OmniSound™

OSC-112

USER'S MANUAL

DOC-336

Rev. 1.0

Med Associates recommends reading this manual prior to operating this product.

Med Associates Inc.
P.O. Box 319
St. Albans, VT 05478
www.med-associates.com

Phone: 802.527.2343

Fax: 802.524.2110

CONTENTS

CHAPTER 1 INTRODUCTION

The OSC-112 OmniSound + OmniCtrl Connection Panel provides 12 Med Associates Input/Output lines and two audio outputs. The OSC-112 connects to the IC-124 OmniControl Card via HDMI cable. The OSC-112 can play pure tones, white noise, clicks and wave files via the two output audio channels.

Power Supply and Data Communication Specifications

Input power	28 VDC via M12 circular connector "28 V IN"
Output power	28 VDC via M12 circular connector "28 V OUT"
Upstream Data Connection (to IC-124)	HDMI cable "HDMI COM"
Downstream Data Connection (to daisy-chained OSC-112)	HDMI cable "HDMI AUX"

Digital Input/Output

Digital I/O Lines	12*
I/O Line Interface	3-pin Molex connector
I/O Line Configuration	28 VDC
Max Continuous Current	1.25 A (all outputs combined)

*Each I/O line may be configured as either an Input or an Output in the Med Hardware Configuration Utility.

Audio Output Function

Output Modes	Pure tones, white noise, clicks, wave files
Audio Output Ports	2 @ speaker-level, 1 @ line-level
Speaker-Level Output Ports ("CH1", "CH2")	1/4" stereo speaker jack
Line-Level Output Port ("LINE OUT")	1/8" stereo "headphones" jack
Audio Output Sound Pressure Level (SPL) CH1/2	20 dB SPL to 100 dB SPL in 1 dB increments**
Audio Output Format (Tone, White Noise, Click)	140 kHz, 32 bits
Wave File Storage Media	SD Card, exFAT file system
Wave File Format	44.1 kHz, 16 bits
Wave File Capacity	255 Files
Audio Output Offset Range	-20 dB to +5 dB per channel
Pure Tone Output Frequency Range	1,000 Hz to 70,000 Hz
Pure Tone Duration	1 ms to 65,535 ms in 1 ms increments
White Noise Output Frequency Range	Broadband white noise

White Noise Duration	1 ms to 65,535 ms in 1 ms increments
Click Output Frequency Range	1 Hz to 100 Hz
Click Duration	1 ms to 65,535 ms in 1 ms increments
Rise/Fall Time	0 ms to 1000 ms
Rise/Fall Time Modes	Pure tone, white noise

*SPL obtainable via CH 1/2 with ENV-228AM or ENV-328BM/BW speakers measured at 7.5 cm, 0° axial offset with Med Associates' ANL-930 Sound Pressure Level Measurement device.

Definitions

Rise/Fall Time is the duration in milliseconds for the sound output generator to "ramp up" to the desired sound pressure level output at sound onset and "ramp down" to silence at the end of playing a sound. The rise/fall time helps eliminate speaker "pop," especially at the sound onset. The rise/fall Time duration will be added to the beginning and end of tone or white noise duration. A rise/fall time of 10 ms is a good starting point for determining your desired settings. A one second pure tone with a 10 ms rise/fall time will be 1.020 s long, with one second at full amplitude. Rise/fall time is applicable to pure tone and white noise modes, but not click or wave file modes.

dB: decibels, shortened version of dB SPL, used in acoustics as a unit of sound pressure level

Hz: Hertz, cycles per second

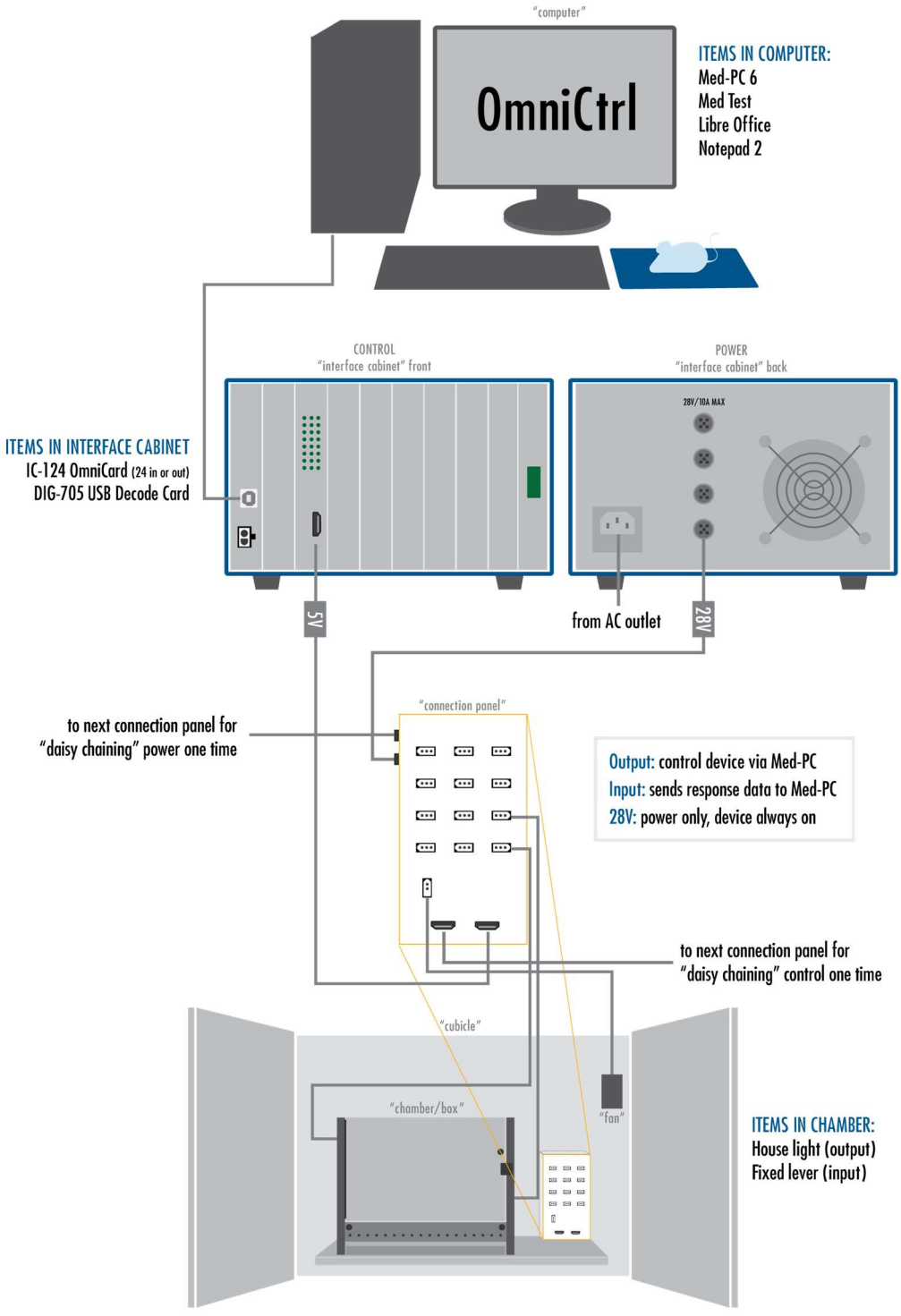
ms: milliseconds, 1/1000th of a second

Overview

This manual covers the sound generating portion of the OSC-112 OmniSound + OmniCtrl Connection Panel. The I/O portion is covered in **DOC-334 OmniCtrl Power and Interface System**.

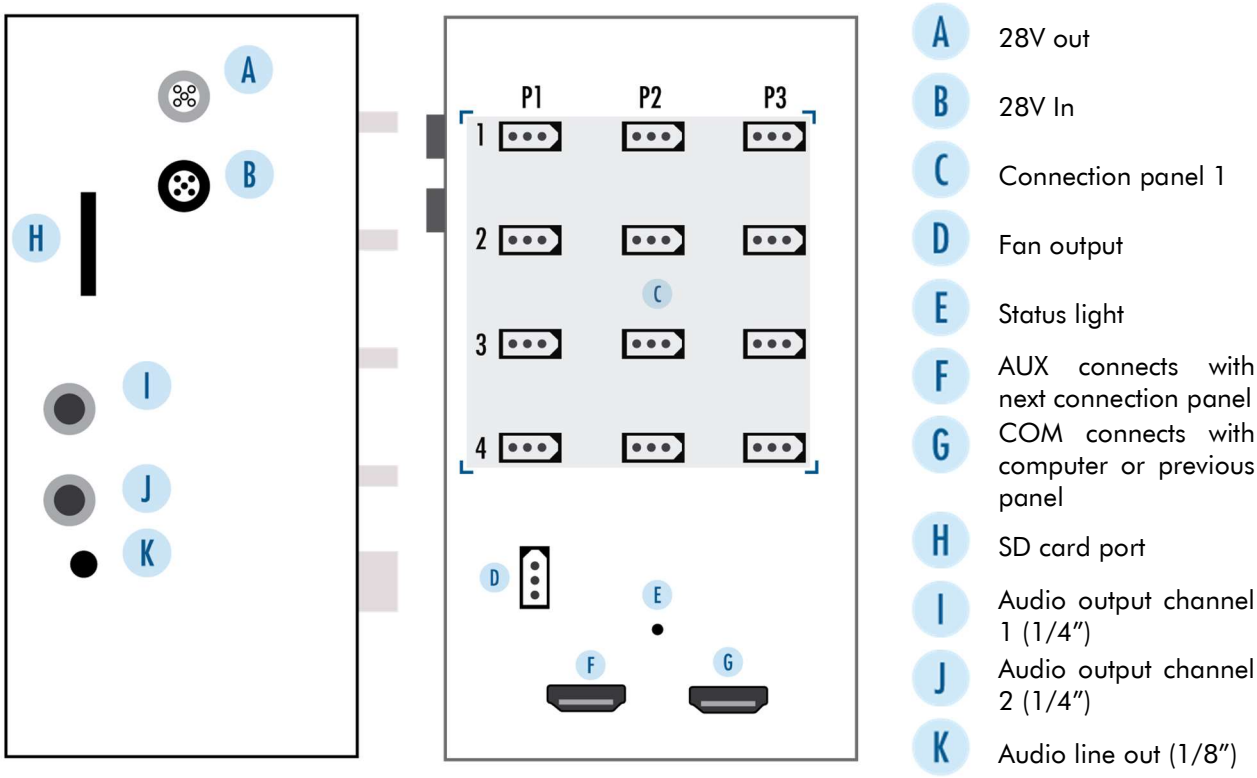
CHAPTER 2 HARDWARE CONFIGURATION

FIGURE 2.1 – OVERALL SYSTEM CONNECTIONS AND COMPONENTS



See **DOC-334 OmniControl Power and Interface Systems** for information on power and data interface connections. This chapter covers the audio connections on the OSC-112.

FIGURE 2.2 -OSC-112



Audio Connectors

The left side of the OSC-112 has two 1/4" speaker jacks, one 1/8" speaker jack, and one SD card slot. The two 1/4" speaker jacks are labeled "CH1" (channel 1) and "CH2" (channel 2). The 1/8" speaker jack is labeled "LINE OUT". The SD card slot is label "SD."

An SD card with wave sound files in the SD card slot is required for playing wave files through the OSC-112. The files must be on the root folder of the SD card and named in the format "WFn.WAV" where "n" is a number 1 through 255. See Wave File Manager section in Chapter 3 for discussion of the utility for naming and moving files to the SD card.

The 1/4" speaker jacks CH1 and CH2 produce amplified monophonic audio output. Both CH1 and CH2 may be used concurrently to produce any combination of tones, clicks, or white noise. For playing wave files, CH1 will generate a mono wave file output. When playing a stereo wave file, CH1 will produce the left channel sounds, and CH2 will produce the right channel sounds.

The 1/8" headphone jack "LINE OUT" produces un-amplified ("Line Level") stereo audio output. Again, CH1 will produce the left channel sounds, and CH2 will produce the right channel sounds.

Sound via Daisy-Chained OSC-112s.

The sound channels of a daisy-chained OSC-112 are duplicates of the upstream OSC-112. The daisy-chained OSC-112 sounds cannot be controlled independently of the parent OSC-112. Therefore, if the daisy-chained OSC-112 is in a separate environment from the upstream OSC-112, be warned that the daisy-chained OSC-112 will play the same sounds at the same time on the same channels as the parent OSC-112.

CHAPTER 3 SOFTWARE CONFIGURATION

OmniSound Modes

The OmniSound system can produce sounds in one of four modes: **Tone**, **White Noise**, **Click**, or **Wave File**. The previous-generation Med Associates' sound generator (ANL-926) supported the Tone, Click and White Noise modes. The wave file mode is introduced with the OSC-112 sound generator.

Wave File Mode

The OSC-112 can play up to 255 sound files in the waveform audio file format ("WAV" or "wave") from an SD card inserted in OSC-112 left side panel. The OSC-112 can play stereo (dual-channel) or mono (single-channel) wave files in the linear pulse-code modulation (LPCM) format sampled at 44.1 kHz with 16 bits per sample. When playing a mono wave file, the channel 1 speaker will produce the sound. When playing a stereo wave file, the channel 1 speaker will produce the left channel, and the channel 2 speaker will produce the right channel sounds.

Wave File Manager

Wave File Manager (WaveFileMgr.exe) is a PC-based tool that allows the user to manage wave files that are to be used on the OmniSound card (within the OSC-112).

The OmniSound card supports playing of wave files that are contained on an SD card inserted into the OSC-112. Selected wave files are specified in MedState Notation protocols and MED Test by specifying a number (1-255) to indicate the desired wave file. The OmniSound card expects the file name of a wave file contained on the SD card to be in the format: *WFn.wav* (e.g. WF1.wav).

Keeping track of the content of each wave file when they are named 'WFn.wav' can be confusing unless one manually records the wave file number assignments. The Wave File Manager helps manage this issue.

The *Wave File Manager* tool allows the user to:

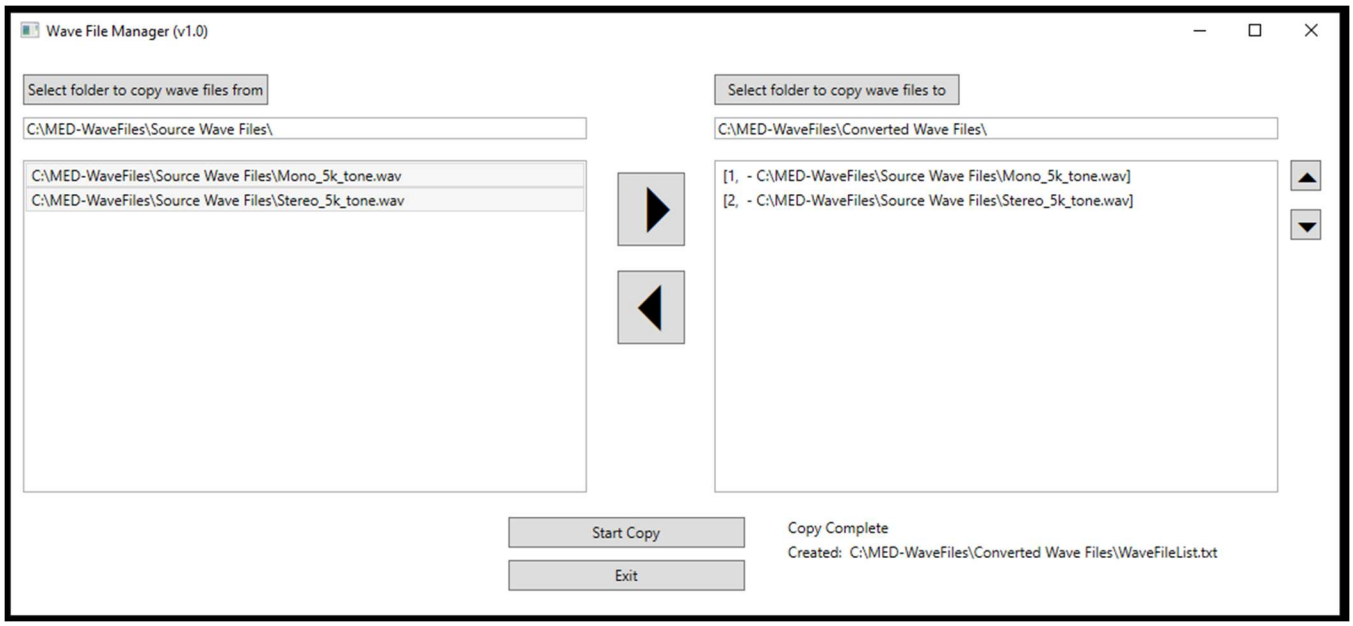
1. Select wave files (referred to as "source" files from a Source folder).
2. Assign a number (1-255) to the selected wave file.
3. Copy the file into the filename format required by the OmniSound card (referred to as "converted" files to a Converted folder).
4. Create a text file in the Converted folder that lists each selected wave file and its assigned wave file number.

The default folders on the PC running *Wave File Manager* for the "source" and "converted" files are as follows (but the user has the option to browse to different folders):

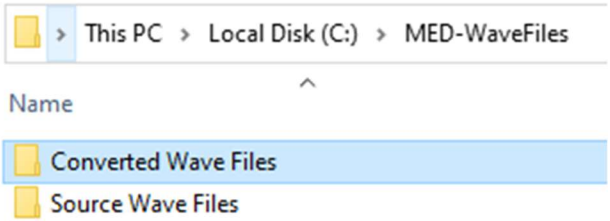
C:\Med-WaveFiles\Source Wave Files\
C:\Med-WaveFiles\Converted Wave Files\

Once the user has completed converting the list of desired wave files, the user can manually copy the files in the "converted" folder to the root directory of an SD card using Windows File Explorer.

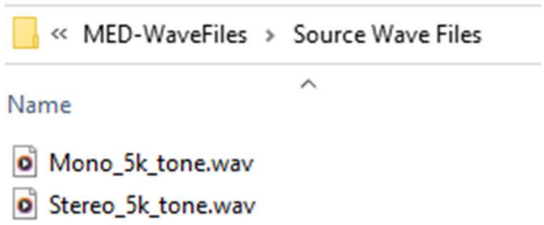
The following is a screenshot of the *Wave File Manager* user interface:



The following is a screenshot of Windows File Explorer displaying the contents in the *C:\MED-WaveFile* folder:

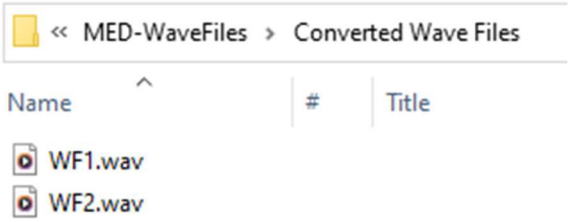


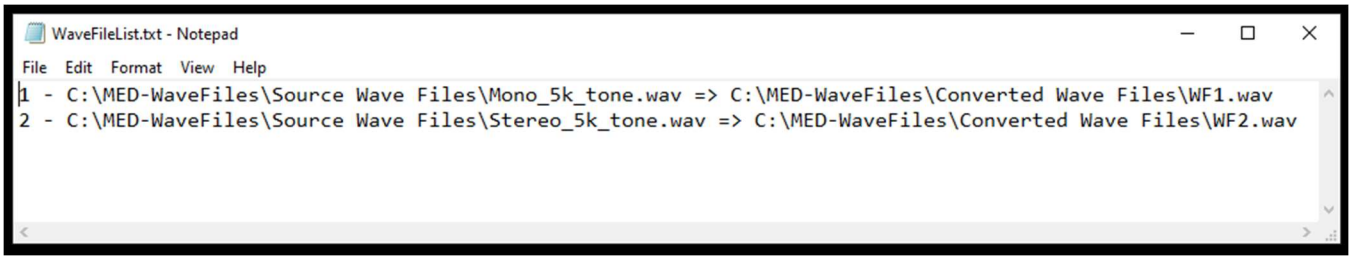
The following is a screenshot of Windows File Explorer displaying the contents in the *C:\MED-WaveFile\Source Wave Files* folder:



The following is a screenshot of Windows File Explorer displaying the contents in the *C:\MED-WaveFile\Converted Wave Files* folder:

The following is a screenshot of the content in the *C:\MED-WaveFiles\Converted Wave Files\WaveFileList.txt*:





Wave File Manager application can be downloaded from the Med Associates web site www.med-associates.com. After downloading the zip file, decompress the contents of the zip file to a temporary folder and run the self-extracting .exe to launch the installation program. The installer will put WaveFileMgr.exe in the desired folder and will create the following default folders to be used by the Wave File Manager,

```
C:\Med-WaveFiles\
```

```
C:\Med-WaveFiles\Source Wave Files\
```

```
C:\Med-WaveFiles\Converted Wave Files\
```

Testing OSC-112 Hardware using MED Test

The MED Test application is installed with the installation of Med-PC. MED Test provides several dialog boxes useful for testing the Input/Output functionality of the OSC-112, as well as the sound generation functionality. From the Windows Start button, navigate to the "M" section, open the Med Associates key, and choose the MED Test x64 application. MED Test x64 uses the 64-bit drivers to operate the connected hardware. Med-PC 6 is a 64-bit application, therefore use MED Test x64 to use the same drivers as Med-PC 6. The MED Test **x86** application uses the 32-bit version of the hardware drivers, for supporting older applications (Med-PC 4, for example).

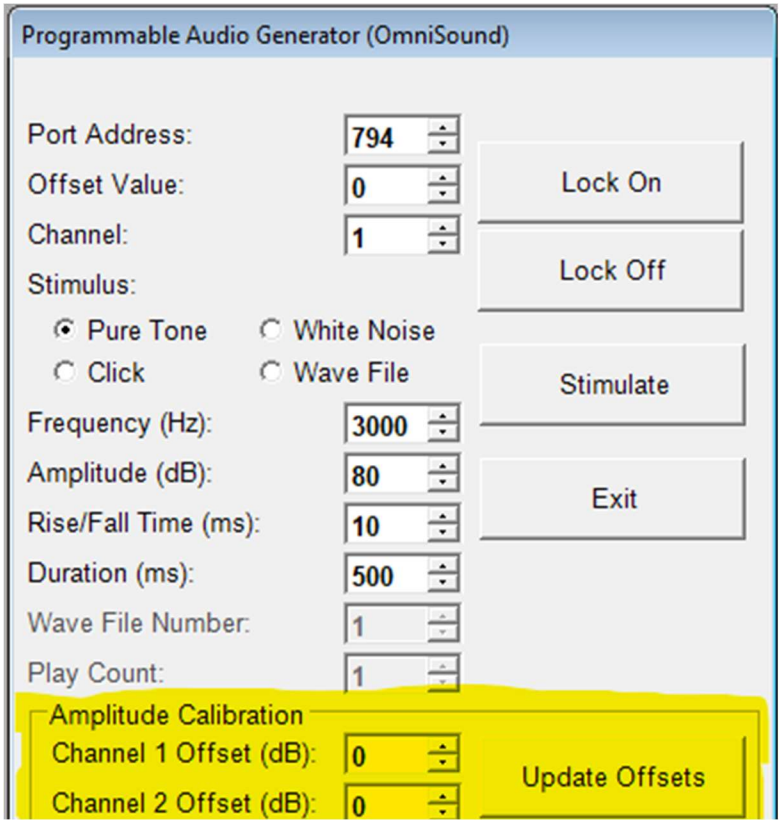
From the MED Test x64 main window, open the "Misc Modules" menu; there are two options here of interest. The first is the "OmniSound" option, and the second is the "OSC-112" option. The "OmniSound" option launches the "Programmable Audio Generator (OmniSound)" dialog. The "OSC-112" menu option will open two dialogs, the "OmniCard (IC-124)" dialog, and the "Programmable Audio Generator (OmniSound)" dialog. Use the "OmniCard (IC-124)" dialog to test the 28 V I/O functionality of the IC-124 OmniCard interface card. Use the "Programmable Audio Generator (OmniSound)" dialog to test the four (4) modes of audio generation on the OSC-112.

Refer to the MED Test User's Manual DOC-200 for details on using the "OmniCard (IC-124)" and the "Programmable Audio Generator (OmniSound)" dialogs. The manual is accessible from the MED Test x64 "Help" menu or the "Med Associates" section of the "Windows Start" menu.

Amplitude Calibration

Each of the two audio output channels may have a sound pressure level offset defined. The sound pressure level of each channel may be adjusted by -20 dB to +5 dB. The sound pressure level offset will be applied to the audio output stream to help achieve consistent sound pressure from each speaker in each chamber. The offset value is applicable to an OSC-112 channel and speaker combination. If a speaker/OSC-112 combination is changed, a new offset value should be defined. To set the offset, run the **MED Test x64** utility from the Windows Start menu or desktop shortcut, open the **Misc Modules** menu, and choose **OmniSound**.

FIGURE 3.1–OMNISOUND DIALOG



To determine the optimal offset value for each channel, play the same tone (frequency) at the same SPL (amplitude) in each speaker in each chamber. When playing the tone, put the ANL-930 Sound Pressure Level Measurement tool's microphone in the exact same location relative to each speaker. Adjust the Channel Offset using the up/down arrows and press the "Update Offsets" button to apply the changes to the hardware. Set the Offset value for each channel so the SPL displayed on the ANL-930 matches the Amplitude set on the MED Test OmniSound Dialog.

Begin setting the Offsets with the following recommended offset values for the two currently shipping speakers:

ENV-228AM: -8 dB

ENV-328BM/ENV-328BW: +1 dB

The offset values are stored in the OSC-112 when the OmniSound dialog's "Update Offsets" button is pressed.

CHAPTER 4 OMNISOUND COMMANDS AND OPERATING INSTRUCTIONS

The OSC-112 plays sounds as directed in a MedState Notation procedure. Use the TRANS utility to write MedState Notation procedures and compile them into programs executed in Med-PC®.

Software Files

The OSC-112 commands are provided by the OmniSound.DLL library files. The library files are distributed with the Med Drivers Installer SOF-MED-DRIVERS and are available on the Med Associates web site: www.med-associates.com. There are two copies of the OmniSound.DLL file, a 32-bit and 64-bit version are installed into the appropriate Windows System folders. The interface to the library file is defined in OmniSound.HED. The .HED file is distributed with Med-PC 6 and is installed into the Med-PC 6 installation directory. If while writing MedState Notation protocols this document is not available, the .HED file can be viewed to show the OSC-112 commands and parameters.

OmniSound Basic Commands Overview

The following are the basic OSC-112 commands for MedState Notation.

```
SetPort_OS  (MG, Port)
SetRack_OS  (MG, Rack)
StartTone_OS(MG, BOX, Channel, Amplitude, Duration, Rise/Fall Time, Frequency)
StartNoise_OS(MG, BOX, Channel, Amplitude, Duration, Rise/Fall Time)
StartClick_OS(MG, BOX, Channel, Amplitude, Duration, Click Frequency)
StartWave_OS(MG, BOX, Amplitude, Play Count, Wave File Number)
StopAudio_OS(MG, BOX, Channel)
```

These seven commands – two "Sets," four "Starts," and a "Stop" – are the basic commands for generating audio in MedState Notation with an OSC-112. There is an extended set of advanced commands covered later in Chapter 6 Legacy ANL-926 Commands. The extended command set is provided for converting MedState Notation protocols written for ANL-926 sound generators to using OSC-112 sound generators. The OSC-112's full functionality is obtainable with the seven basic commands. Any new protocol written for the OSC-112 should use these seven basic commands.

There are four "Start" commands, one for each of the OmniSound Modes (Tone, White Noise, Click, and Wave File): `StartTone_OS`, `StartNoise_OS`, `StartClick_OS`, and `StartWave_OS`. There is a single "Stop" command that ends any playing sound on the specified channel: `StopAudio_OS`. The two "Set" commands are optional and only required if the OSC-112's IC-124 OmniCard has been re-addressed from the default port 794 or if the IC-124 is in a rack other than Rack #1.

OmniSound commands are calls to Pascal programming language functions and **must be prefixed with a tilde (~) and suffixed with a semi-colon and tilde (;~)** within MedState Notation protocols. For example,

```
~SetRack_OS(MG, 2);~      /IC-124 is in Rack #2
```

calls the SetRack_OS command. The second tilde could be followed by another semi-colon to issue another Med-PC® command. For example,

```
~SetRack_OS(MG, 2);~; SET A = 3; /Set rack & assign 3 to var "A"
```

Each OSC-112 MedState Notation command requires a series of parameters separated by commas and enclosed by parentheses. Each command will take at least one of the following parameters. Several commands share the same parameters. A discussion of the OmniSound Command parameters follows.

OmniSound Command Parameters

PARAMETER	DESCRIPTION	VALID VALUES
MG	MedGlobal callback pointer. See next section.	MG
BOX	Test chamber identifier placeholder. See next section.	BOX
Channel	Speaker port to activate on OSC-112	1 or 2
Amplitude	Sound pressure level in decibels	20 through 100
Duration	Length of time for sound to play in milliseconds. Value of 0 plays tone until Stop command is issued.	0 through 65,535
Rise/Fall Time	Duration of ramp-up-to and ramp-down-from indicated amplitude in milliseconds. 10 ms is a common rise/fall time. The rise/fall time is added to the duration.	1 through 1000 in 1 ms increments
Frequency	Pure tone frequency in cycles per second (Hertz). To generate white noise, frequency parameter value is 0.	1000 through 70,000 in 1 Hz increments, or 0 for white noise
Click Frequency	Click frequency in cycles per second (Hertz).	1 through 100 in 1 Hz increments
Play Count	Number of times to play specified wave file. Specify 0 to repeat wave file until StopAudio_OS command is issued.	0 through 255
Wave File Number	Numerical component of wave file's filename (e.g., specify value "1" to select "WF1.WAV".)	1 through 255

Note that either the StartNoise_OS command or StartTone_OS command with 0 for the Frequency parameter may be used to generate white noise.

MG and BOX Parameters

The first parameter of all the OSC-112 basic commands is a MedGlobal callback pointer referenced by reserved keyword **MG**. This pointer is used by Med-PC to receive error messages from the running instance of the protocol.

Most OSC-112 commands take the reserved keyword **BOX** as the second parameter. BOX is replaced by Med-PC at runtime with the test chamber number the program is controlling. The BOX keyword and the MG keyword are both used to return information from the instance of the protocol running in each box to Med-PC.

Initialization Commands: SetPort_OS and SetRack_OS

There are two optional Initialization commands for the OSC-112: `SetPort_OS` and `SetRack_OS`.

The commands are **not required if** the OSC-112's IC-124 OmniCard is using the default auto-configuration settings in a single interface cabinet (rack). The default IC-124 settings are Port = 794 and Rack = 1.

If an IC-124 OmniCard has its port changed from auto/default 794, the MedState Notation protocol must call `SetPort_OS` command with the new port setting as the second command. Likewise, if the IC-124 is in a rack other than number 1, the `SetRack_OS` command must be called with the rack number (as defined in the Med-PC Hardware Configuration Utility) as the second parameter. The Initialization commands need to be called only once per program before playing sounds.

COMMAND	**SetPort_OS**
Usage	`SetPort_OS(MG, Port)`
Description	Tells the OSC-112 driver the port address of the IC-124 card connected to the OmniSound device.
Parameters	**MG**: MedGlobal callback pointer. **Port**: Physical port address of the IC-124 card. Valid values are even numbers 780 794. Default value is 794.
Example	`~SetPort_OS(MG,794);~ \IC-124 port address is 794`

COMMAND	**SetRack_OS**
Usage	`~SetRack_OS(MG, Rack);~`
Description	Tells the OSC-112 driver which rack (interface cabinet SG-6080, SG-7308, etc.) houses the IC-124 card connected to the OSC-112.
Parameters	**MG**: MedGlobal callback pointer. **Rack**: Value 1 through 4 identifying the interface cabinet housing the IC-124 controlling the OSC-112. Identifiers are defined in the Med Hardware Configuration Utility.
Example	`~SetRack_OS(MG,1);~ \IC-124 card is in rack #1`

Sound Commands: StartTone_OS, StartNoise_OS, StartClick_OS, StartWave_OS

There is a "Start" command available to play each of the four possible audio types: Tone, Noise, Click and Wave.

COMMAND	StartTone_OS
Usage	StartTone_OS(MG, BOX, Channel, Amplitude, Duration, Rise/Fall Time, Frequency)
Description	StartTone_OS plays a pure tone or white noise via the specified channel at the specified amplitude, for the specified duration.
Parameters	**MG**: MedGlobal callback pointer. **BOX**: Test chamber identifier. **Channel**: Valid values are 1 or 2. **Amplitude**: Sound pressure level output in decibels. Valid values are 20 through 100. **Duration**: Length of time for sound to play in milliseconds. Valid values are 0 through 65,535. Value 0 plays tone until stop command is issued. **Rise/Fall Time**: Duration of ramp-up-to and ramp-down-from indicated amplitude in milliseconds. Valid values are 1 through 1000 in 1 millisecond increments. 10 ms is a common rise/fall time. The rise/fall time is added to the duration. **Frequency**: Pure tone frequency in cycles per second (Hertz). Valid pure tone values are 1000 through 70,000 in 1 Hz increments. To generate white noise, frequency parameter value is 0.
Example	\StartTone_OS(MG, BOX, Channel, Amp, Dur, R/F Time, Freq) ~StartTone_OS(MG, BOX, 1, 70, 5000, 10, 1000);~ \play 1000 Hz tone at 70 dB for 5 seconds with a 10 msec \rise/fall time on channel 1

COMMAND	StartNoise_OS
Usage	StartNoise_OS(MG, BOX, Channel, Amplitude, Duration, Rise/Fall Time)
Description	StartNoise_OS plays a broad-spectrum white noise via the specified channel at the specified amplitude, for the specified duration.
Parameters	**MG**: MedGlobal callback pointer. **BOX**: Test chamber identifier. **Channel**: Valid values are 1 or 2. **Amplitude**: Sound pressure level output in decibels. Valid values are 20 through 100. **Duration**: Length of time for sound to play in milliseconds. Valid values are 0 through 65,535. Value 0 plays tone until StopAudio_OS command is issued. **Rise/Fall Time**: Duration of ramp-up-to and ramp-down-from indicated amplitude in milliseconds. Valid values are 1 through 1000 in 1 millisecond increments. 10 ms is a common rise/fall time. The rise/fall time is added to the duration.
Example	\StartNoise_OS(MG, BOX, Channel, Amp, Dur, R/F Time) ~StartNoise_OS(MG, BOX, 1, 85, 2500, 10);~ \play white noise on channel 1, at 85 dB, for 2.5 seconds, \with a 10 ms Rise/Fall time.

COMMAND	StartClick_OS
Usage	StartClick_OS(MG, BOX, Channel, Amplitude, Duration, Click Frequency)
Description	StartClick_OS plays a click sound via the specified channel at the specified amplitude and click frequency.
Parameters	**MG**: MedGlobal callback pointer. **BOX**: Test chamber identifier. **Channel**: Valid values are 1 or 2. **Amplitude**: Sound pressure level output in decibels. Valid values are 20 through 100. **Duration**: Length of time for sound to play in milliseconds. Valid values are 0 through 65,535. Value 0 plays click until StopAudio_OS command is issued. **Click Frequency**: Frequency in cycles per second (Hertz). Valid click frequency values are 1 through 100 in 1 Hz increments.
Example	\StartClick_OS(MG, BOX, Channel, Amp, Dur, Click Frequency) ~StartClick_OS(MG, BOX, 1, 80, 1500, 10);~ \10 Hz click on channel 1 at 80 dB for 1.5 seconds

COMMAND	StartWave_OS
Usage	StartWave_OS(MG, BOX, Amplitude, Play Count, Wave File Number)
Description	StartWave_OS plays the specified wave file at the specified amplitude for the indicated number of repetitions. A mono wave file will be output on channel 1. A stereo wave file's left channel will be output on channel 1 and right channel will be played on channel 2.
Parameters	**MG**: MedGlobal callback pointer. **BOX**: Test chamber identifier. **Amplitude**: Sound pressure level output in decibels. Valid values are 20 through 100. **Play Count:** Number of times to play wave file. Valid values are 0 to 255. Value 0 plays wave file until StopAudio_OS command is issued. **Wave File Number**: Valid values are 1 to 255.
Example	\StartWave_OS(MG, BOX, Amp, Play Count, Wave File#) ~StartWave_OS(MG, BOX, 70, 2, 3);~ \play Wave file #3 two times at 70dB

Stop Command: StopAudio_OS

There is a single command to stop any OSC-112 audio type: StopAudio_OS.

COMMAND	StopAudio_OS
Usage	StopAudio_OS(MG, BOX, Channel)
Description	Stops the sounds emanating from the indicated channel (1 or 2). This command will supersede any duration or repeat values specified on audio start.
Parameters	**MG**: MedGlobal callback pointer. **BOX**: Test chamber identifier. **Channel**: Valid values are 1 or 2.
Example	\StopAudio_OS(MG, BOX, Channel) ~StopAudio_OS(MG, BOX, 1);~ \End the audio playing on channel 1

Note on Variables vs Fixed Values as Command Parameters

MedState Notation uses the single letters A through Z as variables. A letter variable may hold a single real number or an array of real numbers. For example, the MedState Notation command

```
SET A = 1.302
```

assigns the value 1.302 to the variable "A", and the commands

```
DIM B = 2;   \Array   "B"   dimensioned   to   hold   3   elements   0,   1,   and   2
SET B(0) = 2.732; SET B(1) = 2.43; SET B(2) = 5; \Assign values
```

create an array named "B" of size 3 elements (indexes 0 through 2) and assigns values 2.732, 2.43, and 5 to the array elements. Note the array elements are accessed in MedState Notation using parentheses. However, inside a Pascal command (like the OSC-112 commands) **square brackets [ ] instead of parentheses ( )** are used to access array elements.

MedState Notation example of fixed values to achieve 3200 Hz tone for 1000 ms at 85 dB on channel 1 with a 10 ms rise/fall time:

```
~StartTone_OS(MG, BOX, 1, 85, 1000, 10, 3200);~
```

MedState Notation example of variables (array variable "A" and simple variable "F") to achieve the same results:

```
DIM A = 3;
...
SET A(0)=1; SET A(1)=85; SET A(2)=1000; SET A(3)=10; \note ()
SET F=3200;
~StartTone_OS(MG, BOX, A[0], A[1], A[2], A[3], F);~ \note []
```

When passing values to the OSC-112 commands via the function parameters, either "hard-coded" fixed values or variables may be passed. The advantage of using variables over fixed values becomes apparent when the parameters to the command will change throughout the program execution.

CHAPTER 5 OMNISOUND SAMPLE MEDSTATE NOTATION PROTOCOLS

The following MedState Notation (MSN) Protocol examples show the methods to control the sounds produced via the OmniSound OSC-112 connection panels. These MedState Notation examples assume the IC-124 OmniCards controlling the OSC-112 OmniSound + OmniCtrl connection panels are housed in a single rack, this single rack is rack #1, and the IC-124 OmniCards are set to "auto configure".

Example 1: Turn on pure tone frequency

```
S.S.1,
S1,
  \ Play 3 kHz tone on Channel 1, at 70 dB, for 2.5 seconds
  \ plus, 10 ms rise/fall times
  1": ~StartTone_OS(MG,BOX,1,70,2500,10,3000);~ ---> STOPDISCARD
```

The example above shows the StartTone_OS command playing a 3 kHz (3000 Hz) tone for 2.5 seconds (2500 ms), on channel 1's speaker at 70 dB with a 10 ms rise/fall time to mitigate speaker pop.

This is the shortest possible code snippet that can be compiled (in TRANS) and executed (in Med-PC) to generate a pure tone with the OSC-112.

Example 2: Play wave file

```
S.S.1,
S1,
  \Play Wave File #3, two times, at 80 dB
  1": ~StartWave_OS(MG, BOX, 80, 2, 3);~  ---> STOPDISCARD
```

The example above shows the StartWave_OS command playing WF3.wav (3), twice (2), at 80 dB.

"OmniSound Freq.MPC" and "OmniSound Example.MPC"

Two MedState Notation (MSN) Protocol examples are installed with the installation of Med-PC 6: "**OmniSound Freq.MPC**" and "**OmniSound Example.MPC.**"

"OmniSound Freq.MPC" is based on the "A926FREQ.MPC" MSN Protocol for use with ANL-926 audio generators. "A926FREQ.MPC" example is distributed with Med-PC 4, 5, and 6. Both programs utilize K-pulses[1] 1 through 5 for changing tone frequency and audio modes. The "OmniSound Freq.MPC" protocol uses additional K-pulses 6 and 7 for exemplifying the wave file mode. As the ANL-926 only had one audio output channel, and "OmniSound Freq.MPC" is almost a direct conversion of "A926FREQ.MPC", "OmniSound Freq.MPC" only generates audio on a single channel (channel 1).

The **"OmniSound Example.MPC"** protocol uses the seven basic OSC-112 commands to illustrate the four audio modes of the OSC-112, as well as some advanced and legacy commands for adjusting the tone frequency, amplitude, duration, and rise/fall time. "OmniSound Example.MPC" generates audio on both output channels 1 and 2. Similar to the "A926FREQ.MPC" and "OmniSound Freq.MPC" protocols, "OmniSound Example.MPC" uses K-pulses 1 through 19 for controlling state transitions. The states accessed via K-pulses 1 through 7 use the basic OSC-112 commands, and K-pulses 8 through 18 access states that use the legacy commands. K-19 ends the demo.

Both "OmniSound Freq.MPC" and "OmniSound Example.MPC" feature extensive comments on usage and command parameters. Open these files using the TRANS utility or any text editor. The files will be installed to the **MPC** subfolder of the MED-PC installation directory, which is **C:\MED-PC** by default. To run these protocols in Med-PC, first translate and compile the MSN code in the TRANS utility.

[1] K-pulses are discussed in the TRANS Help file (TRANS utility, Help menu, Help For Trans option) in the "Keyboard Signals (#K)" section, and in the Med-PC Programmer's Manuals DOC-003 or DOC-301.

CHAPTER 6 LEGACY ANL-926 COMMANDS

The OSC-112 OmniSound + OmniCtrl connection panel features a programmable audio generator. The previous-generation Med Associates programmable audio generator was model number ANL-926. The OSC-112 may be controlled via the OSC-112 OmniSound Commands discussed above (StartTone_OS, StartNoise_OS, StartClick_OS, and StopAudio_OS), or via "legacy" commands based on their analogous ANL-926 commands. The legacy commands allow a MedState Notation protocol written for ANL-926 to be easily converted for use with OSC-112 OmniSound + OmniCtrl connection panels.

The legacy commands differ from their analogous ANL-926 commands in three ways:

1. The suffix **_OS** is added to the command.
2. Channel number is added to the parameter list.
3. "RP" commands which specify the Rack and Port have an additional "O" for Offset and address offset parameter added.

Med-PC programs issue commands to the OSC-112 programmable audio generator using a combination of **command type** prefixes combined with **sound attribute keywords**. The OmniSound commands have an underscore OS (_OS) suffix to differentiate them from their analogous ANL-926 commands. See section Converting ANL-926 Protocols to Use OSC-112 for converting ANL-926 programs to use OSC-112 hardware.

Command Types:

`Set` Updates the sound attribute keyword value but does not produce any sounds.

`On` Updates the sound attribute keyword value **and** produces a sound.

Sound Attribute Keywords:

`Freq` Sets frequency value (`SetFreq_OS`) or sets frequency value and starts a pure tone or white noise (`OnFreq_OS`). Values must be positive integers (whole numbers) in the range of 1,000 Hz to 70,000 Hz (70 kHz) for Pure Tones. Specify zero (0) Hz to produce white noise. The currently set amplitude, duration, and rise/fall time values are unaffected.

`Amp` Sets the amplitude (`SetAmp_OS`) or sets the amplitude and starts the last pure tone sound, white noise, or click (`OnAmp_OS`). Amplitude value must be in range of 20 dB to 100 dB. The currently set frequency, duration, and rise/fall time values are unaffected.

`Dur` Sets the duration (`SetDur_OS`) or sets the duration and starts the last pure tone sound, white noise, or click (`OnDur_OS`). Duration value must be in the range of 1 ms to 65,535 ms. The currently set frequency, amplitude, and rise/fall time values are unaffected.

`RF` Sets the rise/fall time (`SetRF_OS`) or sets the rise/fall time and starts the last pure tone sound or white noise (`OnRF_OS`). Rise/fall time value must be int the range of 1 ms to 1000 ms. The rise/fall time is added to the beginning and end of the currently set duration. The currently set frequency, amplitude, and duration values are unaffected.

`Click` Sets the click frequency (`SetClickFreq_OS`) or sets the click frequency and starts the clicker (`PulseClick_OS`). Click frequency value must be in the range of 1 Hz to 100 Hz. The currently set tone frequency, amplitude, duration, and rise/fall time values are unaffected.

Legacy Command Default Values

If an "On" command is issued before any "Set" commands are issued, the following default values will be used by the OSC-112:

ATTRIBUTE	DEFAULT VALUE
Frequency	1 kHz (1000 Hz)
Amplitude	64 dB
Duration	1 second (1000 ms)
Rise/Fall Time	10 ms (0.010 s)
Click Frequency	10 Hz (10 clicks per second)

Legacy "RPO" Commands

The ANL-926 command set included several "RP" commands. These RP commands accepted two parameters to specify the Rack and Port. In a MedState Protocol for ANL-926s, using the RP commands allowed the SetRack and SetPort commands to be omitted, and the rack and port could be specified in each sound generating command. This was especially necessary if the ANL-926 cards of a system were in separate interface card cabinets (racks).

The OSC-112 supports these commands with the legacy commands that end with **RPO_OS**. In addition to the rack and port parameters, the offset value needs to be specified for OSC-112 "RPO" commands. The offset value may be calculated from the BOX value.

The offset for each OSC-112 is equivalent to (BOX-1)*4.

ANL-926 "RP" TO OSC-112 "RPO" CONVERSION EXAMPLE:

Here is the format and parameter list for the ANL-926 SetFreqRP command:

```
~SetFreqRP(MG, Rack, Port, Box, Frequency);~
```

An example of the ANL-926 command usage:

```
~SetFreqRP(MG, 2, 794, BOX, 2000);~
```

Format and parameter list for OSC-112 command SetFreqRPO_OS:

```
~SetFreqRPO_OS(MG, Rack, Port, Offset, BOX, Freq, Channel);~
```

ANL-926 command example converted to use an OSC-112:

```
~SetFreqRPO_OS(MG, 2, 794, ((BOX-1)*4), 2, 2000, 1);~
```

Legacy Frequency Commands

The OSC-112 can produce tones at frequencies from 10 Hz to 70 kHz in 1 Hz increments. Actual frequencies produced will be determined by the speaker's frequency response curve and limitations. Set the frequency value to 0 Hz to produce a wide band white noise.

SetFreq_OS(MG, BOX, Frequency, Channel): Sets the frequency to be produced, but does not issue a sound.

```
Ex: ~SetFreq_OS(MG, BOX, 2500,1);~
```

SetFreqRPO_OS(MG, Rack, Port, Offset, BOX, Frequency, Channel): Sets the desired frequency as well as the rack, port, and offset address to use but does not produce a sound.

```
Ex: ~SetFreqRPO_OS(MG, 2, 794, (BOX-1)*4, BOX, 2500, 1);~
```

OnFreq_OS(MG, BOX, Frequency, Channel): Sets the desired frequency and produces a sound. The sound will continue until the set duration is reached, or another command is issued.

```
Ex: ~OnFreq_OS(MG, BOX, 2500, 1);~
```

OnFreqRPO_OS(MG, Rack, Port, Offset, BOX, Frequency, Channel): Sets the desired frequency, the rack, port, and offset to use and produces a sound until the set duration is reached, or another command is issued.

```
Ex: ~OnFreqRPO_OS(MG, 2, 794, (BOX-1)*4, BOX, 2500, 1);~
```

Legacy Amplitude Commands

The amplitude is entered as a whole number in decibels (dB). The available range is 20 dB to 100 dB in 1 dB increments.

Amplitude, or sound pressure level (SPL), can vary with the frequency being produced, the distance from the speaker the measurement is taken, and the speaker's frequency response curve. If a precise output value is required, the system should be calibrated using a SPL meter and MED Test.

SetAmp_OS(MG, BOX, Amplitude, Channel):

Sets the amplitude of the sound but does not produce a sound.

```
Ex: ~SetAmp_OS(MG, Box, 85, 1);~
```

SetAmpRPO_OS(MG, Rack, Port, Offset, BOX, Amplitude, Channel): Sets the amplitude and the rack, port, and offset address to be used but does not produce a sound.

```
Ex: ~SetAmpRPO_OS(MG, 2, 794, (BOX-1)*4, BOX, 85, 1);~
```

OnAmp_OS(MG, BOX, Amplitude, Channel): Sets the amplitude to the desired level and produces a sound for the set duration or until another command is issued.

```
Ex: ~OnAmp_OS(MG, BOX, 85, 1);~
```

OnAmpRPO_OS(MG, Rack, Port, Offset, BOX, Amplitude, Channel): Sets the amplitude, rack, port, and offset to use and produces a sound until the set duration is reached or another command is issued.

```
Ex: ~OnAmpRPO_OS(MG, 2, 794, (BOX-1)*4, BOX, 85, 1);~
```

Legacy Rise / Fall Time Commands

The rise/fall time is a whole number entered in milliseconds. This value sets the duration of the gradual turn on and turn off period used to suppress the "pop" that may result when a transition from silence to a significant volume (or vice-versa) occurs. The rise/fall time is added to the beginning and end of a sound's duration. A sound with duration of 1 second and 10 ms rise/fall time will take a total of 1.020 seconds to play. The available range is 1 to 1000 milliseconds in 1 millisecond increments.

SetRF_OS(MG, BOX, Rise/Fall Time, Channel): Sets the rise/fall time but does not produce a sound.

```
Ex: ~SetRF_OS(MG, BOX, 5, 1);~
```

SetRFRPO_OS(MG, Rack, Port, Offset, BOX, Rise/Fall Time, Channel): Sets the rise/fall time as well as the rack, port, and offset values, but does not produce a sound.

```
Ex: ~SetRFRPO_OS(MG, 2, 794, (BOX-1)*4, BOX, 5, 1);~
```

OnRF_OS(MG, BOX, Rise/Fall Time, Channel): Sets the rise/fall time and produces a sound until the set duration is reached or another command is issued.

```
Ex: ~OnRF_OS(MG, BOX, 5, 1);~
```

OnRFRPO_OS(MG, Rack, Port, Offset, BOX, Rise/Fall Time, Channel): Sets the rise/fall time, the rack or port values and produces a sound until the set duration is reached or another command is issued.

```
Ex: ~OnRFRPO_OS(MG, 2, 794, (BOX-1)*4, BOX, 5, 1);~
```

Legacy Duration Commands

The sound duration is adjustable from 1 to 65,535 milliseconds. The duration set with one of the SetDur commands is the duration that is used when OnFreq_OS, OnAmp_OS and OnRF_OS commands are issued.

SetDur_OS(MG, BOX, Duration, Channel): Sets the duration of the sound when an OnFreq, OnAmp or OnRF command is issued, but does not produce a sound.

```
Ex: ~SetDur_OS(MG, BOX, 2000, 1);~
```

SetDurRPO_OS(MG, Rack, Port, Offset, BOX, Duration, Channel): Sets the sound duration as well as the rack, port, and offset to be used, but does not produce a sound.

```
Ex: ~SetDurRPO_OS(MG, 2, 794, (BOX-1)*4, BOX, 2000, 1);~
```

OnDur_OS(MG, BOX, Duration, Channel): Sets the duration of the sound and produces a sound for that duration.

```
Ex: ~OnDur_OS(MG, BOX, 2000);~
```

OnDurRPO_OS(MG, Rack, Port, Offset, BOX, Duration, Channel): Sets the sound duration as well as the rack, port, and offset to be used, and produces a sound for that duration.

```
Ex: ~OnDurRPO_OS(MG, 2, 794, (BOX-1)*4, BOX, 2000, 1);~
```

Legacy Tone On/Tone Off Commands

A ToneOn command locks the audio signal on until an OnFreq_OS, OnAmp_OS, OnRF_OS or OnDur_OS ("On" command) is issued or the ToneOff command is issued. This command ignores any previously set duration.

ToneOn_OS(MG, BOX, Channel): Locks the audio signal on using the values for frequency, amplitude, and rise/fall set previously.

```
Ex: ~ToneOn_OS(MG, BOX, 1);~
```

ToneOnRPO_OS(MG, Rack, Port, Offset, BOX, Channel): Locks the audio signal on using the specified port in the specified rack until an "On" command or ToneOff command issued.

```
Ex: ~ToneOnRPO_OS(MG, 2, 794, (BOX-1)*4, BOX, 1);~
```

ToneOff_OS(MG, BOX, Channel): Turns off an audio signal that has been locked on or interrupts a timed audio signal.

```
EX: ~ToneOff_OS(MG, BOX, 1);~
```

ToneOffRPO_OS(MG, Rack, Port, Offset, BOX, Channel): Turns off an audio signal that has been locked on or interrupts a timed audio signal using the specified port and rack.

```
Ex: ~ToneOffRP_OS(MG, 2, 794, (BOX-1)*4, BOX, 1);~
```

Legacy Click Commands

The click frequency is the number of clicks per second. The possible range is 1 to 100 clicks per second.

SetClickFreq_OS(MG, BOX, Click Frequency, Channel): Sets the click frequency but does not issue a sound.

```
Ex: ~SetClickFreq_OS(MG, BOX, 10, 1);~
```

SetClickFreqRPO_OS(MG, Rack, Port, Offset, BOX, Click Frequency, Channel): Sets the click frequency and the rack and port to be used, but does not produce a sound.

```
Ex: ~SetClickFreqRPO_OS(MG, 2, 794, (BOX-1)*4, BOX, 10), 1;~
```

PulseClick_OS(MG, BOX, Click Frequency, Channel): Sets the click frequency to be used and produces a sound for the set duration.

```
Ex: ~PulseClick_OS(MG, BOX, 15, 1);~
```

PulseClickRPO_OS(MG, Rack, Port, Offset, BOX, Click Frequency, Channel): Sets the click frequency and the rack and port to be used and produces a sound for the set duration.

```
Ex: ~PulseClickRPO_OS(MG, 2, 794, (BOX-1)*4, BOX, 15, 1);~
```

ClickOn commands will lock on the click signal until any other command is issued. A SetClickFreq_OS or SetClickFreqRPO_OS command must be issued before a ClickOn_OS command can be used.

```
Ex: ~ClickOn_OS(MG, BOX, 1);~ or ~ClickOnRPO_OS(MG, 2, 794, (BOX-1)*4, BOX, 1);~
```

ClickOn_OS(MG, BOX, Channel)

ClickOnRPO_OS(MG, Rack, Port, Offset, BOX, Channel)

ClickOff commands turn off a click signal that was locked on and can be used to interrupt a timed pulse of clicks.

```
Ex: ~ClickOff_OS(MG, BOX, 1);~ or ~ClickOffRPO_OS(MG, 2, 794, (BOX-1)*4, BOX, 1);~
```

ClickOff_OS(MG, BOX, Channel)

ClickOffRPO_OS(MG, Rack, Port, (BOX-1)*4, Box, Channel)

CHAPTER 7 CONVERTING ANL-926 PROTOCOLS TO USE OSC-112

The OSC-112 replaces the previous generation programmable audio generator model: ANL-926. MedState Notation code protocols written for use with ANL-926 audio generators must be altered to use the OmniSound commands before using with OSC-112 audio generators.

Each ANL-926 command has an analogous OmniSound command. The OmniSound commands differ in three ways from the equivalent ANL-926 command:

1. Addition of **_os** to the end of the command (for **O**mni**S**ound)
2. Addition of channel number to parameter list (**1** or **2**)
3. "RP" commands which specify the Rack and Port have an additional "O" for Offset and address offset parameter added.

The ANL-926 has a single audio output channel and the OSC-112 has two audio output channels. Each OSC-112 audio channel has a ¼" stereo output connector on the left side of the OSC-112 case. The top connector is channel 1 and the bottom connector is channel 2. The desired output channel must be specified in the function call parameter list when generating sounds with the OSC-112.

Example:

ANL-926 Command	OSC-112 Command
`~OnFreq(MG, BOX, 2000);~`	`~OnFreq_OS(MG, BOX, 2000, 1);~`
`\Turn on 2 kHz tone`	`\Turn on 2 kHz tone on Channel 1`

When converting ANL926 compatible code for use with OSC-112, note the ANL-926 required an initialization command (`InitANL926` or `InitANL926RP`). There is no "init" command for the OSC-112. However, there are two initialization commands still available for the OSC-112: `SetPort_OS` and `SetRack_OS`.

Once any of the initializing commands are called, they do not need to be used again in the program.

To convert an ANL-926 protocol to use with OSC-112:

1. **Open** the protocol in **TRANS** utility.
2. **Add** _OS suffix to each ANL-926 command.
3. **Add** 0_OS suffix to each ANL-926 "RP" command.
4. **Add** the desired output speaker **channel number** parameter to each ANL-926 command.
5. **Add** the desired address **offset** and **channel number** to each ANI-926 "RP" command.
6. **Delete** the **InitANL926** or **InitANL926RP** command.
7. **Save** the protocol with a new name to differentiate from the ANL-926 version.
8. **Translate** the program to create a Med-PC program.

The ANL-926 allowed sound amplitude specified in 0.5 dB increments, e.g., 70.5 dB. The OSC-112 allows sound amplitude to be specified in whole-number increments only, e.g., 70 dB or 71 dB. When converting ANL-926 code for use with OSC-112, the compiler will ignore fractional sound amplitudes and round down to the next whole number (e.g., 70.5 dB becomes 70 dB.) To make the code perfectly clear for others to understand, we recommend removing the fractional portion of any sound pressure amplitude settings, e.g., change `70.5` to `70` or `71`.

Conversion Key

To convert MedState Notation for an ANL-926 to use an OSC-112, convert each command according to the following guide.

ANL-926 Command	OSC-112 Command
InitANL926	<Not Applicable>
SetPort(MG, Port)	SetPort_**OS**(MG, Port)
SetRack(MG, Rack)	SetRack_**OS**(MG, Rack)
SetFreq(MG, Box, Frequency)	SetFreq_**OS**(MG, Box, Frequency**, Channel**)
SetAmp(MG, Box, Amplitude)	SetAmp_**OS**(MG, Box, Amplitude**, Channel**)
SetRF(MG, Box, Rise/Fall)	SetRF_**OS**(MG, Box, Rise/Fall**, Channel**)
SetDur(MG, Box, Duration)	SetDur_**OS**(MG, Box, Duration**, Channel**)
SetClickFreq(MG, Box, Frequency)	SetClickFreq_**OS**(MG, Box, Frequency**, Channel**)
OnFreq(MG, Box, Frequency)	OnFreq_**OS**(MG, Box, Frequency**, Channel**)
OnAmp(MG, Box, Amplitude)	OnAmp_**OS**(MG, Box, Amplitude**, Channel**)
OnRF(MG, Box, Rise/Fall)	OnRF_**OS**(MG, Box, Rise/Fall**, Channel**)
OnDur(MG, Box, Duration)	OnDur_**OS**(MG, Box, Duration**, Channel**)
PulseClick(MG, Box, Frequency)	PulseClick_**OS**(MG, Box, Frequency**, Channel**)
ToneOn(MG, Box)	ToneOn_**OS**(MG, Box**, Channel**)
ToneOff(MG, Box)	ToneOff_**OS**(MG, Box**, Channel**)
ClickOn(MG, Box)	ClickOn_**OS**(MG, Box**, Channel**)
ClickOff(MG, Box)	ClickOff_**OS**(MG, Box**, Channel**)
InitANL926RP(MG, Rack, Port)	<Not Applicable>
SetFreqRP(MG, Rack, Port, Box, Frequency)	SetFreqRP**O**_**OS**(MG, Rack, Port, **Offset,** Box, Frequency, **Channel**)

ANL-926 Command	OSC-112 Command
SetAmpRP(MG, Rack, Port, Box, Amplitude)	SetAmpRP**O_OS** (MG, Rack, Port, **Offset,** Box, Amplitude**, Channel**)
SetRFRP(MG, Rack, Port, Box, Rise/Fall)	SetRFRP**O_OS** (MG, Rack, Port, **Offset,** Box, Rise/Fall**, Channel**)
SetDurRP(MG, Rack, Port, Box, Duration)	SetDurRP**O_OS** (MG, Rack, Port, **Offset,** Box, Duration**, Channel**)
OnFreqRP(MG, Rack, Port, Box, Frequency)	OnFreqRP**O_OS** (MG, Rack, Port, **Offset,** Box, Frequency**, Channel**)
OnAmpRP(MG, Rack, Port, Box, Amplitude)	OnAmpRP**O_OS** (MG, Rack, Port, **Offset,** Box, Amplitude**, Channel**)
OnRFRP(MG, Rack, Port, Box, Rise/Fall)	OnRFRP**O_OS** (MG, Rack, Port, **Offset,** Box, Rise/Fall**, Channel**)
OnDurRP(MG, Rack, Port, Box, Duration)	OnDurRP**O_OS** (MG, Rack, Port, **Offset,** Box, Duration**, Channel**)
PulseClickRP(MG, Rack, Port, Box, Frequency)	PulseClickRP**O_OS** (MG, Rack, Port, **Offset,** Box, Frequency**, Channel**)
ToneOnRP(MG, Rack, Port, Box)	ToneOnRP**O_OS** (MG, Rack, Port, **Offset,** Box**, Channel**)
ToneOffRP(MG, Rack, Port, Box)	ToneOffRP**O_OS** (MG, Rack, Port, **Offset,** Box**, Channel**)
ClickOnRP(MG, Rack, Port, Box)	ClickOnRP**O_OS** (MG, Rack, Port, **Offset,** Box**, Channel**)
ClickOffRP(MG, Rack, Port, Box)	ClickOffRP**O_OS**(MG, Rack, Port, **Offset,** Box**, Channel**)

CONVERSION EXAMPLE

For example, below is a portion of the ANL926FREQ.MPC file distributed with Med-PC 4, 5, and 6:

```
S.S.1,
S1,
  0.01 ": SET A(^Freq) = 1000, A(^Amp)   = 100, A(^RF) = 10;
         SET A(^Dur)  = 1000, A(^Click) = 10 ---> S2

S2,      \ First Statement: Wait for START signal and then issue
         \ starting stimulus
         \
         \ Second Statement: Update screen display with default values
         \ for Control Variables. This will show any changes made via
         \ the "Configure | Change Variables" Window prior to START.
  #START: ~SetRack(MG, 1);~;              \ ANL-926 cards are in Rack 1
          ~InitANL926;~;                  \ Reset ANL-926
          ~SetFreq(MG, BOX, A[0]);~;      \ Initialize Frequency
          ~SetAmp (MG, BOX, A[1]);~;      \ Initialize Amplitude
          ~SetRF  (MG, BOX, A[2]);~;      \ Initialize Rise\Fall Time
          ~SetDur (MG, BOX, A[3]);~;      \ Initialize Duration
          ~OnFreq (MG, BOX, A[0]);~;      \ Issue Starting Stimulus
          ---> S3
  1     ":      SHOW      1,Frequency,A(^Freq),      2,Amplitude,A(^Amp),
3,Rise/Fall,A(^RF);
      SHOW 4,Duration,A(^Dur),   5,Pulse Click,A(^Click) ---> S2
```

And here is the code converted for use with OSC-112:

Note `_OS` added to each command, channel parameter (", `1`") added to each command, and the removal of the `InitANL926` command.

```
S.S.1,
S1,
  0.01 ": SET A(^Freq) = 1000, A(^Amp)   = 100, A(^RF) = 10;
         SET A(^Dur)  = 1000, A(^Click) = 10 ---> S2

S2,      \ First Statement: Wait for START signal and then issue
         \ starting stimulus
         \
         \ Second Statement: Update screen display with default values
         \ for Control Variables. This will show any changes made via
         \ the "Configure | Change Variables" Window prior to START.
  #START: ~SetRack_OS(MG, 1);~;           \ OSC-112 cards are in Rack 1
          ~SetFreq_OS(MG, BOX, A[0], 1);~;  \ Initialize Frequency
          ~SetAmp_OS (MG, BOX, A[1], 1);~;  \ Initialize Amplitude
          ~SetRF_OS  (MG, BOX, A[2], 1);~;  \ Initialize Rise\Fall Time
          ~SetDur_OS (MG, BOX, A[3], 1);~;  \ Initialize Duration
          ~OnFreq_OS (MG, BOX, A[0], 1);~;  \ Issue Starting Stimulus
          ---> S3
  1     ":      SHOW      1,Frequency,A(^Freq),      2,Amplitude,A(^Amp),
3,Rise/Fall,A(^RF);
      SHOW 4,Duration,A(^Dur),   5,Pulse Click,A(^Click) ---> S2
```

ANL-926 Protocol Command	OSC-112 Protocol Command	Note
`~ToneOn(MG,BOX);~`	`~ToneOn_OS(MG,BOX,1);~`	Turn on the tone configured with a previous "Set" command on channel 1.
`~SetClickFreq(MG,BOX,20);~`	`~SetClickFreq_OS(MG,BOX,20, 2);~`	Set the click frequency to 20 Hz for channel 2.
`~SetRF(MG,BOX,10);~`	`~SetRF_OS(MG,BOX,10,1);~`	Set the rise/fall time to 10 msec on channel 1.

Note: in MedState Notation, capitalization is ignored. `ToneOn_OS` is equivalent to `toneon_os`, `tOnEoN_oS`, etc.

ANL-926 and OSC-112 Together

If the Med-PC system consists of both ANL-926 and OSC-112 programmable audio generators, there are two possible MedState Notation Protocol scenarios. Either maintain separate programs for the specific hardware and load the appropriate program per box based on that box's hardware, or in a single program use IF statements to call separate commands based on the BOX number (and therefore hardware type).

SEPARATE PROGRAMS

If the existing ANL-926 program is called "Habituation.MSN" and the converted-for-OSC-112 program is called "Habituation OmniSound.MSN", be sure to load "Habituation.MSN" into the boxes with ANL-926 cards, and "Habituation OmniSound.MSN" into the boxes with OSC-112 audio generators.

SINGLE PROGRAM

A single program would utilize the reserved keyword BOX to determine the hardware, and call the correct command based on that hardware. This single program could be loaded into any box. If, for example, we have a system with two (2) boxes numbered 1 and 2 with ANL-926 audio generators, and two (2) boxes numbered 3 and 4 with OSC-112 audio generators, the code would look like:

```
\Boxes 1&2 have ANL926, boxes 3&4 have OSC-112
S1,
     1": IF (BOX <= 2) [@ANL926, @OSC112]
          @ANL926: ~OnFreq(MG, BOX, 2000);~ ---> S2
          @OSC112: ~OnFreq_OS(MG, BOX, 2000, 1);~ ---> S2
```

ANL-926 vs OSC-112 Capabilities

When converting code for ANL-926 for use with OSC-112, keep in mind that the two models have their own set of limitations.

Field	ANL-926	OSC-112	Note
Frequency	10 Hz to 35,000 Hz	1,000 Hz to 70,000 Hz	Minimum 10 -> 1,000
Amplitude	20 dB SPL to 100 dB SPL in 0.5 dB increments	20 dB SPL to 100 dB SPL in 1 dB increments	Whole dB increments
Port	790	794	Default port

ANL-926 Command	OSC-112 Command
InitANL926	<Not Applicable>
SetPort(MG, Port)	SetPort_**OS**(MG, Port)
SetRack(MG, Rack)	SetRack_**OS**(MG, Rack)
SetFreq(MG, Box, Frequency)	SetFreq_**OS**(MG, Box, Frequency**, Channel**)
SetAmp(MG, Box, Amplitude)	SetAmp_**OS**(MG, Box, Amplitude**, Channel**)
SetRF(MG, Box, Rise/Fall)	SetRF_**OS**(MG, Box, Rise/Fall**, Channel**)
SetDur(MG, Box, Duration)	SetDur_**OS**(MG, Box, Duration**, Channel**)
SetClickFreq(MG, Box, Frequency)	SetClickFreq_**OS**(MG, Box, Frequency**, Channel**)
OnFreq(MG, Box, Frequency)	OnFreq_**OS**(MG, Box, Frequency**, Channel**)
OnAmp(MG, Box, Amplitude)	OnAmp_**OS**(MG, Box, Amplitude**, Channel**)
OnRF(MG, Box, Rise/Fall)	OnRF_**OS**(MG, Box, Rise/Fall**, Channel**)
OnDur(MG, Box, Duration)	OnDur_**OS**(MG, Box, Duration**, Channel**)
PulseClick(MG, Box, Frequency)	PulseClick_**OS**(MG, Box, Frequency**, Channel**)
ToneOn(MG, Box)	ToneOn_**OS**(MG, Box**, Channel**)
ToneOff(MG, Box)	ToneOff_**OS**(MG, Box**, Channel**)
ClickOn(MG, Box)	ClickOn_**OS**(MG, Box**, Channel**)
ClickOff(MG, Box)	ClickOff_**OS**(MG, Box**, Channel**)
InitANL926RP(MG, Rack, Port)	<Not Applicable>
SetFreqRP(MG, Rack, Port, Box, Frequency)	SetFreqRP**O**_**OS**(MG, Rack, Port, **Offset,** Box, Frequency, **Channel**)
SetAmpRP(MG, Rack, Port, Box, Amplitude)	SetAmpRP**O**_**OS** (MG, Rack, Port, **Offset,** Box, Amplitude, **Channel**)
SetRFRP(MG, Rack, Port, Box, Rise/Fall)	SetRFRP**O**_**OS** (MG, Rack, Port, **Offset,** Box, Rise/Fall, **Channel**)

ANL-926 Command	OSC-112 Command
OnFreqRP(MG, Rack, Port, Box, Frequency)	OnFreqRP**O_OS** (MG, Rack, Port, **Offset,** Box, Frequency**, Channel**)
SetDurRP(MG, Rack, Port, Box, Duration)	SetDurRP**O_OS** (MG, Rack, Port, **Offset,** Box, Duration**, Channel**)
SetClickFreqRP(MG, Rack, Port, Box, Frequency)	SetClickFreqRP**O_OS** (MG, Rack, Port, **Offset,** Box, Frequency**, Channel**)
OnAmpRP(MG, Rack, Port, Box, Amplitude)	OnAmpRP**O_OS** (MG, Rack, Port, **Offset,** Box, Amplitude**, Channel**)
OnRFRP(MG, Rack, Port, Box, Rise/Fall)	OnRFRP**O_OS** (MG, Rack, Port, **Offset,** Box, Rise/Fall**, Channel**)
OnDurRP(MG, Rack, Port, Box, Duration)	OnDurRP**O_OS** (MG, Rack, Port, **Offset,** Box, Duration**, Channel**)
PulseClickRP(MG, Rack, Port, Box, Frequency)	PulseClickRP**O_OS** (MG, Rack, Port, **Offset,** Box, Frequency**, Channel**)
ToneOnRP(MG, Rack, Port, Box)	ToneOnRP**O_OS** (MG, Rack, Port, **Offset,** Box**, Channel**)
ToneOffRP(MG, Rack, Port, Box)	ToneOffRP**O_OS** (MG, Rack, Port, **Offset,** Box**, Channel**)
ClickOnRP(MG, Rack, Port, Box)	ClickOnRP**O_OS** (MG, Rack, Port, **Offset,** Box**, Channel**)
ClickOffRP(MG, Rack, Port, Box)	ClickOffRP**O_OS**(MG, Rack, Port, **Offset,** Box**, Channel**)

CONVERSION EXAMPLE

For example, below is a portion of the ANL926FREQ.MPC file distributed with Med-PC 4, 5, and 6:

```
S.S.1,
S1,
  0.01 ": SET A(^Freq) = 1000, A(^Amp)   = 100, A(^RF) = 10;
         SET A(^Dur)  = 1000, A(^Click) = 10 ---> S2

S2,      \ First Statement: Wait for START signal and then issue
         \ starting stimulus
         \
         \ Second Statement: Update screen display with default values
         \ for Control Variables. This will show any changes made via
         \ the "Configure | Change Variables" Window prior to START.
  #START: ~SetRack(MG, 1);~;              \ ANL-926 cards are in Rack 1
          ~InitANL926;~;                  \ Reset ANL-926
          ~SetFreq(MG, BOX, A[0]);~;      \ Initialize Frequency
          ~SetAmp (MG, BOX, A[1]);~;      \ Initialize Amplitude
          ~SetRF  (MG, BOX, A[2]);~;      \ Initialize Rise\Fall Time
          ~SetDur (MG, BOX, A[3]);~;      \ Initialize Duration
          ~OnFreq (MG, BOX, A[0]);~;      \ Issue Starting Stimulus
          ---> S3
  1     ":      SHOW      1,Frequency,A(^Freq),      2,Amplitude,A(^Amp),
3,Rise/Fall,A(^RF);
      SHOW 4,Duration,A(^Dur),   5,Pulse Click,A(^Click) ---> S2
```

And here is the code converted for use with OSC-112:

Note `_OS` added to each command, channel parameter (", `1`") added to each command, and the removal of the `InitANL926` command.

```
S.S.1,
S1,
  0.01 ": SET A(^Freq) = 1000, A(^Amp)   = 100, A(^RF) = 10;
         SET A(^Dur)  = 1000, A(^Click) = 10 ---> S2

S2,      \ First Statement: Wait for START signal and then issue
         \ starting stimulus
         \
         \ Second Statement: Update screen display with default values
         \ for Control Variables. This will show any changes made via
         \ the "Configure | Change Variables" Window prior to START.
  #START: ~SetRack_OS(MG, 1);~;           \ OSC-112 cards are in Rack 1
          ~SetFreq_OS(MG, BOX, A[0], 1);~;  \ Initialize Frequency
          ~SetAmp_OS (MG, BOX, A[1], 1);~;  \ Initialize Amplitude
          ~SetRF_OS  (MG, BOX, A[2], 1);~;  \ Initialize Rise\Fall Time
          ~SetDur_OS (MG, BOX, A[3], 1);~;  \ Initialize Duration
          ~OnFreq_OS (MG, BOX, A[0], 1);~;  \ Issue Starting Stimulus
          ---> S3
  1     ":      SHOW      1,Frequency,A(^Freq),      2,Amplitude,A(^Amp),
3,Rise/Fall,A(^RF);
      SHOW 4,Duration,A(^Dur),   5,Pulse Click,A(^Click) ---> S2
```

ANL-926 Protocol Command	OSC-112 Protocol Command	Note
`~ToneOn(MG,BOX);~`	`~ToneOn_OS(MG,BOX,1);~`	Turn on the tone configured with a previous "Set" command on channel 1.
`~SetClickFreq(MG,BOX,20);~`	`~SetClickFreq_OS(MG,BOX,20, 2);~`	Set the click frequency to 20 Hz for channel 2.
`~SetRF(MG,BOX,10);~`	`~SetRF_OS(MG,BOX,10,1);~`	Set the rise/fall time to 10 msec on channel 1.

Note: in MedState Notation, capitalization is ignored. `ToneOn_OS` is equivalent to `toneon_os`, `tOnEoN_oS`, etc.

ANL-926 and OSC-112 Together

If the Med-PC system consists of both ANL-926 and OSC-112 programmable audio generators, there are two possible MedState Notation Protocol scenarios. Either maintain separate programs for the specific hardware and load the appropriate program per box based on that box's hardware, or in a single program use IF statements to call separate commands based on the BOX number (and therefore hardware type).

SEPARATE PROGRAMS

If the existing ANL-926 program is called "Habituation.MSN" and the converted-for-OSC-112 program is called "Habituation OmniSound.MSN", be sure to load "Habituation.MSN" into the boxes with ANL-926 cards, and "Habituation OmniSound.MSN" into the boxes with OSC-112 audio generators.

SINGLE PROGRAM

```
\Boxes 1&2 have ANL926, boxes 3&4 have OSC-112
S1,
     1": IF (BOX <= 2) [@ANL926, @OSC112]
          @ANL926: ~OnFreq(MG, BOX, 2000);~ ---> S2
          @OSC112: ~OnFreq OS(MG, BOX, 2000, 1);~ ---> S2
```

A single program would utilize the reserved keyword BOX to determine the hardware, and call the correct command based on that hardware. This single program could be loaded into any box. If, for example, we have a system with two (2) boxes numbered 1 and 2 with ANL-926 audio generators, and two (2) boxes numbered 3 and 4 with OSC-112 audio generators, the code would look like:

ANL-926 vs OSC-112 Capabilities

When converting code for ANL-926 for use with OSC-112, keep in mind that the two models have their own set of limitations.

Field	ANL-926	OSC-112	Note
Frequency	10 Hz to 35,000 Hz	1000 Hz to 70,000 Hz	Minimum 10 -> 1000
Amplitude	20 dB SPL to 100 dB SPL in 0.5 dB increments	20 dB SPL to 100 dB SPL in 1 dB increments	Whole dB increments
Port	790	794	Default port

CHAPTER 8 OMNISOUND SPEAKER FREQUENCY RESPONSE TABLES

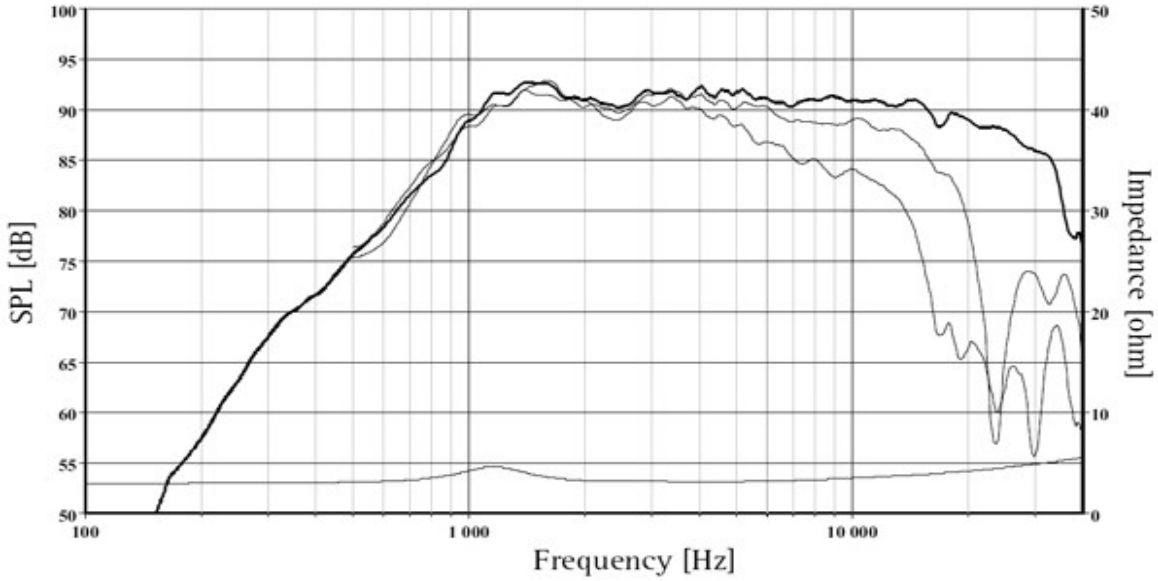
The following graphs display the frequency response from the currently supported rat and mouse speakers. The rat speaker model number shown is ENV-228AM, and the mouse speaker model number is ENV-328BW or ENV-328BM. The speakers used to produce these models may change at any time, and the graphs may not represent the speakers in your system. These charts are useful to illustrate the dramatic drop-off in sound pressure level of higher frequencies when measured at wider angles (0°, 30°, and 60° shown) off-axis from the speaker's centerline.

Always measure sound pressure levels at the same distance and angle from the speaker to ensure consistent values for comparison between chambers, frequencies, and environmental conditions.

Med Associates recommends using the ANL-930 Sound Pressure Level Measurement device for calibrating sound pressure levels amongst multiple sound attenuating cubicles (SACs) and test chambers.

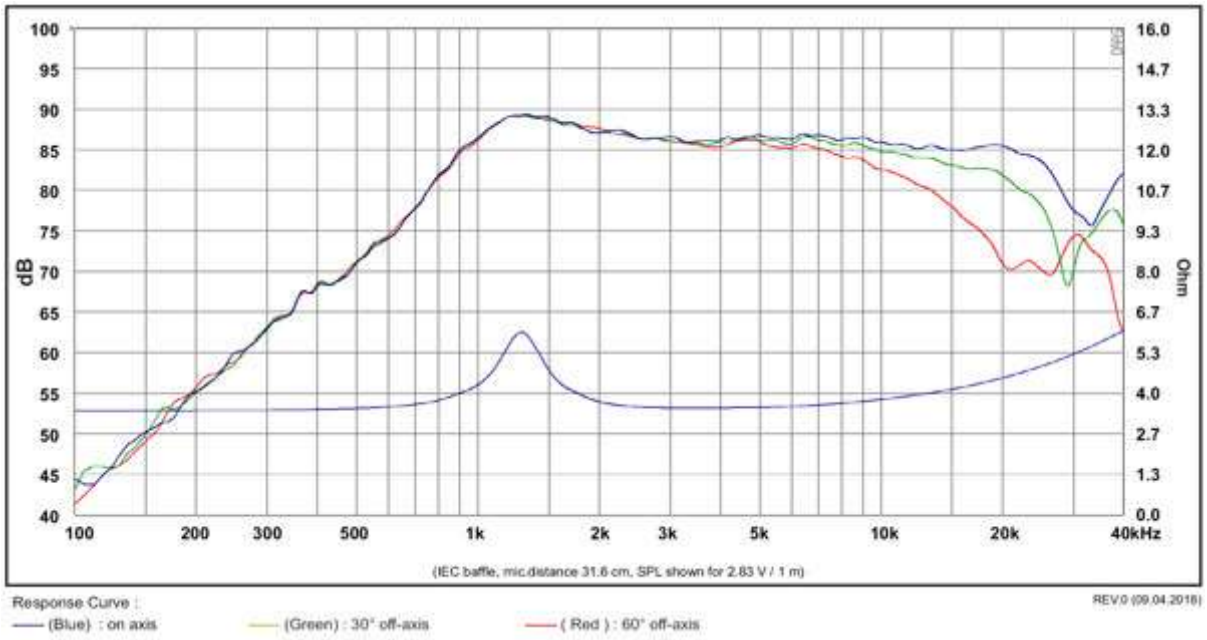
Note: Speaker models may vary. Med Associates presents these graphs from the speaker manufacturers as examples of degradation of sound pressure levels at higher frequencies and greater off-axis angle measurement.

FIGURE 8.1- FREQUENCY RESPONSE: ENV-228AM CAGE SPEAKER FOR RAT CHAMBER



The frequency responses above show measured free field sound pressure in 0, 30, and 60 degrees, mounted in a 0.6m by 0.8m baffle. Input 2.83 Vrms, microphone distance 0.5m, normalized to SPL 1m. The impedance is measured without baffle using a 2V sine signal.

FIGURE 8.2- FREQUENCY RESPONSE: ENV-328BW CAGE SPEAKER FOR ENV-307A CLASSIC MOUSE AND ENV-328BM CAGE SPEAKER FOR ENV-307W WIDE MOUSE.



CHAPTER 9 SAFETY, FAIL-SAFES, AND REPAIR

Cleaning

Do not submerge any portion of the device in liquid. Clean the device case using a soft cloth that is dampened with water and detergent. The device case is usually aluminum or plastic polymer. Should fluids enter the device internals disconnect power and invert the device to drain. Set the device in a warm dry area to air dry thoroughly before restoring power.

DO NOT AUTOCLAVE

DO NOT SUBMERGE

Warranty & Repair

This device is protected under a limited manufacturer warranty for two years from the date of purchase.

Med Associates continues to support and repair products outside the warranty period indefinitely. For a fee, we will repair our products as long as we are logistically able. We are proud to continue to support instruments manufactured in our opening production line from 1972.

If you are interested in further information about this device, its component parts, accessories, or integrating it within your application, please contact Med Associates Support.

Contact Information

Please contact Med Associates, Inc. for information regarding any of our products.

Med Associates Support is available to answer technical questions weekdays between 8:00am and 4:30pm ET by phone at 1-802-527-2343, or email at support@med-associates.com. We look forward to collaborating with you.

This and other manuals are available at our website.

Visit: https://med-associates.com/resources/manuals/ or use your device to follow the QR code link held by the rat on the back cover of this manual.

Visit our website at www.med-associates.com. For pricing inquiries, email sales@med-associates.com.

NOTES:

NOTES:

NOTES:

This manual was prepared by:
Med Associates Staff
With special thanks to the contributions of

Andy Lebeis
Software Engineer

Kevin Deutermann
Graphic Designer

Seth Truran
Sr. Application Scientist

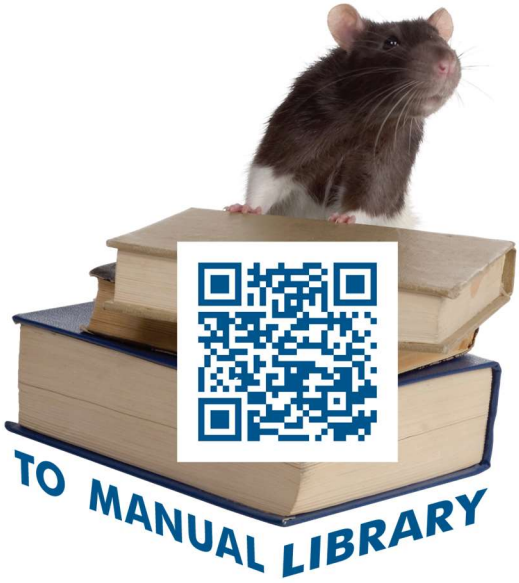
Christophe Stoelinga
Electrical Engineer

Gary Bamberger
Software Engineer

Jon Beyor
Support Manager

Bobbi Bushey
Sales Manager

Chris Johnson
Engineering Manager



Med Associates Inc.
P.O. Box 319
St. Albans, VT 05478
www.med-associates.com

Phone: 802.527.2343

Fax: 802.524.2110