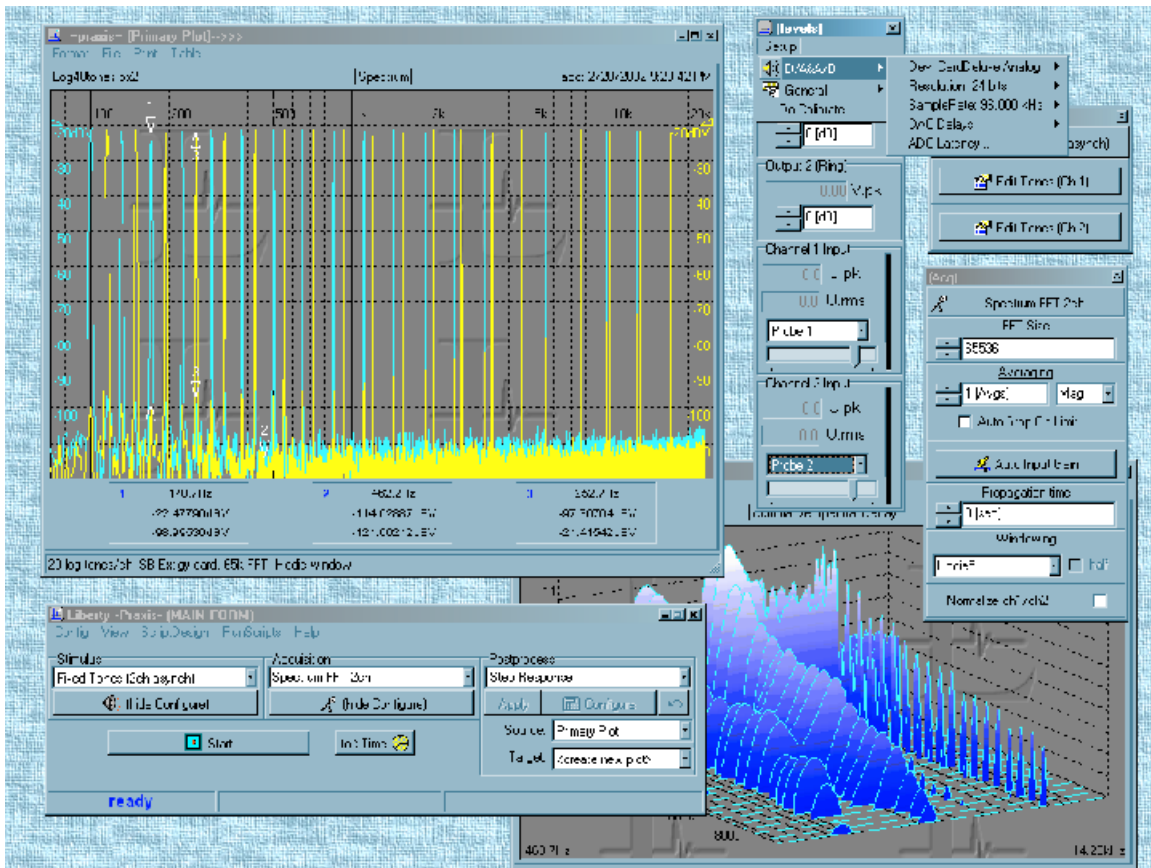


# praxis v2.2



January, 2005

Copyright © 2001-2005 Liberty Instruments, Inc. All rights reserved.

## Contents

---

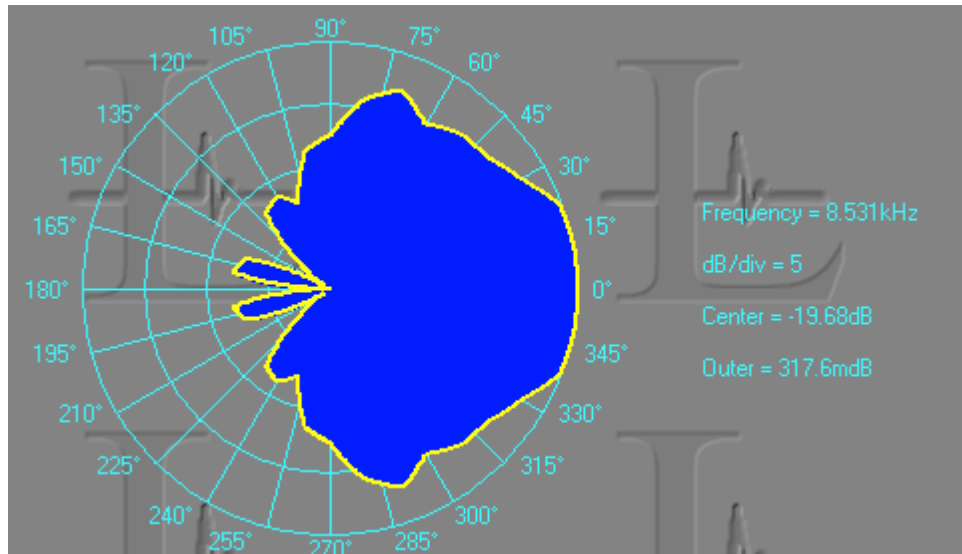
Praxis Description .....	6
Capabilities .....	7
Hardware and Software Requirements .....	8
Praxis Program Design Considerations .....	9
How to purchase an AudPod for Praxis "Full" operation .....	11
Working with Praxis .....	12
Installation.....	12
Connections.....	14
About "Windows Mixers" .....	16
Getting Help.....	17
Installing and Zipping Up Script Files .....	19
The Stimulus .....	21
Stimulus Parameters .....	21
Envelopes.....	22
Stimulus Types .....	22
What are Synchronous, Asynch, ReSync Stimuli?.....	23
Chirp (ReSync) Stimulus .....	25
MLS and MLS/Pink (ReSync) Stimuli .....	26
Chirp (Synchronous) Stimulus.....	27
Impulse (Synchronous) Stimulus.....	27
MLS and MLS/Pink (Synchronous) Stimuli .....	27
Fixed Tone (Synchronous) Stimulus .....	28
Fixed Tones (Asynch) Stimuli.....	28
The Waveform Component Editor.....	28
Stepped Tones (Synchronous) Stimulus.....	30
Stepped Tones (Asynch) Stimuli .....	31
The SteppedTone Editor .....	32
The List File Editor .....	33
Frequency List Files .....	33
WAV File (Synchronous and Asynchronous) Stimuli .....	35
White Noise and Pink Noise (Asynch) Stimuli.....	35
Wavelet Burst (Synchronous) Stimulus.....	35
The Acquisition .....	36
Acquisition Parameters.....	36
Windowing.....	37
Automatic Input Adjustment .....	39
Find Window Edges Button .....	39
Averaging .....	40
Auto Stop.....	41
Propagation Time .....	41
Triggering .....	42
Isolation of Distortion.....	42
Acquisition Types .....	44
Time Domain (1ch and 2ch) Acquisitions.....	44
Impulse Response via MLS Acquisition .....	45
Impulse Response via Chirp Acquisition .....	45
Spectrum FFT (1ch and 2ch) Acquisitions .....	45
Spectrum RTA (1ch and 2ch) Acquisitions.....	46
Normalize ch1/ch2 and the Coherence Function .....	47

Frequency Response Mag and Phase Acquisition.....	48
Freq Response Mag (1ch and 2ch) Acquisitions.....	49
Freq Response Live Monitoring Acquisitions .....	49
Impulse Response Live Monitoring Acquisitions .....	50
Impedance Acquisition .....	51
Component Track (1ch and 2ch) Acquisitions.....	52
The Component Tracking Configuration Editor.....	52
Frequency Response and IMD Acquisitions .....	54
Spectrum 2ch FFT Complex Acquisition .....	55
Wavelet Maximum Level Capacity .....	55
Wavelet Single Frequency Acquisition .....	56
Impulse Response Asynch (1ch or 2ch) Acquisition .....	56
Post Processing Operations .....	56
PostProcess Parameters.....	56
PostProcess Types.....	57
PostProcess: ReMap to New Frequencies.....	58
PostProcess: Filter .....	58
PostProcess: Math .....	59
PostProcess: Paste Over .....	59
PostProcess: Smoothing .....	60
PostProcess: FFT .....	60
PostProcess: Inverse FFT .....	60
PostProcess: Thiele/Small Parameter Extraction.....	60
PostProcess: CSD Waterfall.....	61
PostProcess: Step Response .....	62
PostProcess: Hilbert Transform.....	62
PostProcess: Schroeder Curve and Room Acoustics .....	63
PostProcess: Group Delay .....	63
PostProcess: Polar Compilation .....	64
PostProcess: Combine FreqResponse and Impedance .....	65
PostProcess: Analytic Signal (ETC) .....	65
PostProcess: Power Cepstrum.....	65
PostProcess: Quasi-Anechoic Bass Response Restoration .....	65
PostProcess: Progressive Averaging .....	66
PostProcess: Tail Correction .....	66
Postprocess: Time Domain Math .....	66
Postprocess: Time Domain Channels .....	67
Postprocess: Time Domain Length .....	68
Using Scripts in Praxis .....	69
Praxis Scripts.....	69
The Script Launcher .....	71
Forms Used to Control Praxis.....	71
Praxis Forms.....	71
Using Praxis' Numerical Value Controls.....	72
Engineering Notation .....	73
The Main Form .....	74
(MainMenu) Config   Load Config File .....	74
(MainMenu) Config   Save As .....	74
(MainMenu) Config   Use Startup Config File .....	74
(MainMenu) Config   Hardware Specific .....	74
(MainMenu) Config   Preference   Units.....	75
(MainMenu) Config   Preference   Backdrop.....	75
(MainMenu) Config   Preference   Backdrop.....	76
(MainMenu) Config   Mixer .....	76
(MainMenu) Config   Show Hints .....	76
(MainMenu) Config   Auto Acquisition Start Timer.....	76

(MainMenu) View .....	77
(MainMenu) ScriptDesign.....	77
(MainMenu) Config   RunScripts .....	77
(MainMenu) Help.....	77
Stimulus Controls .....	77
Acquisition Controls.....	77
PostProcess Controls.....	78
The Start/Stop Button.....	78
The Time/Frequency Button.....	78
The Levels Form.....	79
(Levels Menu) Setup   D/A&A/D.....	79
(Levels Menu) Setup   General .....	80
(Levels Menu) Setup   Do Calibrate .....	80
(Levels Menu) Setup   Check Calibration.....	80
(Levels) Output Controls .....	80
(Levels) Input Controls .....	81
Plot Forms .....	81
The Primary Plot.....	84
Secondary Plots .....	84
(Plot Menu) Format .....	84
(Plot Menu) File.....	84
Plot Format Files .....	85
(Plot Menu) Print .....	85
(Primary Plot Menu) Clone.....	85
(Primary Plot Menu) Hold.....	85
(Plot Menu) ClearMulti.....	86
(Plot Menu) Table.....	86
(Plot Menu) Notes .....	86
Plot Markers .....	86
Formatting Plots .....	87
Auto Scaling .....	87
The Time Format Tab.....	88
The Frequency Format Tab .....	88
The Traces Format Tab .....	89
The VScale Format Tab .....	89
The VRef Format Tab .....	90
The Markers Format Tab .....	90
The Windowing Format Tab.....	90
The Weight Format Tab .....	91
The Delay Format Tab .....	92
The Weight File Editor.....	92
The Title Format Tab.....	92
The Colors Format Tab .....	92
The "3D" Format Tab .....	93
Configuration Forms ("Configure Forms") .....	95
Files Used in Praxis .....	95
Praxis File Types .....	95
Praxis Configuration Files.....	95
Measurement Considerations.....	96
Measurement Channel, Cal Channel .....	96
Computer Resource Availability .....	97
Measurement Units .....	97
Time, Frequency, Size ...and Noise .....	98
Processing Times .....	99
Chirp vs. MLS vs. Stepped Sine.....	99
Level Adjustments (Free/Demo Mode).....	100

Level Adjustments (AudPod Mode).....	100
Multitone Testing .....	101
Pass/Fail Evaluation .....	102
Praxis Operating Modes .....	103
Free/Demo Mode.....	103
Probes (Free/Demo Mode) .....	103
RTA Operation in Demo Mode.....	105
AudPod (Full) Mode.....	106
AudPod Description.....	106
USB (Universal Serial Bus) .....	107
Probes (Full or AudPod Mode).....	108
Microphones .....	109
Credits and Tools Used to make Praxis .....	112

## Praxis Description



**PRAXIS** is a versatile audio measurement system intended for professional use. Utilizing the high performance data acquisition hardware of quality modern computer soundcards, Praxis offers time-coherent, wide band, and highly customizable measurement capabilities for research, design, production control and system verification. It can provide full complex frequency response data, quasi-anechoic measurements, impulse responses, true complex impedance, room characteristics, toneburst measurements including peak SPL power capability, tracking distortion of HD, THD, or various orders of IM, transducer polarity measurement, sound pressure, RTA, single or dual channel spectra, triggered measurements, polar response measurements, Thiele/Small parameters, and a vast range of other capabilities. All these features are further enhanced with Praxis' built-in scripting capabilities, allowing Praxis to be programmed using DelphiScript or VBScript for custom measurements or programmable operations.

Through use of the Praxis USB "**AudPod**" key/interface device, Praxis overcomes the limitations encountered by other measurement systems that have attempted to use soundcards for sophisticated measurements. And by operating with current and ever-improving future soundcard technology, Praxis will work with future computer systems and is not constrained by any requirement for an existing custom hardware board or an obsolete hardware expansion architecture or bus adapter.

The AudPod allows soundcard gains and levels to be calibrated, allows input sources to be automatically switched and controlled and allows the acquisition dynamic range to be optimized and customized using its internal step attenuators. The AudPod also provides differential inputs for 2 probes, and low voltage bias and microphone preamplification for versatile electret based measurement microphones and capsules. Praxis with its AudPod is able to make use of calibration files for transducers, and can calibrate around the soundcard converters to provide data in terms of physical units such as Volts, SPL, g, or Pascals.

Praxis' data graphing and formatting capabilities are unequalled anywhere. Unlimited overlaid plots can be generated, and you can use custom backgrounds, markers with search and tracking functions, fully adjustable scaling and grid settings, weighting functions, fixed and flexible width

time data windows, and “notes” fields. The measured data, once collected, can be modified or transformed via a set of powerful “PostProcess” operations for in-depth analysis or interpretation.

Detailed calculation of room acoustics characteristics (such as Schroeder curve, STI, Objective Clarity, RT60, etc.) is provided along with efficient, wide dynamic range measurement capabilities using techniques such as logarithmic sweeps (chirp) and shaped wavelet tonebursts.

The Praxis measurement system provides for intelligent automation, through its support of visual Scripts. A Praxis Script is an application (program) that can run within Praxis, complete with a Graphical User Interface, and is able to operate the Praxis’ main features (as well as other general computer operations). Praxis includes a range of application and demonstration scripts, and others can be developed by users with minimal programming ability. Included is the Liberty Script Designer, a visual RAD development tool (patterned after Borland's Delphi system) to streamline development of custom Praxis Scripts, using drop and drag techniques and coding of event handlers in DelphiScript or VBScript.

Praxis operates in one of two modes:

- the Full operating mode, in which all capabilities are enabled. This mode requires purchase of the AudPod with the Praxis Software.
- and the Free (or Demo) mode, which is provided free. It allows inspection, formatting, processing and printing of Praxis-obtained data. Free mode also provides several usable measurement functions, including Thiele/Small parameter measurement of woofers, and RTA monitoring of audio signals. And, of course, Free Mode also allows you to explore the features of Praxis.

See also: Capabilities, Hardware and Software Requirements, Program Considerations, How to Purchase an AudPod

## ***Capabilities***

Praxis is able to generate and acquire audio data in 16bit and 24bit formats, and at sample rates to 192kHz. It can operate with one or two channels in input and output. Although Praxis can work with WAV files as input or output, Praxis normally generates its stimulus signal in real time, rather than being constrained by finite pre-generated wave files. It can, for instance, generate an arbitrary multitone signal set continuously or generate very long sweeps on the fly.

Praxis is a very versatile measurement instrument. The many possible combinations of its Stimulus, Acquisition, and PostProcess types, and the nearly infinite extensibility provided by Praxis' Scripts, prevent a full listing of its capabilities from being presented here. However, some of the more obvious applications of Praxis include:

- Complex Frequency response measurements (magnitude and phase) of electronic and acoustical systems
- Complex Impedance measurements
- Extraction of Thiele-Small parameters of loudspeaker drivers.
- Measurement of in-room quasi-anechoic frequency response of loudspeakers, with fixed or flexible time windows.
- Obtaining the impulse response of acoustical and electronic systems
- Measuring sound room characteristics
- RTA measurements
- Spectral Contamination measurements
- Multitone generation and measurements
- Harmonic distortion measurements
- Intermodulation distortion measurements (of nearly any practical order)

- Shaped tone burst measurements
- Sound hardware evaluations
- Time domain waveform investigation
- Vibration analysis
- Quality control testing of loudspeakers, transducers, electronic devices
- Measuring acoustic characteristics of rooms such as Speech Transmission Index, Reverberation Time, or Clarity
- Crossover design and optimization
- Waterfall (spectral decay) plot generation
- Compilation, formatting and display of Polar radiation pattern data
- Pass/fail Evaluation of measured data compared to easily generated limit curves
- Alignment of noise masking systems
- signal generation: sinusoids, noise, bursts, sweeps
- sound level monitoring and measurement of frequency response using live program material
- Determination of peak output capability of sound systems or loudspeakers
- Timed measurements (starting and/or finishing)
- room equalization
- room and loudspeaker sound convolution
- Isolating the distortion components from the linear components in the output signals of audio components and loudspeakers

## ***Hardware and Software Requirements***

### **Soundcard requirements:**

- Supports true full duplex at 16 bits (24 bits also supported), stereo mode
- Sample rate at 48kHz (rates to 96kHz are supported with Windows 2000 and to 192kHz are supported with Windows XP).
- Must have stereo Line input and stereo line output connectors. For most portable laptop-based systems, a quality external USB type soundcard is recommended. This is because the built-in sound hardware in laptop computers is usually of very low grade and these machines seldom provide line input jacks. Recommended USB soundcard models can be found at [www.libinst.com](http://www.libinst.com)
- Must NOT have hardwired record-->play or play-->record monitoring (such features, if included, must be capable of being disabled via the system mixer applet).
- The soundcard's software mixer should have adequate control to center the gain levels to accommodate signal levels of approximately 1.5V peak (most pro-level or quality soundcards meet this requirement).
- Have good, reliable, and fully debugged Windows 32 bit drivers.

### **Software requirements:**

- Windows 2000, XP, or later. Windows 98SE and ME are usable, but not generally recommended because of resource limitation and stability issues.

### **Computer Hardware Requirements:**

Praxis relies on the power of your main processor (which is also busy running the Windows operating system), so faster is better! The following are minimum and recommended characteristics of a computer for Praxis:

- Clock: 333MHz minimum; 500MHz or higher recommended; Athlon or PentiumII/III/IV processors are preferred.
- RAM: 128MEG or more recommended



- Video: (video speed helps with Praxis). AGP recommended.
- Use of Praxis with 256 (8 bit) color graphics may cause graph drawing problems in some systems. 16bit, 24bit or 32 bit graphics is recommended.
- Other than for Free Mode, must have at least one free working USB port (for the AudPod). When used with a USB-type soundcard, at least two free working USB ports are required. A USB hub may be used to expand the number of ports.

## ***Praxis Program Design Considerations***

Praxis is the third generation of audio measurement software from Liberty Instruments, preceded by **IMP** (1992) and **Liberty Audiosuite (LAUD)** (1994). The previous systems were 16 bit DOS-based. Praxis, is a 32-bit Windows program (or "application", per the modern jargon), and was developed entirely from a new start.

Being DOS based, IMP and Liberty Audiosuite benefited by being able to count on almost having the computer's full attention. Scheduling and latency were not major concerns and the audio collection hardware was virtually under direct control. On the other hand, the rather primitive 16bit memory architecture limited the data sizes which could be efficiently handled, and sharing of data between the DOS based programs and other programs was often inconvenient. Availability of compatible hardware was an ongoing issue for both programs as well.

Praxis makes full use of the 32 bit "flat" memory model, and also provides Windows interface features with which most users are now familiar. Praxis can use a wide variety of Windows soundcards and can be installed on (or quickly transferred between) computers without removing panels or rebooting.

While Liberty Audiosuite used DSP-based audio cards to assure the availability of enough bandwidth for its signal processing operations, Praxis uses the computer's main Pentium-grade processors for this purpose. The main difference, of course, is that processors are now much faster -- when LAUD was developed, 16MHZ machines were common, but now 1GHz clock speeds are considered to be only moderately fast.

### **Design Goals and Philosophy**

Two characteristics stressed during Praxis' development were (1) Flexibility and (2) Ease of Use.

Flexibility of course makes the program more valuable and promotes its adoption for more purposes. It also allows the program's capabilities to remain up to date.

Ease of Use means less time spent in support calls for both the developer and the user. This is an important consideration when the staff is as small as Liberty Instruments' staff!

Liberty Instruments addressed Flexibility by enabling very adaptable graphing windows ("Plots"), by providing a wide array of stimulus, acquisition, and postprocess types, and by opening up an entire API ("Application Programming Interface") for controlling Praxis from user-written scripts.

We addressed Ease of Use by providing extensive Help features and manual, and by designing the controlling forms to, as much as practical, show controls and options only when they could be relevant to the operation being performed. For instance, the Configuration Forms for the various Acquisition types will change depending on the chosen Stimulus and Acquisition, to show only the appropriate controls for the situation. And you can hide these Configuration windows when further adjustment is not needed, and quickly bring them back up when you need them.

Praxis is designed assuming the typical laboratory sequence of:

- 1) Stimulate the device being tested: "Stimulus"
- 2) Record the device's response: "Acquisition"
- 3) Transform and interpret the response, in relation to the stimulus: "PostProcess"

To these, Praxis adds another layer of control: "Scripting" programmability. This is somewhat like having a laboratory assistant available to do the tests over and over again, to modify the process depending on results, and to further transform the results to new data forms as they are developed.

Scripting, of course, also allows for easy-to-perform measurement processes to minimize the required skill level of a measurement operator.

***How to purchase an AudPod for Praxis "Full" operation***



**Contact:**

Liberty Instruments, Inc.  
P.O. Box 1454 West Chester, OH 45071  
Phone/Fax: (513) 755 0252

[www.libinst.com](http://www.libinst.com)

Please see our web site for contact and availability information

# Working with Praxis

## *Installation*

**If you are installing in Full Mode (that is, if you will be using Praxis with the AudPod), you may want to check the Liberty Instruments web site to verify that you have the latest version of Praxis --if not, it can be downloaded from the [www.libinst.com](http://www.libinst.com) website.**

A). **If you will be using an AudPod for full PRAXIS operation**, connect it to your computer as shown below. If you will be running in "Free" (or demo) mode, skip to step B.

Your computer may report "new hardware found" and search for drivers. It will normally be able to install and use drivers built-in to Windows.

**FOR WINDOWS 98 ONLY**, an "AudPod.Inf" driver file can be found on our web site. Use this ONLY if Windows is unable to find a usable built-in driver -- (but *never* use this inf file for Windows ME, 2000, or XP!).

B). **Insert the PRAXIS CD into your CDROM drive.** If the installation program starts automatically, follow the instructions. If it does not start, use the Explorer (accessible from your "My Computer" icon) to browse to the CDROM's directory, and then double-click on the file "PraxisInstall.exe" to begin installation. Newer versions of PRAXIS can be installed directly over older versions without uninstalling.

C). **Start the PRAXIS application.** The first time this is done, it will present a Soundcard Mixer matching utility, asking you to verify that Praxis has correctly identified the mixers that control the detected record and play soundcard devices and which record devices are associated with which play devices of the same soundcard(s). (If you have questions about Windows Mixers or how to use the "Device-Mixer Matching" form, use the Help menu of the Main Form, select "Help", then in the Help window, click the Contents button, and browse "Liberty Praxis Overview....About Windows Mixers").

After correcting these (if necessary), click "OK" to approve the matching of these and a new window (or "form") will appear, titled "**-praxis- Script Launcher**", with a number of forms behind. This top form is a device that lets you select and start various automated "script" programs for training, information, or to run measurements. You can choose to have the Script Launcher appear each time Praxis is started, or you can bring it up when needed using menus on Praxis' "Main Form".

D). You should now **run the "Soundcard Capabilities" script** as follows:

- In the Script Launcher, click on the top control labeled "Select Script". A list of scripts will drop down. Find the one titled "**Soundcard Capabilities**" and click on it.
- Click on the LAUNCH SCRIPT button and the Script Launcher will be replaced with a new form titled "Sound Card Claimed Capabilities".
- One of the forms in the background is titled "**(levels)**". Find this Levels Form (you can click on visible regions of forms to bring them to the front, if necessary) and on the Levels Form, locate its Setup menu control text near the top. Click on the word "**Setup**" to open the menu.
- A list of submenus will appear. Click on the one labeled "**D/A&A/D**", then click on the top entry in the next list to appear, and you will see a list of all available soundcard devices on this computer. Select the one you wish to use with PRAXIS.
- Click on a visible portion of the "**Sound Card Claimed Capabilities**" script form to bring it to the front, then click on its "**Check Sound Device**" button. After a short pause, the table should be filled with reported capabilities of your soundcard, and perhaps a

recommendation at the bottom about the setting of something called "**WAVE\_EXT**". (Note that for many lower-grade soundcards, these claimed capabilities are often exaggerated!! This script does not perform a test, it only asks the card to report its claimed capabilities).

- The "WAVE\_EXT" setting can be found on a menu on the "Liberty Praxis (**MAIN FORM**)" on your screen. Click the "Config" menu, then "Hardware Specific", and then "WAVE\_EXT" will appear near the bottom of the next list. For most situations the WAVE\_EXT option should be left NOT checked. If you need to change its setting, click on the menu word "WAVE\_EXT" to toggle it between checked and unchecked states.
- Then close the "**Sound Card Claimed Capabilities**" script form by clicking on the small "x" at the top right of the form

E). **Load your microphone correction (calibration) files.** (If you do not have an AudPod and a calibrated microphone with data file, skip to step F). Identify your microphone's data file (it should come with your microphone and be a file ending with a ".dat" extension, such as "B12345.dat". Copy this file to your Praxis installation directory (normally, "C:\Program Files\Praxis"). Then, use the Main Form menu of Praxis to load the microphone data by the options "Config, Hardware Specific, Mic/Accel1, Load" (to use the microphone at the AudPod's Mic1 input -- this is the proper setting for most installations) or "Config, Hardware Specific, Mic/Accel2, Load" (to use the microphone at the AudPod's Mic2 input).

Self-powered microphones or microphones using external bias/preamplification can alternately connect using the "Probe1" jack of the AudPod. With this connection, the microphone data should be loaded as if it were for the Mic1 connection.

An accelerometer calibration file, if available, can be loaded by the same processes.

If your microphone does not have a calibration file, but you do know its characteristics, you can use Windows Notepad to open, edit, rename and save the file "ExampleMic.dat" (included in your Praxis directory), which also shows the required format.

F). **Calibrate Praxis for this soundcard.**

- If you do not have an AudPod, click the large "Information and Demo Scripts" button on the Main Form to again bring up the Script Launcher. Run the "Probe Cal" script, the "Ground Resistance" script and the "RTA Cal" scripts, as required for your intended application.
- If you **do** have the AudPod, again find the "(Levels)" form, click its "Setup" menu and select "Do Calibrate". Then follow the instructions for the calibration process. You can find information about special requirements or needed settings for some soundcards at <http://www.libinst.com/SoundCardCal%20Main.htm>. The first time you calibrate be sure to read the presented instructions and check the mixer settings and available controls thoroughly. If you later recalibrate, you can suppress the wordy explanations by using the checkbox presented at the beginning of the calibration wizard.

G). **Use Praxis.**

- If you have an AudPod, you can find detailed step-by-step instructions for many measurement needs in the Measurement Guides script (easily accessed using the Main Form's "Help" menu). Other scripts for general usage or Introduction can be found using the "RunScripts" menu.
- If you don't have an AudPod (Free mode), you will find automated scripts to help you measure Thiele/Small parameters or use the Free RTA, from the "Information and Demo Scripts" button. You can also read the descriptions provided in the Measurement Guides script (though most of these measurements themselves are not available without an AudPod). And be sure to check out the "Introduction" script to check out some of Praxis' capabilities.

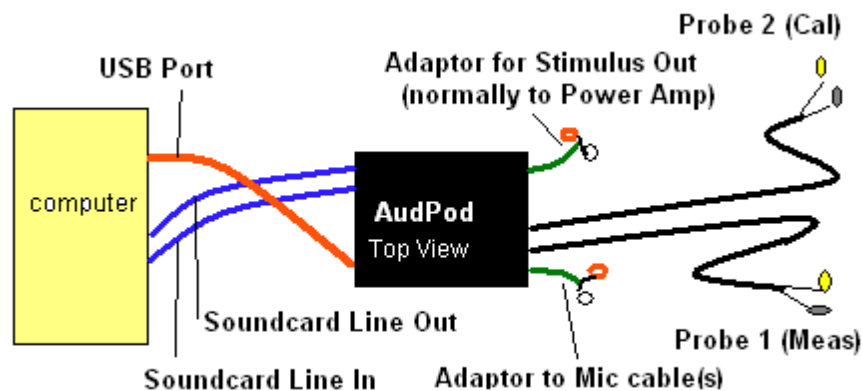
## Problems?

- **...with calibration?** Make certain that all other sound or sound-controlling applications are closed. In particular, allow no "3D Sound", "EAX", "Environmental Audio" or similar processing to be active during Praxis use. Be sure that no monitoring functions are enabled which might be sending record audio into the playback channel or playback audio into the record channel. Make sure that the "Mixers and I/O Matching" (see [C], above) has been properly done. Also, make sure your soundcard drivers are up to date. Early release drivers can often be buggy.
- **...with selection of sample rate or operation with certain rates?** Many sound cards are less than honest about their reported capabilities. In some cases, you may need to determine yourself which rates are actually supported. Most cards, however, can be relied on to support 48kHz sampling.
- **...with selecting 24bit resolution?** Some soundcards or drivers will not support the 24bit mode unless called with the "WAVE\_EXT" option checked. Also, many cards that report 24bit support can't really do it (or even emulate it properly) -- check your soundcard's documentation.
- **...with operating Praxis controls?** See PRAXIS' extensive help facility. Also see the built-in scripts for tips on using Praxis.
- **...with anything else?** Check the FAQs at [www.libinst.com/praxfaq.htm](http://www.libinst.com/praxfaq.htm) for tips or links to getting updated versions of Praxis.

## Connections

### For Full Mode (With AudPod):

Attach the AudPod to your computer and probes/microphones as shown:



In the diagram above, the blue lines represent stereo cables with male-male 3.5mm phone plugs. These connect the line inputs and outputs of your soundcard with the AudPod.

The red line represents the USB cable.

The short green lines are 3.5m stereo-phone to dual-RCA adapters. These typically have a red RCA jack and a white RCA jack. For the Stimulus Output adapter, the white jack is the output "tip" connection and the red jack is the output "ring" connection. The output adapter typically will connect to the line, aux, or "CD" input of a power amplifier (for "synchronous" type measurements, only the white or "tip" connection is to be used -- in fact, for most measurements,

the "ring" output is rarely needed).

The Mic Input adapter converts both mic channels to RCA type cables (which are easily obtained to extend and connect with a calibrated cable biased electret microphone such as the Girardin microphones). The White adapter jack is normally the Mic#1 input and the Red jack is the Mic#2 input. In most cases, only the Mic#1 input will be used. These jacks can also be used for input from accelerometers. You can alternately connect an externally powered preamplified microphone or accelerometer using the AudPod's "Probe1" input jack (see the section on "Praxis Operating Modes: Microphones").

The Probes connect as shown. The AudPod probes are differential, and contain two clips on the end that attaches to devices being tested.

- The **black (or green or blue)** clip is the **inverting input** and is typically connected to ground at the measurement point. It can also be left unconnected (but should always be used in impedance or Thiele-Small measurements). Or it can be connected to the point at which the measurement input is to be referred.
- The white (or yellow, or red) clip is the **non-inverting input** and is the usual sense or "hot" input

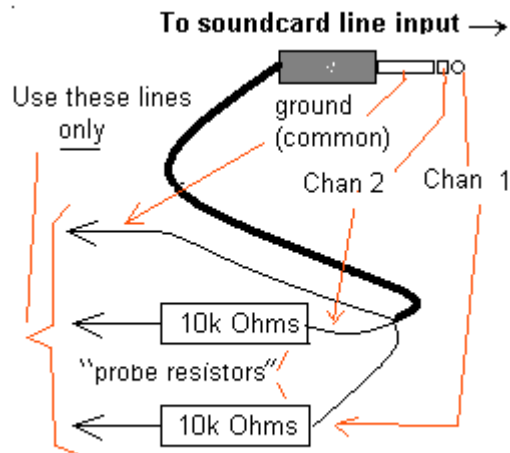
Although both probes are electrically the same, the one with a white or yellow probe clip is typically used as "Probe 1" and the one with a red probe clip is typically used as "Probe 2". The plug end of each probe cable is marked with a color band to help in making proper connection to the AudPod.

PLEASE READ AND OBSERVE THE CAUTIONS ABOUT THE PROBES DESCRIBED IN THE "PROBES" DESCRIPTION.

**For the Free (Demo) Mode (Without AudPod):**

In this mode, the probes are used only for making impedance measurements of woofers, in order to extract Thiele-Small parameters. If you use these type probes, **you do so entirely at your own risk**; incorrect connection or operation can damage your computer or other hardware -- please see all cautions given in the Demo Mode Thiele-Small Parameter Measurement Script. **Liberty Instruments will not be responsible in any way for any undesirable results.**

The probes can be constructed as shown. The 3.5mm stereo phone plug is to be inserted into a STEREO LINE INPUT jack of your soundcard. (Note that in some sound cards, the channels may be swapped -- if strange impedance measurements are obtained, try switching the connections of the resistor probes).



If you are using the RTA facility of the Free Demo Mode, you can connect a microphone into your computer's microphone jack. However, the measurement cannot be calibrated, and the frequency response characteristics of the microphone preamplifier (and any software equalizers being used) in your soundcard may have unexpected effects on the measured results.

**About "Windows Mixers"** Note: You can usually open the Windows mixer applet for your selected soundcard (if it has one) by using the "WinMixer" menu option of the Levels Form.

When you start Praxis the first time (or after installing a new sound card or sound device), a small form labeled "Device-Mixer Matching" will appear. (You can start this form at other times by selecting "Mixer and I/O Matching" from the Main Form's "Config" menu).

**Input->InMixer** Edit

Input Devices	Matching Mixer
CardDeluxe Analog	NONE
CardDeluxe Digital	NONE
SB Live! Wave Device	SB Live! Wave Device

**Output->OutMixer** Edit

Output Devices	Matching Mixer
CardDeluxe Analog	NONE
CardDeluxe Digital	NONE
SB Live! Wave Device	SB Live! Wave Device

**Output<>Input Match** Edit

Output Devices	Matching Input Dev.
CardDeluxe Analog	CardDeluxe Analog
CardDeluxe Digital	CardDeluxe Digital
SB Live! Wave Device	SB Live! Wave Device

Apply Cancel Ok About

**Be sure to check all rows (use sliders at right if needed)**

The Device-Mixer Matching form shows how Praxis assumes that your "mixers" are assigned to your various soundcards or sound chips (in case you have more than one). "Mixers" are the small Windows program forms that let you adjust your soundcard's volume and gain slider controls. There is usually one for each card's "record" (or "Input") function and one for its "play" (or



"Output") function. You can usually tell which mixer is intended for which soundcard, but Praxis cannot always detect this correctly by itself. Also, some soundcards (such as many professional recording grade cards) do not have any mixers (or have only "playback" but not "record" mixers), so use the option "No Mixer" as appropriate.

To change the setting, highlight the "Mixer" entry that is wrong by clicking on it. Then click on the closest "Edit" button above the entry and a list of available mixers will pop up. Select the proper mixer (or "No Mixer") from the list. Click [Ok] when the matching looks correct. Similarly, use the "Output<>Input Match" portion of the form to verify (or change, if necessary) that each output sound device is associated with the proper input sound device. The devices associated should reside on the same sound card.

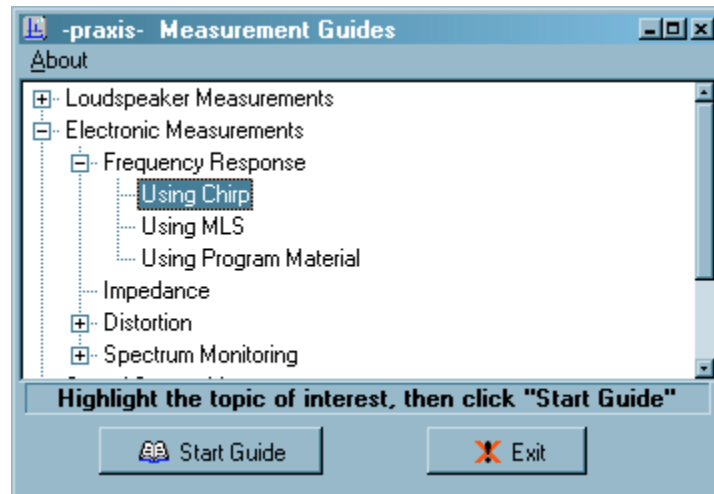
Similarly, use the "Output<>Input Match" portion of the form to verify (or change, if necessary) that each output sound device is associated with the proper input sound device. The devices associated should reside on the same sound card.

The reason why it is important to properly identify the device pairs and their Windows mixers (if any) is that other applications or users may make adjustments to those mixers at other times (when PRAXIS is not being used). The mixer settings for sound devices used by PRAXIS must be restored to the same state as they were when PRAXIS' calibration was performed, or the effects of the calibrations will not be valid. Identifying the mixers allows PRAXIS to take a "snapshot" of the mixer settings at calibration and to restore the proper settings when PRAXIS again selects that sound device.

## **Getting Help**

Praxis includes several types of built-in assistance:

- For general Help in understanding the Praxis system, use the Help menu on the Main Form. This will lead to the Windows Help facility, from which you can refer to the table of contents or search in the index for the topic in which you are interested. The help pages often contain links to other related help topics. These links are underlined. You can follow the links, if you wish, by double clicking on them.
- You can find answers or solutions to some questions and problems at <http://www.libinst.com/praxfaq.htm>
- For making a wide variety of measurements using the AudPod, Praxis includes a special Script that can serve as a step-by-step on-screen Guide. This Script is also available from Free/Demo mode to show examples of Praxis use. This "**Measurement Guide**" Script provides detailed instruction concerning connections, parameter settings, and procedures for accomplishing numerous measurements. These instructions are not configured to perform the measurements automatically, but instead lead you through the steps to perform functional measurements. This quickly builds familiarity with the system and enables a user to modify or extend each process for his own specific needs. The Measurement Guide Scripts can be accessed by using the Main Form menus "Help, Measurement Guides". (Note that the Measurement Guide Scripts cannot operate while another Praxis script is being used).



**For Registered Users of the Praxis "Full Mode" (AudPod Owners):**

If you cannot find the software support information you need from the help files, you can contact us with questions (**please have the serial number from your AudPod available when contacting us. Sorry, we cannot supply direct technical support for users of the Praxis Free/Demo version**).

**Support questions are best sent via Fax** (USA phone number **513 755 0252**), as that method brings the question to our more immediate attention, avoids time availability conflicts, and also allows for a more well thought-out answer. Be sure that you have a valid return fax number clearly visible on the your fax cover sheet! If the question is less urgent, you can contact us over the internet via email addresses that can be found on our web page at [www.libinst.com](http://www.libinst.com).

Suggestions, bug reports, and comments are welcome. However, we regret that we cannot provide specialized versions of our software for individual users, and appreciate your understanding that we must evaluate new feature requests in terms of the expected usefulness to our users in general and to possible side effects of implementing such features, as well as the required effort in implementation.

**Please note** that Liberty Instruments and Praxis do not attempt to address general engineering training to any significant degree. It is beyond the scope of Praxis (or any single software program, for that matter) to instruct about signal theory, electronics, acoustics, programming, or other general technical fields. We are a very small company, with limited (and overworked!) personnel -- **please ask direct support questions only on subjects related to Praxis software use**. For information on more general subjects, we can, however, suggest a few specific places to start:

- **Loudspeaker Theory:** Loudspeakers, An Anthology (Four Volumes), Raymond E. Cook, Editor. [Audio Engineering Society](#). Also available through [AudioXpress](#).
- **Audio in general:** AES, the Journal of the [Audio Engineering Society](#). 10 issues per year. A CD set covering nearly all past articles and papers is also available from AES.
- **Audio Electronics and Loudspeaker Design and Testing:** A number of good books available through [AudioXpress](#).
- **Acoustics:** The Master Handbook of Acoustics, by F. Alton Everest, ISBN 0-8306-4437-7
- **Delphi (Object Pascal) Programming:** The Borland Delphi # Developers Guide (comes with the [Delphi development software](#)). or Mastering Delphi 5, by Marco Cantu, ISBN 0-7821-2565-4 . The Delphi 5 Help files can also be downloaded from Borland's web site

[www.borland.com](http://www.borland.com) and can be very helpful when developing custom DelphiScript programs for PRAXIS.

- **PascalScript Programming:** documentation on the components used for Praxis's script operation ("Dream Scriptor"), and on particulars on the PascalScript subset of Object Pascal can be found at [www.dream-com.com](http://www.dream-com.com).

There is also a wealth of information, including references to other texts, available via the internet. You can search on particular subjects and keywords using a search engine such as [Google](http://www.google.com).

### ***Installing and Zipping Up Script Files***

PRAXIS installs with a number of built-in scripts, but you may wish to install scripts from other sources, such as custom scripts developed on a different computer or written by another PRAXIS user or posted on the internet.

The files comprising a PRAXIS script are grouped into a folder (the "Script Folder" or "Script Directory"). PRAXIS includes a facility to "zip" an existing script into a compact form that can be copied or emailed to another computer. PRAXIS also includes a related facility to install (or unzip) this compressed script to the machine into which you wish to install the script.

A PRAXIS "zipped script" has the extension **".zis"**. It can be created using PRAXIS' Main Form menu "Scripts->Zip Up A Script". This will present a form that will ask you for the name of the Script Directory that you wish to zip, where you suggest it should be installed in the new machine, and where you want the ".zis" file saved to for later transfer to other computers.

To install a "zipped script", use the PRAXIS Main Form's menu "Scripts->install zipped script". In the form presented, enter the path and name of the zis file you wish to install, and the directory under which you wish to install it and activate.

## Stimulus, Acquisition, PostProcess... and Scripts

Praxis classifies its measurements based on three primary option categories, which you select via the Main Form:

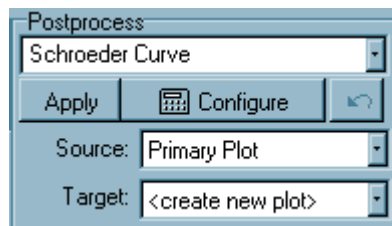
- The **Stimulus** is an audio band electrical signal (if any) used to stimulate the device being tested. The Stimulus could be an MLS burst, a WAV file, a chirp (sinusoidal sweep), a steady set of tones, or any of a number of other signals. In a measurement setup, the Stimulus should be selected first.



- The **Acquisition** is the collection of the result that the device generates in response to the stimulus. The Acquisition types which are usable at any time depend on the currently selected Stimulus. Select the Acquisition type after first selecting the Stimulus.



- A **PostProcess** is a transformation or objective interpretation made from the response. While any of the PostProcess types can be selected at any time, a given type can be applied only if it is compatible with the data type that is contained in the current PostProcess "Source" Plot. The "Apply" button (in the PostProcess area of the Main Form) will be grayed-out if the operation is not applicable.



Stimulus, Acquisition, and PostProcess each have an associated Configure button which allows hiding or showing a form that is used to set relevant values specific to the selected type(s). These three characteristics (and the many possible settings selectable from their Configure forms) define a basic measurement. There are literally thousands of possibilities, but is usually simple to find the setting appropriate to your requirement.

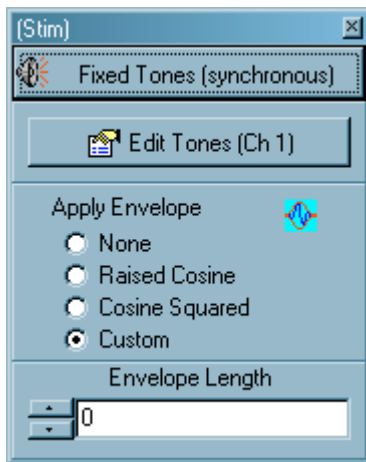
More complex or higher level operations are further supported by Praxis' **Scripts**. Scripts can change the process depending on previous results or current conditions, or can apply further processing to the data. A Script can guide a less technical user with setup diagrams, instructions, and with visual controls such as buttons, text boxes, sliders, or even AVI movie viewers. Custom user-written scripts can be developed by those familiar with Delphi or Visual Basic programming (and with Praxis operation).

# The Stimulus

## ***Stimulus Parameters***

The available parameters (settings), which you can select in the Stimulus Configure Form, vary with the type of Stimulus that you have selected. The controls and labels inside the Stimulus Configure Form will change, and the height of the form may change to adapt to the changed contents (you may need to move the form to view all of its offered settings).

You can hide the form by clicking on the 'X' button at its upper right (or clicking on the "Hide Configure" button for Stimulus). The Configure Form may be unhidden again by clicking on the "Configure" button.



Some of the buttons on Configure windows will bring up other forms (some of them quite detailed, such as the Waveform Component Editor). Others are simply single integer or floating point values.

Some synchronous Stimuli, such as sine waves (or sets of tones) can be subjected to an "Envelope" before being output by the sound card.

There are also controls for specifying the gain and peak voltage capability of the power amp (if any) used to boost the stimulus output signal.

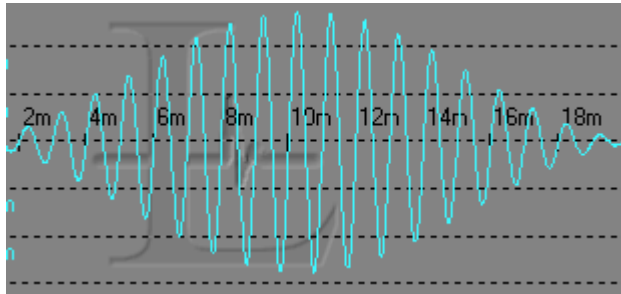
Some stimulus types provide an option to "Invert Alternate Stims". This option causes PRAXIS to invert the polarity of the stimulus signal on every other stimulus stream, and to reinvert the data again on the acquisition stream. When used with averaging, this results in cancellation of DC offsets in the measurement chains. This option should NOT be used when measuring distortion, as even order distortion components will also tend to be cancelled during averaging. The stimulus bursts are synchronized with the line frequency so that interference at the AC power line frequency (and its harmonics) also tends to cancel in the process.

When "Invert Alternate Stims" is not checked, or when it is not offered as an option, PRAXIS will, when possible, synchronize identical stimulus output bursts so that they occur at an odd number of half cycles of the AC Power line frequency. When used with averaging, this will tend to cancel effects of AC Power line components in the measurement, minimizing contamination from this source.

For the power line cancellation to work properly, you must first make sure that the parameter set in the Main Form's "Config, Hardware Specific, Line Frequency" menu is set to the power line frequency for your locale.

## Envelopes

An envelope is, mathematically speaking, a waveform that is multiplied by the output stimulus signal. For instance, if the output were a sine wave, and a "raised cosine" envelope were applied to it, the resulting output signal would resemble this:



Envelopes are used to for applying gentle tone bursts to devices (usually loudspeaker drivers or transducers) for time domain investigations. Their effect in the frequency domain is similar to that of a Window (applied to the Acquisition signal). Since an Envelope modified signal is very time dependent and also tapers toward zero signal level, it makes sense only for Stimuli that are non-continuous and Synchronous . In Praxis, they are applicable to synchronous Fixed Tones and Stepped Tones (either of which can include simultaneous multiple tones).

Usable Envelopes as of this writing are Raised Cosine (a  $\cosine(x)+1$  waveform) or a Cosine Squared waveform (Raised Cosine squared). Custom envelopes are not yet supported. The Envelope type and length (number of samples) can be set from the Stimulus Configure Form

The Acquisition Size (and FFT Size, if applicable) should be set to a value that is sufficiently longer than the Envelope Length in order to make use of the entire waveform.

The Wavelet Burst type stimuli provide a different and wider selection of envelope shapes and additionally allow the length of the envelope or burst to be specified in terms of the number of cycles of the stimulus tone.

## **Stimulus Types**

(Also see "What are Synchronous, Asynch, ReSynch Stimuli?")

Praxis provides the following Stimulus Types:

- **Chirp (ReSync) Stimulus:** Also known as a "sweep" stimulus.
- **MLS and MLS/Pink (ReSync) Stimuli**
- **Impulse (Synchronous) Stimulus**
- **Chirp (Synchronous) Stimulus**
- **MLS and MLS/Pink (Synchronous) Stimuli**
- **Fixed Tone (Synchronous) Stimulus**
- **Fixed Tones (Asynch) Stimuli**
- **Stepped Tones (Synchronous) Stimulus**
- **Stepped Tones (Asynch) Stimuli**

- **WAV File (Synchronous and Asynchronous) Stimuli**
- **White Noise and Pink Noise (Asynch) Stimuli**
- **Wavelet Burst (Synchronous) Stimuli**

## ***What are Synchronous, Asynch, ReSync Stimuli?***

These terms are used within Praxis to describe three distinctly different classes of Stimulus (and Acquisition). The class to be used will depend on the selected Stimulus type. The Acquisition operation, if applicable, will be automatically adapted by Praxis to match the Stimulus' class.

In brief:

- A **Synchronous** Stimulus/Acquisition is precisely timed, but can measure from only one input at a time. It can provide full complex data, and is time coherent.
- An **Asynch** (or asynchronous) Stimulus/Acquisition does not track exact time relationships. The output is assumed to be "ongoing" and continuous. Complex response measurements can be made, however, provided that no echoes need to be removed and that speed of measurement is not a concern.
- A **ReSync** Stimulus/Acquisition performs an asynchronous acquisition with **two input channels** and uses the time relationships (between data from the two input channels) to regenerate or "resynchronize" a result which is equivalent to Synchronous data. This can be very fast and can provide very good results (and its use is highly recommended whenever it may be applicable).

A more detailed explanations follow, for those who may be curious.

---

### **Synchronous:**

This is a class in which the Stimulus tone is applied, and the Acquisition is begun, in exactly the same way and in exactly the same time relationship with each acquisition burst. This class can be used in Praxis only with the AudPod.

Such synchronization is not a simple thing to do using Windows soundcards, because, in Windows, a "record" and a "play" are two separate processes. The operating system does not enforce any correspondence between them. For example, if the "Play" is sending out a sequence of 1000 samples while "Record" is recording 1000 samples, one can not precisely know *which* Play sample number was playing at the time that the 100th record sample was being recorded. The small delay between them can be different with each attempt (and often is, with many soundcards), which can cause great havoc if measurements are being "coherent averaged", or if precise knowledge of time delays is required.

Praxis accomplishes Synchronous operation by using one of the Record and one of the Play channels to act as a timing reference loop for the channel that is being used to do the measurement. The stimulus "Tip" (usually, the soundcard Left channel) is used for the measurement channel and the "Ring" (usually the Right channel) is used for the timing reference. A coherent combination of psuedo-noise streams is emitted at the timing reference output channel and is decoded by Praxis to precisely determine the relative record and playback timing. A disadvantage of this method is that this leaves only one channel free for doing the measurements.

Using the synchronizing process for response or impedance requires additional time. For example, to measure frequency response, first the AudPod must select the "Cal" input into the one channel, then the stimulus is sent out and a "Cal" acquisition is made, then the pod selects

the Measure channel, then the stimulus is again sent out and a "Measure" acquisition is made. Then the software processes the results, or repeats the process if averaging is needed.

A characteristic of this method is that stimuli tend to be in bursts, after which the system processes the data it has, and then sends out another synchronized burst.

This synchronous technique is the only practical way to do shaped tone burst measurements (which are usually single channel, and so are not hindered in the efficiency of their acquisition).

---

### **Asynch:**

This is a class in which the output signal begins emitting the output stimulus (tones, white or pink noise, WAV files) and then the input acquires the required number of samples of the result (such as an FFT or RTA spectrum analyzer style plot of the resulting spectrum), without particular concern for the two events occurring in any precise time relationship. The only requirement is that the acquisition is collected during the time that the effect of the output stimulus, on the device being tested, can be expected to be ready for measurement. If you expect there to be significant delay before this effect is ready, you can adjust a "propagation time" parameter (available in the Acquire Configure Form) to cause Praxis to wait. A series of tones can be emitted in order, and separately measured and tracked in order, using Asynch operation, as is done in "Stepped Tones (Asynch) Stimuli".

In Asynch operation, there can be one or two output channels, along with one or two input channels all in use simultaneously. Most measurements made in Asynch mode do not provide phase data, since phase implies consideration of time characteristics.

In cases where reflection or echo effects do not need to be dealt with or measured, it is possible, however, to make complex measurements (those providing both magnitude and phase information). This is done by determining, via Discrete Fourier Transform (DFT) the relative magnitude and phase measured by each of the two channels, and thus calculating the magnitude and phase response of the unit under test. This process is somewhat similar to the handling in the Resynch class, except it operates on only one frequency at a time (and is thus rather slow). This technique can be used to measure impedance or frequency response (provided that there is no requirement to measure the response in any particular time relative to the applied stimulus).

---

### **ReSynch:**

There are two forms of "ReSynch" operation.

**The ReSynch form used with MLS** (or MLS/Pink) relies on the fact that MLS stimulus is inherently periodic -- the stimulus sequence repeats, and (mathematically) repeats forever. You can obtain the same impulse response (IR) data of a unit being tested by acquiring and processing **any** sequence of (MLS Length) samples from it while it is being subjected to this cycle of stimulus inputs. You then lack only one thing: knowledge of where the IR response begins (your result is also periodic, starting over again after its end). This beginning point can be determined, however, if you also have a clean MLS measurement made of the Stimulus signal itself -- the highest peak of the reprocessed IR from the Stimulus signal corresponds to the beginning of the sequence.

ReSynch MLS thus makes two MLS measurements at the same time and compares them to obtain what is equivalent to Synchronous impulse response data. From this data, all normal MLS measurement results can be obtained. In fact, there are very few cases (if any) where this ReSynch MLS cannot be used to obtain the same result as a Synchronous MLS measurement. Using this technique further saves time by eliminating the need for "warm-up" cycles when



averaging MLS acquisitions-- a new acquisition can begin immediately after the previous one has been processed, since the stimulus runs uninterrupted.

**The ReSynch form used with Chirp Asynch** is somewhat similar to the process for MLS. However, a chirp stimulus is not periodic. Praxis gets around this by obtaining an acquisition data set that is longer than the chirp's time length (the "propagation time" parameter can be used to assure it is adequate). The data from the two channels is then compared in its spectral content to recreate the impulse response or complex frequency response of the device being measured. A characteristic of Chirp measurements done this way is that impulse responses, when they are obtained, are first (and only) available with all Cal and Microphone data corrections already applied -- the IR data is derived from frequency domain data, rather than the other way around. However, to allow echoes to be windowed out of a frequency response, the impulse response is than FFT'd back into the frequency domain.

This process would have been impractical for most purposes in the recent past -- the calculations and data processing required would have been too time consuming, and the memory required is large (MLS, in contrast, is very computation effective and very memory efficient). But with 1GHz processors and cheap RAM memory, this can now be easily handled.

The chirp asynch method can provide very good noise immunity over the normal audio band, due to its efficient use of the hardware's dynamic range. However, to avoid excessive low frequency noise contamination from the normalization process, the reference (channel 2 input) must be from a path having very good low frequency response. If a power amplifier is used to boost the stimulus signal, and the power amplifier response does not extend to under 5Hz, the reference signal should be taken from the stimulus output signal that feeds the power amplifier. For most impedance measurements (in particular, Thiele/Small parameter measurements), the power amplifier response *must* extend to very low frequencies. Chirp Asynch measurements can have poor dynamic range at very low frequencies because of this effect, or at very high frequencies near half the sample rate because of the anti-aliasing filter of the soundcard, which has a cutoff near this "Nyquist frequency". In the time domain impulse response, the low and high frequency effects can be manifest as noise or rippling in the tails of the impulse response. These effects are not usually visible on a normal scale, but when viewed in a log scale (such as in a Schroeder plot), they can limit the potential dynamic range.

In Praxis version 2, there is also provided a synchronous version of the Chirp stimulus. This is useful for when an un-normalized impulse response measurement is needed to achieve maximum possible dynamic range, such as for room acoustics work.

### ***Chirp (ReSync) Stimulus***

A chirp is a frequency sweep stimulus, which is internally synchronized by referring the acquired Measurement channel to the acquired Cal channel data. To use this stimulus, both input channels must be used in the acquisition, and the "Cal" channel should see a clean sample of the applied chirp. Being a "ReSynched" type stimulus, chirps can be used to measure complete complex response data including true phase measurement.

**This stimulus is usually the best choice** for speed and noise immunity in frequency response or impedance measurements. See "Chirp vs. MLS vs. StepSine".

It also allows measurement of a system impulse response, which allows echoes to be windowed out of the measurements (yielding a quasi-anechoic response of a loudspeaker, at least at higher frequencies, without need for an anechoic chamber). A chirp-derived impulse response measurement can provide a high dynamic range for optimum determination of room acoustic parameters. **A chirp derived impulse response accounts for microphone calibration data, and for any response aberrations in the stimulus channel, giving a completely "calibrated"**

**impulse -- this is the only impulse response measurement in Praxis that provides these features directly** (with other stimuli, the microphone correction and stimulus normalization happen only after data is transformed into the frequency domain).

For Impedance chirp measurements, you can select the frequency range of the measurement. For all other measurements, the frequency range will be from near-DC to the "Nyquist Frequency" (one half the sample rate).

You can select the chirp to be a "log" type or a "linear" type. A linear chirp has a "white" spectrum, that is, it ideally has equal energy per Hz over all covered frequencies. A log chirp has a "pink" spectrum, that is, the energy per Hz drops at a rate of 3dB per Octave of increasing frequency. The log chirp is usually the best one to use, as it concentrates more test energy into the lower frequencies (where noise is typically more of a problem).

You can also select the length of the chirp. Longer chirps will increase the frequency resolution and noise immunity, but will increase the processing time and file size required for saving the data.

Chirp Stimuli used with acquisition types of "Impulse response via Chirp" or "Frequency Response Mag and Phase" result in the acquired data being transformed into an impulse response. This allows echoes to be windowed out, and such an operation is assumed in "Frequency Response Mag and Phase" measurements made with chirps. For this reason, Praxis allows an FFT Size to be selected in that case that does not need to be as long (in time) as the chirp -- this allows for much smaller data file sizes, and can further improve noise immunity.

For maximum dynamic range impulse responses, there is also a synchronous form of the chirp that acquires data from only one channel (without a "cal" reference channel).

### ***MLS and MLS/Pink (ReSync) Stimuli***

These use Maximum Length Sequence (MLS) type stimuli, and rely on special mathematical properties of the MLS for rapid calculation of impulse responses with improved immunity to noise. MLS stimuli sound very noise-like, but repeat in a very precise manner. MLS measurements can provide an impulse response curve, which allow windowing out of echoes for quasi-anechoic measurements.

The ReSync types use the Cal channel to determine synchronization of the Measurement channel, and generally offer more efficient operation than do the Synchronous versions. Also, the stimulus sound from this type is continuous, which may be less annoying during testing.

MLS has a "white" spectrum (equal energy per Hz of bandwidth), while MLS/Pink filters the MLS spectrum to an approximately "pink" spectrum (-3dB per Octave slope, for energy observed within fixed bandwidths). MLS/Pink therefore focuses more of the test energy into the lower frequencies, and this can be advantageous for overcoming noise such as room noise in acoustical measurements. In most cases, the MLS/Pink stimulus is to be preferred over the normal MLS stimulus. In general, longer MLS lengths provide greater noise immunity but longer measurement times.

The impulse response derived from MLS/Pink (using "Impulse Response via MLS Acquisition") will also include the effects of this applied "pink" spectrum, and therefore will not be the true impulse response of the system being measured. But, when impedance measurements or frequency response measurements are made using MLS/Pink, the non-flat applied spectrum cancels out in the result.

For most measurements, particularly those in which an impulse response is desired, a better choice is to use the Chirp stimulus (see "Chirp vs. MLS vs. StepSine").

### ***Chirp (Synchronous) Stimulus***

This stimulus provides the same test signal as the Chirp (ReSync) stimulus, but acquires data from only one source at a time. Compared to the ReSync form, this provides for impulse responses of higher dynamic range because the response performance of the soundcard and/or a power amplifier cannot degrade the dynamic range of the measurement from the measurement channel.

Impulse responses obtained with this stimulus will not be "normalized", that is, they will be in units of Volts, Pa, g, etc. Use of a Chirp (Synchronous) stimulus is most appropriate when the resulting impulse response will be used for Room Acoustics (Schroeder Curve) postprocesses.

### ***Impulse (Synchronous) Stimulus***

A classic narrow pulse stimulus. It is a rather slow method of measurement and has poor immunity to noise. To obtain usable results, a number of averages typically must be made for each measurement to reduce the effects of noise.

Impulse stimulus is seldom used for measurements, except perhaps as an alternate technique when the results using other techniques seem suspect for some reason.

Like all Synchronous Stimulus types, all data acquired (both "Cal" and "Measure") with this stimulus is processed through the same soundcard channel. The other channel (usually the right channel) is used only for synchronizing the process, and is routed back to the soundcard via the AudPod.

### ***MLS and MLS/Pink (Synchronous) Stimuli***

These use Maximum Length Sequence (MLS) type stimuli, and rely on special mathematical properties of the MLS for rapid calculation of impulse responses with improved immunity to noise. MLS stimuli sound very noise-like, but repeat in a very precise manner. MLS measurements can provide an impulse response curve, which allow windowing out of echoes for quasi-anechoic measurements.

Like all absolute Synchronous Stimulus types in Praxis, all data acquired (both "Cal" and "Measure") with this stimulus is processed through the same soundcard channel. The other channel (usually the right channel) is used only for synchronizing the process, and is routed back to the soundcard via the AudPod.

For most measurements, particularly those in which an impulse response is desired, a better choice is to use the Chirp stimulus (see "Chirp vs. MLS vs. StepSine"). If an MLS-based measurement result is preferred over a chirp, the MLS or MLS/Pink (ReSync) types are generally more efficient than the rigid "Synchronous" types and provide essentially the same result.

MLS has a "white" spectrum (equal energy per Hz of bandwidth), while MLS/Pink filters the MLS spectrum to an approximately "pink" spectrum (-3dB per Octave slope, for energy observed within fixed bandwidths). MLS/Pink therefore focuses more of the test energy into the lower frequencies, and this can be advantageous for overcoming noise such as room noise in acoustical

measurements. In most cases, the MLS/Pink stimulus is to be preferred over the normal MLS stimulus. In general, longer MLS lengths provide greater noise immunity but longer measurement times.

The impulse response derived from MLS/Pink (using "Impulse Response via MLS Acquisition") will also include the effects of this applied "pink" spectrum, and therefore will not be the true impulse response of the system being measured. But, when impedance measurements or frequency response measurements are made using MLS/Pink, the non-flat applied spectrum cancels out in the result.

### ***Fixed Tone (Synchronous) Stimulus***

A tone or set of tones which can be repetitively applied in the same time and phase relationship. This stimulus can apply an Envelope effect acting on the stimulus output, if desired.

A typical use for this stimulus is for "spectral contamination" investigations, in which a large number of tones are applied to a device being tested, and the production of other frequencies "between the sticks" is observed with an FFT analyzer. Because this stimulus is Synchronous, each repeated stimulus burst is identical, so the output result can be Coherently averaged, thus allowing noise to be reduced.

Like all absolute Synchronous Stimulus types in Praxis, all data acquired (both "Cal" and "Measure") with this stimulus is processed through the same soundcard channel. The other channel (usually the right channel) is used only for synchronizing the process, and is routed back to the soundcard via the AudPod.

The tones to be output are configured using the Waveform Component Editor. The number of tones which can be simultaneously used is a function of the computer speed and the related hardware, but typically 40 or more tones can be handled.

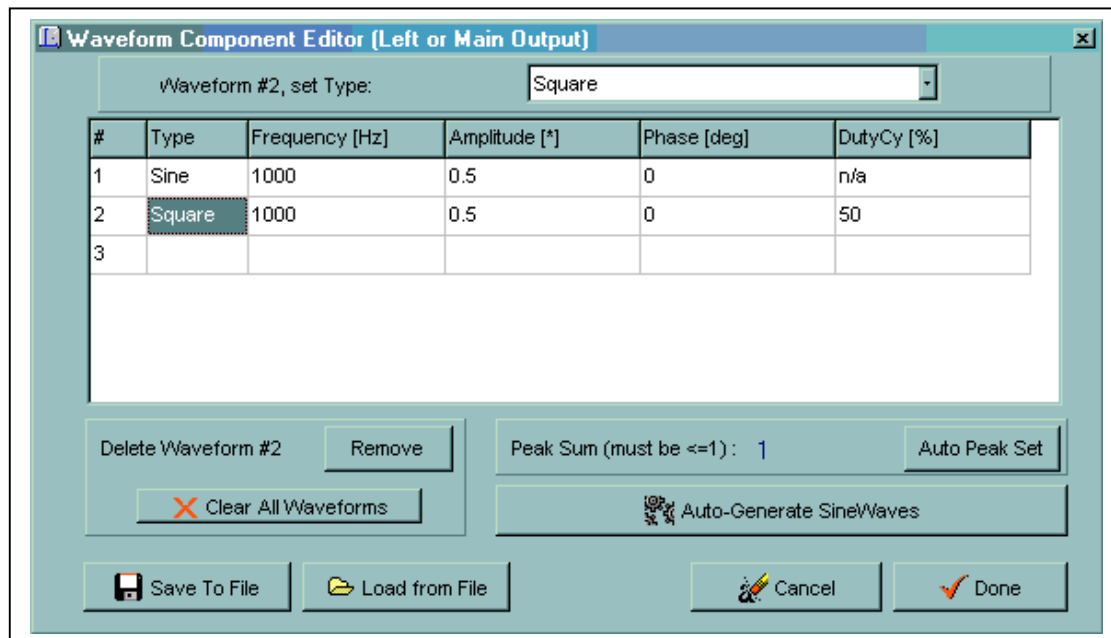
### ***Fixed Tones (Asynch) Stimuli***

A tone or set of tones which can be applied continuously.

A typical application for this is for stimulating harmonic or intermodulation distortion products and monitoring them via and FFT type Acquisition. It can also be used as a general purpose signal or function generator, or for experimenting with various harmonic series.

The tones to be output are configured using the Waveform Component Editor. The number of tones which can be simultaneously used is a function of the computer speed and the related hardware, but typically 40 or more tones can be handled. A variety of waveforms can be combined, including sine waves, square waves or pulses, and triangle waves.

The Waveform Component Editor



This Editor allows you to select the number, frequencies, levels, starting phases, and duty cycles of the stimulus tones for the Fixed Tones (Synchronous and Asynchronous) Stimulus types. You can specify each tone individually, save or retrieve the specification for the set of tones to a files, or configure the tones by specifying starting and stopping frequencies (and/or levels), number of tones, and the spacing of the tones.

(In Free/Demo mode, you can work with the editor, but cannot use any of the created sets in a Stimulus. WAV files can, however, be generated and saved with the tool).

Each sine wave, , etc., is called a "Waveform Component". To use the Waveform Component Editor to modify individual Components, click the mouse in the grid space for the Component number and the property you wish to change. For instance, to change the frequency for Component #3 (assuming it exists), click in the third white row, under the "Frequency" heading. The control at the top of the form will change to a Numerical Value Control for selecting the Frequency. If you change this, the value in the grid will change to match.

To add a new Component click the mouse in any cell of the last (empty) row in the grid. You can then edit its properties as desired. To remove a component, highlight one of the cells in its row (by clicking the mouse), and then click on the [Remove] button. To remove all the waveforms, click on the [Clear All Waveforms] button.

The amplitudes specified here are relative. Their sum can NOT exceed a value of 1.00. To automatically scale all the values so their sum is 1.00, click the [Auto Peak Set] button. This will give you the option of scaling all values in their current proportions, or giving them all the same amplitudes. You can, of course, adjust the amplitude of each Component manually, instead, if you prefer. But you will not be able to exit with the new values activated until their sum is not greater than 1.

Rather than enter the characteristics for a large array of sine waves manually (which might take quite a while), you can instead describe the set and let Praxis do it with the [Auto Generate sine waves] button. This brings up a form which is rather self-explanatory. The form will ask you to specify how the frequencies (and/or amplitudes and/or phase) are to vary from component to component, and the maximum number of sine waves it is to use over the specified range (it will stop when the first limit is reached, unless a starting value is the same as the stopping value for a property). Then, click [Generate], and you will return to the Waveform Component Editor, with the

new value filled in as specified (they will be added to the components that were already in the Editor). Using the Auto Generate feature, you can easily configure sets of waveforms for custom "Spectral Contamination" stimuli.

After you specify a waveform set to your liking, you can save the state of the Waveform Component Editor to disk for later retrieval using the provided [Save to File] and [Load from File] buttons. You may also generate a WAV file from the specified waveform set.

### ***Stepped Tones (Synchronous) Stimulus***

A sequence of successive tones or sets of tones which can be repetitively applied in the same time and phase relationship with each sequence. For example, the sequence could perhaps be a succession of tones in the following pattern:

*100Hz,200Hz,300Hz...19900Hz,20000Hz*

which could then be used with an acquisition to give a data point at each 100Hz spacing from 100Hz to 20kHz. There can be more than one tone per step, and the tones need not be in ascending order (nor in any order at all, if a list file is used). This type stimulus has the often stated advantage of being usable to generate frequency domain data that can utilize on any set of frequency points (that is, it is not restricted to a large number of linear spaced points, as are simple FFT based methods). However, this advantage is not significant with modern systems, since you can easily PostProcess frequency domain data (from any type measurement) to ReMap it onto any desired set of frequency points -- in general, use of a Chirp or MLS type stimulus will be much more time efficient.

When used to measure frequency response, the output from the device being measured is analyzed, by the Discrete Fourier Transform (DFT) for energy at the one applied frequency, with each step in the sequence. The Acquisition process rearranges the measured data in order of increasing frequency during the measurement. Because analysis can be made to detect only at the applied stimulus frequency, this method of measurement gives perhaps the best immunity to noise and distortion due to the large amount of applied energy and long time in analyzing. It is an extremely slow method, and can apply considerable stress on devices being tested because of the large amount of energy applied.

This stimulus can also be used to apply multiple tones to generate distortion-vs.-frequency plots, or to send tone "clusters" through a transducer while watching its spectrum (using an FFT acquisition) for breakup due to rubbing or other breakup conditions.

This stimulus can apply an Envelope effect to act on the stimulus output, if desired. (An envelope cannot be used to make impedance measurements, however). Because this stimulus is entirely synchronous, acquisition can be configured to window out (or gate) echoes or reflections from acoustic measurements.

Like all absolute Synchronous Stimulus types in Praxis, all data acquired (both "Cal" and "Measure") with this stimulus is processed through the same soundcard channel. The other channel (usually the right channel) is used only for synchronizing the process, and is routed back to the soundcard via the AudPod.

## Stepped Tones (Asynch) Stimuli

A sequence of successive tones or sets of tones applied in a specified sequence, with a measurement point typically made and recorded at each step. The Acquisition of data is not synchronized with this type stimulus. However, some types of complex response data (such as impedance) can still be obtained using Asynch Stepped Tones with a two-channel acquisition.

The sequence of tones or sets of tones is specified using the SteppedTone Editor (which can be brought up using a button in the Stimulus Configure Form).

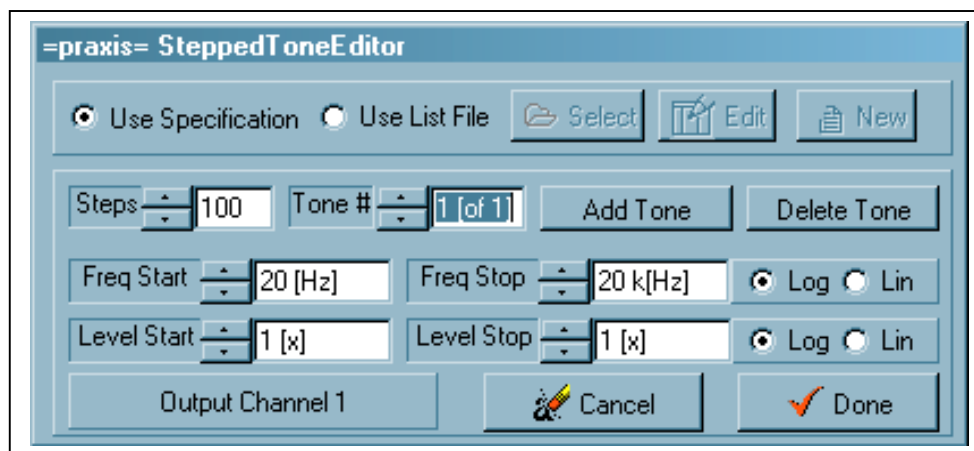
As an example, the stepped sequence could perhaps be a succession of tones in the following pattern:

*100Hz,200Hz,300Hz...19900Hz,20000Hz*

which could then be used with an acquisition to give a data point at each 100Hz spacing from 100Hz to 20kHz. There can be more than one tone per step, and the tones need not be in ascending order (nor in any order at all). This type stimulus has the often stated advantage of being usable to generate frequency domain data that can utilize on any set of frequency points (that is, it is not restricted to a large number of linear spaced points, as are FFT-based methods). However, this advantage is not significant with modern systems, since you can easily PostProcess frequency domain data (from any type measurement) to ReMap it onto any desired set of frequency points -- in general, use of a Chirp or MLS type stimulus will be much more time efficient than will stepped methods.

When used to measure frequency response, the output from the device being measured is analyzed, by the Discrete Fourier Transform (DFT) for energy at the one applied frequency, with each step in the sequence. The Acquisition process rearranges the measured data in order of increasing frequency during the measurement. Because analysis can be made to detect only at the applied stimulus frequency, this method of measurement gives perhaps the best immunity to noise and distortion due to the large amount of applied energy and long time in analyzing. For usable results, full data windowing is required. It is an extremely slow method because each frequency is measured individually, and can apply considerable stress on devices being tested because of the large amount of energy applied.

This stimulus can also be used to apply multiple tones to generate distortion-vs.-frequency plots, or to send tone "clusters" through a transducer while watching its spectrum (using an FFT acquisition) for breakup due to rubbing or other breakup conditions.



## The SteppedTone Editor

The SteppedTone Editor allows you to specify the stepped tones by either giving a specification or by presenting a file with a formatted list of frequencies. Any changes to this list do not take effect until the [Done] button is used (you can use [Cancel] to void any changes).

To **Use Specification** to define your tone sequence, select that option at the top of the form. Then define the number of steps in the sequence (if you are using more than one tone per step, each tone must cover the same number of steps).

Then, select which tone you wish to edit (in the "Tone #" control). If you want to add another tone to the set, click the [Add Tone] button. To delete the tone currently displayed, click [Delete Tone]. For each tone, you can then define the start and stop values of both the Frequencies and the Levels, and the way you wish the values to progress. (The "Levels" value is comparable to the "Amplitude" value used in the Waveform Component Editor which is used when defining Fixed Tones). The adjustments are made using Praxis' Numerical Value Controls, which can be adjusted in a number of convenient ways.

"Log" means that successive values are calculated by applying a multiplier to generate a value based on the value of its preceding step -- for frequency, an example (which generates 4 steps in log fashion from 1kHz to 8kHz) might be:

1000Hz, 2000Hz, 4000Hz, 8000Hz.

The same range covered by "Lin" (using equal steps spaced 2333 Hz) would be:

1000Hz, 3333.3Hz, 5666.6Hz, 8000Hz

For Levels, values are displayed and controlled as a scaling factor, with the maximum allowable value being 1.00. "Log" will give a constant dB increment per step, while "Lin" will step in relative increments. The levels settings will usually have the same start and stop values in most situations. In use, the final overall output level can be adjusted using controls on the Levels Form. **At each step, the sum of the levels applied from all tones will be automatically adjusted to be 1 or less, by scaling the values. This is done to avoid overloading the soundcard outputs.**

To **Use List File**, select that option at the top of the form. You can use the provided buttons to **Select** an existing list file (\*.lfl) from disk, create a **New** file or **Edit** an existing file (after it is Selected from disk).

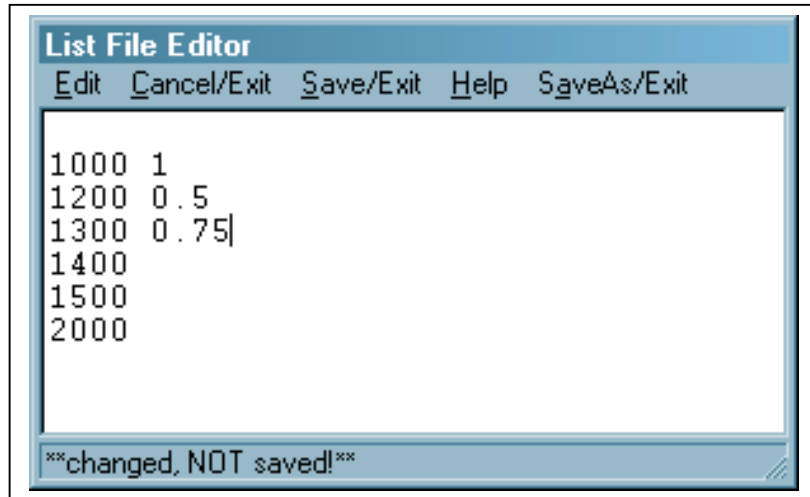
Clicking the [Edit] button brings up yet another form, the List File Editor. Use this editor to specify the order and various characteristics of the sequence of tones which you wish to apply. **In a list file that specifies Stepped Tones, you are responsible for assuring that the sum of (the absolute values of) all applied tone levels for each step remains less than 1.00 -- otherwise, Praxis may adjust the levels to ensure this, or may output distorted tones.**

See the description of the Frequency List Files for details on how to write properly formatted List Files



### The List File Editor

This is a simple text editor. Simply type the text you wish into the provided area. To generate a list file that will be understood by Praxis' Stepped Tone Stimulus generators, please see "Frequency List Files".



The **Edit** menu provides options to allow you to cut and paste between different regions (or other windows applications).

#### To leave the editor, use one of the following:

- **Cancel/Edit** menu: leaves the editor and discards all changes.
- **Save/Exit** menu: saves the file, with the changes you have made, to the existing file name, then exits
- **SaveAs/Exit** menu: allows you to change the file name, save the data (with changes) to the new file name, and then exit. The data under the previous filename (if any) will not be changed. *If you are using this from the [SteppedToneEditor](#) and want to use the newly saved file for Stepped Tones, you must use that Editor's [Select] button to load the new filename -- otherwise, it will still use the file as previously named.*

### Frequency List Files

Frequency List files (with extension ".lfl") are rather simple text files, and are not difficult to make, particularly if "header lines" are not needed and only frequencies (and not varying tone levels) are to be used.

A Frequency List File can be as simple as a series of frequencies, one value per line, such as:

```
_____  
2000  
1200.23  
1000  
50.1  
_____
```

Do not use Engineering notation (such as "2.000kHz") in Frequency List Files. Write each entire value out in standard decimal form.

The List files will be interpreted by Praxis' Stepped Tone Stimulus Generators as follows:

### The Header Line

The first line can be an **optional** "Header line", which, if given, must start with the pound-sign character "#". This line can be used to define much more complex waveform sets for each step.

Headers are somewhat complicated, and if you need only to specify a sequence of frequencies, you can skip using a header altogether. **If the header line is not given, all values in each succeeding line are assumed to represent simultaneous frequencies**, for monophonic output at each step. The level for each tone will be assigned the value: 1/( the number of tones used in that step).

An example header line may look like this:

**# f a p w f a f w / f a...**

- **The first character (after the #) must be an "f"**, otherwise the header will be ignored. An "f" in the (n+1)th position of the header indicates that the nth value in each line is to be interpreted as a Frequency for a new tone. Each time an "f" appears on the same line, another tone is generated per the value found in the corresponding position on the succeeding "Step Lines". All header characters ("a", "p", "w") until another "f" or a "/" (or the end of the line) indicate parameters of the tone for the previous specified frequency value.
- If the next character is "a", then values in that position in succeeding lines will be interpreted as being Levels (these values should be kept at or below 1.00). If the next character after an "f" is NOT "a", then the level (amplitude) of the tone at the last specified frequency will be assigned the default value "1". **All level values for each tone in a step, for each channel, will be scaled by Praxis, if necessary, so that the sum of their values is less than 1.**
- A character "p" specifies that values in the corresponding position are to be interpreted as the starting phase for the tone. The default value (if not specified) is 0.
- A character "w" can be used to specify the wave type for the tone (default is "0" for sine wave). For this, the value will be rounded and interpreted as follows: 0=sine, 1=square wave, 2=step (phase=delay), 3=impulse, 4=triangle wave.
- All values after the character "/", and until the end of the line, will be interpreted as specifications for tones in the Right channel (usable with Asynch Stimulus only!). The first character after the "/" must be an "f", for a new frequency.

### Step Lines

- **Each line (except a header line) represents one step in the stepped tone sequence.** The lines will be used in the order presented (from top to bottom, and if the sequence is not stopped, will start again at the top).
- **The first value in each non-header line is always the first tone frequency** for the Left channel (or both channels for single channel outputs).
- (for Asynch Stepped Tone Stimuli) **If no separate data is given for the Right channel, its output will be the same as for the left channel.**
- **If a Header is given in the first line**, all values in succeeding lines will be interpreted (if possible) in terms of the given header.
- **If no Header is found**, all values in each line will be interpreted, if possible, as frequencies for individual tones in the single (or both) channels.

- The Stimulus generator will attempt to use the file as best it can, and will generally not report format errors.

### ***WAV File (Synchronous and Asynchronous) Stimuli***

WAV files use the standard sampled audio data format for digital audio sounds. They are commonly used in the Windows operating system for sound effects and short voice prompts.

To use them as Stimuli in Praxis, the WAV File must meet certain requirements:

- It must have been recorded at the sample rate currently specified for Praxis' D/A (output) . The sample rates, in Praxis, are selected via the Setup | D/A&A/D menu in the Levels Form.
- The WAV files must be 16bit types (24 bit WAV files are not supported at this time), and the Praxis D/A must be set for 16bit resolution.
- The WAV files must be Stereo types.

The WAV file stimuli are provided primarily for completeness. They are seldom useful for measurements, but do offer extreme flexibility (with much inconvenience).

### ***White Noise and Pink Noise (Asynch) Stimuli***

These stimuli are computer generated noise. They are designed to have a very long repeat cycle (on the order of one day). In Praxis, this noise is generated continuously (rather than playing from a finite WAV file).

- **White Noise** has (ideally) equal energy per Hz of bandwidth. White noise, viewed on an FFT type spectrum analyzer (and averaged over a long time) will show a flat spectrum. Viewed on an RTA type analyzer (where each "bin" is of different Hz bandwidth), it will show a spectrum that rises at 3dB/octave. It is similar to MLS noise.
- **Pink Noise** has (ideally) equal energy per Octave of bandwidth. When viewed on an FFT type analyzer, it will tend toward a spectrum that falls at 3dB per Octave. When viewed on an RTA type analyzer, it will tend toward a flat spectrum. It is similar to MLS/Pink noise.

In practice, these noises will not precisely have the ideal spectral characteristics stated above, as there will be effects from the frequency responses of the hardware, and errors in the DSP filtering used to modify the signals. Typically, flatness within several dB is achievable.

When using an RTA to investigate or monitor in-room frequency response of an audio system, Pink Noise is usually used as the stimulus. This is provided in Praxis Free "Demo Mode".

When using an FFT analyzer to investigate frequency response, White noise should be used (however this can be quite uncomfortable to experience, and can potentially damage high frequency drivers of loudspeakers). -With Praxis, this can be done much better and faster with a Chirp based frequency response measurement.

### ***Wavelet Burst (Synchronous) Stimulus***

This Stimulus is used to create series of shaped tonebursts. The lengths of these tonebursts is scaled by the burst nominal frequency so that the burst consists of a specified number of cycles of the nominal frequency.

The Wavelet Burst Stimulus can also, depending on the Acquisition type it is used with, step through a series of log-spaced nominal frequencies and/or iterate through (or search for) different output levels. Wavelet Burst is intended for use in peak power capability determination or other short term distortion or compression testing. It can also be used for investigations of ringing in the time domain.

The stimulus can be configured (via the "Edit Wavelet" button on its Stimulus Configuration form) to select:

- Starting and Stopping Frequency for the test (for single frequency distortion vs. level testing, only the Starting frequency is used)
- Frequency Step size (in fractions of an octave)
- Starting level
- Level step size and search resolution (when applicable)
- Name of a relative "weight format" file that can be used to limit the output level over specified frequency ranges. There is also a button which allows you to create or edit this file.
- The length of the bursts (in terms of number of cycles, in half cycle increments)
- The envelope shape of the bursts. (This is not connected with the envelope parameter available with some other synchronous stimuli). The envelopes are similar to data windowing functions, affecting the spectrum and time distribution of the energy in the bursts.

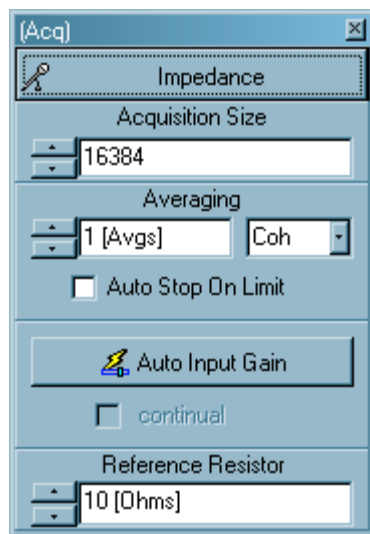
When the length and the envelope shape of the bursts are configured, the Wavelet Burst Stimulus Editor will calculate and display the -3dB bandwidth and the -1dB bandwidth. These values are shown to help in selection of appropriate acquisition parameters.

## The Acquisition

### ***Acquisition Parameters***

The available parameters (settings), which you can select in the Acquisition Configure Form, can vary with the type of Acquisition (and also with the type of Stimulus) that you have selected. The controls and labels inside the Acquisition Configure Form will change, and the height of the form may change to adapt to the changed contents (you may need to move the form to view all of its offered settings). Praxis will usually show only parameters that may be related to the measurement situation you have selected.

You can hide the form by clicking on the 'X' button at its upper right (or clicking on the "Hide Configure" button for Acquisition). The Configure Form may be unhidden again by clicking on the



"Configure" button.

Some of the buttons on Configure windows will bring up other forms (some of them quite detailed, such as the Component Tracking Editor). Others are simply single integer or floating point values.

#### **Some characteristics which may be controlled from this Form are:**

- Windowing: determines how time domain data (or acquired sample sets) will be pre-processed to avoid degradation of frequency resolution or to remove the effects of echoes or reflections from measurements.
- Automatic detection of Edges of the "Anechoic" region of an Impulse Response (used with a "Half Window").
- Automatic Input Adjustment of the levels, as controlled by the AudPod.
- Averaging to reduce the degrading effects of noise in measurements.
- Automatic Stop after a defined number of acquired or averaged data curves, or after a preselected amount of time.
- Propagation Time to ensure that data collection time is adequate to account for delays between application of the stimulus and the arrival of the result to be acquired (Asynch stimulus, only).
- Triggering, so that acquisitions can begin on a preconfigured signal edge. ·

#### **Windowing**

Windowing refers to a process in which a time domain data set is tapered, at one or both ends, before a Discrete Fourier Transform (DFT) or a Fast Fourier Transform (FFT) is applied to convert it into frequency domain data. This minimizes the generation of "spectral splatter" (apparent energy appearing in inappropriate frequency ranges) from an abrupt start or stop of a time domain waveform. The collection of data samples must be finite, with starts and stops, in real-world measurement systems.

#### **Setting Window Edges**

Window Edges define the region of time domain data which will be transformed. A starting edge is defined at the left, a stopping edge at the left, and only data between these edges is used in the transformations (except with "**Flex**" windows -- see below). Window edges are most often used to exclude the effects of reflections and echoes in a measurement.

The window edges can be set graphically or numerically via controls that are within tabs of the Plot Forms. To see this Windowing tab, an applicable time domain trace must be visible in the plot -- in some cases, you may need to do a "practice" acquisition of the type needed, then use the Time/Frequency button to change to time domain to see a trace you can use for window edge placement. For new acquisitions, the window edges must be set in the Primary Plot. For postprocess operations requiring window edges, they should be set within the Postprocess Source Plot.

#### **Half Windows**

One type of time domain data, the Impulse Response, has abruptness as an essential part of its nature. When processing an Impulse Response for spectral or frequency response data, the tapering operation of the window should be applied ONLY to the latter part of the range that will be used. The early part should start at full amplitude. This type of window is called a "**Half**" window. Praxis will select this type in acquisitions when it is appropriate. For some PostProcessing operations, you may have to select between the half and the full type windows, depending on your needs.

## Removing Echoes and Reflections

When using data windows to avoid effects of acoustic reflections, the leading and ending edges of the window are placed using features in the Windowing Tab of the Plot form. Alternately, the Find Window Edges button can be used in some cases to place these edges.

The leading edge of the window is usually placed just before the point where the data goes from a flat line (zero) and begins to show a response. The ending edge of the window is usually placed just before the response to the first echo or reflection is noticed. The frequency resolution (and lowest frequency at which any meaningful data can result) will no better than  $1/(\text{time extent of the window})$ .

In Praxis, the position of the leading window edge does **NOT** affect the phase of frequency response measurements. The phase effects due to time-of-flight (from speaker to microphone, for example) can be separately removed by setting the Delay Compensation in the Plot Form. Time of flight phase shifts are not changed by the left window edge. ***This is different than the way that IMP and LAUD operate.***

## Types of Data Windows

Praxis provides the following types of data windows (in both half and full types):

- **No Window:** the entire data set (over the FFT Size, or the Acquisition Size) is used without tapering either end or zeroing any samples.
- **Rectangular:** values outside the window range are set to zero, but values within the range are not tapered at all.
- **Blackman Harris 4:** tapers one or both edges using the 4-term Blackman-Harris window. This type is recommended for most frequency response measurements.
- **Hodie 5:** a 5-term Hodie window, best used for FFT measurements with narrower width but not as good ultimate noise rejection as the 7-term Hodie window.
- **Hodie 7:** a 7-term Hodie window, recommended when FFT measurements are made using 24-bit acquisition hardware. This provides the best possible dynamic range for viewing tone components.
- **Hamming Window:** low performance, but commonly used window type.
- **Bingham Window:** very gentle, low performance, window. Sometimes useful for CSD Waterfall plots.
- **FlatTop Window:** window type recommended for measuring levels of discrete tones, provides accurate results even if tone frequency is not centered on an FFT bin.
- **Hann Window:** commonly used simple window type based on raised cosine shape.

## "Flex" Windows

When window edges are used to remove reflections or echoes, a resulting side effect is the imposition of a limit on frequency resolution. When only brief sections of time domain data are used, no low frequency information can be obtained from them and the ability to resolve between closely spaced frequencies is degraded. For example, if you have only 2 milliseconds of time between the window edges, a transformation using strict application of windows will not be able to provide you any information about behavior below 500 Hz ( $=1/2E-3$ ), nor be able to discern between behavior at 1000Hz and 1500Hz (because  $1500\text{Hz}-1000\text{Hz}=500\text{Hz}$ ).

The Flex Windowing option provides a compromise between the desire to reject echoes and the need for low frequency information. It is, in effect, a blending of the effects of several windowing operations. For frequencies above which the defined window edges provide sufficient time data, the resulting frequency domain data is the same as that using strict windowing. For frequencies

between 50Hz and the strict-windowing limit, the results are a weighted blend between window settings as defined and as are needed to resolve data at 50Hz. Below 50Hz (where the room is expected to be essentially reactive), all of the available time domain record is used for greatest resolution.

The result of using the Flex Window option is that upper frequencies are presented as anechoic, mid frequencies are a gradual blend between anechoic and from data in the first 20 to 100 milliseconds, and very low bass frequencies are given without any reflections removed. The graph essentially gives information of separate regions of performance combined into a single graph. At higher frequencies, the measurement is essentially that of the loudspeaker, while at lower frequencies the measurement gradually becomes that of both loudspeaker and room combined. This use of a gradual blend is not intended to imply that the effects of the echoes are less important or can be safely ignored at lower frequencies -- they are simply not removable at lower frequencies, so they are included to the level needed to obtain frequency data resolution.

Similar windowing techniques have been claimed, without proof, to "more accurately" represent the way a loudspeaker system in a room sounds. This claim, however, seems to fly in the face of common experience that room treatments (carpets, wall coverings, furniture) are most sonically effective at *higher* frequencies -- i.e., in precisely the regions where a flexible type window removes those audible, but passively treatable, room effects!

However, it is also true that the effect of each utilized room cannot in practice be ignored at low frequencies (the room is essentially part of a real speaker). And the higher frequency effects of reflections can be strongly dependent on the exact location of the microphone during the measurement, while such position dependency is gradually reduced at lower frequencies. Use of the flex window can therefore perhaps be considered to be an idealization or generalization of the loudspeaker at high frequencies combined with a presentation at lower frequencies that takes into account the harder realities of physics.

## Automatic Input Adjustment

### (Full mode only)

In any measurement system, optimum use of the available dynamic range is important. All measurement (or soundcard) hardware has limitations in both the largest signals it can handle without creating excess distortion, or the smallest signals which it can detect from the noise floor.

In Praxis' Full Mode, the input levels are controlled using features on the Levels Form. The optimum gains of the soundcard Mixer are determined and stored during Praxis calibration processes, for retrieval on each use of Praxis. The controls should be set so that the signal levels applied to the soundcard's A/D converters are just below the maximum setting which does not cause high distortion. (For some distortion measurements, this may need to be adjusted manually for lowest residual distortion, depending on the soundcard hardware).

For some measurement processes, Praxis can do this adjustment for you, by use of the Automatic Input Adjustment. With this feature, you can have Praxis perform a series of test runs of the measurement, and attempt to adjust its input levels for optimum dynamic range. **Note that this will not be successful in all cases. In some circumstances** the adjustment must be made manually by the user.

## Find Window Edges Button

This button can be used with Impulse response data, in some cases, to locate the optimum positions for the leading and ending edges of the data window used for removing echoes from the

data to be converted. For time domain data which is not an Impulse response, this button should not be used. In general, an initial data acquisition (usually of an Impulse Response type) should first be performed.

You can inspect the placement of these edges after an impulse response data waveform is acquired (use the Time/Frequency button if Time Domain data is not currently shown in the Primary Plot).

This facility can be fooled by excess noise in the measurements, or by a large number of reflections (including cabinet effects) following soon after the main impulse response. Optimum placement of the ending edge may also be a matter of judgement -- you should verify that you are satisfied with the placement of such markers before using the result regularly.

### Averaging

Averaging is a process of combining the data from a number of (usually identical) measurements. Similar to averaging is a decay time parameter that can be selected for certain spectrum measurements. Decay time determines how quickly effects of previous spectral content pass from the displayed data.

To use Averaging or decay, select the type to use from the drop-down box in the Averaging control, and the number of samples to average in the Numerical Value Control. The "Average/Decay ON" checkbox must be checked for these to have any effect. When the number is reached, the acquisition process will automatically stop (if "Auto Stop" has been selected) or will continue (but will show data which is mostly due to only the selected number of acquisitions, or within the specified decay time).

To Restart an Averaging Process, click on the underlined "reset" label on the Acquisition Configuration Form.

### **For Noise Reduction:**

Averaging is commonly used to minimize the effects of noise in measurements. This is particularly effective with the Coherent type of averaging (usable on Synchronous or ReSynch type measurements, which are Time Coherent). Averaging reduces noise in a measurement because repeated stimulus signals and acquisition processes can be expected to be identical for each occurrence, but the noise and transient interferences will be different each time. Averaging therefore reinforces the "real" results (which are the same each time) and smooths over noise and interferences which are likely different each time.

With Coherent Averaging, the signal to noise ratio can be expected to increase on the order of 3dB each time the number of averages doubles. (This will be limited of course by the overall resolution of the soundcard hardware).

### **For Transient Smoothing:**

When used in modes other than "Coherent", averaging can combine an ongoing series of acquisitions to provide an overall "typical" result. For example, if you are monitoring the noise spectrum in a room using an RTA, and a telephone were to ring during only one frame of the RTA, that frame would not be representative of the general room noise. But if the RTA were to be averaging a large number of such frames, the spectrum might then be considered to be an average of the expected noise energy.

### **For Spatial Averaging:**



When measuring frequency response of a loudspeaker or audio system in a room, the placement of the microphone (or listener) has a strong effect on the result. To obtain a more typical result, without particular oddities due to a specific placement of the microphone, a series of measurements can be combined using non-coherent averaging, while moving the microphone over various regions. This can also be accomplished with a series of measurements in a Progressive Average postprocess.

### Types of Averaging:

- **Log**: the result derives from the anti log of the (average of the) Logarithms of the raw values. Phase values, if any, are set to zero.
- **Mag**: the result derives from the average of the magnitudes of the raw values. Phase values, if any are set to zero.
- **Rms**: the result derives from the root of the sum of the squares of the raw values, divided by the number of values processed.
- **Coh**: this is a coherent process similar to "Mag", but operating on synchronized data. It is usable only with time domain data or with complex frequency domain data (i.e., frequency domain data having valid magnitude and phase values). Phase values, if present, are also averaged. This type averaging can greatly reduce noise (in a synchronous or ReSynch measurement).
- **Decay**: the result simulates an analyzer using analog detectors having a time constant of the specified value. The result will be more accurate for longer decay times, relative to the update rate of a given measurement.

### Auto Stop

The number of samples to average is specified in the Average control of the Acquisition Configure Form.

When that number of acquisitions is reached, the acquisition process will automatically stop if the "Auto Stop On Limit" checkbox has been checked. This will occur ONLY if the "Averaging/Decay ON" checkbox is checked.

You can also configure some acquisitions to automatically stop after a predetermined time. For this, select the AutoStop Timer radio button and configure for the desired time period which you wish the measurement to run. This option is usually used with averaging set to "Decay" mode.

### Propagation Time

This value can be set in the Acquisition Configure Form for Asynch type acquisitions (and also for some Chirp-based measurements and for "Impulse Response Live Monitor" or "Frequency Response Live Monitor" type measurements). The value of propagation time is specified in units of seconds using a Numerical Value Control. It is not used in Synchronous measurements.

A requirement in measurements is that the Acquisition is collected during the time that the effect of the output Stimulus, on the device being tested, can be expected to be ready for measurement. If you expect there to be significant delay before this effect is ready, you can adjust a "propagation time" parameter (available in the Acquire Configure Form) to cause Praxis to wait before collecting the result.

Accurate setting of the Propagation Time is important when using random stimuli such as noise or musical program material, with "Normalize ch1/ch2" spectrum analysis or "Live Monitoring" measurements.

Triggering Acquisitions can be configured to start on rising or falling edges of either input, when passing through a specified level, with the trigger point positioned at a specified percentage of the time frame of the acquisition. Using this, Praxis can be configured to wait for a transient edge, and when found, to record it in time or frequency domain.

If the trigger is used to find the beginning of a longer data collection (such as a max-hold FFT or a long decay time RTA, along with an Auto-Stop on Timer setting), you can also configure the acquisition to continue from that point ("Free Run after Trigger").

### Isolation of Distortion

When a synchronous type Stimulus is used and the Acquisition type is either FFT 1ch or Time Domain 1ch, an option appears on the Acquire Configuration form to allow you to "Isolate Distortion". This is a very sophisticated feature that requires some explanation.

When a loudspeaker or other device processes an input signal, the output signal can be thought to consist of two parts:  
the "**linear part**" and the "**noise and distortion part**".

The "linear" part is that which is explainable by the device's Frequency Response (or equivalently, by its Impulse Response). This linear part can be calculated knowing stimulus signal and the device's Impulse Response. In signal processing terms, the linear part is equal to the time domain Stimulus after it has been convolved with the Impulse Response of the device.

The "noise and distortion" part is the part of the output signal which can't be predicted by knowing the Stimulus and the Impulse Response of the device. If the device being measured is time invariant, then the noise will be different with every measurement, and the distortion will be repeatable. The distortion can consist of harmonic distortions, intermodulation distortions, spectral contaminations, etc.

PRAXIS can calculate the linear part using convolution, as described above. PRAXIS can then isolate the distortion and noise by subtracting this linear part from the signal that we actually measure at the output. The noise component can be reduced, if desired, by averaging.

To use the "Isolate Distortion" option effectively, certain details must be observed.

- You must first define the setup arrangement and *keep it constant* throughout the test. For loudspeaker measurements, this must be very exacting, and includes the placement of the device, the placement of the microphone, and the position of all moderate or large sized objects or surfaces in the room -- *including the human operator!* Room temperature and humidity can also have effects. It is best that the impulse response measurement (see below) be made immediately before the FFT or Time Domain acquisition, to minimize changes in conditions.
- A special Impulse Response measurement must first be made for the test, under the same conditions as will be used for the FFT or Time Domain acquisition. The Impulse Response must be measured using the "Chirp (synchronous)" stimulus and the "Impulse Response via norm Chirp" acquisition. This is because we need to determine the impulse response of the system all the way back to the raw digital Stimulus input, not just to the amplifier or soundcard output. Be sure to choose the proper input source for the measurement.
- The impulse response should be measured at a low drive level, to assure that it correctly characterizes the linear response of the device being measured. But the gain of all components **MUST** not be changed after this impulse response has been measured. Therefore, use a low level (such as -30dB) for the Output 1 level, as controlled by the Levels

- Form, when measuring the impulse response. The level can then be turned back up as needed to reach desired test levels when the FFT or Time Domain measurements are made.
- The impulse response should be made using a long (perhaps 4 seconds) chirp, to minimize noise in its determination.
  - It is best (but not absolutely necessary) if the Measure Input gain (on the Levels Form) is also left constant throughout the tests. This will help to minimize any gain errors during the test.
  - After the impulse response is made, save it to a ".px2" file. It will be called back later.
  - Then, change the stimulus to the desired synchronous type, typically "WAV file (synch)" for a Time Domain acquisition or "Wavelet Burst" for an FFT acquisition. Choose the desired acquisition type, and then in the Acquisition Configure" window put a check mark in the box labeled "Isolate Distortion". You will be prompted to open a "Reference IR file", so select the impulse response measurement that you made above.
  - Whenever you change any PRAXIS settings for the acquisition (including windowing edges), be sure that the "Calc new expected data" checkbox is also checkmarked so that the "linear" part of the signal is recalculated to include any changes you have made.

After you do the above, and make a new Time Domain or Frequency Domain acquisition with your synchronous stimulus, the result will have several parts. The first field will be the "linear" or "expected" signal as calculated from the Stimulus and impulse response. There will also be an additional field, showing the noise/distortion part. For a low distortion, low noise device, this field should be rather low in level.

In FFT measurements, the level of this noise/distortion part will be higher than the "linear" part in frequency regions where the stimulus had low content. This is probably mostly from noise, but could also be from spectral contamination products.

In Time Domain measurements, the result will be in two channels. The channel 1 (Left) will be the "linear" part and the channel 2 (Right) will be the "noise and distortion" part. This is most meaningfully investigated using headphones, to listen to the two components with the "File, Listen" menu of Primary Plot form. You can also export the resulting file as another WAV file. This is most instructive to help determine subjectively how much of a loudspeaker's sound is from nonlinear distortion, and how much is from frequency response and room effects.

### Wavelet Acquisition Parameters

There is a set of special acquisition parameters that are used for Wavelet type acquisitions. These may be accessed using the "Analysis Parameters" button of the Acquisition Configuration Form. These parameters are:

- Limit Condition: This is the type of distortion that is to be analyzed. Currently, the available options are Compression, Total HD (harmonic distortion), 2nd HD, and 3rd HD.
- Use Time Domain Windowing, and "cycles min": This should normally be "checked" when measuring loudspeakers or transducers to minimize noise, but will have no effect for input sources other than microphones. When selected, the Windowing for this acquisition is applied so that it will follow the user-defined edges, except when the length of the window is less than the number of test frequency cycles that are specified in the control at the right of the checkbox. In that case, the right window edge will be moved to the right so that the specified number of cycles are maintained in the window length, up to a maximum of the specified FFTSize. Usually the "cycles min" value should be about double the number of cycles ("Shape Length") that are being used for the Wavelet Burst Stimulus.

To set the windowing edges, start an acquisition and stop it after the first update of the Primary Plot Form. Then click the yellow clockface icon at the bottom of the Primary Plot to show the time domain data, and use the Format menu and the Time and Windowing tabs to set the windowing edge. The starting edge should be just before the beginning of the time domain response to the burst stimulus, and the stopping edge should usually be very close after the starting edge (so that the "cycles min" value is used for all test frequencies).

- **Bandwidth:** This specifies the bandwidth, in fractional octaves of the stimulus test frequency, that are to be analyzed for each harmonic of the tone. For example, "3" indicates 1/3rd octave. It should normally be set to approximately the same value as is reported for the -3dB bandwidth in the "Shape" panel when configuring the Wavelet Burst Stimulus.
- **Max Harmonic:** For "Total HD" only, specified the highest harmonic that will be analyzed. Remember that only harmonics that are below half the sample rate can be detected, so if consistent results are desired across the measured stimulus frequency range, this value should be kept to a low value (recommended: 3).
- **Relative Distortion Acceptance Limit:** For HD, this is the dB value of the analyzed distortion components, relative to the expected level at the stimulus frequency. For example, -40dB would be equivalent to 1%, -34dB would be equivalent to 2%, and -30.5 is 3%. For Compression, this is the maximum acceptable change relative to the expected level at the stimulus frequency. Further testing beyond the applied test level will not be continued for this frequency after this limit is no longer met.
- **Averaging:** Averaging can be used to reduce noise effects in the measurement. The averaging can be Constant for all applied levels and frequencies; can be forced to even numbers (for optimum rejection of power line frequencies); or can be scaled with level and/or with the strength of the frequency response at the test frequency. These scalings can help in reducing the test time when averaging is used.
- **Reference Response:** This is the filename of the frequency response measurement described above. It should be made before the testing is done, with the same equipment, microphone placement and room arrangement.

## ***Acquisition Types***

Praxis provides the following Acquisition Types:

- Time Domain (1ch and 2ch) Acquisitions
- Impulse Response via MLS Acquisition
- Impulse Response via Chirp Acquisition
- Spectrum FFT (1ch and 2ch) Acquisitions
- Spectrum RTA (1ch and 2ch) Acquisitions
- Frequency Response Mag and Phase Acquisition
- Freq Response Mag (1ch and 2ch) Acquisitions
- Frequency Response Live Monitor Acquisition (not yet supported)
- Impulse Response Live Monitor Acquisition (not yet supported)
- Impedance Acquisition
- Component Track (1ch and 2ch) Acquisitions
- Frequency Response and Harmonic Distortion Acquisition
- Frequency Response and IMD Acquisition
- Spectrum 2ch FFT cpx Acquisition
- Wavelet Maximum Level Capacity
- Wavelet Single Frequency Acquisition

## ***Time Domain (1ch and 2ch) Acquisitions***

This acquires raw time domain data, with no processing at all applied. The result is similar to that of an oscilloscope.

For all stimulus except "Impulse", this does NOT give the impulse response of the system being measured -- MLS and Chirp stimuli require further processing to obtain that.

This stimulus is useful for seeing just what signal is being seen by the soundcard, and to check for overloading of devices in the system.

It is also useful for tone burst investigations (used with Envelopes).

### ***Impulse Response via MLS Acquisition***

This type presents the Impulse Response (IR) derived from applying an MLS or MLS/Pink (Synchronous or ReSynch) stimulus.

It is not the raw result from the unit under test (because the stimulus was not a simple impulse), but is processed to achieve the equivalent impulse response with improved immunity to noise contamination.

Note that if a "Pink" type stimulus is used, the time domain IR seen is also "Pink" weighted toward lower frequencies. This is normally used, however, in impedance measurements or frequency response measurements, so the effect of this spectral weighting cancels in the normalization processes inherent in those measurements.

### ***Impulse Response via Chirp Acquisition***

(Using a cal probe input)

This provides measurement of an Impulse Response (IR), a time domain data set which contains all information about a linear system's frequency response. Deriving the frequency response via an IR allows echoes to be windowed out of the measurements (yielding a quasi-anechoic response of a loudspeaker, at least at higher frequencies, without need for an anechoic chamber).

Using a Chirp can provide a high dynamic range for optimum determination of room acoustic parameters. **A chirp derived impulse response using this acquisition type accounts for microphone calibration data, and for any response aberrations in the stimulus channel, giving a completely "calibrated" impulse -- this is the only impulse response measurement in Praxis that provides these features directly** (with other stimuli, the microphone correction and stimulus normalization happen only after data is transformed into the frequency domain).

To obtain an chirp stimulus impulse response that does not correct for cal channel response (but which also provides even greater dynamic range and the capability of simultaneous dual-channel results), use the "Impulse response Asynch (1ch or 2ch)" type acquisition.

### ***Spectrum FFT (1ch and 2ch) Acquisitions***

These Acquisition types perform a Fast Fourier Transform (FFT) on the (perhaps Windowed) data, and present the frequency domain results in the Primary Plot. This provides what is known as an FFT type Spectrum Analyzer.

The resulting data is magnitude only (there is no meaningful phase content). This acquisition is useful for monitoring signals in the frequency domain. It is also used in spectral contamination investigations.

FFT data is inherently reported in equally spaced frequency "bins", which are spaced  
 **$(\text{sample rate})/(\text{FFT Size})$  [Hz]**

from each other. The resolution is enhanced, and the data file size is increased, with use of larger FFT sizes.

Single tone levels will be affected little by changes of resolution (FFT Size), but noise will drop as resolution become finer -- this is because a lower portion of the broadband noise will be in each bin as the bins become narrower (while, for a tone, it is essentially either within a given bin or it is outside of it).

This type acquisition can also be used in 2 channel mode to measure frequency response magnitudes using the Normalize ch1/ch2 option and a wideband signal source (including program material such as music). To measure frequency responses this way while also obtaining phase information and/or with capability of removing echoes, use a Frequency Response Live Monitor Acquisition.

In single channel acquisitions, you can also use Max Hold or Min Hold to collect and display curve maxima or minima.

## ***Spectrum RTA (1ch and 2ch) Acquisitions***

These Acquisition types derive an "RTA" type Spectrum Analyzer, utilizing data from a high resolution FFT. The term "RTA" stands for "Real Time Analyzer", because analog versions of this type analyzer were often used for monitoring signals in real time. However, the RTA in Praxis (and other software based analyzers) does not analyze the data continuously, but in batches. This is seldom a concern, but it does imply that the RTA can not be used as something like a "gunshot detector".

The resulting data is magnitude only (there is no meaningful phase content).

RTA data is inherently reported in frequency "bins", whose width is a constant fraction of an octave, when compared to their center frequency. The bins get proportionally wider as the frequency increases. You can select the fraction of an octave which will be resolved, and this setting, along with the FFT size, results in a minimum bin frequency at which a meaningful result can be reported. The lowest frequencies that can be then resolved improves with use of larger FFT sizes.

Single tone levels will be affected little by changes of resolution, but noise will drop as resolution become finer -- this is because a lower portion of the broadband noise will be in each bin as the bins become narrower (while, for a tone, it is essentially either within a given bin or it is outside of it).

When using an RTA to investigate or monitor in-room frequency response of an audio system, Pink Noise is usually used as the stimulus. This is provided in Praxis Free "Demo Mode".

This type acquisition can also be used in 2 channel mode to measure frequency response magnitudes using the Normalize ch1/ch2 option and a wideband signal source (including program material such as music). To measure frequency responses this way while also obtaining phase information and/or with capability of removing echoes, use a Frequency Response Live Monitor Acquisition.

In single channel acquisitions, you can also use Max Hold or Min Hold to collect and display curve maxima or minima.

### ***Normalize ch1/ch2 and the Coherence Function***

**(this feature is available only with the Audpod -- it is not enabled in Demo mode):**

When 2 channel (2ch) acquisitions are being taken with the Spectrum FFT or Spectrum RTA Acquisitions, an option is available to "Normalize ch1/ch2" . This means that the data from the measured Channel1 is normalized (magnitude divided, or subtraction of decibels) during measurement, by the data from Channel2. This gives a type of frequency response result, provided that the spectral data in channel 1 can be assumed to be due to a response to spectral data sensed by channel 2.

To measure frequency responses in a similar way, while also obtaining phase information and/or with capability of removing echoes, instead use a Frequency Response Live Monitor Acquisition.

To use the Normalize ch1/ch2 option effectively, averaging of acquisitions is required. This allows reduction of noise and accumulation of sufficient energy in each spectral region. The signal supplied to channel 2 must also have adequate energy at the frequencies being measured (it is not possible to measure response at a frequency without supplying sufficient signal at that same frequency to the tested item!). Praxis' averaging operation, when in this mode, uses a weighted factor determined by the energy in the frame and at each frequency -- this allows Praxis to reject invalid data on a frequency and frame basis to minimize contamination which would otherwise be caused by insufficient spectral coverage.

Similarly, the samples in the time frame sensed by channel 1 must correspond to the samples in the time frame resulting from the stimulus. In cases of considerable time delay between loudspeaker and microphone, for example, care must be taken that the response sensed by the microphone was caused by the signals existing earlier in the electrical signal supplied to the loudspeaker. To accomplish this, the Propagation Time parameter should be set to the expected amount (in seconds) of delay.

#### **Coherence Function:**

Any spectral effects in the Channel 1 data that are not linearly attributable to the corresponding spectrum in the channel 2 data will give erroneous results in a normalized measurement. Causes of this contaminating could be noise, distortion, mismatched time frames (propagation delays), or insufficient resolution.

When these type measurements are being made, the second curve of the Primary Plot will show a trace labeled "Coherence Function". This can be used as an aid in determining when and where the measurement data can be trusted.

The Coherence function is effectively a measure of consistency in the measured response. For instance, if in succeeding average frames, the spectral line at 1kHz were to increase as sensed in channel 1, while it decreased as sensed in channel 2, that would indicate some reason to suspect contamination in the measurement at 1kHz -- a fall in the input energy should cause a fall in the output energy at the same frequency, not an increase. Phase effects should also be consistent -- if the normalized phase response were to shift greatly in succeeding averages, that would be a sign of something being amiss.

The coherence function gives a numerical result related to the consistency of the result with each averaging frame. A value of 1 means "high confidence", a value of 0 means "no confidence". With a single average, the coherence function is always 1, because it, of course, completely agrees with itself (though you still shouldn't trust it, because you have only one average -- one "opinion" -- of the response!). But after, say, 10 or more frames are averaged, you can use the Coherence Function trace to help you determine whether your measurement is accomplishing what is needed.

#### **Example usages of Normalize ch1/ch2 measurements:**

An example from loudspeaker or sound control would be to feed channel 2 input from program material that is going into a loudspeaker while feeding channel 1 from a microphone sensing the loudspeaker's output. This would give a frequency response (magnitude) of the loudspeaker in the room. Better sophistication and ability to remove echoes is available by using a Frequency Response Live Monitor Acquisition.

An example from vibration studies would be to use two accelerometers, one (channel 2) attached to the surface of a vibration table and the other (channel 1) applied to an object such as a circuit board that is being investigated for suspected resonances. The vibration table can be driven with some broadband stimulus such as shaped noise, and the resulting plot will clearly show resonances of the board in response to vibrations from the table.

### ***Frequency Response Mag and Phase Acquisition***

This Acquisition type provides the functions of a complete audio band network analyzer. You can use it to measure both frequency response Magnitude (usually in dB) and Phase (in degrees).

This Acquisitions needs measurement results from two channels, sometimes referred to as the "Measure Channel" and the "Cal Channel". A common situation will use a microphone as the input to the Measure Channel. Probe#2 is nearly always used as the input to the Cal channel (this may need to be selected in the Levels Form, or it may automatically be selected by Praxis using the RefCalSource setting available in

(MainMenu) Config | Hardware Specific.

To form the Frequency Response result, the spectrum from the Measure channel is normalized (i.e., divided) by the spectrum from the Cal channel.

For some type Stimuli (MLS and Chirp) you can make quasi-anechoic measurements of loudspeaker frequency response by first placing window edges to avoid echoes and reflections in the Impulse Response.



## ***Freq Response Mag (1ch and 2ch) Acquisitions***

This Acquisition type provides the functions of a magnitude-only audio band network analyzer.

This Acquisitions needs measurement results from two channels, sometimes referred to as the "Measure Channel" and the "Cal Channel". A common situation will use probe as the input to the Measure Channel. Probe#2 is nearly always used as the input to the Cal channel (this may need to be selected in the Levels Form, or it may automatically be selected by Praxis using the RefCalSource setting available in

(MainMenu) Config | Hardware Specific.

For 2channel Frequency Response measurements, two inputs can be normalized against a single Cal input. This is done using two separate acquisitions by the AudPod, and relies on there being no time dependence on the stimulus being used (normally, Stepped Sine).

To form the Frequency Response result, the spectrum from the Measure channel or channels is normalized (i.e., divided) by the spectrum from the Cal channel.

## ***Freq Response Live Monitoring Acquisitions***

A Frequency Response Live Monitor acquisition enables you to measure system frequency responses using musical program material or other arbitrary signals which derive from a common source. It is similar to doing an FFT measurement with "Normalize ch1/ch2" selected, except that it does provide phase information, and it also allows for windowing out of echoes in acoustical measurements (by setting window edges in the associated impulse response).

For this type measurement, the output stimulus is usually selected to be "None", because the stimulus is obtained from external signals. The external signal must be sensed, however, either through a probe or by routing a sample of the program material to the "Line" input of the AudPod. **Be careful not to overdrive the Line Input of the AudPod, as this can cause distortion, interference, or damage to the AudPod. Also, note that the two AudPod line inputs can each be selected only to different single input channels -- use the "ring" connection (channel 2) of the Line Input when using that input to sense the program material during "Live Monitoring". See: (Levels) Input Controls.**

A Frequency Response Live Monitor acquisition operates by simultaneously measuring the spectrum of the source program material (using Input 1) and the spectrum of the output of the system being measured (typically using a microphone in the room). From these two spectra, an impulse response of the system is calculated. Then the impulse response is again windowed and FFT'd back to a frequency response. Obviously, this involves a lot of calculation and can require considerable amounts of time with large FFT sizes or with slower computers.

This process can in principle provide full complex frequency response results and seems elegant at first glance. It has become a popular technique in recent years. However, it is generally quite noisy, inefficient, and slow when compared to response measurements made using chirp or MLS methods. Frequency response determination by this method should be used only when other techniques are impractical (such as in a live music venue).

For best noise immunity and cleanest results, the "Propagation Time" parameter on the Acquisition Configure Form should be set first, with information obtained by an Impulse Response Live Monitor Acquisition. This is so that both measurement channels use data samples and spectral information that are truly related to each other in determining the results.

Even with such adjustment, this Live Monitoring is subject to degradation from a number of often unavoidable factors. For instance, in live venues, any signals in the microphone channel which are not caused by signals from the source program (such as distortion, crowd noise, signals from other channels in multichannel amplification systems, loud percussion sounds propagating acoustically around the PA system, and long time reverberations) will result in degraded results. Any frequency ranges which are not adequately represented in the source program (often in the low bass or very high frequencies) will result in poor results in those frequency ranges -- when using a source program with restricted frequency content, such as from soloists, it may be advisable to mix in some Pink Noise with the program material. In multichannel setups, each channel should be adjusted separately (using mono source, with other channels temporarily silenced) if at all possible. Considerable averaging of acquired results is often required with Live Monitoring measurements to obtain trustworthy results.

Note that setting the propagation time value will affect the delay (or phase) characteristics of the resulting frequency response.

## ***Impulse Response Live Monitoring Acquisitions***

An Impulse Response Live Monitor acquisition enables you to measure system impulse responses using musical program material or other arbitrary signals passing through the system being tested.

An Impulse Response Live Monitor acquisition operates by simultaneously measuring the spectrum of the source program material (using Input 1) and the spectrum of the output of the system being measured (typically using a microphone in the room). From these two spectra, an impulse response of the system is calculated. This calculation utilizes FFTs and an IFFT, so the FFT parameter should be set to an appropriate length beforehand.

For this type measurement, the stimulus is usually selected to be "None", because the effective stimulus is obtained from external signals, usually program material, being handled by the system being tested. The external signal must be sensed, however, either by a probe or by routing a sample of the program material to the "Line" input of the AudPod. **Be careful not to overdrive the Line Input of the AudPod, as this can cause distortion, interference, or damage to the AudPod. Also, note that the two AudPod line inputs can each be selected only to different single input channels -- use the "ring" connection of the Line Input when using that input to sense the program material during "Live Monitoring". See: (Levels) Input Controls.**

The result of this acquisition is an impulse response trace. Although the resulting impulse trace could in principle be used for a large number of purposes and postprocess calculations, it is generally rather noisy when compared to impulse responses made using chirp or MLS methods. Impulse response determination by this method should be used only when other techniques are impractical (such as in a "live music" venue) or in adjustment of the "propagation time" parameter for other monitoring or "Normalize ch1/ch2" measurements (see below).

The impulse response from this measurement can be used to aid in time alignment of a multi-loudspeaker setup or installation. For this use, care must be taken to obtain a sufficiently clean measurement so that the impulse edges can be discerned from noise peaks. If possible, during alignment of loudspeakers in multichannel amplification systems you should adjust one channel at a time, with the other channels silenced. Ideally, these kind of adjustments should be made **before** the room is occupied by the audience, and using more robust means of obtaining impulse responses (such as chirp or MLS).

A primary use for an Impulse Response Live Monitor acquisition is to provide information that you can use to set the "Propagation Time" parameter on the Acquisition Configure Form, prior to making a "Frequency Response Live Monitor" acquisition, or a 2-channel FFT or RTA measurement using the "Normalize ch1/ch2" option. Properly setting the propagation delay assures that these types of measurements, which are made by comparing one spectrum to another, use related time records for calculating these spectra. For example, in an acoustic measurement, the signal picked by a microphone, from a distant loudspeaker, will be delayed from the electrical source signal because of the speed of sound propagation. If the spectra from the source and from the mic were each made from a small number of samples taken simultaneously, it would be possible that the samples taken from the microphone were nearly all due to signal that happened **before** the time of the samples recorded from the source. If this were to occur, the measurements obviously could not give an accurate indication of frequency response. The propagation time parameter effectively causes the channel 2 input to wait for the specified time before collecting its time samples.

To set the propagation time parameter for this purpose, first set the propagation time to zero and the FFT Size to a moderate value (like 8192) or so that its time length,  $FFTSize/SampleRate$ , is considerably longer than the expected delay between the microphone and the program material. Then make a measurement using perhaps 10 averages -- adjusting input levels, first, of course, and formatting the display of the Primary Plot to allow you to clearly see the impulse response. Click on the underlined "Averaging" label in the Acquisition Configure Form after adjusting the levels to average a new set of samples so you can obtain a discernable impulse response. Then, position one of the markers on the Primary Plot to the left side of the trace just before the impulse response activity happens. Read the time value of the marker (it will be displayed at the top of the marker readout area below the Primary Plot's graph), and copy this value to the Propagation Time control in the Acquisition Configure Form. Then make another measurement. When enough acquisitions have been averaged, verify that the impulse response now is displayed to the far left of the Primary Plot (i.e., that it begins soon after "time 0").

Live Monitoring is subject to degradation from a number of often unavoidable factors. For instance, in live venues, any signals in the microphone channel which are not caused by signals from the source program (such as distortion, crowd noise, signals from other channels in multichannel amplification systems, loud percussion sounds propagating acoustically around the PA system, and long time reverberations) will result in degraded results. Any frequency ranges which are not adequately represented in the source program (often in the low bass or very high frequencies) will result in poor results in those frequency ranges -- when using a source program with restricted frequency content, such as from soloists, it may be advisable to mix in some Pink Noise with the program material. In multichannel setups, each channel should be adjusted separately (using mono source, with other channels temporarily silenced) if at all possible. Considerable averaging of acquired results is often required with Live Monitoring measurements to obtain trustworthy results.

Note that setting the propagation time value will affect the resulting impulse response delay and the delay (or phase) characteristics of frequency domain values calculated from the impulse response. This is different from the behavior of the other methods of obtaining impulse response (such as via Chirp or MLS).

### ***Impedance Acquisition***

This Acquisition type uses data from two channels to determine the electrical impedance of a network (or of a transducer). The two channels should each be fed from a separate Probe (Demo Mode, Full Mode).

The measurement setup requires that the network to be measured be wired in series with a fixed resistor of known value (the "Reference Resistor"). A stimulus is applied to the combination and a measurement is made, using probes, of

(1) the voltage across both the network and the Reference Resistor in combination.

and

(2) the voltage across only the network for which the impedance curve is sought.

From these measurements, the relative voltage across the network and the relative current through it can both be found, and from these, the complex impedance (magnitude and phase) can be determined. The Praxis system must first be informed of the value of the Reference Resistor using a Numerical Value Control in the Acquisition Configuration Form.

Impedances can be measured using a number of Stimulus types, but the best choice is usually a Log Chirp type.

Impedance curves of low frequency drivers (usually in two separate states of loading) can be postprocessed to obtain the driver's Thiele-Small parameters.

### ***Component Track (1ch and 2ch) Acquisitions***

This type Acquisition tracks frequency components (energy concentrated at certain frequencies) when a given stimulus is applied. It is used with Stepped Tone stimuli (Synchronous or Asynch). This type Acquisition is able to analyze the response, of the unit being tested, for energy at frequencies **other** than those being applied in the Stimulus.

The most common use for this capability is to measure distortion products. A sequence of tones or sets of tones can be applied the unit being tested, and the levels of distortion products generated are isolated, measured, and plotted onto the graph in the Primary Plot.

The frequencies being tracked (analyzed) are defined using the Component Tracking Configuration Editor. They are defined relative to multiples (or sums and differences of multiples) of the frequencies being applied in the Stimulus. A "Total Harmonic Distortion" figure can also be tracked, if configured in this editor.

Distortion data obtained with the Component Track Acquisitions can be viewed in either dB format (relative to a measured level of a stimulus tone in the resulting spectrum) or as a percent (%) value. This setting can be selected in the Plot Form (a Distortion Tab will appear in the Format controls when distortion data is viewed in the form).

#### **The Component Tracking Configuration Editor**

Component Track data (distortion data, in most cases) is plotted versus the frequency of an applied tone from the Stimulus. This must be the first tone (Tone #1, as configured in the SteppedTone Editor) for one of the output channels. You can select the channel to use (if both are used) in the check buttons at the top of the form.

Up to 2 curves can be generated and plotted in each sequence of measurement steps. For each curve, select:

- An "A" tone (one of the applied frequency tones from the stimulus, in either channel)
- A "B" tone (can be the same as the A tone)

- The multiple of each tone (A and B), the sum of which will be the frequency where the distortion component will be tracked.
- Which tone (A or B) will be used as a reference for the results.

For example, to track the third harmonic of a single applied tone, choose:

A= tone 1, and use 3\* Frequency(A)

B= tone 1, and use 0\* Frequency(B)

**Component Tracking Configuration**

Plot measurements versus parameter

Frequency  Level of Tone #1 of Output Channel #:  1  2

**for curve #1, analyze Input Channel 1 at frequency equal to**

\* Frequency(A) A=Tone  of Output Channel  1  2

+

\* Frequency(B) B=Tone  of Output Channel  1  2

Analyze THD

Reference result to measurement at Frequency  A  B  None

**for curve #2, analyze Input Channel 2 at frequency equal to**

\* Frequency(A) A=Tone  of Output Channel  1  2

+

\* Frequency(B) B=Tone  of Output Channel  1  2

Analyze THD

Reference result to measurement at Frequency  A  B  None

Max Harmonic for THD=

To track a lower third-order intermodulation product resulting from two applied tones, choose:

A= tone 1 (the one lower in frequency), and use 2\*Frequency(A)

B= tone 2 (the one higher in frequency), and use -1\*Frequency(B)

This tracks products at frequency 2A-B, a third-order product.

If the Reference tone was selected to be A, then a result of -40dB for a given product would mean that the component level detected was 40dB below (1% of the amplitude) the level at tone A that is measured in the same input spectrum.

Alternately, a THD analysis can be made, in which all of the harmonics (up to the defined MaxHarmonic, or to half the sample rate) are combined in RMS fashion.

#### Frequency Response and Harmonic Distortion Acquisition

This acquisition provides a technique to automatically measure both frequency response shape and Harmonic Distortion using a log-frequency chirp stimulus. This is accomplished using a technique pioneered by Angelo Farina (see AES Preprint #5093) in which the chirp response is converted to an impulse response, which is separable into portions that represent the linear response to the stimulus (frequency response) and the harmonic outputs due to the stimulus (harmonic distortion).

The configurable parameters of the Chirp stimulus change when this acquisition type is used:

- Max Order: the highest harmonic distortion product to be analyzed. Curve traces will be given up to this product order, as well as for the Total Harmonic Distortion of all products of this order or lower. These curves will be available only in the second trace (trace[B]) of the resulting plot.
- Time: This is the time length of the log chirp stimulus. When measuring devices with long decay times, this chirp length must be proportionally increased. For loudspeakers, echoes, reflections and room effects should be minimized.
- Compensate for Response: This uses the measured frequency response to compensate the measured values of distortion for the effects of the response at the harmonic product frequencies.
- Maximize Windowing: when selected, the windowing (around the impulse response sections corresponding to the frequency response and the harmonic products) is set to allow the maximum time window for each that could keep them isolated. The degree of isolation achieved will also depend on the decay rate of the main impulse response.

### ***Frequency Response and IMD Acquisitions***

This acquisition provides a technique to automatically measure both frequency response shape and intermodulation difference distortion, using a similar technique to that for the Frequency Response and Harmonic Distortion Acquisition. ***This technique is experimental at this time.*** The stimulus uses two simultaneous, closely-spaced log-swept (chirp) stimuli. The two stimulus frequencies maintain a constant frequency ratio throughout the sweep.

The configurable parameters of the Chirp stimulus change when this acquisition type is used:

- Freq Ratio: the ratio of the lower frequency to the higher frequency in the stimulus. It should normally be set to approximately 0.92.
- Max Order: the highest intermodulation distortion product to be analyzed. Curve traces will be given up to this product order. The curves are for the levels of the single difference frequency tones resulting from the measurement. These curves will be available only in the second trace (trace[B]) of the resulting plot.
- Time: This is the time length of the log chirp stimulus. When measuring devices with long decay times, this chirp length must be proportionally increased. For loudspeakers, echoes, reflections and room effects should be minimized.
- Compensate for Response: This uses the measured frequency response to compensate the measured values of distortion for the effects of the response at the harmonic product frequencies.
- DeNormalize: When checked, the distortion levels will not be shown in dB relative to the response. When unchecked, the distortion levels are normalized.

- **Maximize Windowing:** when selected, the windowing (around the impulse response sections corresponding to the frequency response and the harmonic products) is set to allow the maximum time window for each that could keep them isolated. The degree of isolation achieved will also depend on the decay rate of the main impulse response.

## ***Spectrum 2ch FFT Complex Acquisition***

This acquisition type is similar to the FFT 2channel type, except that after the acquisition the result also provides a trace containing the relative phases of the two measured spectra.

## ***Wavelet Maximum Level Capacity***

This acquisition process steps through a series of log-spaced frequencies of a Wavelet Burst stimulus, and adjusts each output burst from low level to the highest level for which the device under test still meets some distortion limit. The input gain setting and averaging are automatically handled, and the frequency response of the system is taken into account in evaluating distortion. The result is a plot of Maximum Peak Level Capacity vs. Frequency, which appears in the Auxiliary Plot. The purpose of this acquisition type is to find out how loud a speaker or system can play, on a short term basis. There is a script in the Praxis installation that demonstrates and further automates this.

This is a very powerful and useful test, but it requires some care in its execution. Background noise, in particular, can be a problem. This can be mitigated by using higher amounts of averaging (at the cost of considerably higher measurement time) and by using time domain windowing (see below). Windowing in this instance is not intended to remove echoes from the response (which is not generally practical at most frequencies for this kind of test), but to eliminate excess noise from intruding during the decay time after the applied burst.

Also, the characteristics of the test microphone must be considered. It should have adequate level handling of its own so that it does not degrade the result of the loudspeaker being measured. It should also use a Mic Correction file so that determined levels are analyzed correctly.

You need to first provide a Reference Response file for the system to be measured. The Reference Response is a frequency response measurement that is used (along with the **Power Amp Gain** parameter of the Stimulus Configuration form) to relate the soundcard's drive level to the expected output from the device under test. This Reference Response should be measured using the Chirp (ReSynch) stimulus and the Frequency Response Mag and Phase acquisition, and should *include* effects of echoes (in other words, use a very long time domain window with "flex windowing" enabled, or don't use windowing at all) during the measurement of the Reference Response. Save the response to a data file before beginning the Wavelet Maximum Level Capacity measurement.

The testing uses an algorithm that increases the test level in fixed dB Steps until the distortion Acceptance Limit is exceeded, and then the levels are again reduced by the dB Resolution until the distortion value is within the Limit. These Step and Resolution values (as well as other Stimulus parameters) are controlled from the Configuration form for the Wavelet Burst Stimulus.

The results of the test show on a graph in the "Aux Plot" showing output level capability as a function of frequency. The horizontal axis is frequency, and the vertical axis is output level. Two curves, one above and one below the measured curve, will display as the test is begun. These curves define the range of levels over which the test is conducted. If the measured curve lies on either of these lines at a frequency, the measured result should be interpreted to mean "at or above" (for the upper curve) or "at or below" (for the lower curve).

There is a set of special acquisition parameters for Wavelet type acquisitions. For descriptions of these, please see the section Acquisition Parameters.

### ***Wavelet Single Frequency Acquisition***

The Wavelet Single Frequency type generates a plot of distortion or compression vs. output level, at a single nominal burst frequency. The frequency of test is the "Start Level" of the Wavelet Burst Stimulus, and the test begins at the starting level, increasing successively by the step size. The test will stop when the distortion or compression no longer meets the Distortion Acceptance Limit (see Wavelet Acquisition Parameters).

This test is useful for investigating how distortion or compression increases with applied level of a short-term stimulus. It is also useful for helping to define a limit to specify for the Wavelet Maximum Level Capacity test.

### ***Impulse Response Asynch (1ch or 2ch) Acquisition***

This provides measurement of an Impulse Response (IR), using a chirp stimulus, but without using a "Cal" probe input. This has advantages when acquiring Impulse Responses that must have maximum dynamic range, such as when analyzing room acoustics. The use of a single analog input to make the measurement avoids noise and bandwidth limiting effects from the Cal probe input and the power amplifier. Instead of a Cal input, a mathematical representation of the stimulus signal is used internally by PRAXIS to transform the signal to the corresponding uncalibrated impulse response.

The 2-channel version of this acquisition type also provides a way to acquire two-channel (binaural or stereo) impulse responses, which can be used with the "Time Domain Math->Convolution" postprocess to perform binaural room auralization simulations.

There are two disadvantages of this type of acquisition, however, compared to the "Impulse Response via Chirp" Acquisition:

- The response of the power amplifier, microphone, or any soundcard response variations, are NOT removed from the measurements, so you must make sure that these have a flat enough response in the frequency range of interest. With good quality (non-consumer) soundcards, condenser or electret measurement microphones, and unequalized power amplifiers, this requirement is easily met.
- The resulting impulse response does not provide information about total delay or sensitivity -- time and level are both relative, not absolute. You cannot, for instance, use the IR from this type acquisition to infer the distance between the signal source and the measuring microphone nor the absolute SPL sensitivity of the system. However, with the 2-channel version, the levels and delays between the channels will still track each other, so relative delays and gain between binaural channels will be in proper relationships.

## **Post Processing Operations**

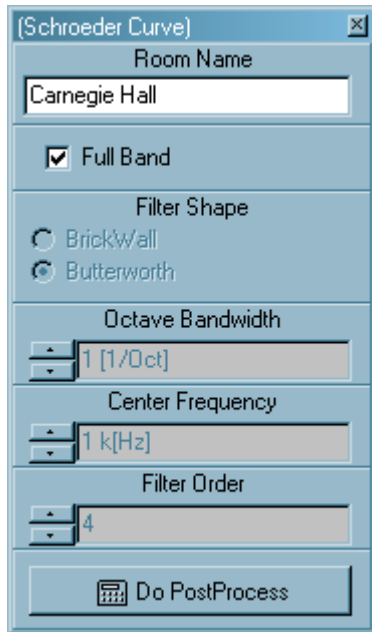
### ***PostProcess Parameters***

The available parameters (settings), which you can select in the PostProcess Configure Form, can vary with the type of PostProcess that you have selected. The Configure Form can not be made to appear at all if the data type currently in the PostProcess Source Plot is not compatible with that PostProcess.



The controls and labels inside the PostProcess Configure Form will change, and the height of the form may change to adapt to the changed contents (you may need to move the form to view all of its offered settings). The appearance of the PostProcess Configure Form will vary greatly with selection of PostProcess type. The name of the PostProcess type will be displayed in caption of the form.

You can hide the form by clicking on the 'X' button at its upper right (or clicking on the "Hide Configure" button for PostProcess). The Configure Form may be unhidden again by clicking on the "Configure" button.



At the bottom of most of this form, there will usually be a "Do PostProcess" button. This button duplicates the effect of the Main Form's PostProcess [Apply] button.

## ***PostProcess Types***

Praxis provides the following PostProcess Types:

- ReMap to New Frequencies
- Filter
- Math
- Paste Over
- Smoothing
- FFT
- Inverse FFT
- Thiele/Small Parameters
- CSD Waterfall
- Step Response
- Hilbert Transform
- Schroeder Curve and Room Acoustics
- Group Delay
- Polar Compilation

- Combine Frequency Response and Impedance
- Analytic Signal (ETC)
- Power Cepstrum
- Quasi-Anechoic Bass Response Restoration
- Progressive Averaging
- Tail Correction
- User Defined (not yet supported)

### ***PostProcess: ReMap to New Frequencies***

This process can take a set of frequency domain data (impedance, frequency response, or spectrum) from the Target Plot and remap the data onto a different set of frequency points. The data points are fitted using a cubic spline. This is not usable with RTA data.

You can select the new set of frequency points to be:

- **FFT format:** The points will be evenly spaced from 0Hz to 1/2 of the given Sample Rate (as specified on the PostProcess Configure form). This kind of data format can be converted (if it contains genuine phase data) to time domain data via an IFFT.
- **Log Spaced:** the points will be spaced at even multiple from each previous point. This minimizes the files size when the data is such that it will normally be viewed on a "Log" type plot. You must specify the Start and Stop frequencies, as well as the total number of frequency points to use.
- **From Source Plot:** the frequency set from the PostProcess Source Plot will be used to map the data in the Target Plot.
- **From Frequency List:** the values will be mapped to values from a text-based frequency list. Each line of the list should contain one frequency, given in ascending order. Buttons are provided to allow you to select an existing list, to Edit a list, or to generate a new one.

You can also apply a Gain to be added to the data, or a Delay value to remove (applicable only in data with phase fields).

### ***PostProcess: Filter***

Use this process to produce the result of passing the frequency domain data (response or spectrum) from the Source Plot through a filter. The result is put into the Target Plot. This process is not usable with Impedance or RTA data.

This can be used to reject or enhance the data in certain frequency regions. With a "Brick Wall" filter type, it can be used to effectively delete regions of data (set the magnitude to very low values), so that the partial data set can later be combined with other partial data sets of frequency domain data in a Math PostProcess or a Paste Over PostProcess. By this method, separate frequency response plots can be spliced together.

The filter types can be any of: Low-Pass, High-pass, Band-Pass, Band-Reject, Flat Response (useful for simply applying a Gain or a Delay to the data), or "Target to 0dB" (this replaces the entire data array with the value "0dB"). "Target to 0dB" is handy for generating a flat-response dummy data set, to get a clean shape to experiment thereafter with the other filter shapes.

Depending on the filter type, you can also select the corner frequency for the filter or the extent of

the frequency range being affected.

Filter shapes can be any of "Brick Wall" (sheer drops in response outside of the passbands, with no change in phase), Butterworth, Chebyshev (0.1dB or 1dB ripple), or Bessel. If you select "Mag Only", only the response Magnitude will be affected by the filter (there will be no effect on the phase). Filter order affects the steepness of the filters (except for "Brick Wall" which is always steep).

You can also apply a Gain to be added to the data, or a Delay value to remove (applicable only in data with phase fields).

### ***PostProcess: Math***

This process combines the data in the Source and/or the Target plots mathematically, and puts the result into the Target Plot. If the data points in the two sets do not match, the Target plot is first mapped to the frequency points of the Source Plot before the operation. This PostProcess is not usable with Impedance data.

The supported math operations are:

**Sum, Difference:** add or subtract the effects of the data. This is an analog of the process that occurs when output from two loudspeaker drivers combines acoustically in air. The operation occurs on the real and imaginary values of the data.

**Mult, Div:** this is equivalent to adding or subtracting the data values that are in decibel and angle format. The Div process is also called "Normalization".

**MaxMag:** each point in the data set is assigned from whichever point (from Source or Target) has the larger magnitude.

**MinMag:** each point in the data set is assigned from whichever point (from Source or Target) has the smaller magnitude.

**MagInv:** Target values magnitude become the  $1/(\text{mag of the Source})$

**Phase Inv:** Target becomes the Source data, but shifted by 180 degrees.

You can also apply a Gain to be added to the data, or a Delay value to remove (applicable only in data with phase fields).

### ***PostProcess: Paste Over***

The data in the Target Plot's data which is (Above or Below) the Break Frequency is replaced with data from the Source Plot. Other options are to clear the entire target (to very small values) or to replace the entire Target data with the Source data. After this operation, the assigned frequency points are usually combination of the points from both Plots.

This is usable for splicing together Frequency Response plots.

You can also apply a Gain to be added to the data, or a Delay value to remove (applicable only in

data with phase fields).

### ***PostProcess: Smoothing***

This process takes "busy" data curves, and smooths them. You can choose to have the Smoothing also operate on the phase (the success of this depends somewhat on the density of the data).

It works by first assigning the frequency points of the Source Plot to the Target Plot. Then it replaces each point in the Target Data Set with a weighted average of all Source Plot data points that are within the Octave Range Limit of that point's frequency. The weighting is determined by the Filter Shape.

If the Filter Shape is "BrickWall", all points outside the "Octave Bandwidth" are ignored and all within it are weighted equally. This shape results in the faster processing, with good results.

If the Filter Shape is Butterworth, the points are weighted by a curve that follows a Butterworth Bandpass shape (of specified Filter Order). Data Points beyond the Octave Range Limit are given no weight (they are ignored). This usually gives the best result, but is much slower to operate.

### ***PostProcess: FFT***

This process converts Time Domain data to Frequency Domain data via the Fast Fourier Transform. You can choose the FFT Size to use for this Post Process operation, and can Window the data if desired (with half or full windows) before the postprocess.

### ***PostProcess: Inverse FFT***

You can use this process to convert complex Frequency Domain data to time domain (usually to form an impulse response). There are no options or parameters to set for this process.

Praxis does not require the frequency domain data points to be at any particular spacing -- Praxis will map the points as required for its process. However, a useful result will require that the data be somewhat complete, with sufficient detail over all regions of the frequency band.

### ***PostProcess: Thiele/Small Parameter Extraction***

This process analyzes the impedance data made of a loudspeaker driver and extracts the Thiele-Small parameters which correspond to the modeled data.

To use this, very accurate impedance data is required, as is information about the driver cone's Effective Diameter. (The units of measure used can be specified using the Main Form's "Config | Preference | Units..." menu).

For the full Thiele Small set, you need to generate and analyze two impedance curves: one with

the driver in Free Air, and one with the driver's cone weighted by a Mass (usually applied via a soft substance such as modelling clay). The mass of the cone is to be Entered in the provided Numerical Value Control.

The curve fitting operation is quite processor intensive. You can observe its progress by first checking "Animate Curve Fits" before extracting a Model (the animation process will further slow the process slightly).

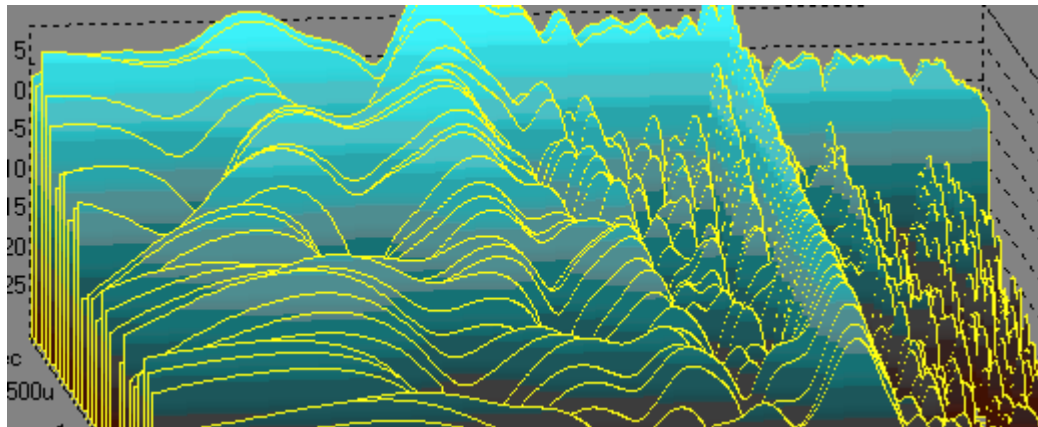
The procedure should be done as follows.

- 1) Click the [clear models] button to remove extraneous data from the modeler.
- 2) Get the impedance curve for the driver in Free Air into the Source Plot (by measurement or loading it from a file). Be sure the Effective diameter is entered correctly, then click [Free Air Model].
- 3) (Optional, for full T/S data): Get the impedance curve for the driver when mass loaded into the Source Plot and make sure the weighed amount of added mass is correctly entered. Click [MassAdd Model].
- 4) When all required data sets have been Modeled, click [Extract]. This will bring up another form showing the results. You can print or save this data to a file (but some operations are not allowed in Free/Demo Mode).

The added mass must be sufficient to drop the extracted resonant frequency by at least 25%. Too much mass, on the other hand, may too severely unbalance the suspension of the driver -- some experimentation may be required.

### ***PostProcess: CSD Waterfall***

This transforms Impulse Response data into the Cumulative Spectral Decay Waterfall plots with which many have become familiar in recent years. A Waterfall plot essentially gives a graphical display of resonance and breakup behavior of a loudspeaker cone. You can find more information about Waterfalls in an articles on the Liberty Instruments web site.



The Waterfall process uses a series of FFTs. You can select the FFT size that will be used, the Windowing that will be applied (set the Window edges using the "Windowing Format Tab", and the Time Increment between successive "sheets" (steps) of the plot. You can also specify the *maximum* number of steps to use in the plot, to limit the processing time.

Praxis lets you rotate and change the viewer's apparent height, from controls on the Plot Form's "3D Format" tab. These controls can be adjusted "live" (though Waterfalls which are very large, dense, or are from too large an FFT size can be infuriatingly slow to adjust this way).

The plot can use solid or gradient colors (to better suggest "height" perspective), but use of gradient colors is recommended ONLY after you have the plot angles positioned -- redrawing a gradient waterfall can be very slow.

The data of a waterfall plot can be saved and retrieved to disk, or dragged/dropped to another plot. Beware, however, that the data sizes of files containing waterfall plots can be very large.

### ***PostProcess: Step Response***

Step Response postprocessing converts a time domain impulse response data set (from the Source Plot) into its equivalent time domain step response (in the Target Plot).

Just as an impulse response is the reaction of system or device to an applied impulse (an abrupt change from zero to a large positive value, and then immediately back to zero) at its input, a step response is the reaction to a "step" (an abrupt change from zero to a positive level).

The impulse response is converted into the step response mathematically. The operation requires integration of the impulse response from its decayed value (its value at "infinity") back to each point.

DC errors from the measurement hardware or offsets in the impulse response data will cause visible errors in the step response (most typically, as "slope" in the "decayed" region where a flat trace, or one oscillating around a fixed level, is then expected). To compensate for this potential problem, the Step Response PostProcess Configure Form provides a Numerical Value Control to allow introduction of a DC offset value. Each change results in a recalculation of the data with the new offset value. It is normally adjusted by trial-and-error.

### ***PostProcess: Hilbert Transform***

The Hilbert Transform transforms a frequency response plot into its **minimum phase** equivalent. It has no effect on the magnitude of the data, changing only the phase data. There are no parameters to adjust for this process (just an [Apply] button).

In Praxis, the data is not required to be at any particular frequency point spacing. The data will be mapped as needed for the calculations, then mapped back to the original frequency data points.

The success of the process will depend somewhat on the completeness of the response data -- data which lacks significant portions of the lower frequencies and/or the higher frequencies is often not well enough defined for the phase response to be well inferred (though a result will be attempted by Praxis). You cannot, for instance, get a trustworthy minimum phase result if you have only a few data points from the middle of the band.

"**Minimum phase**" refers to data for which the phase portion (if unwrapped) is the closest it can be, at each point, to zero, but which still allows the system (that the frequency response represents) to be "**causal**". A causal system is one for which the output response happens only *after* application of the input -- which is obviously a requirement if the data represents any real system.

For instance, merely setting the phase values all to 0 degrees for a response curve would give the lowest phase shift, but if the data were then to be IFFT'd to the time domain (to generate an

Impulse Response or IR), the resulting IR would likely be seen to be active in its "negative time". Negative time is represented in the far right portion of an impulse response --that part can be interpreted as being a combination of the decayed portion of the response (which should be nearly zero) combined with the response before the input was applied (which must be zero in a causal system).

### ***PostProcess: Schroeder Curve and Room Acoustics***

This postprocess transforms impulse response (IR) data (made in a large room with a distantly placed loudspeaker and a microphone) in the Source Plot into a Schroeder curve in the Result Plot. The Schroeder curve is also called the reverberation decay curve.

In the Configure Form for this process, the following parameters can be set:

- a text Room Name (used for saving the data in files or printing reports).
- a check box for selecting Full Band (otherwise, a partial band will be analyzed using the filtering parameters)
- a check box for selecting calculation of the Speech Transmission Index (STI). The calculation requires an impulse response length of 800msec or longer and will be carried out for one octave about the chosen center frequency. If a full-band measurement is selected, the calculation is performed for each octave from 125Hz to 8kHz, and a weighted average (per IEC 60268-16) is given. This calculation can be very time consuming!
- If Full Band is not checked, the filter can be selected to be either a BrickWall or a Butterworth type. The filters used are bandpass types, and have no phase (time delay) effect.
- The filters bandwidth (in partial octaves), Center Frequency, and the order (steepness) of the filter can be selected using Numerical Value Controls.

After the process is performed, two curves will appear in the Target Plot and one report form is provided. The first curve is the Schroeder curve (backward integration of the squared impulse response) for the data, calculated to minimize errors due to the noise floor of the setup. The second curve is a normalized plot of the impulse response magnitude.

The Report form gives values for RT60, Early Decay Time, Initial Time Delay Gap, Objective Clarity, and Strength of Arriving Energy. Any value which cannot be determined is reported as -1. These parameters are presented to correspond to the definitions given in the AES article "A Reverberation Model Based on Objective Parameters of Subjective Perception", by Kim and Ahn (J. Audio Eng. Soc., Vol 49, No. 9, 2001 Sept, p786).

### ***PostProcess: Group Delay***

This postprocess transforms frequency response data containing phase information to a set of data representing group delay. Group delay is the negative of the derivative of phase with respect to frequency.

The operation is performed on the data using algorithms similar to those used to unwrap phase, and like unwrapped phase processes, is subject to some uncertainty in the calculation. This is because the software must make assumptions concerning whether the phase is increasing or decreasing when +/-180degree points are crossed.

The calculated values for group delay are NOT affected by Plot formatting values such as Delay Compensation (adjustable in the Delay tab of Plot Forms). To add or remove delay from a data

curve so that it will affect Group Delay calculation, instead first use the Filter Postprocess operation with Filter Type=Flat Response (other options allow you to then vary gain or delay).

### ***PostProcess: Polar Compilation***

Polar Compilation is used to collect a set of multiple loudspeaker frequency response curves into a single data entity which will be displayed as a polar radiation pattern plot. All response curves must be defined on the same frequency points. The response magnitude curve from the Source Plot is included within the data of the Target Plot, indexed by angle value (which can be edited within the Configuration Form for this postprocess).

The curves to be collected are typically measured using a common setup during a single session, with only the measurement angle (relative to the on-axis position of the loudspeaker) varied during each measurement. The measurements can be made by manually positioning the loudspeaker or the microphone, or an automatic turntable can be controlled by the user or with a custom script program.

The presentation of the polar data can be adjusted using controls on the "Polar" Format Tab of the Plot Form which contains the data.

This PostProcess can also be used in Free (Demo) mode for compiling and displaying polar data from multiple sets of response curves, but please note that the resulting data file cannot be saved to disk unless an AudPod is connected.



### ***PostProcess: Combine FreqResponse and Impedance***

This Post Process is used to combine Frequency Response and Impedance into a single data set. This combined data is for storage and inspection only -- it cannot generally be postprocessed further).

### ***PostProcess: Analytic Signal (ETC)***

This PostProcess function performs an "Analytic Signal" calculation, beginning with time domain data in the Postprocess Source plot. This provides curves for Log Envelope (also known as the "Energy Vs. Time" curve), the instantaneous frequency and the instantaneous phase.

### ***PostProcess: Power Cepstrum***

The Power Cepstrum is a process that can be applied to frequency response data. It produces a time domain curve that is formed from an FFT of the logarithm of the magnitude of the frequency response data. This can provide a visual indication of reflective phenomena in a response curve (discrete reflections cause response ripples the spacing of which are influenced by the distance to the reflecting source).

An article on the concept and application of Power Cepstrum data can be found at <http://www.libinst.com/cepst.htm> . Note that the Power Cepstrum data in Praxis is shown in decibels (with a vertical logarithmic scale), while the Power Cepstrum plots produced by Liberty Audiosuite used a linear power vertical scale.

### ***PostProcess: Quasi-Anechoic Bass Response Restoration***

Quasi-Anechoic Bass Response Restoration operates on impulse response data from a loudspeaker, as measured within a listening room. It is an attempt to reproduce the low-frequency response of a loudspeaker even in the presence of echoes and reflections.

This is based on a technique developed by Laurie Fincham (see "Refinements in the Impulse Testing of Loudspeakers", AES, March 1985), in which the impulse response of a loudspeaker is inverse filtered, using the expected response of the low frequency driver and cabinet. The goal of the filtering operation is an equalized impulse response that decays within the time period before the first reflection occurs. Then, the equalized impulse response is windowed to remove reflections, and converted to the frequency domain by FFT. Lastly, the equalization is undone in the frequency domain, leaving the full-range frequency response of the loudspeaker.

The technique requires prior knowledge or an estimate of the expected characteristics of the loudspeaker, which can be obtained from the driver and box parameters or by a near-field measurement of the low frequency section of the loudspeaker. The configuration form requires you to enter the expected -3dB low frequency cutoff point, the midband frequency against which the cutoff point should be compared (when "Flat Method" is used, the approximate filter shape (Butterworth, Chebychev 0.1dB, Chebychev 1dB, or Bessel), and the filter order. The filter order is normally 3 for a closed box and 4 for a typical vented box design. The window edge in the IR in the Postprocess Source plot should be set to just before the first reflection before applying the postprocess.

When the "Flat Method" checkbox is selected, the equalization attempts to completely remove the highpass response of the loudspeaker before windowing. Noise at very low frequencies is removed artificially, assuming a constant slope from near the cutoff frequency. This often

produces nice looking results, though the graphs tend to be “self-fulfilling” in that the low frequency shape tends to be that which was configured.

For these measurements, good dynamic range in the impulse response is very important. Averaging should be used, and the Chirp (Synchronous) stimulus is preferred.

### ***PostProcess: Progressive Averaging***

Progressive Averaging is a way to combine the frequency responses from multiple measurements (such as from several listening positions in a room) into a single frequency response curve. The averaging can be performed using several methods, but the phase response will be lost for all methods except “Coherent”. Operation of this is simple, just load (or derive from measurement) the data that you wish to include in the Source plot, with the averaging type. Then Apply the postprocess using the button, and the Target plot will include the source. Successive operations with different source data will average those data sets into the Target as well. If you wish to weight some Source plot data curves more heavily, they can be included in the Target ongoing average plot multiple times.

Data to be combined must, of course, be compatible in units, frequency ranges and formats.

### ***PostProcess: Tail Correction***

This Postprocess provides a way to adjust or correct frequency response data at frequency extremes. For example, noise and dynamic range often result in very noisy results at very low frequencies, and response rollofs or soundcard filtering result in unreliable data at frequencies near half the sample rate.

To use this postprocess, first load the Source Plot with the response you wish to correct. Then select the lowest frequency in the data curve that you consider to be reliable and enter this into the postprocess configuration form. Select the known slope for the low frequency rolloff (or use the “Auto” button to estimate it from the slope near the lower frequency in the data. Similarly enter values for the high frequencies (set to a very high frequency if you wish correction at low frequency only). Note that the high frequency slope is negative. Then use the “Do PostProcess” button to apply the result.

### ***Postprocess: Time Domain Math***

This postprocess operates on time domain data from within one or two plots (the Source and/or the Target) to provide a mathematically modified result in the Target plot. You can also specify that an amplification gain be applied to result, or that the result is normalized (set so the maximum amplitude equals 1) on each channel, or so that both channels are normalized together in the same proportion (so the maximum amplitude of the channels equals 1).

The provided functions are:

- Copy the Source data to the Target
- Copy the Source data to the Target, with the polarity inverted
- Sum the Source and Target signals (signal mixing) and put the result into the Target
- Sum the Target signal with the polarity-inverted Source signal (i.e., subtract), and put the result into the Target

- **Convolve** the Source signal with the Target and put the result into the Target.
- **Deconvolve** the Target signal (remove any effects from it that came from convolving with the Source signal) to provide (usually) a system impulse response or a dereverberated sound recording.

**Convolution** is useful for auralization investigations, to simulate the effects on sound of room reverberations or loudspeaker/room responses. To work with auralization (even in Free/Demo mode):

- Acquire a (preferably binaural) impulse response of a room or loudspeaker using an "Impulse Response Asynch (2ch)" Acquisition (or from Deconvolving a measured time domain recording of an in-room response to a Source signal, see below).
- Obtain a recording (a WAV file or time domain px2 data made with PRAXIS and and Time Domain acquisition) of a "dry" signal -- a signal having little or no reverberation content, such as a close-mic'd voice. The sample rate of this recording **MUST** match that of the impulse response data.
- You can then simulate the sound effect of the "dry" signal as it would be perceived in the room or through the loudspeaker by convolving the dry recording with the binaural impulse response into a new file. The impulse response can then, in effect, be used to store the linear "sound" of a room or loudspeaker.
- This is best experienced through headphones or "near-field" loudspeakers (to avoid the effect of the room you are in). PRAXIS v2 provides a "**Listen**" feature in the "File" menu of Plots that contain time domain data. You can use this to select a soundcard device to play back with (preferably different from the one you use to make measurements) and to play back the sounds in the data file before and after convolution.

**Deconvolution** is useful for undoing the effects of convolution. The convolution to be undone might be from a postprocess as described above; or it might come from natural convolution of an audio signal in a system, the effect of the response of a speaker or room on an applied audio input.

For instance, the following technique can be used to obtain a single channel or binarual impulse response of a room or loudspeaker (even using an external recorder and PRAXIS in Free/Demo mode):

- Apply a sufficiently wideband signal (such as a fullband log-chirp) to a system (i.e., play it) and record the result to a Wav file or to a px2 file.
- Load the file (File Open) into a PRAXIS plot using its File Open menu. The data should contain a single tme-isolated wideband stimulus (i.e., one chirp burst) and some sufficient decay time following. You can use a WAV editor or the "Time Domain Length->Use Windowed Section Only" to edit out other sections (excess dead time, multiple bursts) from the time domain data.
- Then open another Praxis plot (using a Plot's "Clone" menu or the Main Form's "View->New Secondary Plot" menu), and load into that a file containing only the original stimulus signal (a single time isolated burst).
- Deconvolve the measured signal in the first plot (the Target) with the stimulus signal in the second plot (the Source). The result should be the impulse response of the system.

## ***Postprocess: Time Domain Channels***

This Postprocess is used to configure a new time domain record from the channels of data in the Source and/or Target plots. You can use this to combine channels, remove channels, swap channels, etc. The length of the file will be that of the longest of the included channels, and if channels from both Source and Target are used, they must be of the same sample rate.

You can also specify that an amplification gain be applied to result, or that the result is normalized (set so the maximum amplitude equals 1) on each channel, or so that both channels are

normalized together in the same proportion (so the maximum amplitude of the channels equals 1).

### ***Postprocess: Time Domain Length***

This Postprocess allows you to add or remove samples at the beginning or end of time domain data records in the Source plot. The result is placed into the Target plot. Added samples are zero-valued. Adding or removing samples changes the length of the data record, while delaying the record preserves the length (though samples are lost, and new ones are zero-valued).

You can also select that the process "Use windowed section only", that is, only the time domain data that is in the selected Window region will be placed into the Target plot.

You can also specify that an amplification gain be applied to result, or that the result is normalized (set so the maximum amplitude equals 1) on each channel, or so that both channels are normalized together in the same proportion (so the maximum amplitude of the channels equals 1).

# Using Scripts in Praxis

## ***Praxis Scripts***

Praxis is able to execute special visual-based Scripts, which are complete applications (programs) in themselves that can, in turn, operate most of Praxis' features. Praxis Scripts can be *extremely* powerful, and can do a virtually unlimited number of tasks.

Scripts can be selected and started easily using the Script Launcher, which appears when Praxis starts up (or by selecting the "Scripts->Run A Script" menu option of the MainForm).

Scripts can change measurement processes depending on previous results or current conditions, or can apply further processing to the data. A Script can guide a less technical user with setup diagrams, photographs, instructions, and with visual controls such as buttons, text boxes, sliders, or even AVI movie viewers.

Script forms appear on-screen along with Praxis' normal forms. Scripts can change parameters in the Praxis forms, just like you can! **So, when a script is running, you should primarily use controls that are on the Script screen, and try not to interfere with its control of Praxis.** If you were to change Praxis' usual controls (in the Praxis Main Form or a Plot) while the script is trying to control them differently, the script (or you!) could get very confused.

Scripts can be those written by Liberty Instruments (and supplied with Praxis, such as the Free/Demo Mode scripts for finding Thiele-Small parameters) or can be developed by a user. The Liberty Script Designer application is provided with both the Free and Full versions of Praxis, allowing visual development of the Scripts in a very efficient and intuitive way. Custom user-written scripts can be developed by those familiar with Delphi or Visual Basic programming (and with Praxis operation, of course). The scripts are event-based and Praxis generates a number of events during its operation, and these events can be handled by Event Handler code in the scripts.

**Note: LAUD scripts can NOT be edited nor made to operate in Praxis, as the two programs are entirely different.**

---

### **How Scripts are Stored and Organized:**

(The following information is provided for those wishing to move or manipulate sets of script files).

All files related to a single script should be contained in a separate subdirectory (also called a "Folder").

In **Full Mode**, the script directories should be subdirectories of the directory named "VCLScripts", which itself resides under the same directory as the Praxis executable file "praxis.exe". This will allow the script to appear in the Script Launcher, for easy access. In most cases, the proper home directory for the script subdirectories will be:

**C:\Program Files\Praxis\VCLScripts**

In **Free (Demo) Mode**, the script directories should be subdirectories of the directory named "DemoScripts", which itself resides under the same directory as the Praxis executable file. This will allow the script to appear in the Script Launcher, for easy access. In most cases, the proper home directory for the script subdirectories will be:

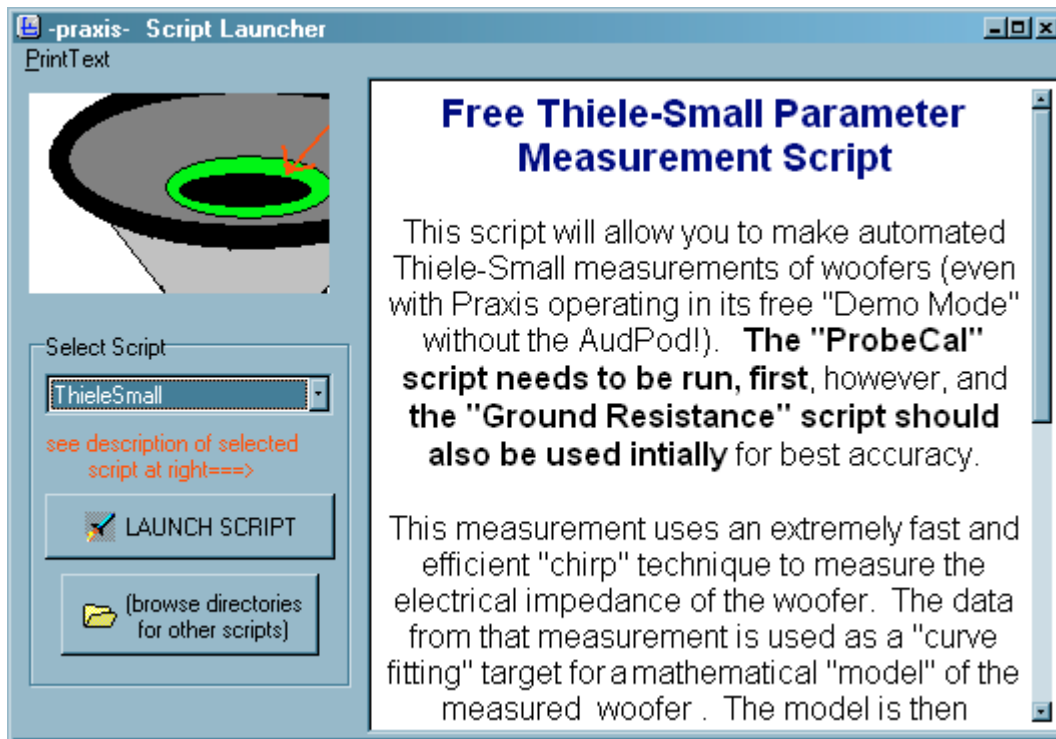
**C:\Program Files\Praxis\DemoScripts**

Ideally, each script subdirectory should contain all the following files:

- **Description.rtf** : a "rich text file", which can be created using the "WordPad" application which comes with Microsoft Windows. This should give a brief description of the script, for display in the Script Launcher.
- **Picture.bmp**: a bitmap file, width=150 pixels, height=100 pixels. This can be easily created using an application such as "Paint" which comes with Microsoft Windows. This picture can give a visual hint about the script in the Script Launcher
- **\*\*\*\*.ipr**: this file is generated using the Liberty Script Editor, and identifies the files that make up the script code and visual forms. There should be ONLY ONE of these ipr files in a script directory.
- **\*\*\*\*.dfm**: this contains data about the visual forms. It is created by Liberty Script Editor. There can be a number of these type files in a complex script.
- **\*\*\*\*.pas** or **\*\*\*\*.vbs** or **\*\*\*\*.js** : these files (generated by Liberty Script Editor) contain the Object Pascal (DelphiScript), VBScript, or JScript files which make up the script code. There can be a number of these files.
- any other bitmap or rich text (or other type) files which may be used in the particular script code.

## The Script Launcher

The Script Launcher is a form which can be used to easily browse, get information about, or use Scripts that have been installed and are available to Praxis. You can start the Script Launcher by using the "Script->Run A Script" menu of the Main Form.



Use the Drop down box to select from the named scripts. A picture and a brief description about the selected script will usually be displayed to tell you about what the script is intended to do.

To Launch (execute) the script, click the provided button.

To find a script in a different location than under the default directory, you can use the Browse button. Browse to the directory that is **above** the desired script's directory.

## Forms Used to Control Praxis

### Praxis Forms

Praxis can show a large number of Forms during its operation.

All Forms can be closed or hidden during Praxis operation **except** for the Main Form -- **closing the Main Form shuts down the Praxis program!** But to minimize clutter on your computer desktop, you will usually want to close or hide any of the Praxis forms which you do not currently need to work with.

Some of the forms (such as editor forms or report forms) will appear or close as requirement for them changes.

Certain of the forms are used more often, as listed below:

- The Main Form
- The Levels Form
- Plot Forms (including the Primary Plot, the Live Time Plot, the Live IR Plot, or the Auxiliary Plot)
- Configure Forms (Stimulus, Acquisition, PostProcess)
- The Script Launcher

## ***Using Praxis' Numerical Value Controls***

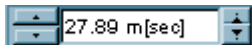
These "NVC" controls look very similar to common "Up/Down" integer controls with which you may be familiar. But they are much more capable (though quite easy to use). Please look over the description below to see how they work.

Some special features are:

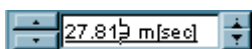
- They are able to control **floating point values** (not just "whole" or integer numbers). They can also be set to allow only whole numbers.
- They can be easily adjusted using the **mouse wheel**, giving control quite similar to having an analog "knob"
- The rate at which the value changes can be adapted to the kind of value being controlled. Or you can **easily configure the change rate** yourself.
- Some of these controls convey and understand **Engineering Notation**. In other words, the value 12345.67 will often be expressed as "1.234567k". Similarly, if you type in the value "987.65m", it will be interpreted as meaning the value: 0.98765.
- They can be set by Praxis to **accept values only within an allowable range**. This saves time and confusion from error messages about invalid errors.
- They can relate basic **Units of measure**. The Units cannot be changed (they are for informational purposes only), and you do not need to enter them.

These controls are very handy and can be adjusted a number of ways. Normally you will use the mouse wheel, or click the "Up/Down" buttons, for rapid, easy control. You can also type the value you want into the box, or cut and paste the value from a different box (using normal Windows techniques). Beginning in PRAXIS version 2, these NVC (or "EditRealSpin") controls can also be designed into custom Scripts using the PRAXIS Script Designer.

When typing values into the control, any of the Engineering Notation characters (p,n,u,m,k,M,G) will delete all other characters to the right, simplifying data entry. If you leave the control and click elsewhere, the new value (if acceptable) will take effect.

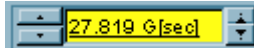


If you are typing in the text area of the control the numerical value will appear underlined until you press the [Enter] key or double-click on the value, or leave the control. THE VALUE CHANGE IS NOT ACTIVATED YET WHEN THE UNDERLINE STILL APPEARS!





You may see the text window turn yellow while you type within it - that means that the value you currently have there, if it were [Enter]ed, would be out-of-range or otherwise invalid, and would not be accepted. But there's no need to worry about it until you have your entire input typed in. If it appears yellow when you are done, though, you'll need to change it to something in-range.



Values in the control (when it is used by Up/Down buttons or the wheel) will change by a selectable mathematical methods. You can modify these methods as you see fit by right-clicking on the Up/Down buttons and by choosing the method you want from the pop-up menu:

- "integer round" will cause the value to go up or down by the integer amount (that you will next be prompted for).
- " 1-2-5 pattern" makes the numbers progress as, for instance: 10m, 20m, 50m, 100m, 200m, 500m, 1,2...etc.
- " high digit[+,-]" alters and rounds using most significant digit, giving progressions like: 60m, 70m, 80m, 90m, 100m, 200m, 300m.....etc. This is often the best general-purpose selection.
- " float[+,-]" increments or decrements on each click using the prompted-for floating point value.
- " float[\*,/]" multiplies or divides on each click by the value you will be prompted for.

If a set of up/down clicker buttons appear to the right of the control (there are little dots on the arrows of these clicker buttons), then these buttons allow you to nudge up or down by one valid data point per click. These will cause the control to jump ONLY to exact data points for which data exists, going to the next (or previous) point in the data progression.

To use your mouse wheel, you must first click or double-click in the text part of the control (or use one of the Up/Down buttons). Then, rolling your mouse wheel will have the same effect as using the configurable Up/Down buttons.

You can also copy, cut, or paste the value from the text area of these controls by first right clicking the mouse over the text box (after highlighting the text). This is useful for transferring values between Numerical Value Controls.

## ***Engineering Notation***

In Engineering, a wider range of numerical values is often encountered than in many other fields. Values are therefore often expressed in a shorthand in which a "power of ten" multiplier is attached to the beginning of the text indicating the Units of Measure.

For example, the value 0.0012 Ohms is normally expressed as 1.2 mOhms ("1.2 milliOhms") and the value 2345000 Ohms is expressed as 2.345 MOhms ("2.345 MegOhms"). This effectively gives a unit of measure which scales itself with the values being encountered, and gives a more intuitive understanding of the relevant scales -- for instance, just using the units "microVolts" conveys the idea that the number is very small, even if the number itself is not fully inspected.

Praxis' Numerical Value Controls normally are set to use this notation. The following abbreviations are used for the multipliers (in place of the usual Greek characters):

- p=  $1 \times 10^{-12}$  ("pico")
- n=  $1 \times 10^{-9}$  ("nano")
- u=  $1 \times 10^{-6}$  ("micro")
- m=  $1 \times 10^{-3}$  ("milli")
- k=  $1 \times 10^3$  ("kilo")

- M=  $1 \times 10^6$  ("Mega")
- G=  $1 \times 10^9$  ("Giga")

.

Therefore, for instance,  $1000\text{m} = 1 = 0.001\text{k}$

## ***The Main Form***

The Main Form, identified by the words "(Main Form)" in its caption, contains the fundamental controls needed for making measurements or accessing other forms.

The Main Form is always open while Praxis is running -- **closing the Main Form shuts down the Praxis program!**

The Main Form contains the following basic controls and features:

- The Main Form Menu
- The Stimulus Controls
- The Acquisition Controls
- The PostProcess Controls
- The Start/Stop Button (to begin Stimulus and Acquisition processes)
- The Time/Frequency Button

.

#### (MainMenu) Config | Load Config File

Use this to load a specific Praxis Configuration File, which has settings and formatting which you have previously saved (and wish to use again). The default Configuration File is "Praxis.ini" which is automatically loaded at each startup, and saved (if you agree, when prompted) when Praxis shuts down.

#### (MainMenu) Config | Save As

Saves the current Configuration settings to a Configuration File which you can specify. To save the settings to the file which will be used when Praxis starts up next time, use the name "Praxis.ini".

#### (MainMenu) Config | Use Startup Config File

The same as using Config | Load Config File, and then specifying "praxis.ini".

#### (MainMenu) Config | Hardware Specific

Presents other menu options concerning the hardware you are using. This does NOT include your soundcard hardware settings (such as Sample Rate or bit resolution) --those are controlled by menus in the Levels Form.

- Loading of **Microphone or Accelerometer** correction data
- **Probe balance**: Allows correction for differences between the two probes being used, for maximum accuracy in impedance measurements. (not used in Free/Demo Mode)
- **Probe Resistors**: Can be used to register the values of the internal series resistors used in the differential probes for AudPod. (not applicable in Free/Demo Mode).
- **RefCalSource**: Select the Input Source (normally Probe#2) that will be used as the "Reference Cal" input when doing "Cal" normalized measurements. THIS SETTING DEFAULTS TO "PROBE2" AND IS NOT SAVED IN CONFIGURATION FILES. This menu option also appears in the Levels Form
- **Ground Resistance**: (Free/Demo Mode Only) This value is the measured resistance in the ground return line to your power amplifier's internal reference. This value is normally set by the "Ground Resistance" script.
- **SyncPulseLevels**: You can adjust these values should you have difficulty in obtaining measurements with "Synch" type stimuli. **InputGain** adjusts the input sensitivity to the synchronization pulses (that are used on one output channel in Synch modes). **OutLevel** adjusts the pulse level that is output from the soundcard for this synchronization. If OutLevel is too high, it can result in a degraded noise floor during synchronous measurements. If OutLevel is too low, it can result in erratic synchronization.
- **Mic1 Uses Probe1 Jack**: allows alternate connection of an externally powered microphone (it allows the Probe1 jack to be used as a balanced or unbalanced input for a microphone, with approximately 12dB lower sensitivity).
- **Line Frequency**: set to this to the AC power line frequency for your locale. This is used to synchronize repetitive measurement stimuli so that line interference will tend to cancel when acquired data is coherently averaged.
- **Load, Save As, Use Startup**: These are used to handle special "Hardware Specific" configuration files that store information about your particular hardware setup.
- **.Priority**: These two options allow you to re-assign the priority given to the Praxis program, and

to the separate execution thread it uses for its audio acquisitions. Setting these too low may prevent the processor from servicing Praxis' data processing in a timely manner. Setting these options too high may hinder the operating system from servicing the soundcard itself (particularly USB connected sound devices) at an adequate rate

• **WAVE\_EXT**: This option determines whether Praxis will communicate with the soundcard using the "WAVEFORMATEX" (WAVE\_EXT unchecked) or the "WAVEFORMATEXTENSIBLE" (WAVE\_EXT checked) structure. This is needed because some soundcard drivers for 24 bit soundcards cannot operate in 24bit mode unless called with the new structure; and some drivers will not work at *all* unless called with the old structure. Praxis includes a built-in script called "Soundcard Capabilities" which should help you decide the proper setting for this menu item.

#### (MainMenu) Config | Preference | Units...

Allows you to select your preference of certain measurement units (American English or Metric).

#### (MainMenu) Config | Preference | Backdrop

Select whether to use a "BackDrop" behind Praxis's forms, to cover distractions from other items on your Windows Desktop. You can also select a bitmap file to use for the backdrop, if you wish.

Using a backdrop is recommended in most situations EXCEPT when developing custom Scripts (in that case, the Script Designer can get hidden too easily).

To access other Windows applications that may be running when you are using a backdrop, hold down the [Alt] key while pressing the [Tab] key to select from the running applications.

#### (MainMenu) Config | Preference | Dashed Gridlines

When this option is checked, all Plot Forms in Praxis will be drawn with the gridlines dotted or dashed, rather than solid. If the option is unchecked, the gridlines will be shown unbroken. Changes to this setting take effect when plots are redrawn after the change.

#### (MainMenu) Config | Preference | Backdrop

When this menu option is checked, PRAXIS scripts will flag and issue an error message for any undeclared variables or procedure names that are encountered during script execution. This is strongly recommended, as it can avoid problems with confusing errors that can result from typographical errors or mis-spelling of variable names.

For script code written in VBScript, you should include the directive "Option Explicit" at the top of the script code listing for the Strict Script Running setting to have effect..

#### (MainMenu) Config | Mixer

This brings up the Device-Mixer Matching form (see "About Windows Mixers").

#### (MainMenu) Config | Show Hints

Enables or disables "Hints" that appear when the mouse is over controls on the Main Form and the Levels Form. You can disable them here if you find them to be distracting.

#### (MainMenu) Config | Auto Acquisition Start Timer

Praxis can be configured to begin measuring at a specific time in the future using the form presented by this menu. For this to work successfully, you must assure that the computer is powered up and running PRAXIS when the start time is reached, and that the computer is "awake" (i.e., that it has not gone into a sleep or hibernation mode).

This feature can be useful for situations such as:

- doing a noise measurement survey at a specific time of the night when you may not be present. To use this, you should also set up for "Auto Stop, Timer" in the Acquire Configuration form for your measurement.
- testing with high signal levels when you may want to give yourself time to leave the room before testing begins.
- when a measurement is very sensitive to vibration or acoustic noise that could result from having human operators in the testing area.

#### (MainMenu) View

Allows you to unhide any Praxis forms including the Configure Forms. You can also create a New Secondary Plot (and bring it into view), or Close All Secondary Plots.

#### (MainMenu) ScriptDesign

This launches the Liberty Script Designer, used to develop custom visual Scripts.

### (MainMenu) Config | RunScripts

Brings up the Script Launcher, allowing you to select and run a Script.

This action is duplicated, in Free/Demo Mode, by the large "Information and Demo Scripts" button.

### (MainMenu) Help

Brings up the Help facility with Praxis' Help information.

### Stimulus Controls



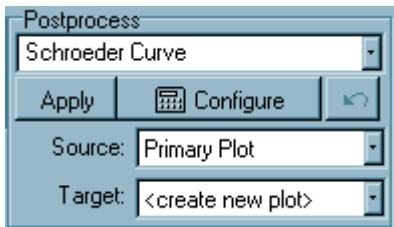
These Main Form controls allow selection of the desired Stimulus and allow you to hide or unhide the Stimulus Configure Form.

### Acquisition Controls



These Main Form controls allow selection of the desired Acquisition and allow you to hide or unhide the Acquisition Configure Form.

### PostProcess Controls



These Main Form controls allow selection of the desired PostProcess and allow you to hide or unhide the PostProcess Configure Form. An "Undo" button is also provided in case a postprocess operation is done in error.

PostProcess operations normally operate on data from the "Source Plot" and put the result data into the "Target Plot". These are defined as follows:

- **Source Plot:** This refers to the single Plot from which PostProcess operations take their source data in most cases (some PostProcess operations use data from both the Source and the Target Plot). You can specify any existing plot to be the Source Plot.

- **Target Plot:** This refers to the single Plot into which the results of PostProcess operations are sent. You can specify any existing plot to be the Target Plot, or you can specify that a new plot be created to be the Target.

The definition of which Plots are to be Source or Target for the next operation can change after a PostProcess completes (this is done to avoid inadvertent modification or overwriting of data). You should always check the Main Form to see which plots will be used for the Source and Target before activating a Postprocess operation.

Most PostProcess operations can be "undone" using the Undo button given on the PostProcess Control of the Main Form. This can remove the effects of only the most recent PostProcess operation.

#### The Start/Stop Button

Use this Main Form button to Start or to Stop a measurement acquisition.

You can also use the [Esc] key to stop measurements.

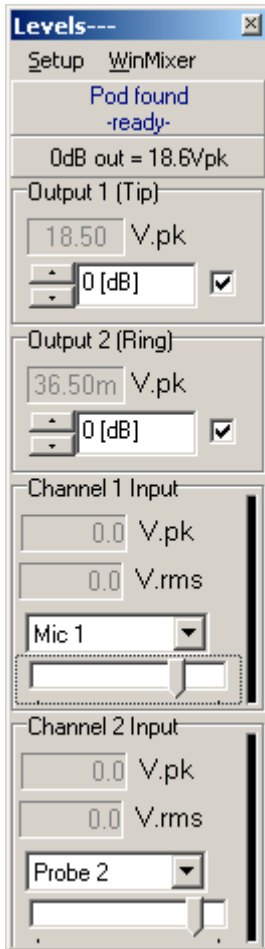
#### The Time/Frequency Button

This Main Form button changes between display of Time Domain data and display of the latest Frequency Domain data (if available), on the Primary Plot. This button also appears, for



convenience, at the bottom of the Primary Plot.

## The Levels Form



This form provides:

- Menus for controlling Analog/Digital (A/D) and Digital/Analog (D/A) conversions on the soundcard
- Menus for selecting special setups (such as having the A/D and D/A be from different cards)
- A menu choice for a hardware calibration Wizard (using the AudPod Full mode)
- Display of the peak output level (in the most recent stimulus buffer). The output voltages are give for the output of any power amplifier following the AudPod's stimulus output, per the Power Amplifier gain setting on the Stimulus forms.
- controls for output levels (relative to Praxis' full scale) and checkboxes to enable or disable outputs
- display of the peak and RMS level values from the most recent input acquisition buffer
- selection of the input source for the relevant inputs channels (Full Mode only)
- slider controls for input levels
- level indicator bars (which light up with **RED** dots if an overload is detected).
- Indicators of input and output levels from last stimulus or acquisition.

**Note:** in Demo mode, lack of an overload indication on the indicator bars does not necessarily mean that overload is not occurring! Some experimentation with the Windows Mixers may be necessary. And **Use caution: APPLYING LARGE VOLTAGES (more than 2V peak) TO YOUR SOUND CARD INPUT MAY CAUSE DAMAGE.**

The level indicator bars and level values are not active EXCEPT when Praxis is Acquiring data.

In Full mode operation, operation well below the input overload point may be required to minimize distortion with some soundcards.

### (Levels Menu) Setup | D/A&A/D

Use this menu to:

- **Select which soundcard** to use for Praxis. (Make sure to select the desired soundcard, if you have more than one before doing the calibrations).
- Select the **Sample Rate** and **Bit Resolution** for the A/D and the D/A (they can be different)
- Apply a value to use for **A/D latency** and for **D/A latency** (these need not be set exactly, and are used only for Asynch or ReSync type measurements, along with any specified Propagation Time, to make sure that the acquisition data is collected only while a response to the stimulus is available).
- You can also specify a value to use to assure that the output electronics have stabilized before data stimulus begins (needed for some lower grade soundcards that generate a "pop" when



starting a new output).

#### (Levels Menu) Setup | General

Here you can make some general selections related to your soundcard or test power amplifier:

- The 48kHz record/play latency. Adjust this to the approximate full-duplex latency of your soundcard to optimize the performance of the triggering operations of the system. If you are unsure about the latency, use the nominal value of 50msec.
- The values of ADDITIONAL series resistors used in each lead of the AudPod's probes, when voltages greater than 40V peak must be handled. of the input when the selected resistors are used in series with the existing 47.5k ohm resistors in the probes.
- Characteristics of the power amplifier: its peak voltage capability and its voltage gain. (in dB).
- You can specify the default RefCalSource. This option is duplicated in the Main Form's "Config -> Hardware Specific" menu.

#### (Levels Menu) Setup | Do Calibrate

In "Full Mode", starts a "wizard" process for calibrating the soundcard gains and levels with the AudPod. All instruction for the process are included in the wizard.

For "Demo Mode", use the calibration script provided in the Demo Scripts, instead.

#### (Levels Menu) Setup | Check Calibration

In "Full Mode", this performs a check of the record and play calibration between the soundcard and the AudPod to verify that the soundcard's mixers or gain control knobs have not been disturbed since the last soundcard calibration. When performing this, remove any cables or loads from the AudPod's Stimulus jack.

#### (Levels) Output Controls

These are Numerical Value Controls which change the output gains (and therefore, the levels) in steps of one significant digit per step. You can also type values directly (integer values, maximum=0dB, minimum=-140dB).

There is also a checkbox to the right of each Output Control. This can be used to temporarily blank or disable the output of the stimulus signal, when the box is unchecked. Make sure this checkbox is checked before making any measurements.

For non-Synchronous type stimuli, you can set the output levels individually for each channel.

The outputs are labeled "Tip" and "Ring" to indicate the connection each channel uses on the min-phone connector of the AudPod. On most soundcards, Output1 is the "Tip", but this may not always be the case.

## ***(Levels) Input Controls***

Readouts are provided to display the peak and the rms values of the signal most recently arriving in acquisition input buffers. The rms values will not be accurate in cases where insufficient samples of a waveform component were in the buffer; it should not be considered an absolute number.

With the AudPod, you control the input sensitivity here with the provided sliders, and select the input source for each input channel.

During an acquisition process (with AudPod only), you can cause PRAXIS to automatically fine-adjust the input sensitivity of the AudPod by double-clicking on the vertical bar graph that indicates input signal level. Alternately, you can right-click on the handle of the horizontal input gain slider to trigger this action. The adjustment is made on the basis of the last input signal acquisition during measurement, and adjusts for 6dB nominal headroom. When the input is overdriven, you may need to do this several times, as it is not possible to predict how far the input is being overdriven until it is reduced below the overdrive point.

### **Input Selection:**

The AudPod allows you to select from a number of input sources for each measurement channel. These are:

- Probe 1
- Probe 2
- Mic 1 (this mic can connect via the Mic 1 jack, or via the Probe 1 jack)
- Mic 2
- Output 1
- Output 2
- Line Input (1 or 2, depending on input channel).

An internal calibration oscillator can also be selected by the AudPod calibration process (but this source is not selectable for general use).

Note: For each input channel, only a specific channel of the Line Input can be selected. Channel 1 can use only the "tip" connection of the Line Input jack and Channel 2 (often used for "Cal" input) can use only the "ring" connection of the Line Input jack.

### **The Ref Cal Input:**

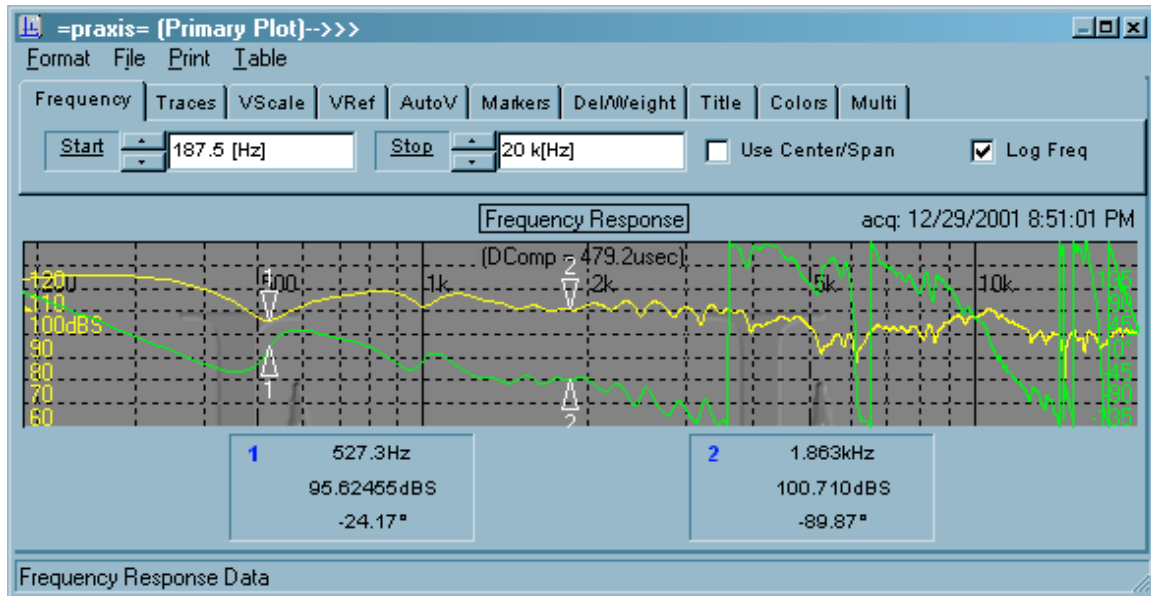
Certain Acquisition types default to a defined "Ref Cal" input for "Cal" measurements.

For example, with many frequency response measurements or impedance measurements, you will find that you cannot directly select the input used for the Channel 2, or "Cal" Input channel. In fact, with Synchronous acquisitions, all measurements inputs use a single A/D channel (Channel 1). The signal source used in these cases is determined by the setting of the "Ref Cal Input". Although RefCalInput can be changed using menu options on the Levels Form or on the Main Form, this Ref Cal Input should normally be "Probe 2", which is the appropriate setting except for very specialized circumstances.

## ***Plot Forms***

A Plot Form is a program component in which data curves can be manipulated. It is used for containing, displaying, formatting, printing and maintaining files of measured audio data.

The Plot Forms are resizable, extremely versatile and are the most complex forms used in Praxis.



There is no built-in limit to the number of Plot Forms you can have (except individual machine limits due to computer resources and memory). The Plots can usually be created at will when needed for data manipulations, or for storing a curve temporarily for comparisons.

**There are several kinds of Plot Forms which are named in Praxis:**

- **The Primary Plot:** This Plot always exists, and it is where measured data from an Acquisition always appears. When you close the Primary Plot, it is merely hidden, and can be made to reappear with its data still intact.
- **Secondary Plots:** This refers to any Plot which not the Primary Plot. When you close a Secondary Plot, it no longer exists and its data is no longer accessible. You can, however, save the data to a file from a secondary plot.
- **Source Plot:** This refers to the single Plot from which PostProcess operations take their source data in most cases (some PostProcess operations use data from both the Source and the Target Plot). You can specify any existing plot to be the Source Plot. The source plot will be labeled with the characters "---->" in its caption bar.
- **Target Plot:** This refers to the single Plot into which the results of PostProcess operations are sent. You can specify any existing plot to be the Target Plot, or you can specify that a new Plot be created to be the target. The target plot will be labeled with the characters "<-----" in its caption bar.
- **Live Time Data Plot.** This plot can be enabled by checking the menu option available from the Main Form's "View" menu. When enabled, this plot will appear during acquisition(if not already visible) and update with unprocessed raw time domain data in its traces, rfrom an in-process acquisition. This allows you to, for example, view the signal from microphones or probes in the time domain, even if the acquisition type is for frequency domain data (which is still displayed in the Primary Plot).
- **Live IR Data Plot.** This plot can also be enabled by checking the menu option available from the Main Form's "View" menu. It will have effect, however, only when the acquisition uses an Impulse response representation in the course of its operation. Using this plot, it is possible, for example to view the time domain impulse response of a device at the same time as the

frequency response is being measured and displayed by the Primary Plot, and/or while the "raw" time domain data is being displayed by the Live Time Data Plot.

- **Auxiliary Plot.** This plot is used for more complicated acquisition types (such as Wavelet acquisitions) in which data is further processed in real time from information being presented in the Primary Plot. The plot will appear when applicable, and will remember its position and size between uses.

Entire data sets can be easily transferred from one Plot to another. Merely hold down the mouse button when the cursor is in the plot's graphing area, and drag to the graphing area of the other plot, then let go of the mouse button.

Similarly, you can transfer the formatting from one plot to another one (provided that the second plot contains compatible data), by dragging from the status bar region at the bottom (where the data title is shown) from the "from" plot to the same bar region of the "to" plot. This is a handy way to set the vertical and horizontal scaling within the plot are set equal when visually comparing data plots. The size of the form containing the plots is NOT altered by this drag'n'drop operation.

#### **A Plot can have these regions:**

- **A Menu:** this is where you select or control Formatting, Filing, Printing, or presentation of the plot's data in Tables.
- **A Set of Formatting Tabs:** these appear (or disappear) when you click on the "Format" menu option. These tabs hold a large number of controls for formatting the graphs shown in the plot. They can be easily hidden when they are not needed.
- **A Description Line:** this displays information such as the file name where the data was loaded from or where it was last saved (if so), the type of data in the plot, and the acquisition time. During averaging in the Primary Plot, the average count is displayed in this line.
- The graphing or **plotting area** where the data curves are displayed.
- **Marker readout area:** values of the frequency (or time) values for the markers (up to 5 may be used), and the curve values at the locations of the markers.
- There can be a number of **text lines giving descriptions for all the "Multi" curves** (you can have very many of such "multi" data curves shown along with the main data in a Praxis plot, should you wish to show more than two curves in one plot).
- A **"status bar"** at the bottom, usually used to display the user-assigned description or title for the data. This sometimes shows other indications, such as the progress in a curve-fitting operation. The status bar also includes a handy auto-scale (lighting bolt) button. The Primary Plot also contains a duplicate of the PRAXIS Start/Stop button and the Time/Frequency button from the Main Form.

#### **PLOT SETS**

Several PRAXIS Plots can be grouped into a PlotSet. This allows the group of plots to occupy the same on-screen form, with the one being shown selected by a series of buttons above the graph area. This can be very helpful for keeping the screen organized when you have many plots active in PRAXIS.

- To make a new PlotSet, choose the first plot that you wish to include in the set, and select its "PlotSets" menu, then select "Start a PlotSet".
- Then to add other plots to the existing PlotSet, simply again select the "PlotSets" menu (of the PlotSet) and then select "Add a Plot to PlotSet". A small form will appear, from which you can select the plots you wish to include in the set. You can also create a new, empty, plot to include into the set.
- To remove a plot from PlotSet (and make it into its own free-standing plot), click the desired plot's button in the PlotSet. Then Right-Click with the mouse on the plot's button and respond to the prompt. The isolated plot will appear, ready to be moved and resized.

When a PlotSet contains multiple Plots, you can select the one you wish to view or work with by clicking the rectangular button with the plot's name, as displayed just below the menu options in the PlotSet.

The Aux Plot, the LiveTime and the LiveIR plots cannot be included in PlotSets. If the PrimaryPlot is in a Plot set, it will always be made viewable when new data is acquired by measurement.

### The Primary Plot

This Plot always exists, and it is where measured data from an Acquisition always appears. When you close the Primary Plot, it is merely hidden, and can be made to reappear with its data still intact.

### Secondary Plots

This refers to any Plot which not the Primary Plot. When you close a Secondary Plot, it no longer exists and its data is no longer accessible. You can, however, save the data to a file from a secondary plot.

### (Plot Menu) Format

Clicking this menu option brings up a set of tabs, each of which contains one or more controls for formatting the plotted data.

The Tabs are(certain tabs may not appear when they are not appropriate for the given data):

- The Time Format Tab
- The Frequency Format Tab
- The Traces Format Tab
- The VScale (Vertical Scale) Format Tab
- The VRef (Vertical Reference) Format Tab
- The AutoV (Automatic Vertical) Format Tab
- The Markers Format Tab
- The Windowing Format Tab
- The Del/Weight Format Tab
- The Title Format Tab
- The Colors Format Tab
- The 3D Curve Format Tab
- The Polar Format Tab
- The Multi Curve Format Tab

### (Plot Menu) File

This menu presents a set of options allowing you to

- Open a data file (and load it into the Plot) from disk. You can select the types of files to load. Praxis can read most of the Liberty Instruments IMP and LAUD file formats.
- Save a data file.
- Save As: save the data file under a different name and/or in a different format.
- Save or Load a Plot Format file. This saves or loads formatting information only, so formats from one data file can be applied to data in the file currently ("Load Format") or in data that is

about to be acquired into the Primary Plot ("Load Pre-format").

- **Listen** to a file (if it is a normal Time Domain format or WAV file), using any soundcard in the computer system.
- Export time domain data in WAV file format
- Import or Export data to or from some other systems
- Allow display of low frequency data below the resolution limit (PRAXIS normally suppresses display of spectral or response data that is below the resolution determined by length of the transformed time record).
- Exit (close the Plot Form).

### Plot Format Files

Plot Format files are essentially data files which contain no data, only the information for formatting data.

They are useful for quickly setting a number of plot format parameters to a set of values which were set in a previously existing Plot.

You can also use them in some cases to "pre-format" the Primary Plot, that is, set up the formatting, in advance, which will be used in an acquisition. This must be done immediately before the acquisition.

The type of the data, to which the format is being applied, must be very similar (same number of curves, same units, types, etc) as the type the Format file was made from.

### (Plot Menu) Print

This menu leads to commands and forms for printing Plots onto hardcopy or to a bitmap graphic file.

You can print in monochrome (lines only), color (lines only), or color fill (color with colored fill regions).

Two drawing resolutions are provided. Depending on the printer and graphic cards being used, both resolutions should be tested for proper proportions and fine detailing.

### (Primary Plot Menu) Clone

This menu option is available only on the Primary Plot. When clicked, this creates a new Plot Form containing a copy of the same data that is in the Primary Plot. The result is the same as if you were to use the Main Form's "View, \*New Secondary Plot\*" menu, and then "dragged and dropped" the data from the Primary Plot to this new plot.

### (Primary Plot Menu) Hold

This menu option is available only on the Primary Plot. If the data in the Primary Plot is currently a Frequency Response or Spectrum type, then the data in Trace B of the Primary Plot will be made into a persistent background curve. This is done by making a file copy of the data and using it as a "Multi Curve" trace.

This can be useful when making measurements during development projects or investigations. The Hold curve can be used to keep a previous measurement result on the screen so that the effects of changes to the device being measured can be easily discerned.

You can now have a series of Hold Curves (rather than just one). To apply an additional Hold curve (without deleting the previous one(s), hold down the [Cntrl] key while clicking the Hold menu. To have the Hold curve replace all previous hold curves, click the Hold menu without using the [Cntrl] key.

To turn off (or hide) the Hold curve, simply click on the "ClearMulti" menu option when Praxis is not acquiring data, or go to the "Multi" Format tab and uncheck the "Show MultiCurves" checkbox.

#### (Plot Menu) ClearMulti

This allows you to quickly clear any assigned or configured multi-curve plots or a Hold curve shown along with the current data in the plot.

#### (Plot Menu) Table

This allows the data from the two main curves of the plot (and possibly other data that is not displayed, such as phase fields) to be displayed in Table form. The Table is scrollable, and the values can be copied for pasting into spreadsheets, word processors, etc.

#### (Plot Menu) Notes

The Notes menu causes a simple text editor form to appear. You can enter information in this editor describing the data and how it was derived. The contents of the notes form will be saved with the data in the form when it is saved with the File menu. However, if new data is placed in the plot by a load or copy operation, or by a new acquisition, the Notes data will be lost (so makes sure to save the data plot to a file after editing the Notes field).

The Notes form also allows you to obtain some parameters of the data currently in the associated plot by using its "About this data" menu. These parameters include number of points, sample rate, description, etc.

#### Plot Markers

The Markers in Praxis' Plot Forms are quite versatile.

- **Up to 5 sets** can be displayed, with one marker for each displayed curve.
- The **markers float** on the data traces, rather than being merely cross-hairs.

.

Markers are not available on waterfall plots.

#### Marker Delta (or Reference) mode:

The markers can operate in "**Delta**" mode -- one marker can be made the "reference", and then the curve values of all the other markers are given relative to that marker. **To activate or deactivate this**, simply click on the marker number at the left side of the readout area (below the graph) for each marker. The reference marker number will change to *red, italics, and will be underlined* (its reported curve values will also become zero).

#### Marker Search and Marker Track:

Each marker can **Search or Track** for minima or maxima in the curves. "Search" moves to the position once. "Track" always tends toward the target position, which can change as the data

changes. The maxima or minima can be "local" (using only adjoining points) or "visible" (the maxima or minima of all displayed points is found).

To select these modes, put the mouse cursor over the value (in the Marker Readout Area) for the desired curve which you wish to track, and right-click the mouse. A small form will pop-up, allowing you to select your choice of modes -- If you click anywhere other than on one of the selections, Search and Track for that marker will be turned off.

### **Placing Markers**

Markers can be located on the data curves by various means. If you click on the Frequency (or Time) domain value, and it will become a Numerical Value Control

. Then, you can:

- Use the Up/Down buttons
- Use your mouse wheel to slide the marker along the data curve.
- Type in the target domain value to which you wish the marker to move. (if the value is not an exact data point, the nearest point will be used, and the readout value will be "grayed out" and the time or frequency will be marked to indicate an inexact value. You can then click on the "dotted" up/down marker to the right to move the marker to a valid data point.
- When you have the Numerical Value Control open, you can move the mouse cursor over the graphing area. You will notice the mouse cursors change to arrows with the marker number between -- place the arrow-head on the curve where you want the marker and left click. (Right-click or press [Esc] to make the mouse cursor leave this mode).

### **Formatting Plots**

Most of the formatting controls for the plots are on page control "tabs" that are made visible via the Format option in the Plot Form.

The two primary formatting options used in plots are adjusting the horizontal and vertical characteristics. These include Time or Frequency (the horizontal "domain" axis), Vertical Scale, Vertical Reference value, Vertical Reference position, and Number of vertical divisions. Automatic scaling features are applied for some of these.

Other formatting options are colors, backgrounds, other data curves ("Multi") on the same plot, Weight Files (a response shape superimposed on the data), Delay and gain offsets, Marker selection, and Title.

In order to facilitate visual placement of data window edges, those parameters are treated as if they were plot formatting options.

### **Auto Scaling**

Auto scaling is a feature built into the Plot Form. It is intended to rapidly adjust the Vertical Scale (VScale) or the Vertical Reference (VRef) to maximize the visibility of the data (that is, so that most of it is not off-scale).

This is activated by use of small "lightning bolt" buttons on some of the tabs.

There is a "master" auto-button at the far left of the VScale format tab. This button will autoscale and autoreference both curves on the plot. Another copy of this master AutoScale lightning-bolt



button can be found near the bottom left of Plot Forms for convenient access at any time.

These are a handy way to get the graph vertically formatted into a comprehensible condition.

### The Time Format Tab

Use the controls on this tab to set either the Start and Stop values, or the Center and Span, for the horizontal (time) axis of the graph. This allows you to easily format the Horizontal range of the plot.

You can select whether to use Start/Stop or Center/Span by using the checkbox at the far right (this checkbox will not stay "checked" --rather it always clears itself and changes its function to toggle between Start/Stop and Center/Span operation).

***You may need to resize the Plot Form to see controls at the far right.***

### Setting to Extremes

When you are using Start/Stop operation, the labels for these Numerical Value Controls are shown underlined -- this is to remind you that these labels can be double-clicked with the mouse to make the control go its extreme setting (that is, double-clicking the Start label causes the Start value to go to its minimum, and double-clicking the Stop label causes the Stop value to go its maximum allowed value).

### Setting Graphically

Besides the normal methods of adjusting the Numerical Value Controls, some controls on this tab sheet can also be set graphically. This allows for very convenient operation.

After clicking the mouse within the Numerical Value control for Start, Stop, or Center, you can then move the mouse cursor within the graph area and then click on the part of the curve you wish to use for that value (if it is on the displayed area). The value will be automatically set within the control and the plot will redraw.

### The Frequency Format Tab

Use the controls on this tab to set either the Start and Stop values, or the Center and Span, for the horizontal (frequency) axis of the graph. This allows you to easily format the Horizontal range of the plot.

You can select whether to use Start/Stop or Center/Span by using the checkbox to the right (this checkbox will not stay "checked" --rather it always clears itself and changes its function to toggle between Start/Stop and Center/Span operation).

Another checkbox at the far right allows you to toggle between Log Frequency (equal number of octaves per horizontal space) or Linear Frequency format (equal number of Hertz per horizontal space).

***You may need to resize the Plot Form to see controls at the far right.***

### Setting to Extremes

When you are using Start/Stop operation, the labels for these Numerical Value Controls are shown underlined -- this is to remind you that these labels can be double-clicked with the mouse to make the control go its extreme setting (that is, double-clicking the Start label causes the Start value to go to its minimum, and double-clicking the Stop label causes the Stop value to go its

maximum allowed value.

### **Setting Graphically**

Besides the normal methods of adjusting the Numerical Value Controls, some controls on this menu can also be set graphically. This allows for very convenient operation.

After clicking the mouse within the Numerical Value control for Start, Stop, or Center, you can then move the mouse cursor within the graph area and then click on the part of the curve you wish to use for that value (if it is on the displayed area). The value will be automatically set within the control and the plot will redraw.

### **The Traces Format Tab**

This tab allows you to select from the available curve data in the data set contained within the Plot. Use this, for instance, to show or hide the phase curves in complex data. The selection is made for each of two "traces", identified as "A" and "B". The A trace has its vertical scale shown at the left, and the B trace has its scale shown at the right.

The available curves are listed in the drop down box. If you select the same curve data in both boxes, it will be removed from the other box (that is, you cannot show the same curve in both traces).

Between the curve label and the drop down box, there is a small color square. If you click on this square, you can select the color to be used for that data trace.

***You may need to resize the Plot Form to access controls at the far right.***

### **The VScale Format Tab**

The controls on this tab allow you to adjust the vertical scale (number of units per division) for both traces (A and B).

You can also select the number of vertical divisions (divided by grid line) to be used in the graph.

The leftmost button (with a small "lightning bolt" graphic) can be used to perform an overall automatic scaling operation of the displayed data in the vertical dimension. This can be useful to make the data visible when newly measured data is current off-scale. The vertical scale Numerical Value Controls also have Auto Scale buttons to their right, for rapidly adjusting the scales (in most cases) into usable ranges.

### **Wrapped and Unwrapped Phase Format**

If the plot contains phase data, you can select whether to show the phase trace as "wrapped" or unwrapped.

Wrapped phase is constrained to be within the range +/-180 degrees. **Unwrapping of phase is an inexact process** -- it is accomplished by seeking the best presentation which makes the available data appear closest to being continuous, after assuming some low-frequency starting phase. It can become confused by "busy" data, noisy data or data with insufficient resolution. The presentation is for improving clarity only -- there is no difference whatsoever, for instance, between phase values of 100degrees and 460degrees.

**Phase should normally be shown as wrapped during acquisition of data**, as the unwrapping process can cause excessive processing time with each update, interfering with measurement continuity.

***You may need to resize the Plot Form to access controls at the far right.***

#### The VRef Format Tab

The controls on this tab allow you to specify which division will be used as the **Vertical Reference division (RefDiv)**, and the **Vertical Reference Value (RefVal)** to be identified with the reference division, for both traces (A and B).

The bottom of the graph is reference division # 0.

The vertical reference value Numerical Value Controls also have Auto Scale buttons to their right, for rapidly adjusting the reference values (in most cases) into usable ranges.

#### The Markers Format Tab

This tab provides a series of checkboxes allowing you to select which of the 5 markers you wish to use.

At the left of the tab sheet is also a colored rectangle. Clicking on this will allow you to select the color to be used for the markers on the graph area. **If you choose pure black (usually the lower left square in the "basic colors" palette that is presented)**, then the markers will use the same colors as the traces on which they lie.

#### The Windowing Format Tab

This tab appears when coherent time domain (usually impulse response) data is shown in the plot. It provides two Numerical Value Controls for defining the range which will be selected to be included with a data window.

Use these to place the starting and stopping window edges that can be used when the shown data (or data that is to be acquired next) is transformed to frequency response data.

Typically, the parts of the trace that are outside the windowed area on the graph will be colored white (the part inside the windowed area will use the current trace color).

#### **Setting Windowing to Extremes**

The labels for these Numerical Value Controls are shown underlined. This is to remind you that these labels can be double-clicked with the mouse to make the control go its extreme setting. Double-clicking the Start At label causes the window start value to go to its minimum, and double-clicking the Stop At label causes the window stop value to go its maximum allowed value. You can also click on the underlined word "All" to cause windowing to include the entire time domain trace (same as double clicking both "Start At" and "Stop At").

#### **Setting Graphically**

Besides the normal methods of adjusting the Numerical Value Controls, the controls on this tab sheet can also be set graphically. This allows for very convenient operation.

After clicking the mouse within the Numerical Value control move the mouse cursor within the graph area and then click on the part of the curve you wish to use for that value (if it is on the displayed area). The value will be automatically set within the Numerical Value Control and the plot will redraw.

### The Weight Format Tab

This "Weighting" tab appears when certain Frequency Domain curve data are used in the Plot. The controls on this tab sheet will vary depending on the data shown in the plot.

This tab sheet provides controls which you can use to modify the ways that magnitude data is presented. The actual data value in the curves are not permanently changed by these controls -- the effects of these can be removed even after the relevant data has been saved and retrieved from a file.

#### Weights:

The button labeled "Weight" allows you to apply the contents of a "weight file" (\*.wgt) onto the magnitude of the plot. A weight is a curve that modifies data as it is displayed on the plot. The effect of the weight curve does not change the measured data, and can be turned "off".

- Clicking on the Load button will open a form allowing you to browse and load a pre-existing weight file.
- Right-Clicking on the **Load button** (or the file name) causes the effects of any weight file to be removed from the plot.
- The name of the currently active weight file will be shown in a label to the right of the Weight button (the name will also appear at the top of the graphing area, if a weight file is being used on the displayed data). If no weight file is being applied, "<none>" will be displayed to the right of the Weight button.
- The Weight files can be edited or created by use of the **Edit button**. This will bring up a text editor for the purpose.
- If the plot currently shows frequency domain data in a decibel format (such as RTA, frequency response, or spectrum), you can use the **Normalize button** to automatically generate a weighting curve file designed to flatten the currently measured curve (the resulting weighting curve will be inverse in shape to the existing curve in Trace A). For non-RTA data, the generated data will be limited to 300 points or less (you can change the number of points by right-clicking --rather than left-clicking -- on the Normalize button).

Weight files are **simple text files**, in which each line gives a frequency, followed by one or more spaces, and then a dB gain value. The frequency values **MUST** be in increasing order. Gain values will be interpolated between these to modify the data at each frequency point in the plot.

For example, if a line in the file were to be

**1000 -3.0**

then magnitude values near 1000Hz would be reduced by approximately 3 decibels in the displayed plot.

Text lines at the top of the plot which start with a quotation mark will be ignored (and can be used for comments or documenting the purpose of the weight file).

## The Delay Format Tab

### **DelayCompensation:**

This value represents the amount of fixed delay to be removed from the traces. Because mathematically "delay is the negative derivative of angular phase with respect to angular frequency", the effects of delay compensation will be noted much more dramatically at higher frequency points in the data. A common use this control is to compensate, in the displayed data, for the "Time of flight" of a signal as it travels from a speaker to the microphone. Double-click the Delay label to cancel any delay compensation value.

### **PhaseInv:**

This checkbox simply causes phase values to be incremented by 180 degrees.

## The Weight File Editor

This is a simple text editor. Simply type the text you wish into the provided area. See "The Del/Weight Format Tab" for information about the structure of Weight files.

The **Edit** menu provides options to allow you to cut and paste between different regions (or other windows applications).

### **To leave the editor, use one of the following:**

- **Load** allows you to load a file to edit (be sure to save it before exiting).
- **Cancel/Edit** menu: leaves the editor and discards all changes.
- **Save/Exit** menu: saves the file, with the changes you have made, to the existing file name, then exits
- **SaveAs/Exit** menu: allows you to change the file name, save the data (with changes) to the new file name, and then exit. The data under the previous filename (if any) will not be changed.

.

## The Title Format Tab

This allows you to enter a text title (or description) for identifying the data in saved files or plot printouts.

When additional data curves ("Multi") are included in a plot, the title of each curve can be made to appear under the graphing area.

## The Colors Format Tab

This tab provides color buttons which can be clicked to allow you to select the colors to be used for:

- The traces (A and B)
- The grid lines used for the vertical and horizontal divisions
- The background (used only if "Use BMP" is NOT checked).

.

### **Plot Backgrounds**

The background of your plots can utilize a bitmap graphic, which can be either stretched or tiled behind your data traces and grids. This allows you to display company logos, for example, or other pictures (even photographs) rather than simple solid color backgrounds.

To use a Plot Background, click on the Use BMP button so that it becomes checked. A form will open allowing you to select the desired bitmap. If you use [Cancel], the default Liberty Instruments embossed gray background will be used.

These backgrounds will not be printed when graphs are printed, except when "color fill" mode is used. The background proportions and tiling positions, however, will in most cases be different from those in the plot as shown on the screen.

### The "3D" Format Tab

This tab appears only when 3D (such as waterfall) data is contained in the plot.

It allows you to adjust your apparent viewing position horizontally and vertically, in relation to the 3-dimensional data curves, using Numerical Value Controls. The plot will be redrawn each time one of these values is changed. The viewed image will also be resized both horizontally and vertically to best utilize the surface area of the graphing area. This should not generally be done when gradient colors (see below) are being used, as redraws can then be quite slow.

You can also select three special colors to use for the image (other colors, such as the colors or the grids or trace edges, can be selected using the Colors Tab). "Fill1" is the color used to fill under traces (when waterfalls are displayed).

When "Fill2" differs from "Fill1", the waterfall will be displayed in Gradient color, which colors the regions so they vary as a function of apparent height (Fill1 will be used for higher regions, and the colors will vary down to Fill2 in the lower regions). USING GRADIENT COLOR CAN GREATLY SLOW THE PLOTTING PROCESS. You can right-click on the "Fill1" or "Fill2" boxes to make both boxes go to the color of the clicked color box (changing 3d plots to non-gradient color).

The Base color is used for the "floor" of the waterfall plot.

### Depth=Q

The checkbox labelled "depth=Q" causes waterfalls to display so that the depth dimension (i.e., going into the plot) is displayed in terms of "number of cycles" or "Q" rather than in units of seconds. This allows evaluation of the degree of resonance, adjusted for the frequency of the resonance. (This mode is based on an idea from Kim Girardin).

### The Polar Format Tab

This tab appears only when the plot contains a Polar response data set. It allows you to select the frequency of the displayed curve and from several options for the data:

- "Normalize" automatically formats the plot so that the largest value (for any angle in the set) lies on the outer edge of the plot.
- "Filled" selects whether the plot shows a solid internal color rather than simply an outside trace. The color of the fill can also be edited. (The color of the trace or outside edge can be selected using the "trace A" color of the Color Format Tab.
- Left/Right Symmetry mirrors any data which is in the range 0 to 180 degrees onto the range 360 down to 180 degrees. This is useful for minimizing data sizes and measurement effort when the transducer pattern is known to be left/right symmetrical.

Front/Back Symmetry mirrors any data which is in the range 270 degrees to 360 (0) degrees to 90 degrees onto the range 270 degrees down to 90 degrees. This is useful for minimizing data sizes and measurement effort when the transducer pattern is known to be front back symmetrical (such as many planar drivers).

- The "Multi" Format Tabs

The controls on these tab sheets allow you to configure and control placement of additional data curves onto the graphing area.

The traces to be included:

- Must be stored in accessible data files, and preferably stored in the ".px2" data format (that format can be read fastest).
- Must be of the same data type (time domain, frequency response, spectrum, etc.) as the "A" trace (or compatible types), in the current data, and must use the same measurement units.
- Should be of small to moderate file size if they are to be displayed in the Primary Plot during data acquisition (very large file sizes can cause significant time lapses).
- Will be scaled using the current scale parameters of the Plot's data.
- Only one trace (the "A" trace, typically) will be used from each data file.
- Can also have their Titles (from the data set) displayed along with a color/symbol legend to help in interpreting the graphs.

You can select the colors and symbol types to use for each added trace, and can also save the set of file names, colors, symbols, etc. in a file for later use.

To Use:

- If a new trace is desired, Click the [Add New>] button. This brings a form you can use to select the files containing your traces.
- You **can** select files which are not compatible with the data resident in this plot, but only curves from compatible files will be shown -- if a selected file results in no traces, that usually means it is not compatible for some reason.
- After selecting a file, it may appear shown in gray in the drop-down box. If this happens, click the Format menu option twice to force the tab sheet to redraw.
- You can check the MultiPlot and Show Legends boxes, as desired, to show the compatible selected traces, and their Title text.
- To remove a trace from the set, highlight it in the drop-down box, and click [>Remove] (and agree to the prompts).
- To select the trace color and the symbol shape for each trace, click on the small color square (between the drop-down box and the [>Remove] button). This will bring up a simple form for selecting color and symbol type.
- To save or retrieve a set of MultiPlot parameters in a ".lil" file, use the buttons at the far right of the tab sheet.

## **Configuration Forms ("Configure Forms")**

These are forms which change size and content with the selected Stimulus, Acquisition or PostProcess. There is typically one Configure Form for each of these.

The Configure Forms can be shown or hidden using buttons on the Main Form.

## **Files Used in Praxis***Praxis File Types*

There are a number of file types which Praxis uses. Listed by their extensions, some are:

- data storage files ".px2". **This is the preferred file format for storing Praxis data.** It is a proprietary binary format, efficient in storage size and fast to read.
- ".pxd". This is a text equivalent to the px2 format. It is somewhat human-readable (using a text editor), but requires larger file storage area, and can not be read as quickly into Praxis.

When Praxis is installed, the "px2", "pxd" and "ipr" file types are "registered" to Praxis. This is **you can open Praxis, with files of these types loaded, by simply double clicking their names or icons within the Windows Explorer.** An "ipr" file causes Praxis to start up running that Script file (if Praxis is not already running). If Praxis is already running on the computer, a new plot will be opened in the existing instance for the "px2" or the "pxd" data files.

- ".imp", ".im2": these are the time domain formats used in IMP and LAUD. These can be loaded into Praxis plots.
- ".frd", ".fr2", ".zma", ".zf2": frequency domain formats from IMP and LAUD. These can usually be loaded into Praxis plots.
- ".dat": these are microphone (or accelerometer) data correction files used by Praxis. They are text based, and can be at any frequency points (the values are interpolated for use). Their format is best understood by inspecting one with a text editor. If the first line ends in "mV/Pa", it is understood to be microphone data. If it is "mV/g", it is understood to be accelerometer data.

**CAUTION: many other software programs (including Microsoft Windows) use the ".dat" suffix for different file types -- do not attempt to load those into Praxis!**

- ".lil": these are list files, used for saving sets of "Multi" curves for display on a plot.
- ".lfl": these are frequency list files, which can be used for controlling Stepped Tone stimuli.
- ".wgt": these are Weight files, used to induce an additional response shape onto frequency response or spectral data.
- ".ini": these can be general configuration files used by Praxis. Be sure to load only Praxis configuration files into Praxis (".ini" is a common file extension used by many programs).
- ".hdi": these are Hardware Specific configuration files.
- ".ltq": these files are specifically used for holding data about available soundcards and mixers
- ".pf\_": Plot format files are used to make one set of data use the formatting saved previously from a similar data plot.
- ".ipr" and various other file types used for creating, configuring, or documenting Scripts.
- ".zis": this format is used to store a "zipped script"-- which is a PRAXIS script directory and all of its contents, compressed to one easily transportable file. The Main Form's "Scripts" directory contains options to zip up a script or to install a zipped script for use by PRAXIS.

## **Praxis Configuration Files**

Praxis stores a large amount of information concerning its setup, such as location of the various Praxis windows, default display parameters for various measurement plots, trace colors,



acquisition settings, stimulus settings, etc. This information is stored in a file, "Praxis.ini", which resides in the same folder as the Praxis program (usually C:\Program Files\Praxis). The information in this file is used to configure Praxis at startup time. This Praxis.Ini file can be transferred between different machines, even those using different hardware types, for those cases where it is desired to duplicate a specific setup. Do not edit the Praxis.ini file directly, as the system reaction to "typos" or invalid parameters cannot be predicted.

#### **Hardware Specific Configurations:**

There is a separate configuration file, also stored in the same directory, called "Praxis.hdi" which stores certain information specific to the hardware in the given machine. This includes settings such as the selected soundcard, the selected sample sizes and sample rates, microphone calibration files, probe values, etc. This file should not be transferred to other machines or operating systems, nor edited directly.

#### **Removing a corrupted Configuration file:**

There have been situations reported in which a PRAXIS configuration file has become corrupted and prevents proper operation of the system. To test for this, or to disable a possibly corrupted configuration file, use this process:

- 1) Exit PRAXIS and browse to the "C:\Program Files\Praxis" folder on your computer.
- 2) In that folder, find the files "Praxis.ini" and "Praxis.hdi"
- 3) Right-click on each of these filenames and use the "Rename" menu that appears to change them by adding a character to each filename. For example, "Praxis.ini" can be changed to "Praxis1.ini". This will prevent PRAXIS from loading that configuration file on startup. PRAXIS will instead create a new, default configuration file to use with the original filename.
- 4) If you wish to go back to the disabled configuration file, exit PRAXIS and then delete the current "Praxis.ini" and "Praxis.hdi" files (or rename them to something else). Then rename the previously disabled filenames back to their original names.

## **Measurement Considerations**

To obtain the best results using Praxis, certain details of the system structure should be understood, and a number of considerations should be kept in-mind.

- Measurement Channel, Cal Channel
- Computer Resource Availability
- Measurement Units
- Time, Frequency, Size...and Noise
- Processing Times
- Chirp vs. MLS vs. SteppedSine
- Level Adjustments (Free/Demo Mode, and AudPod Mode)
- Multitone Testing Considerations

.

### ***Measurement Channel, Cal Channel***

In many measurements made with Praxis, two different signals are sensed, and the raw data from these are combined to form the final "acquired" data result. The following names are used sometimes in Praxis documentation to identify these:

The **Cal Channel** usually used for a probed measurement of the stimulus signal, as it is applied

to the device that you are measuring. For example, in a measurement of loudspeaker frequency response, the Cal channel will usually be made with a probe that senses the high level signal from a power amplifier as it is applied to the loudspeaker terminals. In Full mode (with the AudPod), this input is usually labeled on the Levels Form, and will in most cases use the default input source defined as the "RefCalSource". In Free/Demo Mode, no channel switching is possible under Praxis control (because there's no AudPod), so both channels are simply "inputs".

The **Measure Channel** (abbreviated as just "Meas") is made with a microphone, probe, or line input, and usually senses a signal that is strongly determined by the device you are measuring.

In some situations, such as simple FFT monitoring, these channel designations are not relevant, so the channels are then simply designated as "Channel1" and "Channel 2"

## ***Computer Resource Availability***

Praxis operates in the 32-bit Microsoft Windows operating system, and must share resources with other active programs. Praxis can be very resource-hungry, however, and for best operation need timely access to the resources it needs. The following guidelines should be observed during use of Praxis:

- **Do not allow other applications (or the Windows operating system) to access the soundcard that Praxis is using!** This can cause large errors in measured data. Some applications may change settings of the Windows Mixer controls. If this happens during Praxis use, the calibration for any further Praxis measurements during that session will be invalid. You may need to change settings in you Windows Control Panel to prevent the operating system from generating tones, clicks, and other interruptions through the soundcard.
- Close down any other unnecessary applications during acquisition of data with Praxis. This will allow more memory and processor time to be available for signal processing and presentation.
- Operation of Praxis while the computer is connected to a network may cause acquisition errors, as network activity may occupy the processor's "attention" for excessive time intervals. If problems occur, close the network connection. Arrival of email during an acquisition can cause excessive interruptions.
- Don't browse the internet (or even have a browser open), nor attempt to burn CDRoms during measurements. Praxis is using your computer as test equipment, and needs timely access to it.
- If you are using long Script processes, you may need to disable screensavers (see your Windows documentation for instructions).
- Praxis' resource requirements are much less demanding when it is not acquiring data (such as when it is being only used to work with data to and from disk files).

## ***Measurement Units***

Praxis will try to maintain measurement units when practical. When the AudPod is used ("Full Mode"), the raw measurement units are typically inferred from the selected input source(s). These are typically in units of Volts, Pascals (for acoustic pressure) or g (for accelerometer inputs). The units can be further changed by use of some postprocess operations. In Free/Demo Mode, raw measurement units are unknown.

Measurements shown in the frequency domain are typically shown in "log magnitude" format, otherwise known as "decibel" or "dB". The character(s) after "dB" refer to the value to which the notation refers. For example, a hypothetical "dBY" reading would indicate that the value was calculated by dividing the magnitude value (measured in Ys) by 1Y, then taking the base10 logarithm, and then multiplying by 20:

$$\text{reading} = 20 * \log(\text{Magnitude}/(1*Y)).$$

A common measurement for loudspeaker work is to measure the acoustic frequency response relative to the applied drive voltage. This is done with a microphone and a probe on the loudspeaker system's terminals. In such a case, the magnitude result of the measurement is given in units designated as "**dB**". This is used as a shorthand form to indicate the equivalent SPL for 1V applied. If you add the value 9.03dB ( $=20 \cdot \log 2.828$ ) to this value you would then obtain the often-used "loudspeaker sensitivity" value, in SPL, for **1W (at 8 ohms nominal load) applied. To obtain a graph reading in this manner, use an Offset value (see "The Del/Weight Format Tab") on the plot, of 9.03dB.**

In frequency response measurements, the units of the resulting data curves will depend on the units of the "raw" data from the selected inputs. Praxis will try to use the following guidelines:

- In a frequency response measurement or a Math Postprocess "Div" calculation, if the inferred units from both sources (or plots) are the same, the resulting units will be 'dB' for response magnitude, or degrees for phase.
- Unknown units (from Demo mode, or imported data) are typically indicated by the units "dBU", "dBr", or sometimes just as "dB".
- If either source for a Math Postprocess measurement is in units of "dBU", the ending result will also be in "dBU" or "dB".
- Math PostProcess operations or frequency response normalizations can result in units such as "dBV/g", and these may be denormalized by later PostProcess MULT operations such that they remove the units of the denominator.
- Multiplied units such as "dBV\*V" or dBV\*g will not be generated. If both sources have the same units for a Postprocess Math Multiply operation, the result will use the same units. (Such results are not usually meaningful, however).

## ***Time, Frequency, Size ...and Noise***

When data is measured using Analog to Digital (A/D) converters, there are certain relationships and limitations that result based on the values of the SampleRate and bit resolution. In addition, measurement techniques can have a large bearing on the degree to which noise (in rooms, electronics, hardware, even from data resolution) degrades the measurement results.

- The **length of time** over which valid samples are collected **affects how low** in frequency you can measure. As an absolute, if you collect only **t** seconds worth of samples, you can have no certain knowledge about data at frequencies below  $1/t$  [Hz]. Your measurement technique may provide values at frequencies below this...but they are essentially meaningless. This frequency limit is a best case value -- you should mistrust data as it gets below several times this frequency.
- The **sample rate limits how high** in frequency you can measure. You can measure no higher than one half the sample rate (usually 0.45x is a good value to use with soundcards). Your microphone, its calibration files, or the measurement channel (other electronics) can further limit the upper frequency limit.
- **Very high sample rate and very long sample times mean large data size and long processing times.** Beginning users often immediately go for huge acquisition sizes and maximum sample rates, and spent large amounts of time, make gigantic files, or even lock up their computers. After they learn a little more, these users then begin to use sample lengths no longer than are really needed for the measurement. "Lack of information leads to ignorance. Too much information does this, too." For generality, sample rates of 48kHz are usually adequate for nearly all audio measurements (because lower sample rates than that are troublesome with some soundcard hardware).
- All other things being equal, **longer measurement times usually provide better noise immunity.** But choice of measurement stimulus may be a more significant factor.

- **Too strong a stimulus (or too long) can also cause noise or distortion.** Although a long high level stimulus is normally good for improving the noise floor, the validity of the data can degrade if you begin to overheat, overdrive, or damage what you are testing.
- **Proper adjustment of levels** into your soundcard is essential for low noise in a measurement. The noise inside the soundcard stays at about the same level -- you need to get the signal levels as high above that as you can.
- **Noise in most audio measurements is more of a problem at lower frequencies.** Acoustic insulation and isolation do not work as well at low frequencies and some forms of electronic noise increase with falling frequencies. Try to use a stimulus signal (such as a log chirp, or an MLS/Pink type) that applies more energy into lower frequencies.

## ***Processing Times***

Some of the operations performed by Praxis are very complex and can manipulate huge data structures. These kinds of operations can be very time consuming, particularly if settings are chosen carelessly.

Some things to avoid, if processing time is a concern:

- **PostProcess Smoothing over wide "Octave Range Limits".** A higher number entered in the Octave Bandwidth and Octave Range Limits controls uses a smaller bandwidth, which speeds up the operation. Brickwall filter shape is much faster than is Butterworth filter shape.
- **Excessive Acquisition size.** This number of points must be acquired by Praxis each burst before its processing can even begin, and then takes longer times to process. A doubling of Acquisition size more than doubles the measurement time.
- **Use of "Multi" Plots with many curves.** Praxis is able to make live measurements even with many other traces shown on its Primary Plot. But these do increase the drawing time for the plot, particularly when it is re-formatted, or re-sized.
- **Gradient Colors in waterfalls.** These can look very nice, but greatly slow down the drawing time. The best approach is to do all formatting without Gradient colors, and apply the gradient (by changing one of the colors) last.
- **Too many time slices in waterfalls.** Each slice in a waterfall involves an entire FFT and surface drawing operation. Don't bog down your computer with detail you don't really need.
- **Saving files in ".pxd" format.** This is a text-based format, which uses more disk space and is much slower to read. If you won't need to view the data in text form, save it in the default binary ".px2" format.
- **Unnecessary use of too-high sample rates, too-narrow resolutions, or too-long sample times.** See "Time, Frequency, Size"

## ***Chirp vs. MLS vs. Stepped Sine***

For frequency response measurements with Praxis, the three most useful stimuli are Chirp, MLS, and SteppedSine. They have different characteristics, advantages and disadvantages. Many studies and a number of AES papers have been written about this. An excellent treatment of the subject is given in "Transfer-Function Measurement with Sweeps", by Muller and Massarani, J. Audio Eng. Soc., Vol. 49, No. 6, 2001 June.

Here is a brief summary of some relevant issues:

Chirp:

This stimulus provides good immunity to noise due to its efficient use of the dynamic range of the

device and the stimulus and acquisition channels. It has good immunity to distortion, and tolerates time variations better than does MLS. It requires considerable computer processing power and large memory, however, particularly if impulse response results are needed. This is usually the best choice for use in Praxis.

MLS:

This is a mathematically elegant stimulus. It uses a binary sequence, which, when used to generate an analog stimulus, provides a fast, efficient way to obtain the impulse response of a device or system. It is easily degraded by distortion and time variation, and this degradation tends to be manifested in a limitation of dynamic range. The device being measured requires a "warmup" time while under influence of the stimulus, and this can lengthen the measurement time.

Stepped Sine

This is a technique which is very easy to understand. Tones of each frequency in a data set are applied to a device, and the change in level (and perhaps the phase shift) is recorded on a graph vs. the frequency of the applied tone. In a well-arranged measurement, the acquired signal is first filtered (so only content at the applied frequency is measured). This method give good noise immunity, though with risk of over stressing the device being tested with the long-duration, high level stimulus tones. It is very good at ignoring distortion in the device being measured. And it takes an extremely long time to perform.

### ***Level Adjustments (Free/Demo Mode)***

In Demo Mode, all input selection and level adjustments must be made using the Windows mixer applet.

The goal of level adjustment is to present the soundcard's A/D converter with the highest level signal as possible, such that it never overloads during any portion of the acquisition.

There are bar graphs on the Praxis Levels Form, which indicate relative levels of the applied input. If one of these goes to near full scale (it will display as a blue bar), a red spot will appear indicating overload. Depending on your soundcard hardware, overload may happen before this much level is applied. Some experimentation may be required. These adjustments must be first made, and the Demo Probe Calibration script successfully run before impedance (and Thiele-Small) measurements can be made. Do not readjust these levels without re-running the calibration script.

In Demo RTA mode, amplitudes shown on graphs will increase as the level adjustments are changed. This is because Praxis cannot know about absolute levels in Demo mode -- it can only see the signal levels being applied to the A/D converter, it cannot know where you have the Mixer level controls adjusted.

It is always a good idea to have all available controls of your Windows mixer displayed at first, to make sure inadvertent inputs, outputs, or settings are not being applied.

### ***Level Adjustments (AudPod Mode)***

When adjusting input levels in Full (AudPod) mode using the controls in the Levels Form, Praxis keeps track of the amount of gain or attenuation being applied.

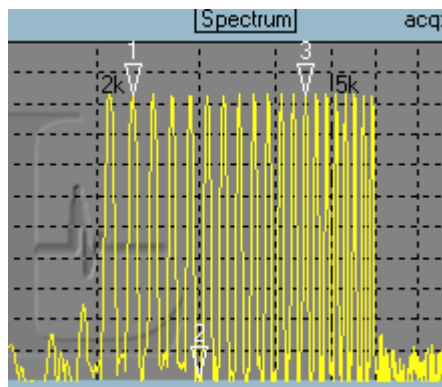
Using this information, along with knowledge of the calibration factors of the soundcard, data is in most cases graphed in terms of physical measurement units. The system attempts to show the levels at the point where they are sensed, rather than just at the A/D converter. This is distinctly different than the operation in Demo Mode, in which, for example, the curves of RTA plots go up and down in response to the settings of the Mixer's level controls.

Note that there may be momentary jumps in displayed data curves (and perhaps other strange effects) if you adjust the input level adjustment sliders, or change the input selection, during an acquisition. In some cases, it may be necessary to stop an acquisition process and then restart it in order to cause the system to reformat the plots correctly for inputs that have changed.

## ***Multitone Testing***

Praxis is able to generate large numbers of tones while simultaneously acquiring input samples for analysis. These kinds of multitone tests are useful for analyzing intermodulation distortions, for "spectral contamination" tests, or simply as "torture tests" for drivers or electronics or codecs to look for breakups and other effects.

A good history and theoretical discussion of multitone distortion testing is given in "**Multitone Testing of Sound System Components - Some Results and Conclusions, Part 1: History and Theory**", Czerwinski et al., *J. Audio Eng. Soc.*, Vol. 49, No. 11, 2001 November.



A spectral contamination test applies a set of tones, sometimes packed within one frequency region, sometime spread over the spectrum, to the unit being tested. Then, generation of other tones is inspected, between or away from the applied frequencies, in the resulting FFT spectrum. With use of external electronic filters, this technique is capable of detecting very subtle distortion effects. The number of tones which can be applied simultaneously by Praxis will be limited by the soundcard type and the processor speed. 40 or more tones can usually be handled without problems.

## **Pass/Fail Evaluation**

Praxis can evaluate most measured curves relative to pairs of predefined limit curves. This feature will most often be used in transducer or loudspeaker production environments for quality assurance.

The Pass/Fail evaluation features are implemented within a Praxis script, usually along with automated measurement functions. The evaluation is performed on the curve within the Primary Plot. See the "QC Pass/Fail" script (in your C:\Program Files\Praxis\VCL Scripts directory, runnable from the Script Launcher) for a simple example of how to implement the pass fail functions. .

The limit files can be easily generated from existing measured or postprocessed curves. A limit curve is declared from the "Trace(A)" curve within a Plot Form or from a data file. The limit curve data type must be compatible in measurement units and type with the curve to be measured. Changes to the appearance of the curve that result from an Offset or Weight (see the "Weight Format Tab") will also apply to the limit curve defined from the plot. This provides an easy way to set limit curves which specify a tolerance range around the measured curve from a reference device. The limit curves are stored as "Multi" curve files, so other multi curves cannot be used during Pass/Fail evaluations.

The evaluation can be specified to be absolute, or alternately, so that Praxis can adjust the overall offset of the curve being evaluated to find if it can then be made to fit. Allowing such adjustment equates to evaluating only the shape of the curve (such as the frequency response) while not evaluating absolute sensitivity.

Any number of limit curves can be specified, indexed by a number. Different sets of limit curves can be saved in alternate directories. The directory name to be used at any time for storage or evaluation is defined by the "**Praxis.LimitFileDirectory**" property.

Script methods provided for Pass/Fail functions include:

```
procedure DefineLimitFile(IsUpperLimit:boolean;LimFileName:string;Limitindex:integer);  
  //assigns/copies limit file from a file.
```

```
procedure DefineLimitPlot(IsUpperLimit:boolean;Source:TDataPlot;Limitindex:integer);  
  //assigns/copies limit file from a plot.
```

```
function IsWithinLimits(var BadFreq:double;Mask:integer;  
EvalAbsolute:boolean;CurveID:integer;LimitIndex:integer):boolean;  
  //Evaluates the curve in the Primary Plot (only over displayed horizontal range).  
  //Bad Freq will return the first frequency found which fails, if any.  
  //Mask is 1 if only the lower limit curve is used; 2 if only the upper limit is used; 3 if both.  
  //CurveID is the "Field number" of the curve to evaluate (as counted from the top in the  
  drop-down boxes found on the "Traces Format Tab" of the Plot Form).
```

```
function ShowLimitCurves(Mask:integer;CurveID:integer;LimitIndex:integer):boolean;  
  //similar to IsWithinLimits, but only shows the limit curves, does not evaluate.
```

## Praxis Operating Modes

Praxis can operate in one of two modes: "Full" (or "Normal") mode, and "Demo" mode.

The operating mode is determined by whether Praxis detects the presence of the AudPod device on the computer's USB port.

### ***Free/Demo Mode***

In Demo mode, Praxis offers a limited set of features in its free installation. Measurements of Thiele/Small values and of RTA spectrum are supported in this mode.

All Praxis data files (and most data files from LAUD and IMP) can be loaded into Praxis "plots", which can be formatted, post processed, displayed or printed.

Measurements made in Demo mode are strictly relative -- there is no way to reliably determine absolute voltage or pressure levels without using the AudPod device, which includes a calibration reference and a trackable input gain control.

Impedance measurements, with results in units of Ohms are possible (and used for Thiele/Small Parameter measurement), through utilization of a separate "reference resistor".

Praxis' Demo mode is able to run "scripts" such as the Demo scripts that are presented when Demo Mode is started, allowing within scripts those operations that are generally supported in Demo mode. The Liberty Script Designer is provided and fully enabled for developing Praxis application scripts (though "Full Mode" Praxis operation will be required to test or operate any scripts that utilize "Full Mode" features).

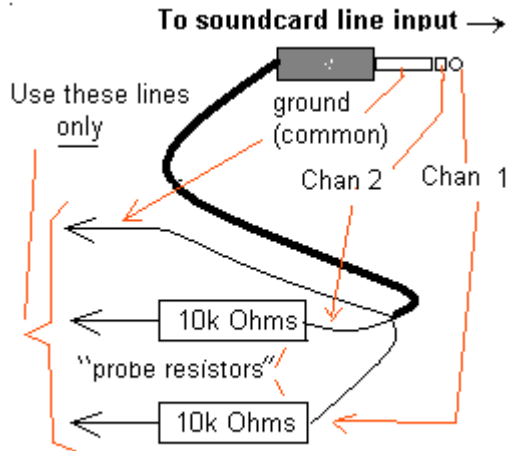
The Demo mode will not timeout nor present any "nag" screens, and does not limit saving, import or export of files. All of PRAXIS' post processing operations are available.

### Probes (Free/Demo Mode)

Probes are used only for making impedance measurements of woofers, in order to extract Thiele-Small parameters. If you use these type probes or use this software in any way to make measurements, **you do so entirely at your own risk**; incorrect connection or operation can damage your computer or other hardware -- please see all cautions given in the Demo Mode Thiele-Small Parameter Measurement Script. **Liberty Instruments will not be responsible in any way for any undesirable results.**

The probes can be constructed as shown. The 3.5mm stereo phone plug is to be inserted into a STEREO LINE INPUT jack of your soundcard. (Note that in some sound cards, the channel may be reversed -- if strange impedance measurements are obtained, try switching the connections of the probes).





### Thiele-Small Measurements in Free/Demo Mode

Praxis' free Demo Mode provides for guided measurement of Thiele-Small parameters, by use of an included Script.

Measurements can also be performed manually (without the script), though this will be less convenient. Only the chirp stimulus is enabled for Thiele-Small measurements in Demo Mode (MLS, MLS/pink, and Stepped Sine methods are also supported in Full mode).

Input adjustments must be first made, and the Demo Probe Calibration script successfully run before Thiele-Small measurements can be made. Do not readjust these levels without re-running the calibration script, when other T/S measurements are to be made.

The Thiele-Small parameter measurement process first performs an impedance measurement of the woofer. **Impedance measurements in Demo mode are limited to frequencies below 2000Hz.**

For these measurements, you will need:

- a woofer to measure
- a power amplifier (capable of a few watts power), which **must have ground referenced outputs**. This means that the "common", "[-]" or "cold" output terminal must be the same electrical circuit node as the input ground shield. Usable results **cannot** be achieved if this is not the case
- a way to orient the woofer so that it is away from surfaces, so that any air vents are not blocked, and so that a small weight can be placed on the dustcap.
- a 10 ohm (approximately) resistor of known resistance, 1/4 watt or higher
- Some modeling clay, and a way to measure the weight of pieces of it
- home-made "probes" (including series resistors). **DO NOT TRY TO DO THIS procedure WITHOUT PROPER SERIES RESISTORS IN LINE WITH THE PROBE CABLES.**
- cables and wiring to connect your soundcard to the power amplifier input, and the woofer to the resistor and these to the power amplifier.
- **a willingness to accept ALL RESPONSIBILITY** for any problems which may occur. Liberty Instruments and its personnel will not accept responsibility for users' interpretations or execution of this measurement procedure.

The instructions for performing this test are given in the Thiele-Small Demo Script.

### RTA Operation in Demo Mode

RTA operation in Demo Mode is much less strict than is Thiele-Small measurement.

For RTA operation, Praxis simply outputs Pink Noise (only this stimulus can be selected) and displays an RTA plot of the spectrum that is directed toward it via the Windows Mixer.

You can use the Acquisition Configure Form and the Primary Plot's Formatting options to adjust the display to your tastes.

## **AudPod (Full) Mode**

In this mode, all Praxis features are enabled. These include all presently supported acquisition and stimulus modes and the ability to save files from postprocessing operations.

With the AudPod, the analog signals to and from the computer soundcard are routed through the AudPod device which controls input levels, switches between different input sources, provides a microphone preamplifier and bias, and provides a calibration tone generator. Measurements made through the AudPod can be expressed in true units (such as Volts, Pascals, or SPL).

Support for development of Praxis "scripts" is provided via the Liberty Script Designer, in which the developer can create his scripts in "RAD" style, via visual as well as text-based manipulations.

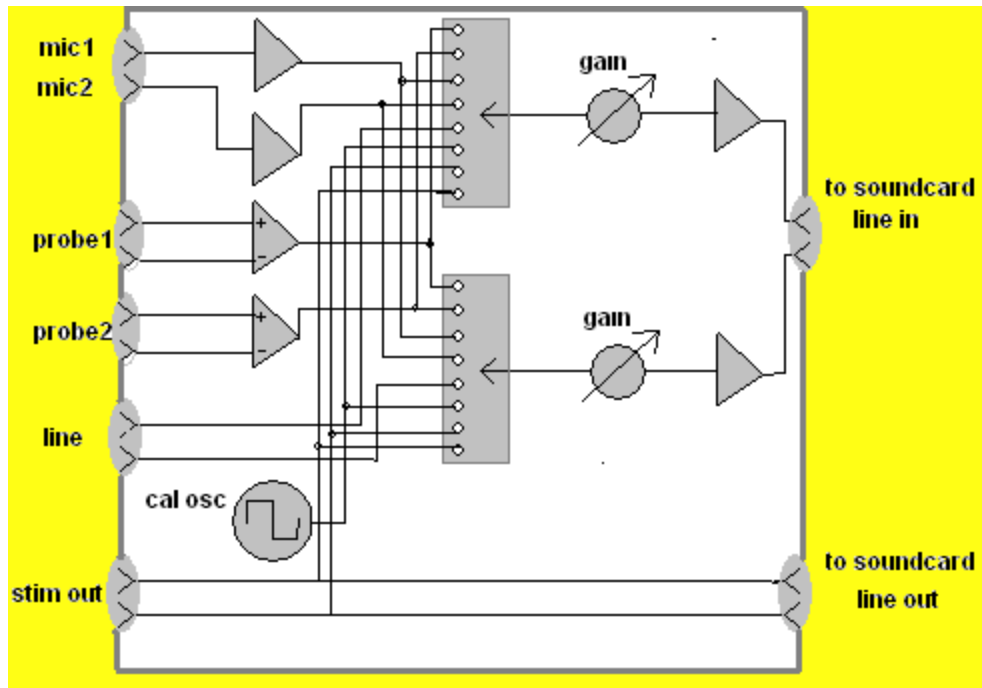
### AudPod Description



The Liberty Instruments AudPod is a digitally-controlled analog interface device, which connects to your computer via its USB port. The AudPod provides a number of advantages for the Praxis system:

- It provides for input source switching (such as between probes, microphones, line level inputs) so that measurement sequences can be automated.
- It provides a high quality, predictable, and precise input gain controller.
- It provides for differential type probe inputs, to minimize interference due to ground currents, to allow for accurate impedance measurements, and to allow voltage measurements to be non-ground referenced. (See Note Below).
- It provides amplification, signal conditioning, and overvoltage protection for the soundcard inputs. The output circuits and the probe input circuitry are DC coupled so that the AudPod will not limit the low frequency capability of your sound hardware.
- It provides bias and precision amplification for electret microphone (or accelerometer) inputs. These inputs are AC coupled, with a corner frequency of less than 1Hz.
- It provides a calibration reference for audio signal levels so that Praxis can identify applied and measured levels in physical units( such as rms Volts or Pascals).
- It provides a hardware key to unlock the "Full" features of the Praxis system.

But the AudPod is not a D/A or A/D converter, nor is it a soundcard replacement. The AudPod works **with** your soundcard to standardize and to extend its input capabilities.



**AudPod Signal Paths**

### USB (Universal Serial Bus)

The Praxis AudPod requires a Universal Serial Bus (USB) interface to connect to the computer. The AudPod also receives its power from this bus connection, and so does not use batteries or any other power supply.

### Automatic Configuration

When your AudPod is connected to your computer, the computer should automatically load the correct drivers for the device. The first time you connect to a given USB port on your computer, however, your computer may ask you to supply a source disk for an "INF" file. The system should be directed to use its internal, built-in drivers.

### Live Connection

USB is a "hot pluggable" interface. This means that the AudPod can be connected to the computer while the computer is running, without the user first needing to turn the computer off. It is important to remember, however, that Praxis cannot properly reconfigure itself for a change in the AudPod's connection state if Praxis is already running. Praxis configures itself for AudPod operation when the program is **started** -- **the AudPod should not be attached to (or detached from) the computer's USB interface while Praxis is running**, or unpredictable results will occur! If you need to attach or detach the AudPod, first shut down Praxis, make the change, and then restart Praxis.

### Computer Power-Up Problems

Under some circumstances (certain computer motherboards with certain operating systems), there have been cases reported where the computer may not power-up while the AudPod is attached. If this condition exists with your computer, first detach the AudPod's USB cable (either at its computer end or at its AudPod end), power up the computer, and then after the computer is

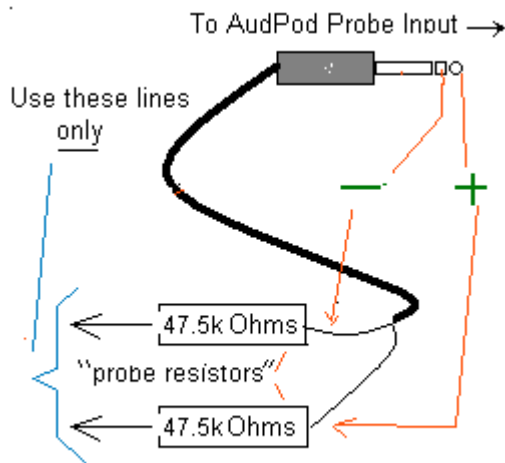
successfully booted, attach the AudPod.

#### USB Connected Soundcards

Praxis is able to work with soundcards that connect via USB, and this allows for use with laptop computers. If possible, give the USB connected soundcards the most direct connection to the computer (rather than through a series of hubs). The AudPod, however, can connect through any hub arrangement.

#### Probes (Full or AudPod Mode)

The probes supplied and used with the AudPod are configured as shown below. The connector shown is a 3.5mm stereo phone plug. The resistors **MUST** be at the far end of the cable from the phone plug (to avoid frequency roll off due to the cable capacitance). The cable lengths should be kept below approximately 20ft. Note that each of these probes serves one channel -- the two inputs shown are used differentially.



The differential nature of the probes is designed to minimize the effects of ground resistance in measurements and to reduce common mode interference. The probes and inputs are not intended, however, to produce a high degree of common mode rejection. The actual rejection will depend on the source impedance of the nodes being probed, as well as on imbalances in probes and circuitry. If higher CMR is required for a particular application, the value of one of the probe resistors can be made slightly adjustable for nulling common mode levels.

If additional resistor values than the ones shown are used (for use with higher voltage levels, for instance), the total resistor value should be identified to Praxis using the "Setup->General->Probe Resistor" menu provided on the Levels Form. The edit form that is presented to enter these values will also show the voltage handling capability of the probe when that series resistor is used.

*The input voltage handling limits (for safety and for linear operation) of the Probes, with respect to ground, are **not** affected by differential connection. The maximum peak voltage that can be applied to either probe contact is **NOT** enhanced or altered by the connection of the other contact of the probe! For linear operation with the standard probes (using 47.5K resistors), **no** signal-plus-DC voltages above **30V peak** should be applied to the probes. The allowable voltage will approximately scale with the value of the probe resistor. **MEASUREMENT OF VOLTAGES APPROACHING OR ABOVE 100V PEAK WITH THE PRAXIS SYSTEM SHOULD NOT BE***

**ATTEMPTED --ANY SUCH OPERATION IS DONE ENTIRELY AT THE USER'S OWN RISK.**

## **Microphones**

Microphones used in Demo mode must be compatible with the inputs to your soundcard.

Microphones used with the AudPod are intended to be based on small electret capsules, such as the Panasonic WM-60 or WM-61 series. Praxis microphones (made by Kim Girardin of Wadinhome Sound) use these capsules, and are calibrated against a laboratory reference microphone for frequencies from midrange up to their upper frequency limit (above 20kHz). At lower frequencies, the mics are typically not individually calibrated, but testing has shown them to have very little variation down to below 10Hz -- Praxis will use the calibration data for the lowest given frequency point for all frequencies below that point. The AudPod provides bias to power the capsules and low noise amplification and impedance buffering for sending the signal to the soundcard. With a standard mic of typically 10mV/Pa sensitivity, the AudPod's electronics will overload at 130dB SPL peak. Special "High SPL" microphones are available from Wadinhome with lower sensitivities and can be used to approximately 142dB SPL peak.

Other microphones can be used with the AudPod. Sensitivities of the microphones should usually be kept to under 15mV/Pa (when loaded by an impedance of 3.3k ohms). If the response curve of the microphone is known, a microphone correction file can be easily created for it using a text editor (see Praxis File Types).

### **Alternate Microphone Connection** (with AudPod, for externally powered microphones):

The "Probe1" input of the AudPod can alternately be used for the "Microphone 1" input to the AudPod. This connection is appropriate when you are using an externally powered and preamplified microphone. When this connection is used, only one probe (Probe 2) can be used simultaneously with the Mic.

To use this connection, go to the "Config, HardwareSpecific" menu of the Main Form and click on "**Mic1 Uses Probe1 Jack**" so that it becomes marked with a check. When a measurement requires use of Microphone 1, Praxis will switch instead to the Probe#1 input for this signal. The currently declared microphone correction file for Mic1 will be used.

The input can be connected as balanced or unbalanced.

- For balanced operation, the "tip" of the plug going to the "Probe1" jack is the non-inverting [+] input; the "ring" of the plug is the inverting [-] input. And the sleeve is, of course, ground.
- For unbalanced connection, use the tip (hot) and sleeve (ground).
- loading of the microphone preamp by either the tip or the ring is 3.6k ohms to ground.

The input sensitivity is 12dB lower when this input is used instead of the AudPod's Microphone input. This gain difference is taken into account by the program, so no change is needed in the Mic1 correction file.

### **Microphone Data Files**

PRAXIS corrects for measured acoustic frequency responses through use of the known response and sensitivity of the microphone that is used for the measurement. PRAXIS obtains this information about the microphone by loading a microphone correction file (with file extension ".dat"). If you are using a microphone of known response and sensitivity, you can easily generate

your own data file for it by use of a text editor. For many laboratory microphones, a data file that indicates a flat response (0dB at every point), and a reported sensitivity is all that is needed.

The data file for a microphone can be loaded using the "Config->Hardware Specific->Mic/Accel" menus of the main form. The files can also be used to provide the measured response of an accelerometer sensor.

The text below is an example of the format needed for a microphone calibration file. The first line needs to begin with a quote (") mark and contain the reference sensitivity value, followed by the letters **MV/PA** (or MV/g for an accelerometer). The line should include the characters "NINV" if the mic is non-inverting (otherwise it will be assumed to be phase inverting).

The value near 1kHz is normally specified to be "0dB", and the reference sensitivity specifies what that 0dB equates to (in millivolts per Pascal). 1Pa rms = 94dB SPL.

Each line that does not begin with a quote is interpreted as follows:  
Frequency (in Hz), response magnitude (in dB), phase angle (in degrees).  
The phase angle is optional (need not be provided).

Frequencies MUST be in increasing order, but may be given at any increment. Values will be interpolated and extrapolated from those provided. Sufficient frequency points should be included to define the response shape, as Praxis will attempt to fit the points to a curve using a spline fit. The last several frequencies should be closely spaced and have the same response magnitude.

Often, the best approach to generating your own microphone data file is to simply start with and edit an existing data file, such as the "ExampleMic.dat" or the "flatmic.dat" file provided in the "C:\Program Files\Praxis" directory after installation of the software or demo. You can use the Windows built-in "Notepad" application to edit this file. Here is an example of the contents of a microphone response file:

---

```
" Example Mic correction data: NINV Ref Sensitivity = 8 MV/PA "  
" the number above is needed to relate the sensitivity.  
" Example is for for a flat mic that is -3dB at 10Hz and 50kHz  
" Note the last two points are closely spaced  
" Hz, dB, degrees  
2.5 -12.0 0  
5 -6.0 0  
7 -4.0 0  
10 -3.0 0  
15 0 0  
17 0 0  
20 0 0  
40 0 0  
80 0 0  
160 0 0  
320 0 0  
640 0 0  
1280 0 0  
2560 0 0  
5120 0 0  
10240 0 0  
20480 0 0  
40960 0 0  
45000 0 0  
47000 0 0  
50000 -3.0 0
```

50001 -3.0 0



## Credits and Tools Used to make Praxis

The Scripting capabilities of Praxis use components from [Dream Company](#). Dream Scripter and Dream Designer are absolutely amazing pieces of software (although often difficult to understand).

The USB host-side access for Praxis's AudPod uses an excellent Delphi component developed by Robert Marquardt, available from [project JEDI](#).

Praxis' Help files were developed and organized using HelpScribble, by Jan Goyvaerts of [Just Great Software](#). I can't say enough about this tool - after spending time with it, I can't imagine developing Help files without it.

The AudPod circuit board was designed using the [Eagle Layout Editor](#). I can highly recommend this layout tool, which is not only easy and powerful, but also inexpensive.

Praxis uses dll files for some auxiliary functions. These files are distributed as either freeware or under the GNU Lesser General Public License, or under terms as described below. No copyrights of these files are claimed by Liberty Instruments. Many thanks to the authors:

- **r8brain.dll** by Aleksey Vaneev (<http://www.voxengo.com/r8brain/>) for sample rate conversion.
- **ZipDll.dll** and **UnzDll.dll** by Chris Vleghert and Eric W. Engler (<http://www.delphizip.net>) for file zipping and unzipping (used to manage our "zipped scripts")
- **WinIO.dll**, copyright 1998-2002, Yariv Kaplan (<http://www.internals.com>) for parallel port access.
- **Get\_IO.dll**, by Graham Bartlett (<http://sourceforge.net>) for identifying addresses of parallel printer ports.

Praxis was written *entirely* in [Borland's Delphi 5](#). And I'm sure that it is no coincidence that many of the exceptional tools listed above were *also* developed using Delphi. Development of Praxis had an initial false start, with much frustration, using the "most popular" development tool -- it was finally restarted using Delphi, and I've never looked back.

The Praxis software and documentation, and the AudPod hardware and firmware were written and/or developed by Bill Waslo at [Liberty Instruments, Inc.](#) Liberty Instruments' daily operations, and customer interface (the friendly voice on our phone) are managed by Carol Stewart.