Button, the EDM will switch to the View Page tab automatically. Please display appropriate signals in the pane to observe time streams (data sources) and computed signals

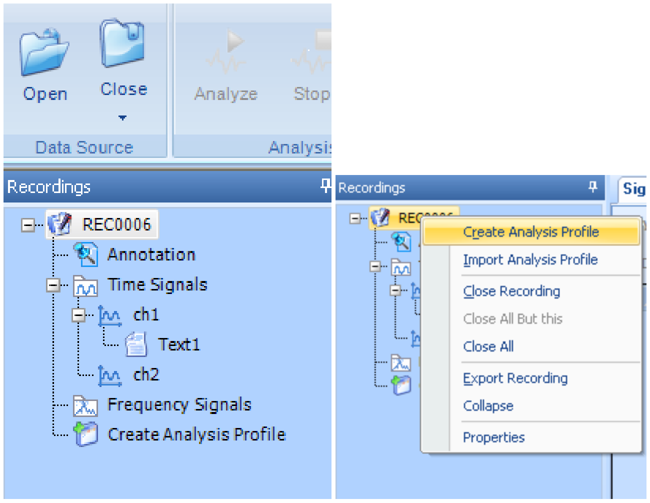
Analysis Profile Operation (in details with examples)

Create the Analysis Profile

1. Load the data file containing the time streams to be used as the input signals of the Analysis Profile. This can be done by clicking on Open icon in the Data Source group, or right click on the Open menu in the Recording tree view area.

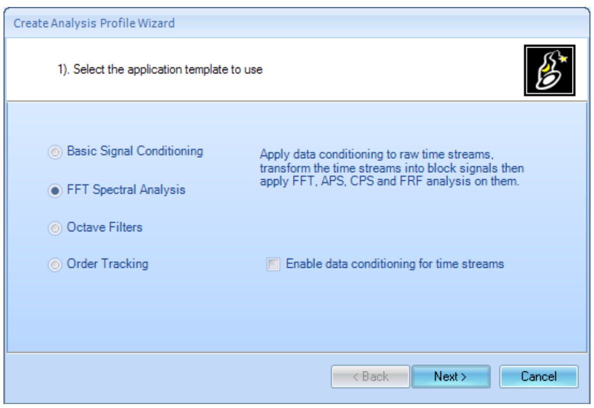


2. After the input signal file is opened, select ***Create Analysis Profile*** Wizard.

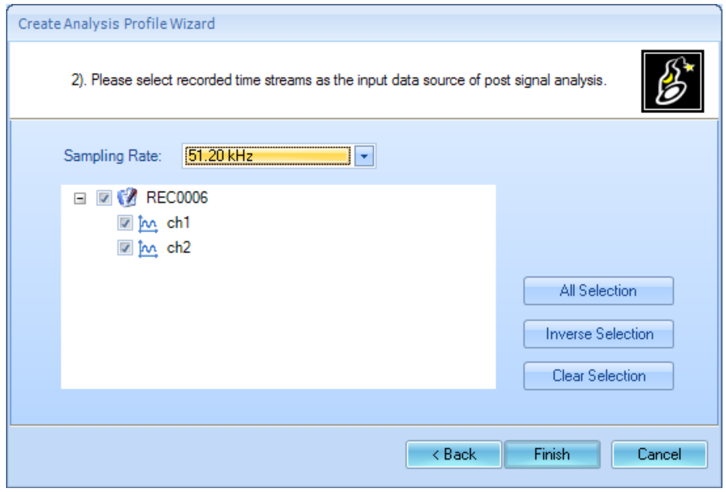


3. In this step, you will select the application template. An application template is a model used to create the analysis process. Please select one of: Basic Signal Conditioning, FFT Spectral Analysis, Octave Filters and Order Tracking.

   When select any template other than the Basic Signal Conditioning item, a check box ***Enable data conditioning for time streams*** will be shown. Check this box if you want to apply data conditioning, such as digital filters to the time stream before they enter the signal processing based on the template.

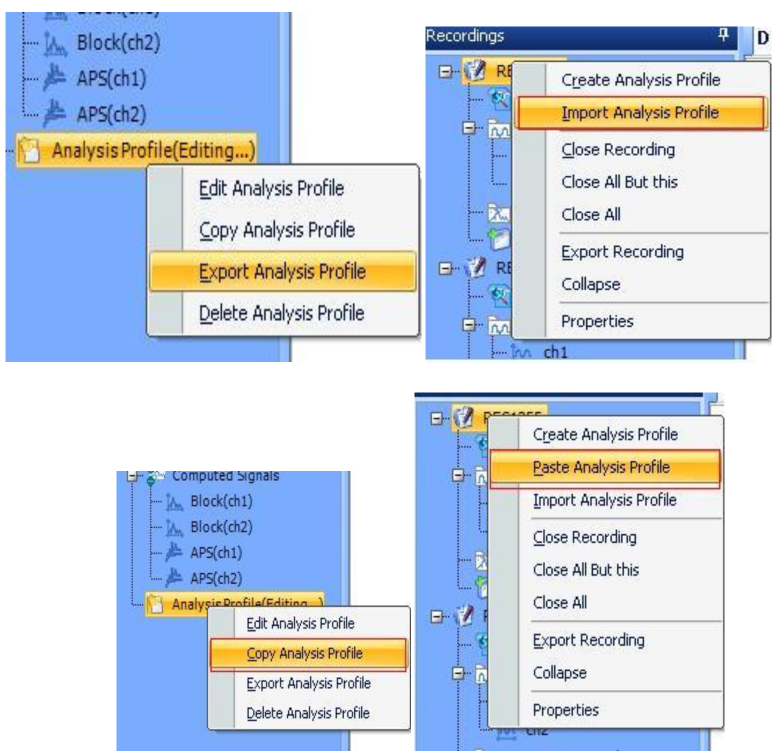


4. Select appropriate sampling rate. Notice that in many cases the sampling rate is fixed based on the rate when data was sampled.



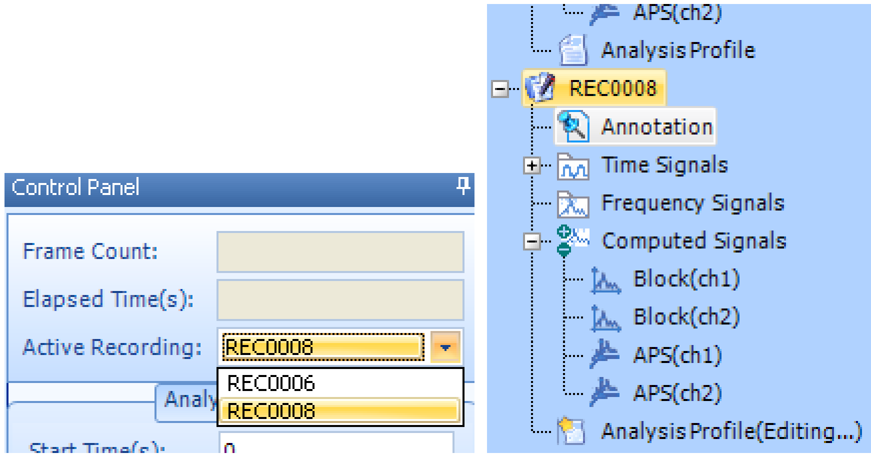
Import and Export the Analysis Profile

The Analysis Profile can be easily imported, exported, copied onto the data files that have the same structure. The "same structure" means these files have the same input streams. For example if you always record the 3 channel signals and saved them into multiple files, you only need to create the Analysis Profile once and copy it over to all the files. Here are some bitmaps:

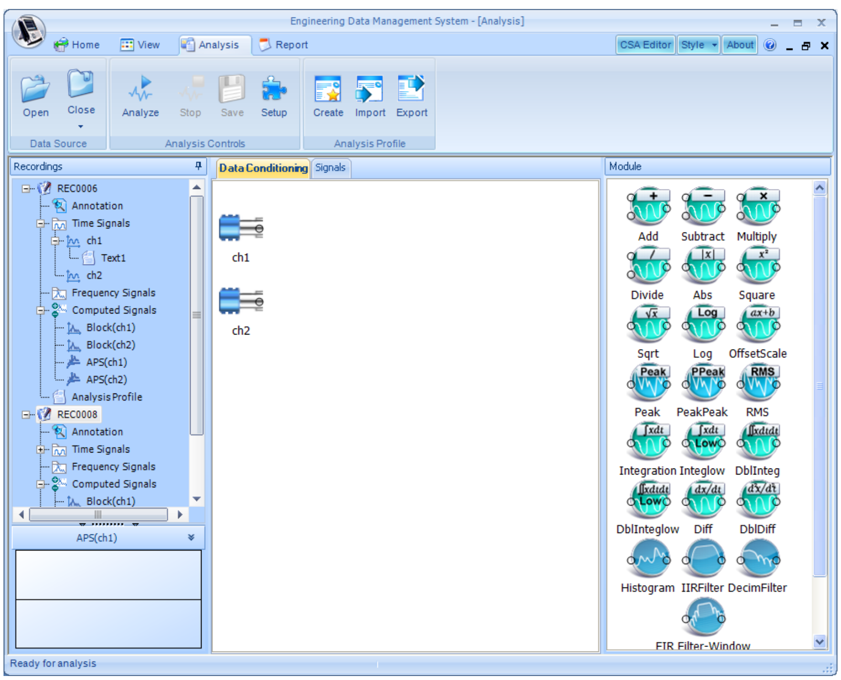


Activate One of Analysis Profiles

Multiple data files can coexist with their own Analysis Profile. Only one can be made active at one time. To activate or switch between them, click on Active Recording from the Control Panel.



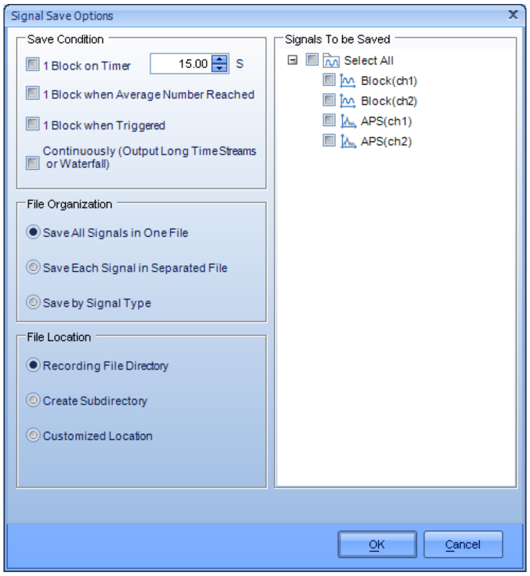
1. Editing the Analysis Profile: Simply switch to the Analysis tab:



2. Delete the Analysis Profile: in the tree view, right click on the Analysis Profile item, then select: ***Delete Analysis Profile***

Setup for the Analysis Profile

Click on the Setup button on the Control Panel:

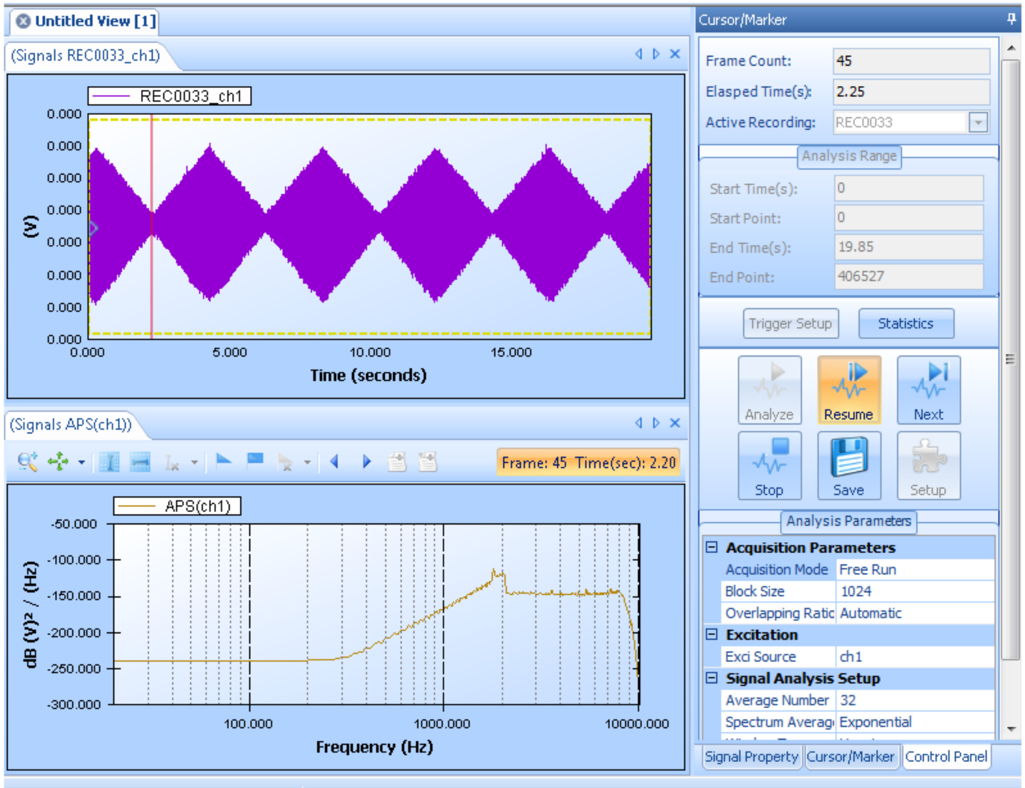


Parameters:

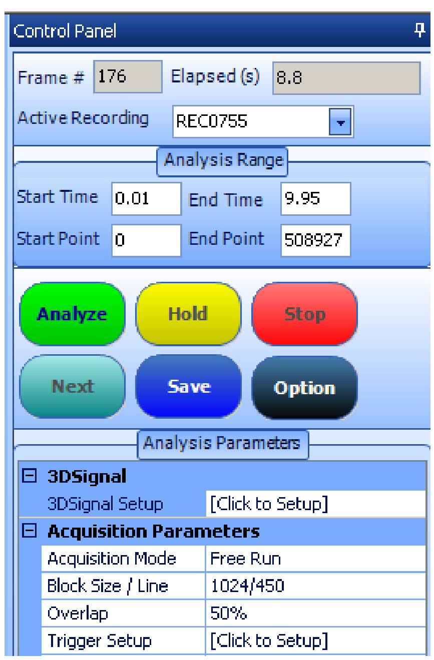
Save Condition: File save conditions
  Timer: Save with a timer
  Average Number Reached: Save when number of average reached
  Trigger Captured: Save when triggered

File Organization: How the data file will be organized when saved
  Save All Signals in One File: Save all signals into one atfx(.dat) file
  Save Each Signal in Separated File: Save signals into separate atfx(.dat) file
  Save By Signal Type: Based on the signal type, group them into different .atfx(.dat) files

Control Panel



Control Panel is used to control the post processing operations. Press the Analyze button to start the processing. The data processing starts at Start Time (or defined by Start Point) and ends at End Time (or End Point)



Frame :the total processed frames so far
Elapsed (s): the total processed time so far
Active Recording: Recording file that is activated
Start Time: In the time stream data signals define the starting time for post processing, in seconds.
Start Point: In the time stream data signals define the starting sample points for post processing
End Time: In the time stream data signals define the ending time for post processing, in seconds.
End Point: In the time stream data signals define the ending sample points for post processing
Trigger Setup: defines the trigger condition for post processing
Analyze: Push this button to start the post-processing
Pause/Resume: Pause and resume the post-processing
Stop: Stop the post processing
Next: process one frame
Save: Save signals
Option: setup related to the save signals.

Order Tracking Post Analysis

Data Recording for Order Tracking

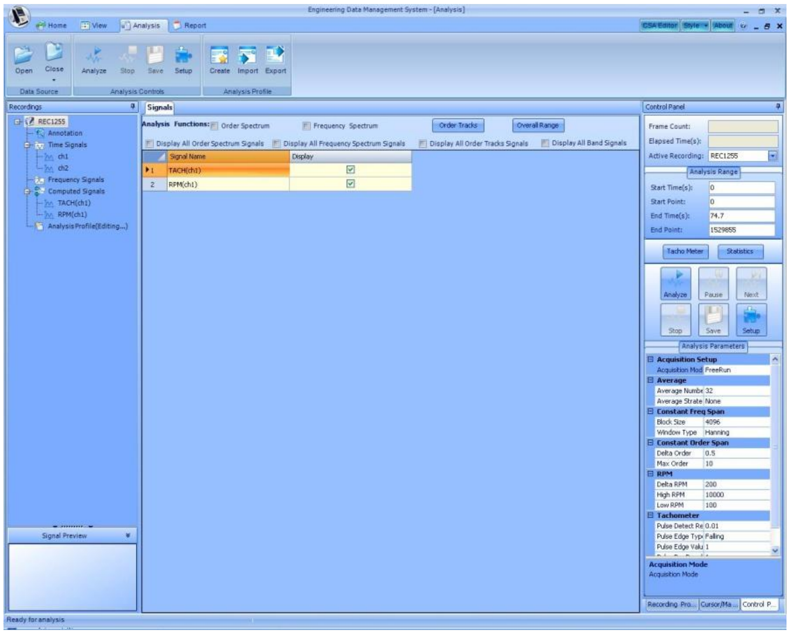
If you plan to conduct the order tracking post processing with recorded data, simply record all the time domain signals as well as the time waveform of the tachometer signal. The tacho signal is usually a chain of pulses. Please verify the engineering units and sensitivity settings of the vibration measurement channels.

The engineering unit of tachometer channel does not matter. It is usually set in voltage measurement.

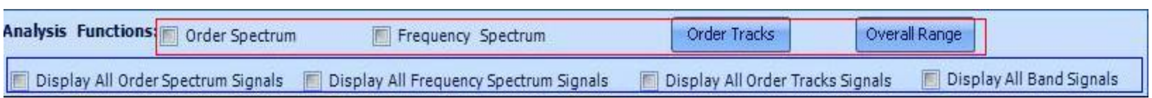
Please set the sampling rate high enough to cover the maximum analysis frequency range of order tracking signals.

Main Setup Page of Order Tracking

According to the instruction above, the Order Tracking post analysis page will be like this:



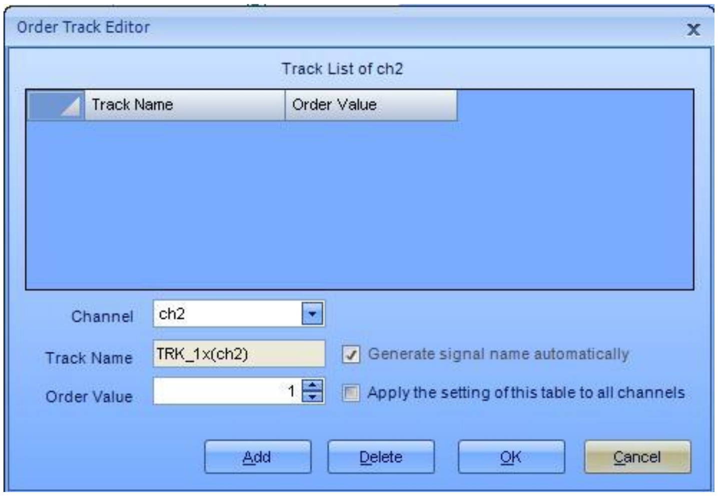
1. Under the Analysis Functions area, there are: Order-Spectrum, Frequency Spectrum, Order Tracks and Overall Range. Check one or more of them if you want to compute their results.



2. In the table please enable these items to be displayed

	Signal Name	Display
1	TACH(ch1)	☑
2	RPM(ch1)	☑
3	RPM_APS(ch2)	☑
4	ORD_Spectrum(ch2)	☑

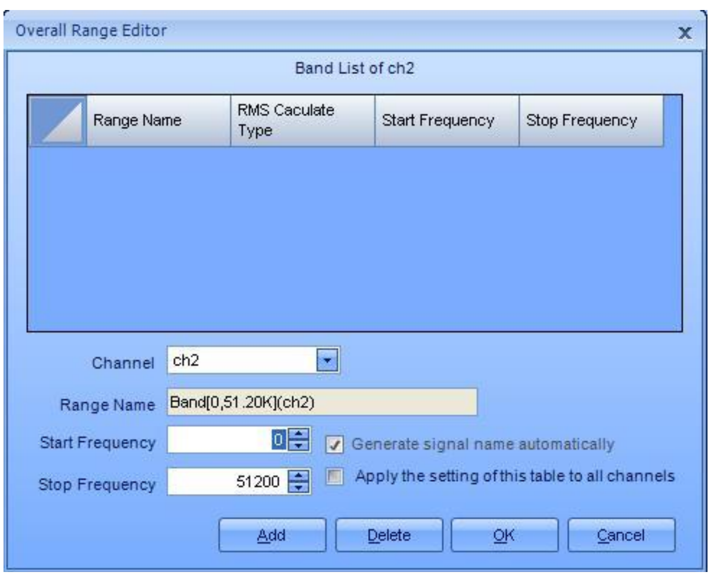
3. Click on ***Order Tracks***, the following dialog box will be shown:



This dialog box defines the order track signals for each input channel.

Channel: select this channel for order tracking analysis
Track Name: Order track signal name
Order Value: The value of orders. For example, enter 1.0 for order 1, 0.5 for half order
Apply the Setting of this table to all channels: Select this option to apply the same setting to all input cannels
Add: add this signal to the list. It will be computed when Analyze button is pushed.
Delete: delete this signal from the list.
OK and **Cancel**: accept or cancel the settings.

4. Click on ***Overall Range*** button to enable the band spectrum analysis. A band spectrum presents the energy in certain frequency band as variable of rotating speed.



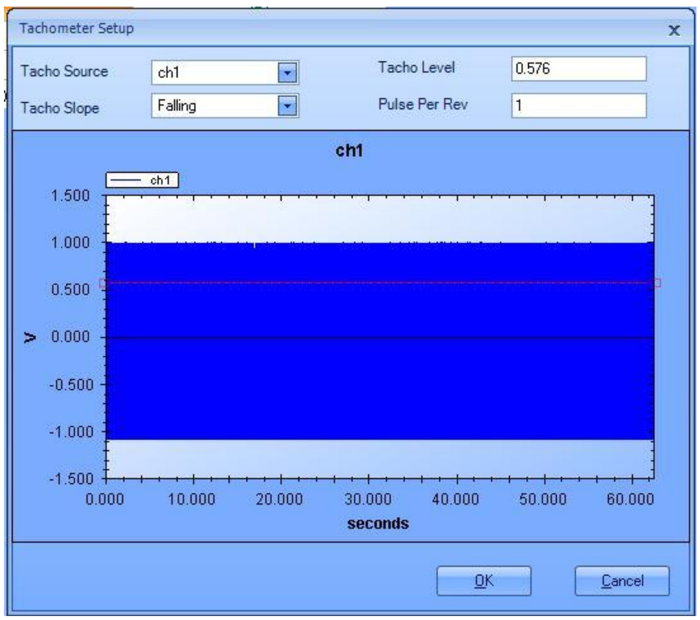
Channel: select this channel for band spectrum analysis
Range Name: the signal name of the band spectrum
Start Frequency: the low cutoff frequency of such band-pass filter
Stop Frequency: the high cutoff frequency of such band-pass filter

Apply the Setting of this table to all channels: Select this option to apply the same setting to all input cannels
Add: add this signal to the list. It will be computed when Analyze button is pushed.
Delete: delete this signal from the list.
OK and **Cancel**: accept or cancel the settings.

Tachometer Setup

Tachometer signal is critical in the order tracking analysis. The tacho signal must be recorded, in its original time waveform, to the CoCo flash memory at the same time when other measurements are taken.

To set up the tachometer signal in the post analysis, simply click on the Tachometer button on the Control Panel, and assign the parameters for pulse detection.



Tacho Source: the input channel that records the tacho waveform
Tacho Level: the threshold of tacho signal detection
Tacho Slope: using rising or falling edge for RPM detection
Pulses per Rev: pulses per revolution

You can use the mouse to drag the level line to make it within the detection range.

OT Analysis Parameters

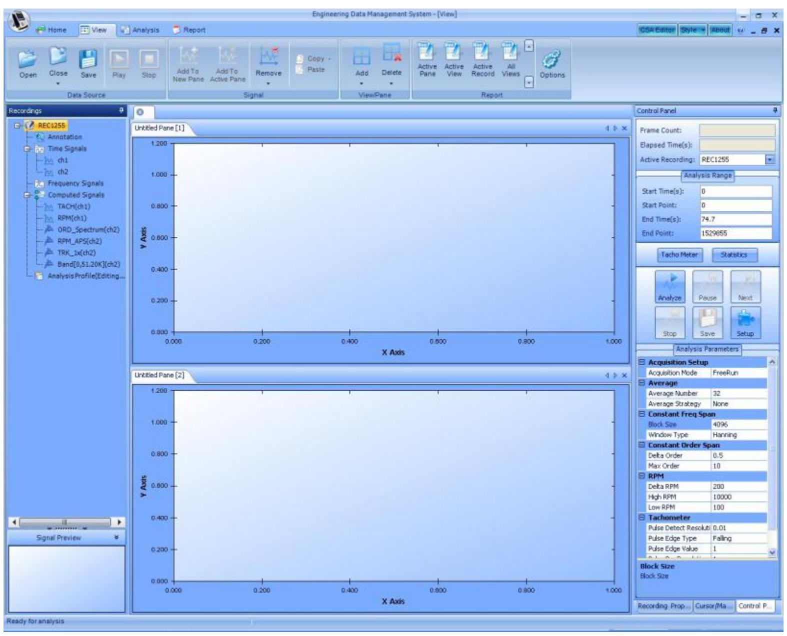
On the Control Panel, the analysis parameters can be set.

Acquisition Mode	FreeRun: Running all the time without using RPM trigger RunUp: From the LowRPM to HighRPM, calculate all the signals once. RunDown: From the HighRPM to LowRPM, calculate all the signals once. RunUpAndDown: Conduct RunUp and RunDown once. RunDownAndUp: conduct RunDown and RunUp once
Average Mode	Average Strategy : None, Exp , Linear, Peak Hold Average Number
Constant Freq Span	Block Size (Lines) Window Type
Constant Order Span	Max Order Delta Order
RPM	Low RPM: The minimum 1x RPM value of interest High RPM: The maximum 1x RPM value of interest

	Delta RPM: The resolution of RPM spectrum or band spectrum.
Tachometer	Pulse Per Revolution: the pulses per revolution. This number can be a non-integer. The non-integer value will be needed when multiple gearboxes with different tooth ratio are involved. Pulse Edge Value: threshold of detection Pulse Edge Type: Rising or falling Tacho Source: source channel number of the recording signal used as tacho signal

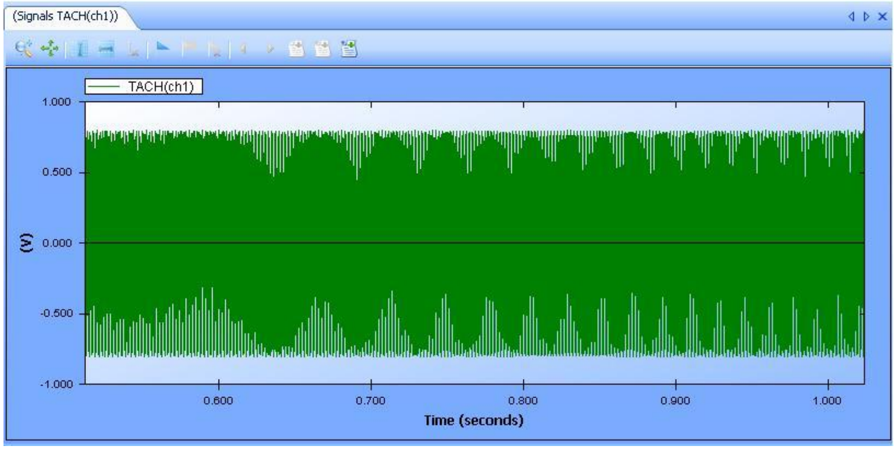
View Order Tracking Computed Signals

Select the View tab:

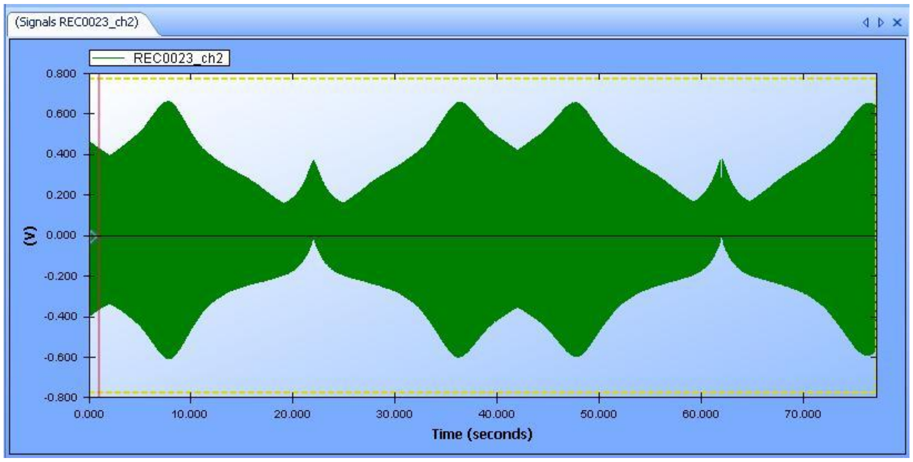


Drag the signals from the left tree view to the empty panes, or create new pane for signal plot. Press Analysis button to process the signal. Followings are the example signals shown in this test.

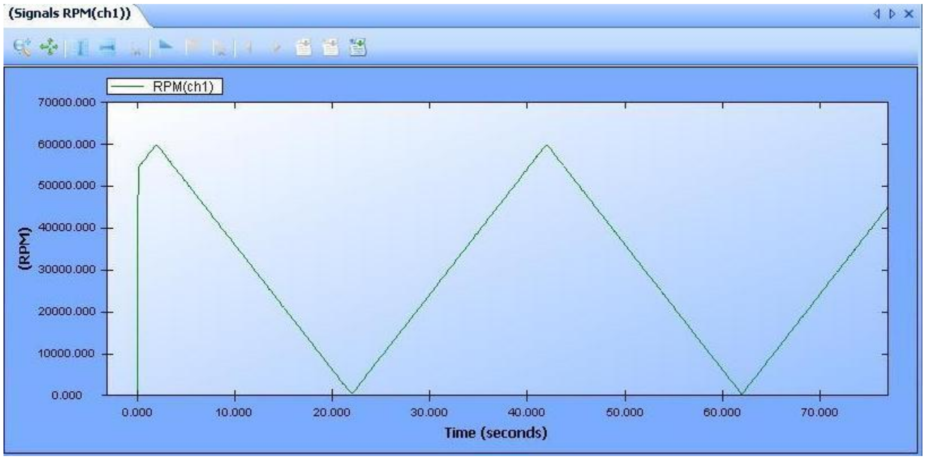
（1） Tacho signal



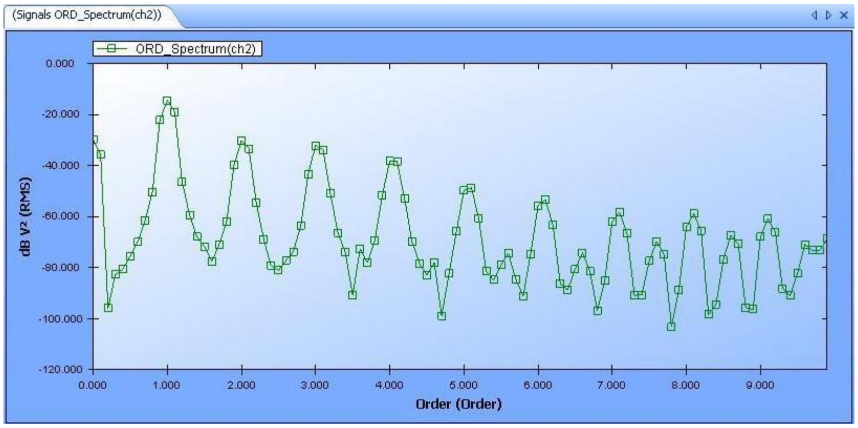
（2） Channel 2 vibration signal

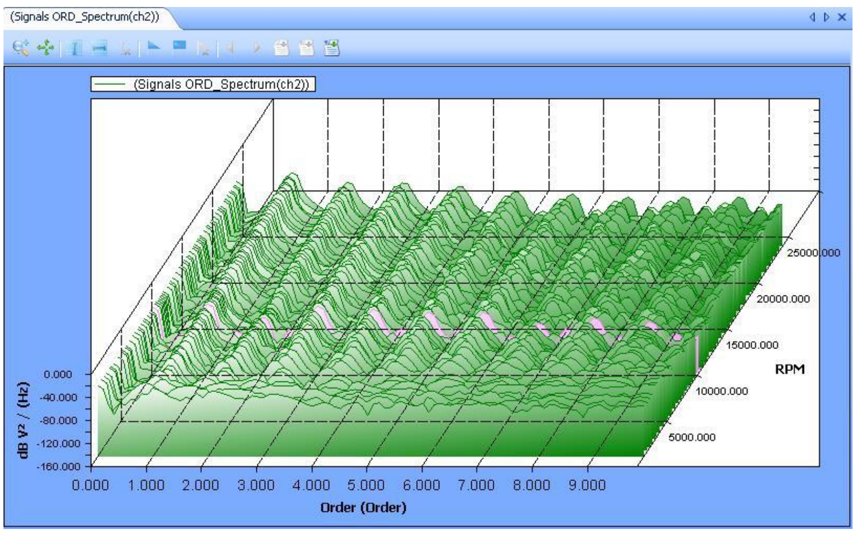


（3） RPM signal



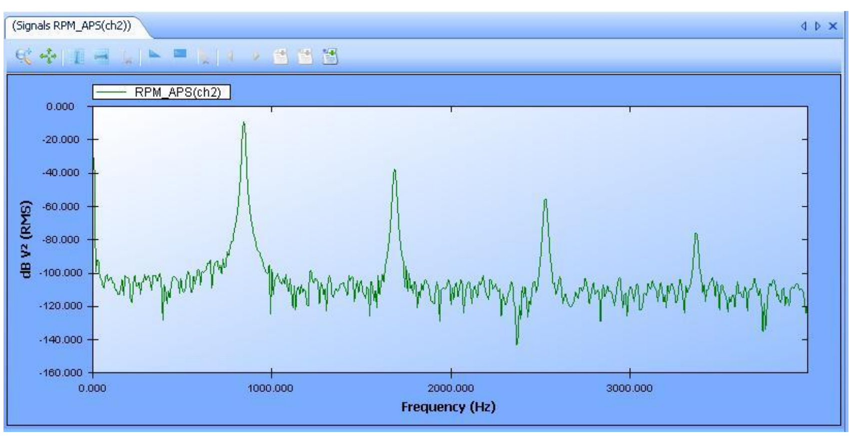
（4） Order Spectrum Signal

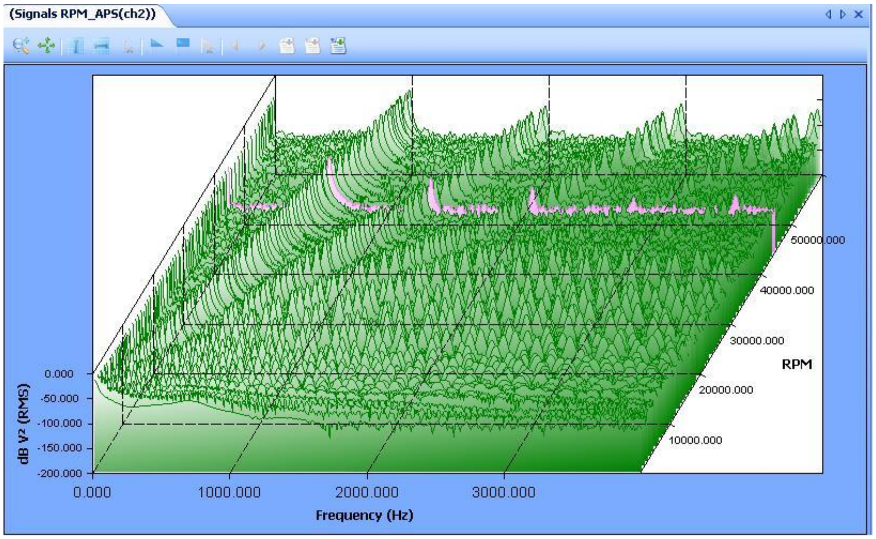




3D display

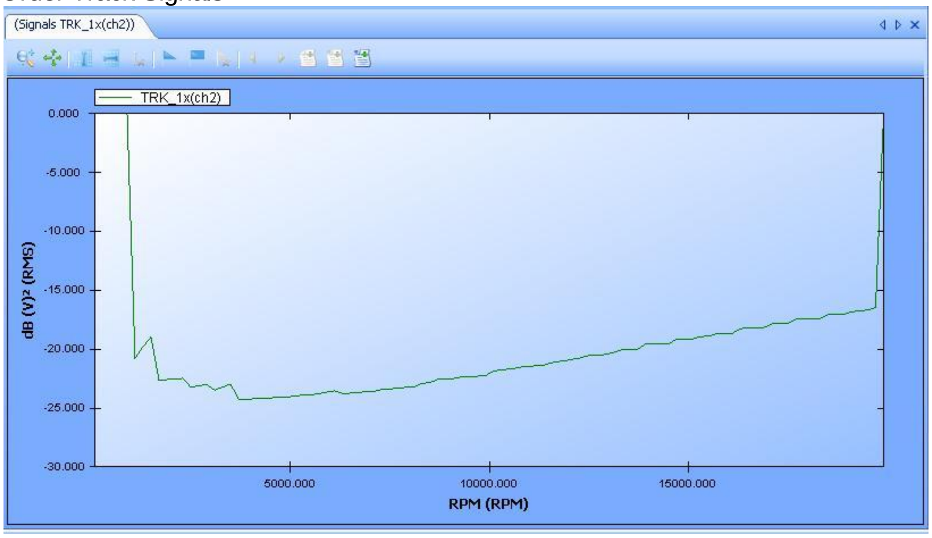
（5） Frequency Spectrum Signals



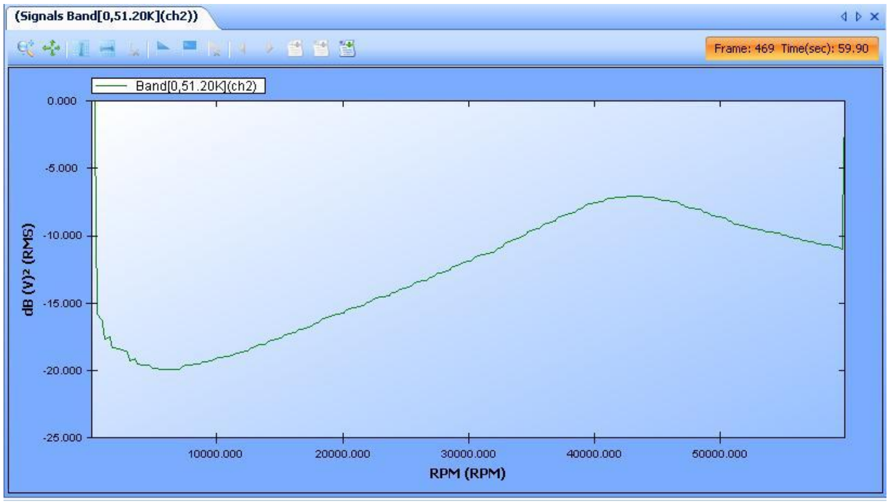


3DSignals

（6） Order Track Signals



（7） Overall Range Signals



Octave Analysis Post Processing

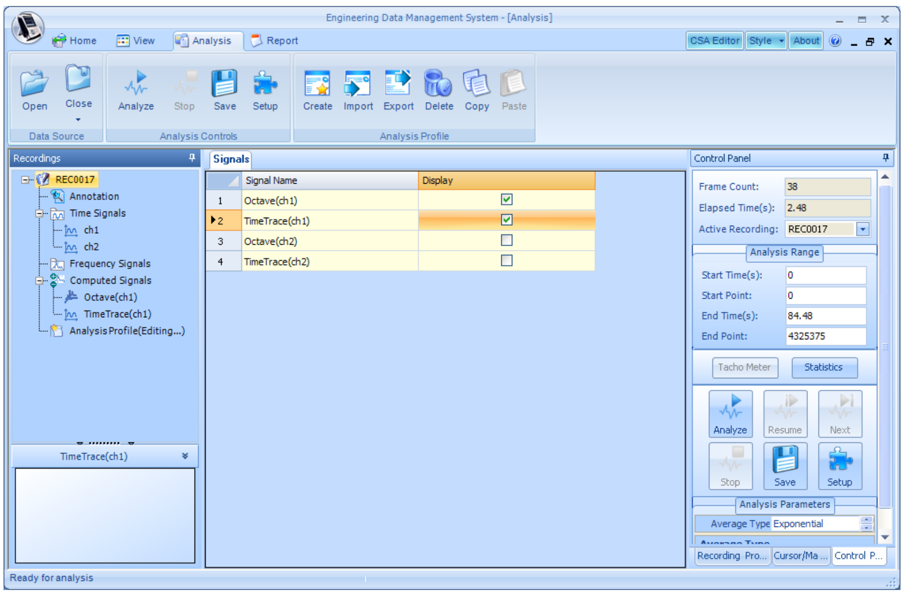
Data Recording for Octave Analysis

If you plan to conduct the octave analysis post processing with recorded data, simply record all the time domain signals. If the octave filters are to be displayed in SPL, please set the measurement quantity in the channel table of CoCo as SPL.

Please set the sampling rate high enough to cover the maximum analysis frequency range of octave signals. The sampling rate supported must be one of: 20.48kHz, 25.6kHz, 40.96kHz, 51.2kHz, 64kHz, 81.92kHz, 102.4kHz

Octave User Interface

After select the Octave Analysis template, the main display page will be like this:

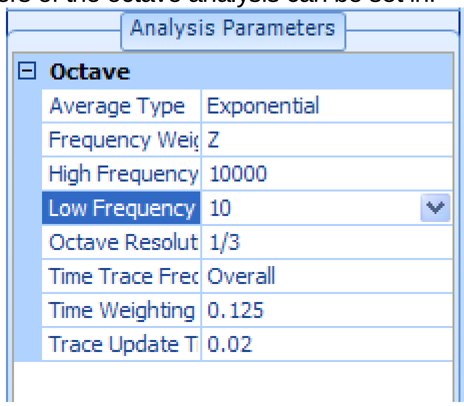


Input signals: the recorded time streams
Octave(chx): the octave spectrum for the chx

TimeTrace(chx): this is the output r.m.s. time trace of the individual filter of the octave spectrum. The center frequency of that filter is controlled in real-time by parameter Time Trace Freq.

Octave Analysis Parameters

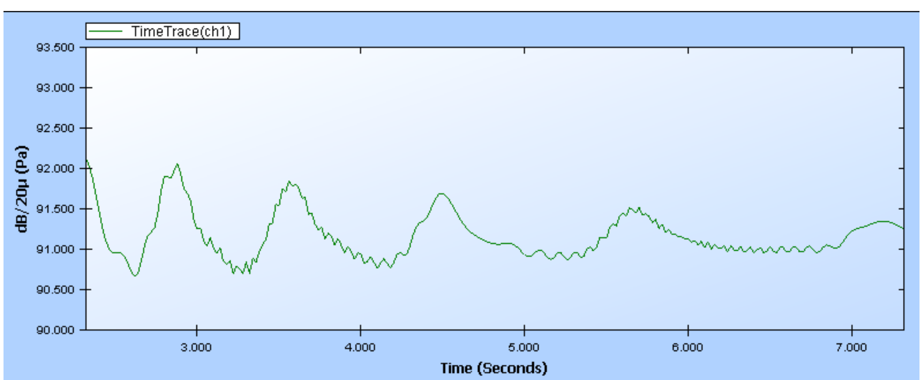
The analysis parameters of the octave analysis can be set in:



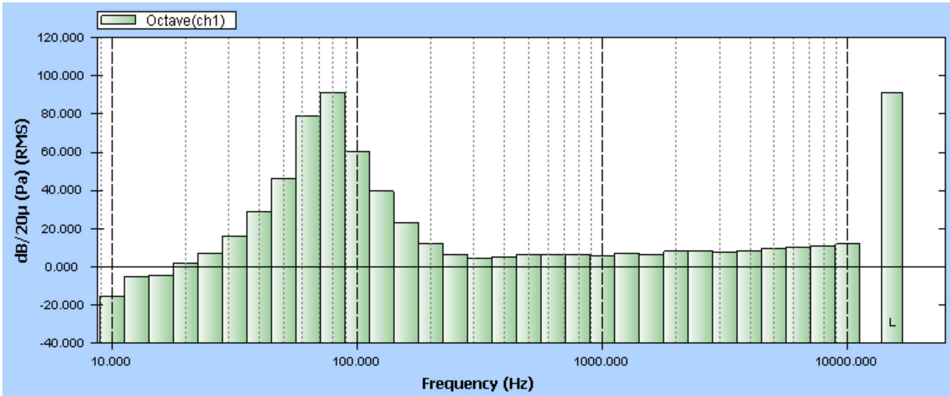
Octave Resolution: the fractional octave factor, 1/1, 1/3, 1/6 or 1/12
Low Frequency: the lowest analysis frequency of interest in octave analysis
High Frequency: the highest analysis frequency of interest in octave analysis
Average type: Linear, Exponential and Peak Hold
Average Time: the averaging time of r.m.s. output after the filter
Weighting Type: Z, A, B, C, frequency weighting
Time Trace Freq: the center frequency of filter for the time trace
Trace Update Time: the update time interval of that time trace

Octave Signal Examples

Time Trace Signal



Octave Spectrum



8. Report Page

The Report Page displays reports that are generated in the Home or Analysis Page. Reports are generated using macros within Microsoft Word and Excel. The MS Word and Excel applications are embedded in the Report Page in the EDM software as shown in **Error! Reference source not ound.**. After the report is generated you can manually edit the document, change the format, save and print using all the commands available in the Word or Excel software. Note the automatic Report feature requires Microsoft Word 2000 or later.

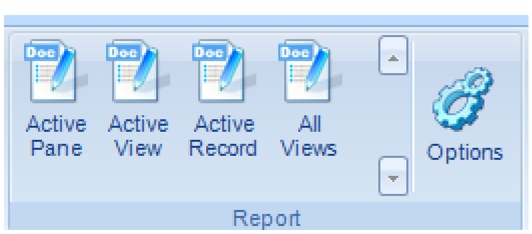
Generating Reports

Reports can be generated from the Home or the View Page.

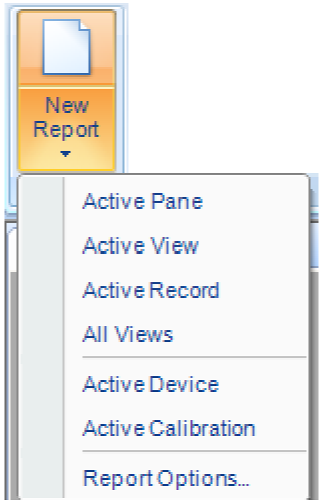
Home Page – click on the Report Button on the Device File View to generate a Device Report.

View Page – a report can be generated from the Analysis page using the following controls. Right click on the pane and select Report this Window of the menu items:

Or click on one of the items in the ribbon:



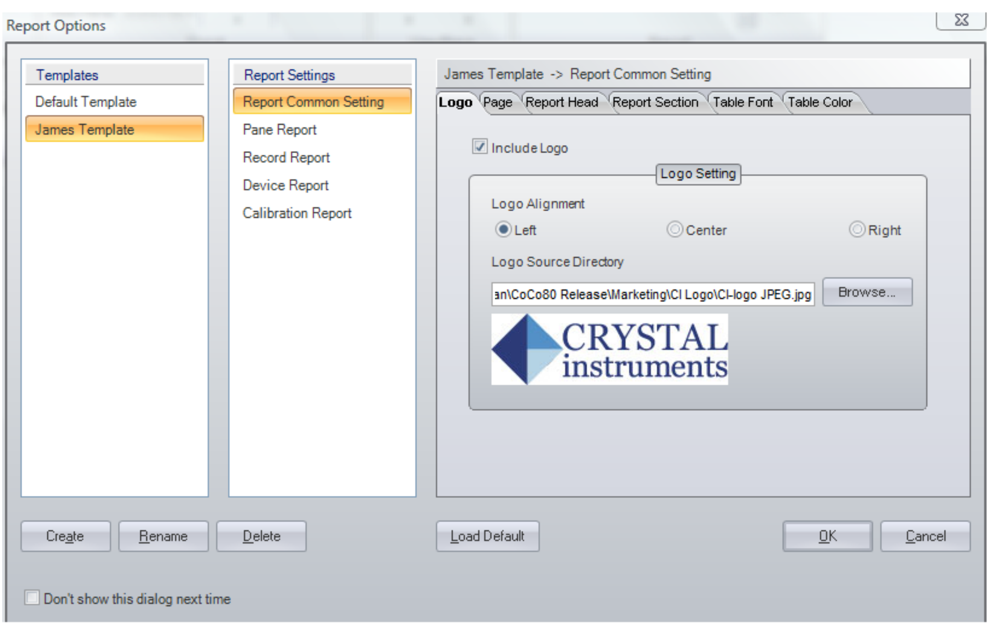
Or go to the Report tab page and select one of the items under ***New Report***:



Note: it will take about 15 seconds in the first time to initiate the report because the EDM software will load in related libraries.

Report Template

Report can be generated based on user-defined template. Report shows active pane, active view, active device etc.. Many fields and attributes are customizable. After report is created and previewed, it can be exported into various file formats.

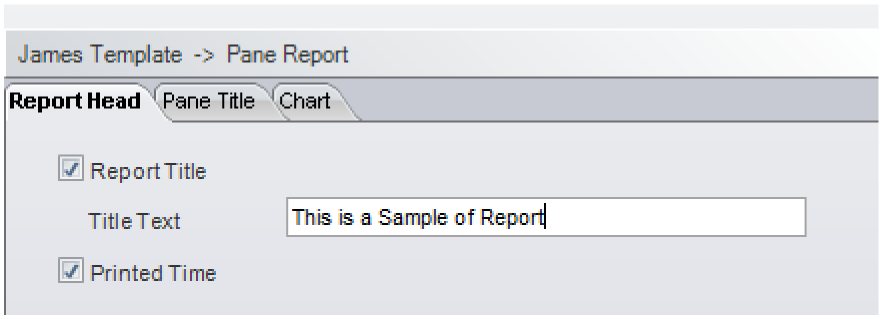


Template Operations

Create, rename, delete and load default.

Customizable Fields and Attributes of Template:

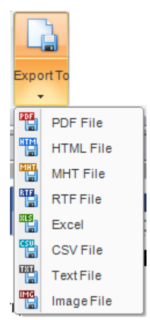
Logo: insert, position and alignment
Page Orientation: Portrait or Landscape
Report Header: Font size, color and style
Report Section Title: Font size, color and style
Report Title: User definable text
Section Title, Signal Property, Cursor, Marker and Annotation: all text can be defined by user
Chart Graph: Title, size and background color can be defined by user



Report Operations

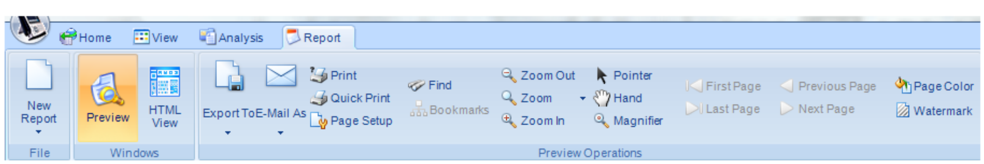
Create new report, Preview, HTML view, Email to, Print, Quick Print, Page Setup, Find, Bookmarks, ZOOM in and out, Go to Last and First Page, Page Color, Watermark

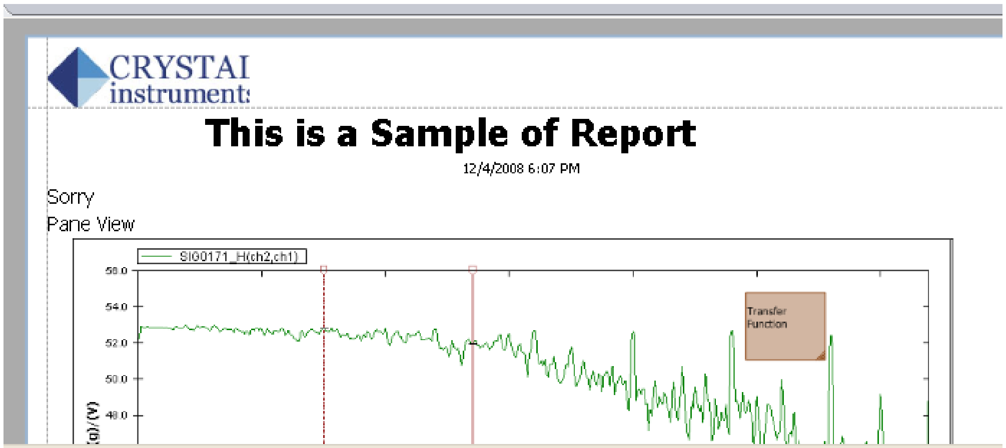
Report Export Format



PDF, HTML, MHT, RTF, Excel, CSV, Text and Image files.

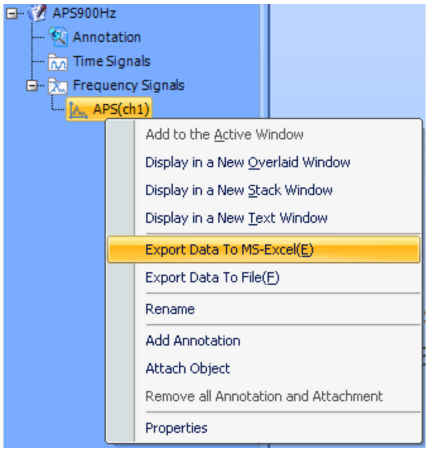
The Report Ribbon displays the most frequently used commands after the report is created:



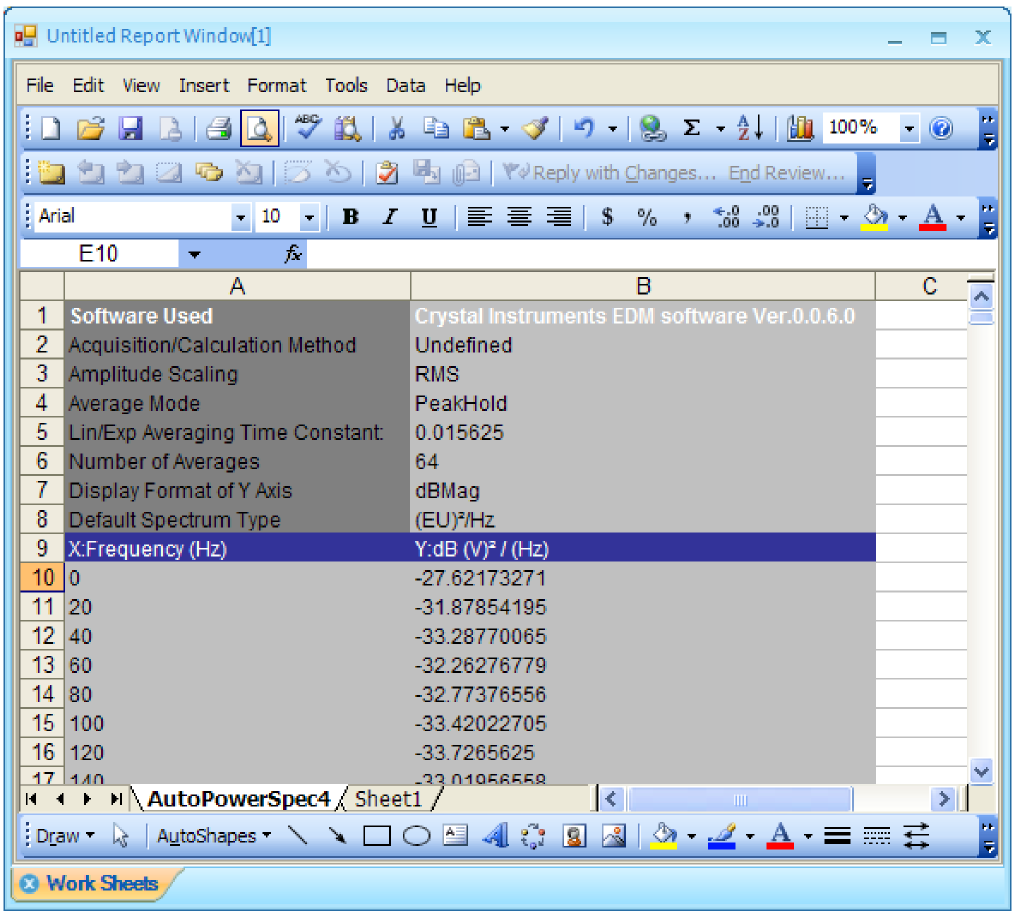


Excel Report

Figure 85 shows a typical Excel Report. This report can be generated by right clicking on a signal name in the Recordings View in the Analysis Page and selecting Report to Excel from the popup menu. The Excel Report shows the data file attributes and the values in a spreadsheet format. You can format the display using the Excel format commands but you can not edit the content of the report.



■ Figure 84. An Excel Report can be generated by right clicking on a signal name in the Recordings View in the Analysis Page and selecting Report to Excel from the popup menu.



■ Figure 85. An Excel Report includes file attributes and data values in a spreadsheet format.

9. CSA Editor Page

This section describes the Configurable Signal Analysis concept that is the basis for the CoCo-80 functionality and allows advanced users to customize the analysis features to suit individual needs. This section gives a brief description that is intended for the basic user. It does not describe writing new or editing CSA Files which is a topic for advanced users. For more on this topic refer to the CSA Editor Guide.

When the CoCo-80 powers up the Welcome screen is shown. From this screen the use must select one of the CSA Files loaded on the CoCo-80. When a CSA File is selected, it defines the settings and analysis functions that are computed by the CoCo-80. These settings include the following:

- Time Stream Data Recording Candidates
- Block Data Save Candidates
- Trace Candidates
- Parameters used by the data conditioning functions such as Add, Subtract, Multiply, Divide, Square, Square Root, RMS, Scale, Offset, Decimate
- Parameters used by the signal analyzer functions such as FFT, Auto Power Spec, Coherence, FRF

The CSA File defines a subset of signal candidates, parameters and functions that will be used on the CoCo-80 Device. This is the key to enabling a wide variety of powerful signal analysis functions on a portable device with a simple user interface. By limiting the signal candidates and parameters in the CSA, the simple user interface is possible and the hardware computational resources are optimized. The list of signal candidates limits what signals can be recorded from the CSA File. You can specify that one, some or all of the signal candidates to be used on the device but you can not select a signal that has not been defined as a candidate in the CSA File.

The CSA File is designed to control how the data is processed, not how the data is acquired. When the CSA File is changed, the processing functions are changed according to the new CSA File, but the data acquisition parameters are not changed. For this reason, the following settings are applied globally and are not part of CSA File:

- Sampling Rate
- Input Channels: sensitivity, coupling, channel labels
- Output Channel: output waveform settings

All pre-programmed CSA Files have predefined parameters that are loaded when the file is selected. You can modify the parameters on the CoCo-80 from the ***Param*** Soft Button in the Display screen. Modified CSA Files can be saved with a different name using the Save As soft Button in the Analysis screen so that the original files are not overwritten.

All pre-programmed CSA Files carry a variable called *Maximum Sampling Rate*. This is the sampling rate that this CSA can safely execute without exceeding its computational resource limit. Maximum Sampling Rate is used to limit the selection of the sampling rates.

Preprogrammed CSA Files

The CoCo-80 is preprogrammed with a set of default CSA Files which provide a wide range of options that meet most users' needs. Additional CSA Files may be downloaded from the Crystal Instruments web site. In addition, by using the CSA Editor the advanced users may edit or develop their own customized CSA Files to meet their specialized needs. A summary of the default CSA Files is given below.

- Table 2. Preprogrammed CSA File Descriptions.

CSA File Name	Description
Add(2)	Add any of two input channels and generate one time stream
AutoPowerSpec(8)	Apply window, FFT, square and average calculation to generate auto-power spectrum for 8 channels
OffsetAndScaling(8)	Apply gain and offset for 8 channels
Capture(8)	Show time stream and transient capture for 8 channels
Coherence(8)	Apply window, FFT, square and average calculation to 8 channels and generate 7 coherence functions
CrossPowerSpec(8)	Apply window, FFT, square and average calculation to 8 channels and generate 7 cross power spectrums
Decimation(8)	Apply n stage of decimation and filters to 8 channels
DoubleIntlow(8)	Compute Double Integration(for low frequency signals) for 8 channels
FRF(8)	Apply window, FFT, square and average calculation to 8 channels and generate 7 Freq Response functions
Integration(8)	Compute Integration value for 8 channels
Integrationlow(8)	Compute Integration(for low frequency signals) value for 8 channels
RMS(8)	Compute RMS for 8 channels
Time(8)	Display the time streams of 8 channels

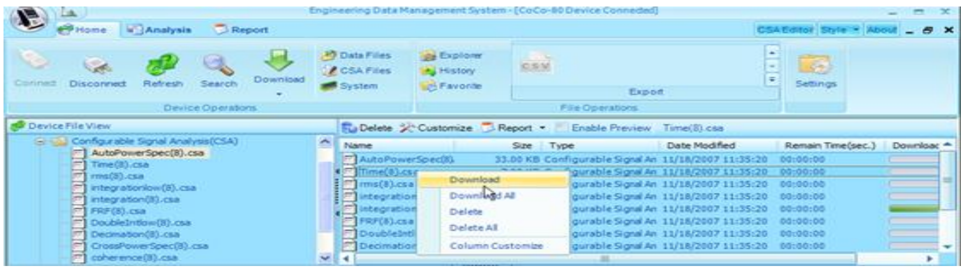
Change CSA Files from the CoCo-80

After a CSA File is selected it can be modified from the CoCo-80 to change its parameters using the ***Trace*** and ***Param*** soft buttons in the Display screen. The modified CSA File can then be saved using the Save As soft button in the Analysis screen. This allows modified CSA File to be saved and used again later.

Editing CSA from the EDM Software

CSA Files can also be edited or a new CSA Files can be created from scratch from the EDM software on a PC. This feature allows the advanced user to create custom analysis functions to suit their special needs. The EDM software includes the CSA Editor Page which guides you through setting up the CSA File and defining the analysis functions. This advanced topic is not covered in this manual. Refer to the CSA Editor Guide for information on editing CSA Files from the EDM software.

CSA Files can be downloaded from the CoCo-80 Device to the PC from the EDM Home Page. Click CSA Files in the Device Operations Ribbon and then select the CSA File in the Device File View. Finally right click on the CSA File and select Download from the popup menu. You can download CSA Files so that the files are backed up on the PC and can be uploaded to the device at a later date. This feature allows you to customize CSA Files for different applications and to reuse the files as needed.

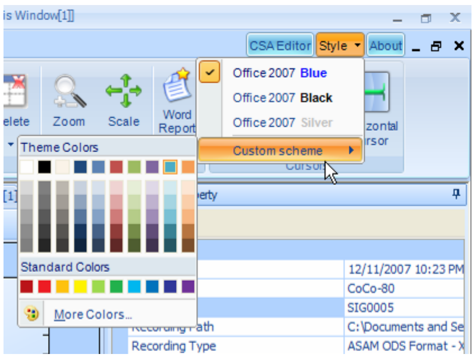


- Figure 86. CSA Files can be downloaded to the PC from the EDM Home Page.

10. Style and About Tabs

Style Tab

The Style Tab defines the general appearance of the EDM software. You can change the color scheme of the software to blue, black or silver or define your own custom color scheme by picking a color from the Theme Color menu. The color scheme is applied to all pages, Views, Panes menus, etc throughout the software.



■ Figure 87. Style Tab.

About Tab

The About Tab displays information about the EDM software including version number, and contact information for Crystal Instruments. To close the About display click anywhere on the About Display. You can click on the website link to open the Crystal Instruments website in your Internet browser. You can click on the E-Mail address to launch your email editor and send an email to Crystal Instruments

■ Figure 88. About Box displays software version and contact information.

11. Data Export Formats

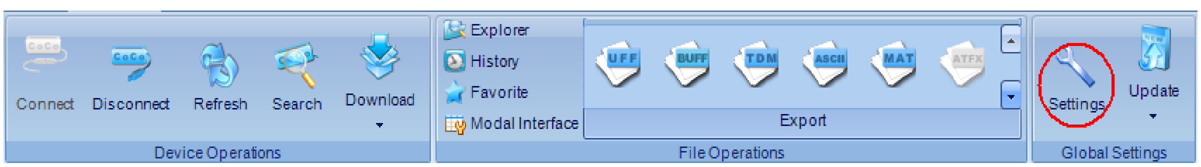
This section describes the process for exporting data from EDM in the ODS format to other software in any of the supported export data formats.

EDM uses the ODS data format as the default internal format. It is recommended that data be maintained in the ODS format unless the data must be shared with an application that is not compatible with this format. ODS is a format that has wide acceptance in the automotive industry and is being adopted by other industries. An ODS file includes meta information, or information pertaining to the data, including environment, dimensions and units, administration, description and security. All meta information is stored together with the actual data in most cases. The standard also allows meta information to be stored in a separate file when the data file is very large and requires a more compact binary data format.

However if data must be shared with another system of software that does not support the ODS data format then EDM can export signals in the following formats: UFF, BUFF, ASCII UFF, MATLAB file, NI-TDM file, User-Defined ASCII file, and .CSV (MS EXCEL).

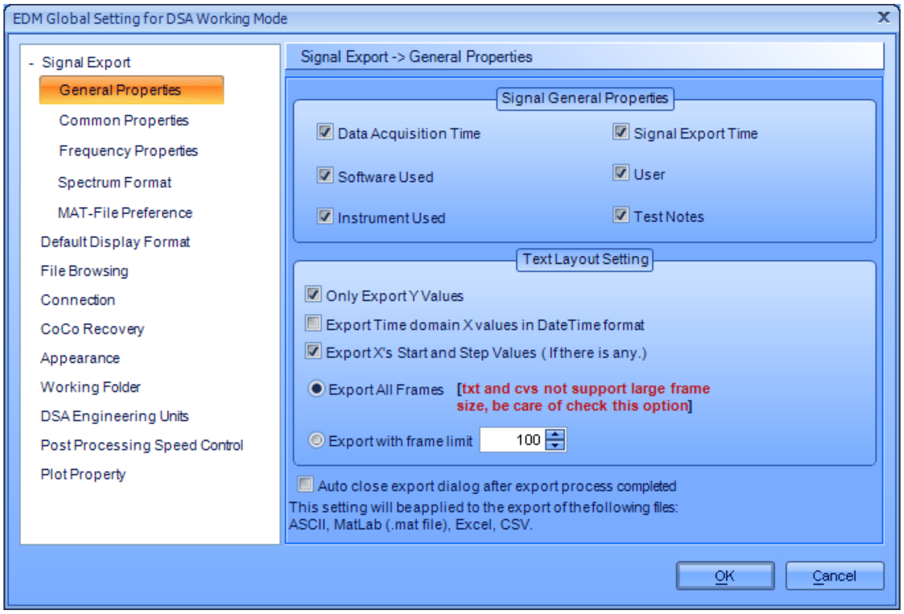
Settings for Signal Export

It is important to choose the attributes to include in the exported data files before exporting data. Open the Settings View from the Home Ribbon.



- Figure 89. Signal Export Settings are accessed from the Setting dialog.

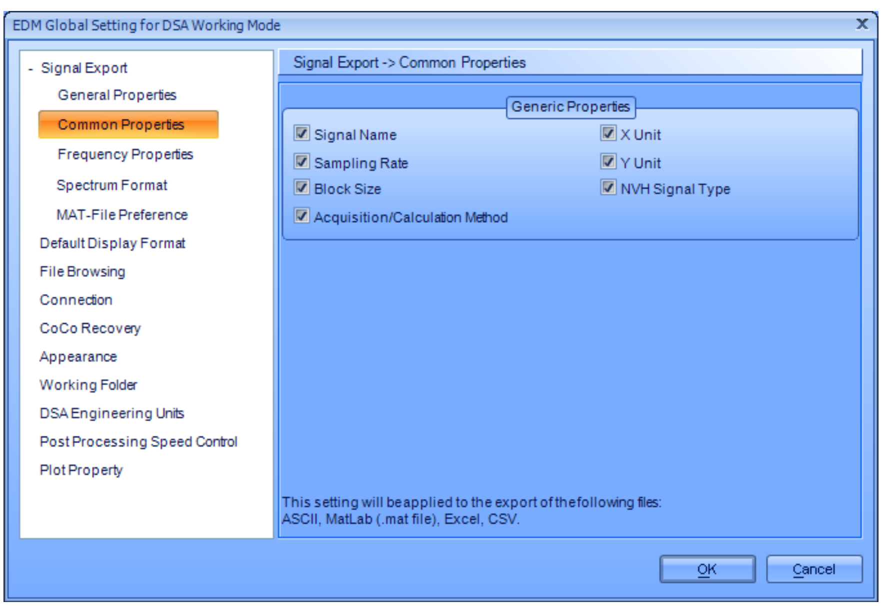
The Settings View includes three sub-tabs under Signal Export. These settings can be applied to User-Defined ASCII, MATLAB, and EXCEL CSV files. These settings do not apply to UFF, BUFF, and NI-TDM because these formats have their own settings that cannot be modified.



■ Figure 90. EDM Settings dialog with Signal Export General Properties Settings.

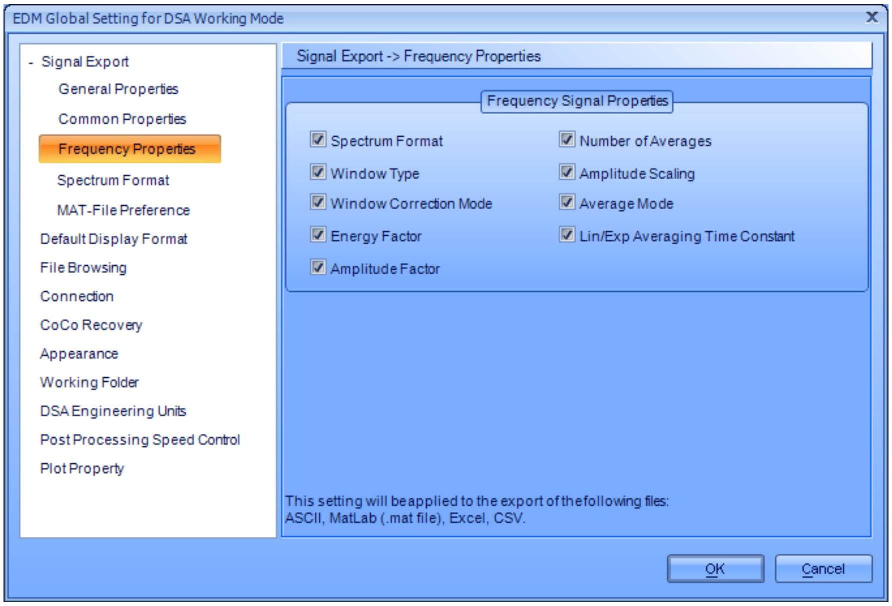
Signal General Properties – defines which of the attributes will be included in the exported data files including: Data Acquisition Time, Software Used, Instrument Used, Signal Export Time, User and Test Note.

Text Layout Settings – defines the layout of text in the data file including: Only Export Y Values or Export X Value Start and Step Value.



■ Figure 91. EDM Settings Dialog with Signal Export Common Properties Settings.

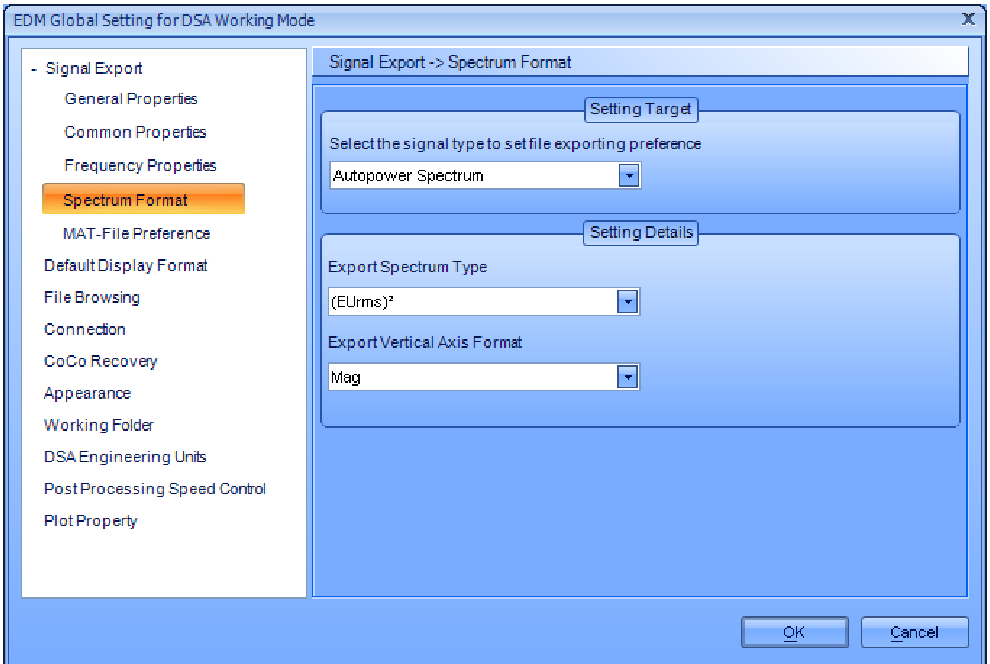
Generic Properties – defines which of the following will be included in the exported data file including: Signal Name, Sampling Rate, Block Size, X Unit, Y Unit and NVH Signal Type.

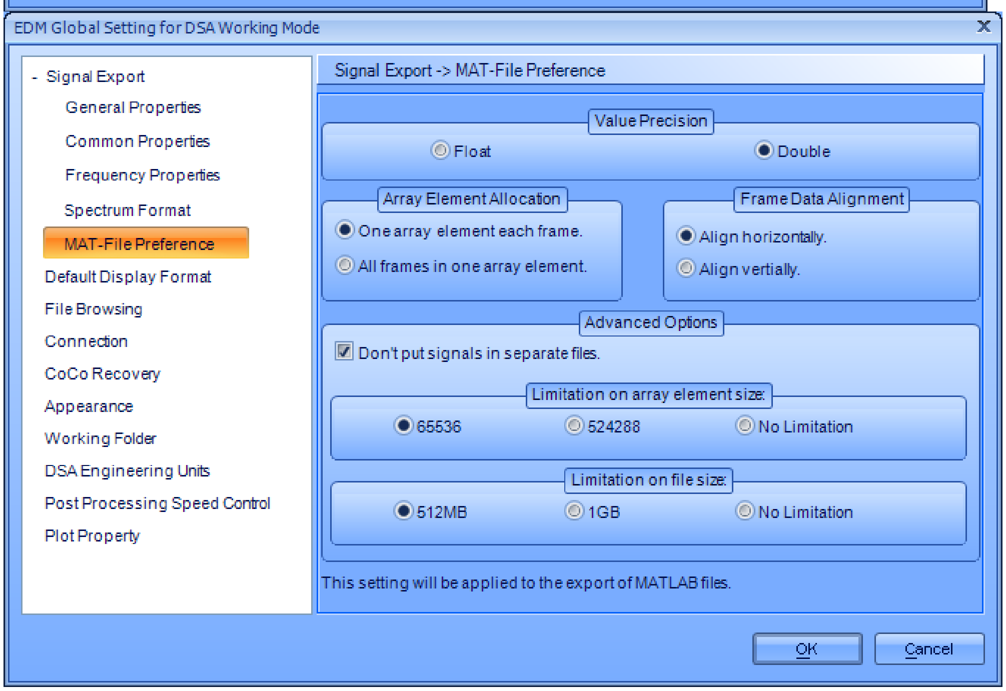


■ Figure 92. EDM Settings Dialog with Signal Export Frequency Properties Settings.

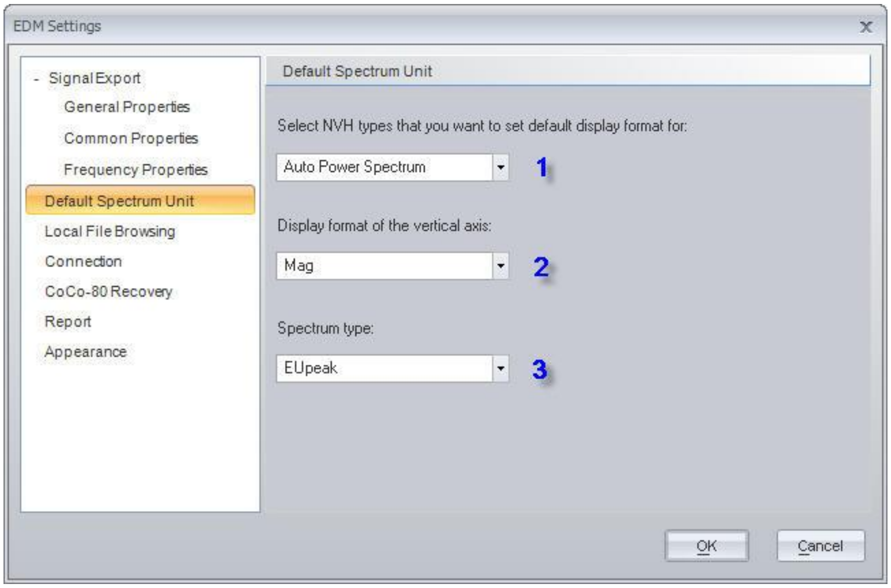
Frequency Signal Properties – defines which of the following attributes will be included with the exported frequency signal including:

- Spectrum Format
- Window Type
- Window Correction Mode
- Energy Factor
- Amplitude Factor
- Acquisition/Calculation Factor
- Amplitude Scaling
- Averaging Mode
- Lin/Exp Averaging Time Constant
- Number of Averages





Default Spectrum Unit Settings – defines the type of spectrum type to apply to NVH Pane. The NVH Pane is a special frequency Pane that automatically includes the spectrum type, vertical axis format and scaling. This is a convenient tool for quickly creating Panes with a consistent format.



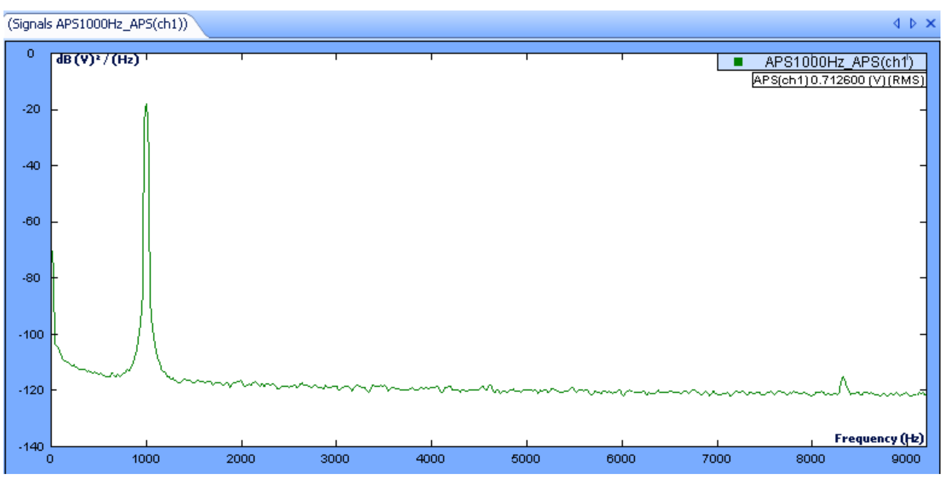
■ Figure 93. EDM Settings Dialog with Default Spectrum Unit Settings.

NVH Type – select one of the frequency bases spectra including: auto power spectrum, coherence, complex spectrum, cross power spectrum and frequency response spectrum.

Vertical Axis Format – choose dBMag or Mag

Spectrum type – choose $(EU)^2$/Hz, $(EU)^2$s/Hz, $(EUrms)^2$, EUpeak or EUrms. The NVH data files exported into ASCII, CSV, Excel and MATLAB will be in the values using the default spectrum unit. This notion is critical because it means that an NVH signal exported in different spectrum units will have different values.

EU^2/Hz - A signal displayed in this spectrum type is power spectrum density (PSD). It provides *power* normalized to a 1 Hz bandwidth and is useful for *wideband, continuous signals*.



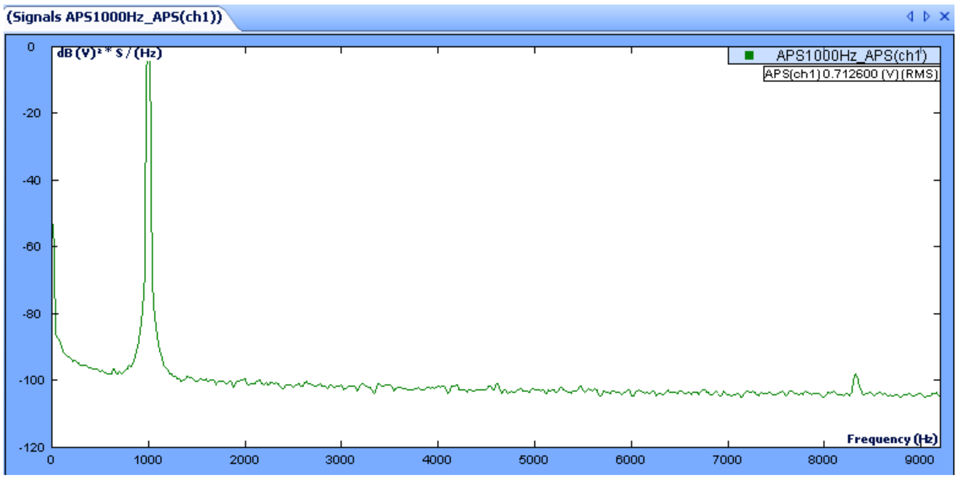
■ Figure 94. Example of a power spectral density signal.

The relationship between the band height, bandwidth and RMS is given by

Band height * Bandwidth = RMS^2

From the above signal, we can see the bandwidth is approximately 10,000 Hz and the height of the band is around 0.0001 $(EU)^2$/Hz. The area under the curve should be $(1\ volt)^2$ because the input signal is white noise with **1** volt RMS.

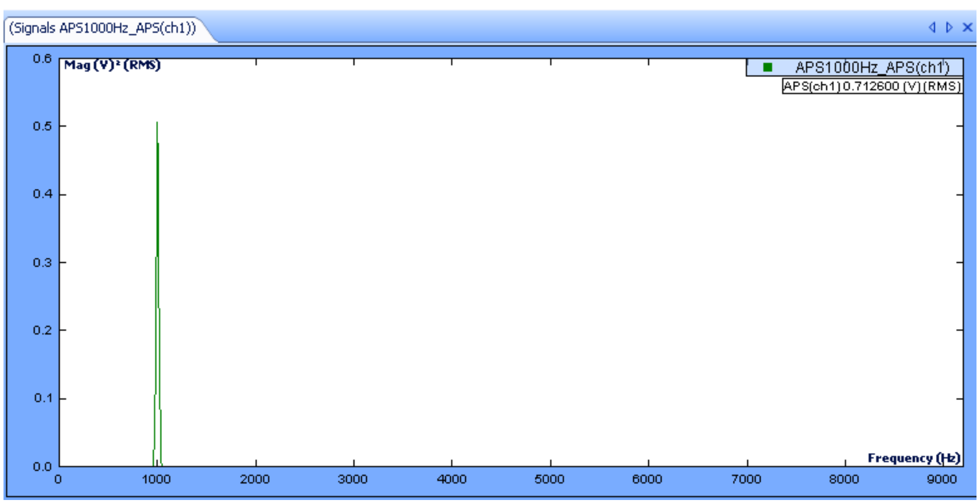
EU^2S/Hz - A signal displayed in this spectrum type is energy spectrum density (ESD). It provides *energy* normalized to a 1 Hz bandwidth and is useful for *wideband, continuous signals, or transient signals*.



■ Figure 95. Example of an energy spectrum density signal.

Values for ESD = values of PSD * Time Period of Data Capture.

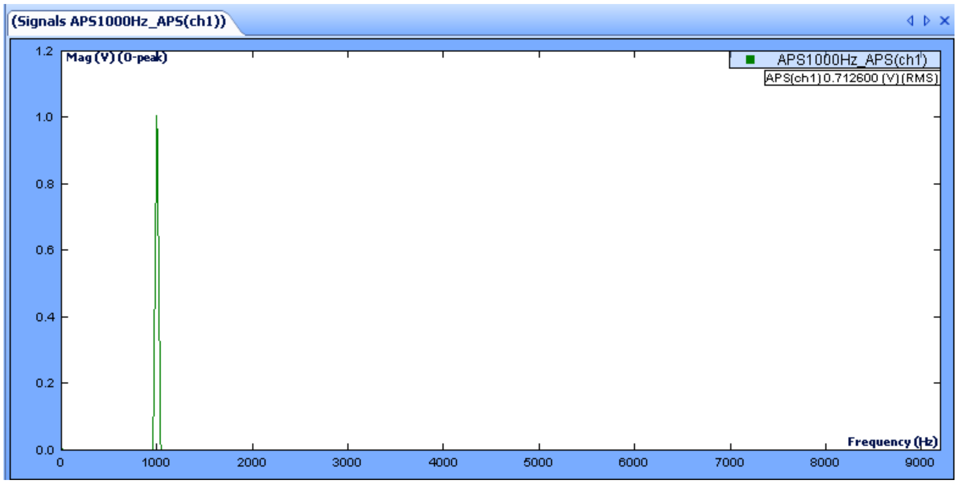
$EUrms^2$ - A signal displayed in this spectrum type is power spectrum. It is useful for measuring the *power of narrow band signals*.



■ Figure 96. Example of a power spectrum signal.

The peak value should be **0.5** volt because the sine wave with 1 volt amplitude is the input.

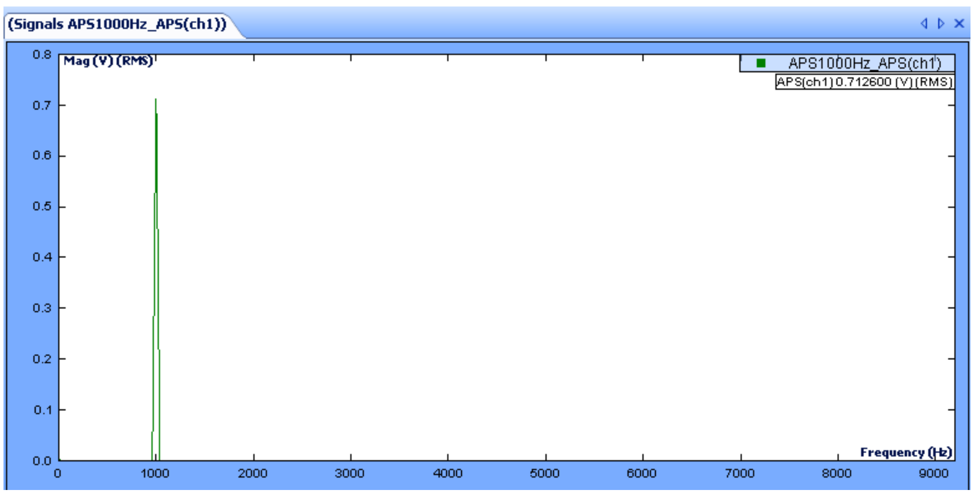
EUpeak - A signal displayed in this spectrum type is linear spectrum. It is useful for *analyzing periodic signals, or narrow band signals*.



■ Figure 97. Example of a linear spectrum signal with EUpeak units.

The peak value should be **1** volt because the sine wave with 1 volt amplitude is the input.

EUrms - A signal displayed in this spectrum type is also linear spectrum. It is useful for *analyzing periodic signals, or narrow band signals*.



■ Figure 98. Example of a linear spectrum with EUrms units.

The peak value should be **0.7071** volt because the sine wave with 1 volt amplitude is the input.

Signal Export

After the Signal Export Settings have been selected the next step is to select one or more files and export them in the desired format. This section describes how to export signals in the supported export formats.

To export one or more signals select the desired signal(s) on the Home Page in the Local File View and click on one of the export icons in the Export section of the File Operations Ribbon. If all export formats are not visible you can use the up and down arrows next the ribbon to scroll though the list. After you click on the desired format icon a dialog box opens prompting you to choose a name and location to export the file. Next, click on the Save Button to complete the export process. A message indicates that the file was exported successfully.

Data Formats

This section gives some information about the format for the export formats.

Universal File Format (UFF)

The CoCo-80 and EDM Software support the Universal File format (UFF). This format was originally developed by the Structural Dynamics Research Corporation (SDRC) in the late 1960s and early 1970s to facilitate data transfer between computer aided design (CAD) and computer aided test (CAT) in order to facilitate computer aided engineering (CAE). SDRC, as part of EDS, continues to support and utilize the UF formats as part of their CAE software. Currently, MTS, Noise and Vibration Division supports and continues to develop IDEAS software in the test area that utilizes UF formats. The formats were originally developed as 80 character (card image), ASCII records that occur in a specific order according to each UF format. As computer files became routinely available, single UF formats were concatenated into computer file structures. Recently, a hybrid UF file structure (UF Dataset 58 Binary) was developed for experimental data that allows data to be stored in a more efficient binary format. Before the introduction of ASAM ODS, the use of the Universal File Format as a de-facto "standard" has been of great value to the experimental dynamics (vibration and acoustic) community, particularly in the area of modal analysis. Both users and vendors have benefited from this de-facto standard.

The EDM software can export the data in UFF (Dataset 58) and BUFF (Dataset Binary 58). For more information on UFF refer to http://www.sdrl.uc.edu/uff/uff.html.

ASCII UFF

The basic (ASCII) universal file format for data is universal file format 58. This format is completely documented by SDRC and a copy of that documentation is on the UC-SDRL web site (www.sdrl.uc.edu/UFF2/58.asc). The universal file format always begins with two records that are prior to the information defined by each universal file format and ends with a record that is placed after the information defined by the format. All records are 80 character ASCII records for the basic universal file format. The first and last record are start/stop records and are always -1 in the first six columns, right justified (Fortran I6 field with -1 in the field). The second record (Identifier Record) always contains the universal file format number in the first 6 columns, right justified.

This gives a file structure as follows (where b represents a blank character):

```
bbbb-1
bbbb58
...
...
...
bbbb-1
```

■ Figure 99. Basic structure for a ASCII UFF format file.

Binary 58 Universal File Format (BUFF)

The Binary 58 universal file format was originally developed by the UC-SDRL in order to eliminate the need to compress the UFF 58 records and to reduce the time required to load the UFF 58 data records. The Binary 58 universal file format yields files that are comparable to compressed files (approximately 3 to 4 times smaller than the equivalent UFF 58 file). The Binary 58 universal file format loads approximately 30 to 40 times faster than the equivalent UFF 58 file, depending upon the computing environment. This format was submitted to SDRC and subsequently adopted as a supported format.

The Binary 58 universal file format uses the same ASCII records at the start of each data file as the ASCII dataset 58 but, beginning with record 12, the data is stored in binary form rather than the specified ASCII format. The identifier record has the same 58 identifier in the first six columns, right justified, but has additional information in the rest of the 80 character record that identifies the binary format (the size of the binary record, the format of the binary structure, etc.).

```
    -1
    58b    x     y        11       2222     0    0         0          0
...
... (11 ASCII header lines) (UFF and BUFF have the identical header.)
...
...
... (zzzz BINARY bytes of data, in format specified by x and y, above)
... (interleaved as specified by the ASCII dataset 58)
...
    -1
Format  (I6,1A1,I6,I6,I12,I12,I6,I6,I12,I12)
        Field 1        - 58 [I6]
        Field 2        - lowercase b [1A1]
        Field 3        - Byte Ordering Method [I6]
                     1 - Little Endian (DEC VMS & ULTRIX, WIN NT)
                     2 - Big Endian (most UNIX)
        Field 4        - Floating Point Format [I6]
                     1 - DEC VMS
                     2 - IEEE 754 (UNIX)
                     3 - IBM 5/370
        Field 5        - number of ASCII lines following [I12]
                     11 - for dataset 58
        Field 6        - number of bytes following ASCII lines [I12]
        Fields 7-10      - Not used (fill with zeros)
```

■ Figure 100. Format of a Binary UFF format file.

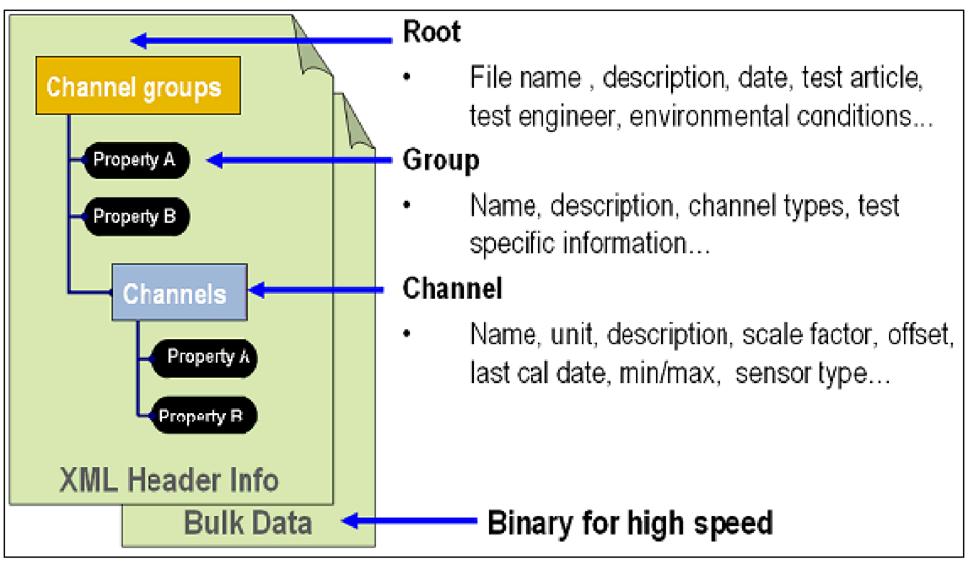
When reading or writing a dataset 58b, care must be taken that the binary data immediately follows the ASCII header lines and the closing '   -1' immediately follows the binary data. The binary data content is written in the same sequence as the ASCII dataset 58 (i.e. field order sequence). The field size is NOT used, however the data type (int/float/double) content is. Note: there is no CR/LF character embedded in or following the binary data

NI-TDM Format

NI TDM (Technical Data Management) file format is a structured, search-ready format that utilizes XML to manage a wide range of attributes associated with each data file. The TDM format is designed to capture and manage all the important information surrounding a measurement or simulation, assuring that the data is self explanatory, reusable, and requires no additional explanation to recreate the conditions in which it was captured.

The TDM data model defines 3 levels of hierarchy, as the Root, Group, and Channel objects. Each file has a single root object, and any number of group and channel objects, each of which can have an unlimited number of user-defined, scalar properties. Each Channel additionally contains a 1D array of data values. Both the properties and the data arrays are stored in your choice of a variety of standard data types. For example in one file you may choose to save the original data in group 1 named “raw data”, and create a second group called “analyzed results” to separate the analyzed results from the original data.

The TDM file format offers several unique benefits such as the ability to scale to your specific project requirements and easily attach descriptive information to your test or simulation data. The use of XML makes TDM files easily searchable and portable between applications.



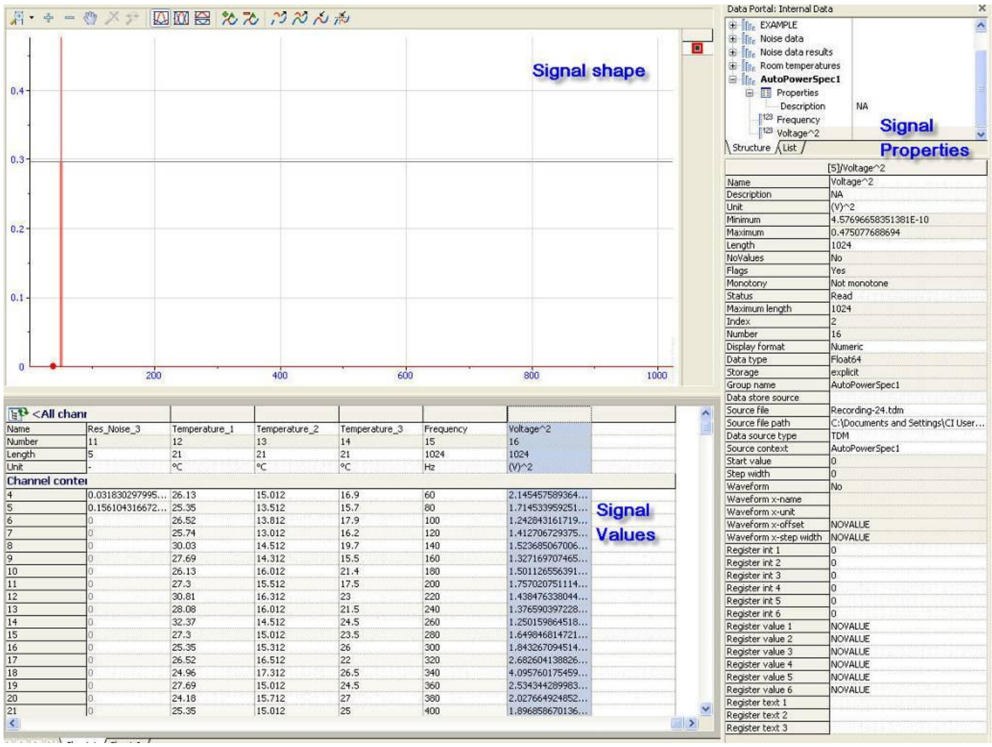
■ Figure 101. NI TDM Data Format.

This is a structured data format that is defined and widely used by the LabView from National Instruments.

For more information, please visit:

http://zone.ni.com/devzone/cda/tut/p/id/3727
http://zone.ni.com/devzone/cda/tut/p/id/3542
http://zone.ni.com/devzone/cda/tut/p/id/2824

An example of exported NI-TDM file:



■ Figure 102. Example of a NI TDM Data File.

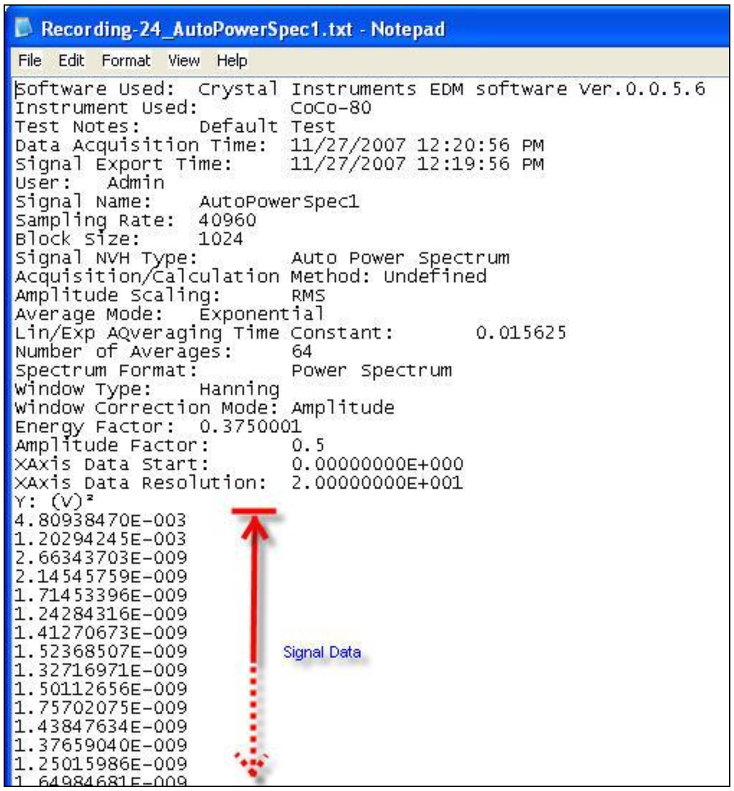
User-Defined ASCII Format

ASCII is the abbreviation for American Standard Code for Information Interchange. ASCII data files contain only the values of variables stored in ASCII format and are sometimes called raw data files because they contain only data and no file attribute information. That is, no variable definition information is included in a raw data file. The ASCII file can include a header with some information but this information can only be used by software that is programmed to expect the header information in a specific format.

User-Defined ASCII files allow users to define its attributes and header format in the Signal Export Settings. The information selected in the Export Settings will appear at the top of the ASCII file followed by the signal data.

For more information about this format, please visit:

http://web.uccs.edu/lbecker/SPSS80/ascii.htm



Recording-24_AutoPowerSpec1.txt - Notepad

File Edit Format View Help

```
Software Used:  Crystal Instruments EDM software Ver.0.0.5.6
Instrument Used:        CoCo-80
Test Notes:     Default Test
Data Acquisition Time:  11/27/2007 12:20:56 PM
Signal Export Time:     11/27/2007 12:19:56 PM
User:   Admin
Signal Name:    AutoPowerSpec1
Sampling Rate:  40960
Block Size:     1024
Signal NVH Type:        Auto Power Spectrum
Acquisition/Calculation Method: Undefined
Amplitude Scaling:      RMS
Average Mode:   Exponential
Lin/Exp AQveraging Time Constant:       0.015625
Number of Averages:     64
Spectrum Format:        Power Spectrum
Window Type:    Hanning
Window Correction Mode: Amplitude
Energy Factor:  0.3750001
Amplitude Factor:       0.5
XAxis Data Start:       0.00000000E+000
XAxis Data Resolution:  2.00000000E+001
Y: (V)²
4.80938470E-003
1.20294245E-003
2.66343703E-009
2.14545759E-009
1.71453396E-009
1.24284316E-009
1.41270673E-009
1.52368507E-009
1.32716971E-009
1.50112656E-009
1.75702075E-009
1.43847634E-009
1.37659040E-009
1.25015986E-009
1.64984681E-009
```


■ Figure 103. Example of a User-Defined ASCII Data File.

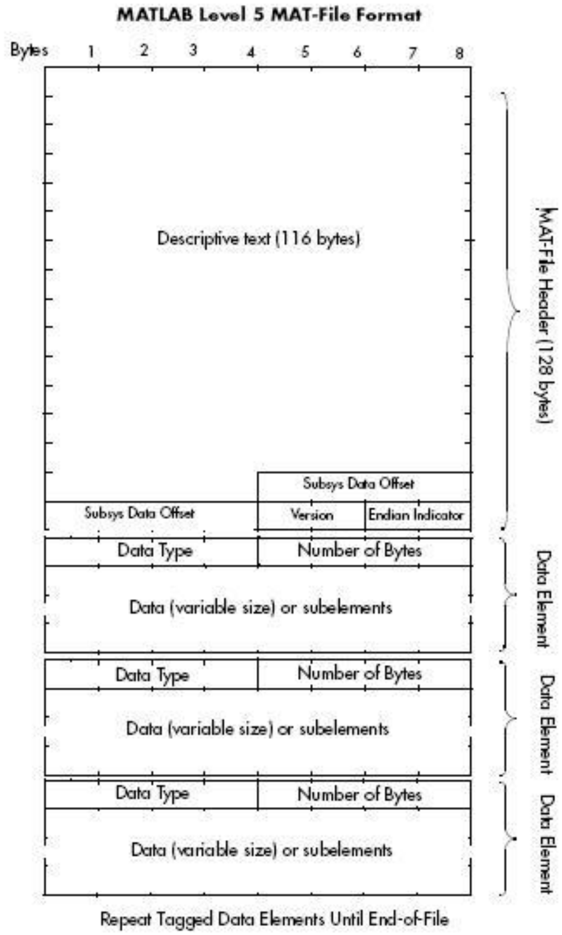
MATLAB Format

There are two different formats of MATLAB files. They are Level 4 and Level 5 MAT-files. Level 4 MAT-files are compatible with versions of MATLAB up to Version 4. Level 5 MAT-files are compatible with MATLAB version 5 and up. ***EDM only exports signals in Level5 MAT-files.***

A MAT-file stores data in binary form with the extension .mat. Files saved in this format can be opened and loaded directly into the MATLAB workspace.

Level 5 MAT-files are made up of a 128-byte header followed by one or more data elements. Each data element is composed of an 8-byte tag followed by the data in the element. The tag specifies the number of bytes in the data element and how these bytes should be interpreted; that is, should the bytes be read as 16-bit values, 32-bit values, floating point values or some other data type.

By using tags, the Level 5 MAT-file format provides quick access to individual data elements within a MAT-file. You can move through a MAT-file by finding a tag and then skipping ahead the specified number of bytes until the next tag.



- Figure 104. MAT Data File Structure.

MAT-file Header Format - Level 5 MAT-files begin with a 128-byte header made up of a 124 byte test field and two, 16-bit flag fields

Header Text Field - The first 116 bytes of the header can contain text data in human-readable form. This text typically provides information that describes how the MAT-file was created. For example, MAT-files created by MATLAB include the following information in their headers: Level of the MAT-file (value equals 1 for Level 5), Platform on which the file was created, and Date and time the file was created.

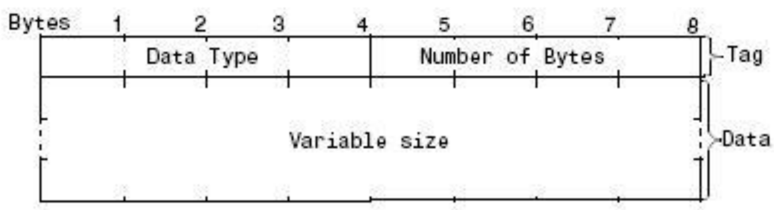
Header Subsystem Data Offset Field - Bytes 117 through 124 of the header contain an offset to subsystem-specific data in the MAT-file. All zeros or all spaces in this field indicate that there is no subsystem-specific data stored in the file.

Header Flag Fields - The last four bytes in the header are divided into two, 16-bit flag fields.

Field	Value
Version	When creating a MAT-file, set this field to 0x0100.
Endian Indicator	Contains the two characters, M and I, written to the MAT-file in this order, as a 16-bit value. If, when read from the MAT-file as a 16-bit value, the characters appear in reversed order (IM rather than MI), it indicates that the program reading the MAT-file must perform byte-swapping to interpret the data in the MAT-file correctly.

Data Element Format - Each data element begins with an 8-byte tag followed immediately by the data in the element.

MAT-File Data Element Format



■ Figure 105. MAT Data Element Format.

The Tag Field - The 8-byte data element tag is composed of two, 32-bit fields:

Data Type - The Data Type field specifies how the data in the element should be interpreted, that is, its size and format. The MAT-file format supports many data types including signed and unsigned, 8-bit, 16-bit, 32-bit, and 64-bit data types, a special data type that represents MALAB arrays, Unicode encoded character data, and data stored in compressed format.

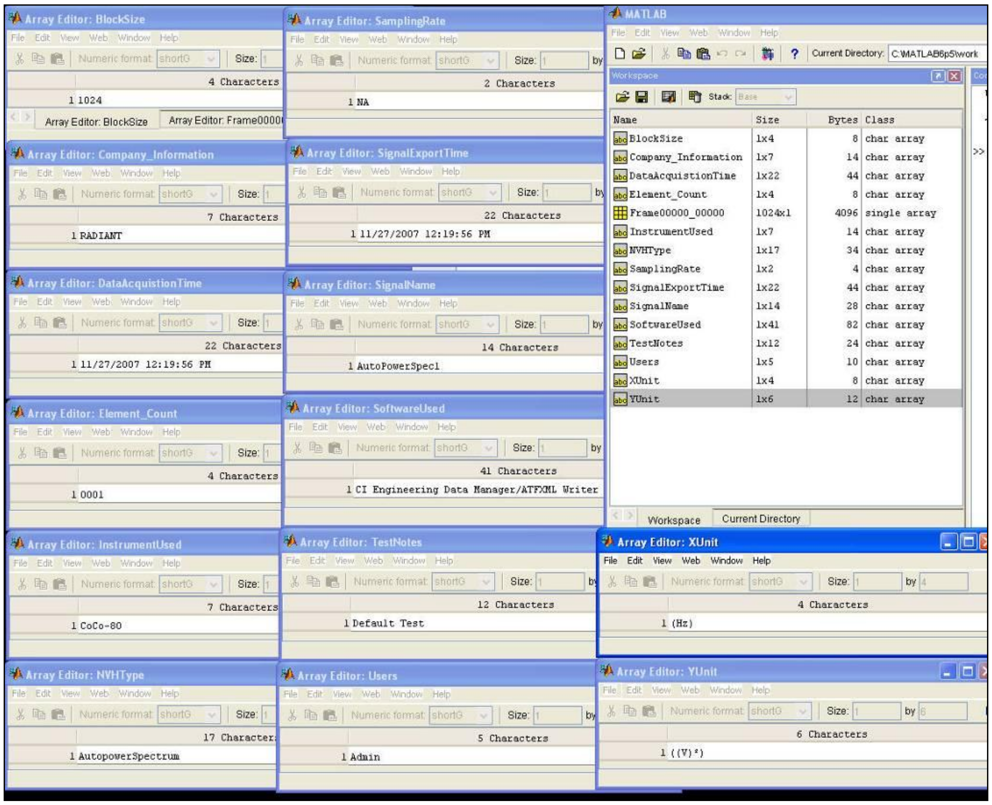
Number of Bytes - The number of Bytes field is a 32-bit value that specifies the number of bytes of data in the element. This value does not include the eight bytes of the data element's tag.

The Data Field - The data immediately follows the tag.

For more detail information, please visit

http://www.mathworks.com/access/helpdesk/help/pdf_doc/matlab/matfile_format.pdf

An example of exported MAT-file (MATLAB) that as been loaded into the MATLAB workspace is shown below:



- Figure 106. Example of a MAT Data File.

CSV (MS EXCEL) Format

CSV, which is the abbreviation of Comma Separated Value, is a format used primarily to transfer basic data between databases and spreadsheets. Each line is considered a record. Fields within each record are divided by a comma. Each line must have the same number of fields (commas). If a comma or leading and/or trailing blanks appear in any field value, the field must be enclosed by quotes (") to indicate the information is data and not a field divider.

This file format, as it is used in MS EXCEL, has become a pseudo standard throughout the industry, even among non-Microsoft platforms. In some cases, particularly if the CSV file is output from a database, there will be some header rows that start the file. These header rows will contain information about the database. The information defined in the Signal Export Settings will appear at the top of the CSV file followed by the signal data.

For more information, please visit:

http://www.creativyst.com/Doc/Articles/CSV/CSV01.htm
http://answers.google.com/answers/threadview?id=379536
http://docs.python.org/lib/module-csv.html

An example of exported CSV file:

Recording-24_AutoPowerSpec1.csv

	A	B	C
1	Software Used:	Crystal Instruments EDM software Ver.0.0.5.6	
2	Instrument Used:	CoCo-80	
3	Test Notes:	Default Test	
4	Data Acquisition Time:	11/27/2007 12:20	
5	Signal Export Time:	11/27/2007 12:19	
6	User:	Admin	
7	Signal Name:	AutoPowerSpec1	
8	Sampling Rate:	40960	
9	Block Size:	1024	
10	Signal NVH Type:	Auto Power Spectrum	
11	Acquisition/Calculation Method:	Undefined	
12	Amplitude Scaling:	RMS	
13	Average Mode:	Exponential	
14	Lin/Exp AQveraging Time Constant:	0.015625	
15	Number of Averages:	64	
16	Spectrum Format:	Power Spectrum	
17	Window Type:	Hanning	
18	Window Correction Mode:	Amplitude	
19	Energy Factor:	0.3750001	
20	Amplitude Factor:	0.5	
21	XAxis Data Start:	0.00E+00	
22	XAxis Data Resolution:	2.00E+01	
23	Y: (V)Â²		
24	4.81E-03		
25	1.20E-03		
26	2.66E-09		
27	2.15E-09	Signal Data	
28	1.71E-09		
29	1.24E-09		
30	1.41E-09		
31	1.52E-09		
32	1.33E-09		

■ Figure 107. Example of a CSV Data File.

Sound .WAV Format

This is the sound wave files that can be played by most of the media players. Due to limited information a wave file can carry, the wave files exported only contain very basic waveform shape and it does not hold any attribute information of ODS. The user is expected to use the .WAV file to listen to its sound effect, instead of for data processing.

12. Index

13. Version History

Date	Version	Comments
12/12/07	0.1	First Draft by CB, incorporated Data Export White Paper.
2/2/08	0.30	Added the definition of Data Conditioning Module
12/2008	0.91	Added Post Processing
1/2010	1.00	Added Spider quick manual

EDM Users Guide Typeface

Headings Arial Black 12 and 11 pt

Body Text: Arial 10 pt

Captions: Arial Narrow 9 pt