



APPLICATION **NOTE**

Sound Manager 2
Third Party Interface
Socket IP Communication

APPLICATION NOTE

THIRD PARTY INTERFACE- SOCKET IP COMMUNICATION

Table of Contents

Third Party Interface	3
TCP/IP	3
RS232.....	3
TPI Protocol	3
Protocol Definition.....	3
SPECTRA i.NET® OP Communication Messages.....	4
iNET Examples.....	7
Emerald OP 24 Channel Communication Messages.....	10
Emerald Examples.....	17
Sound Manager (dB) to TPI Comparison Chart	19

APPLICATION NOTE

THIRD PARTY INTERFACE- SOCKET IP COMMUNICATION

Third Party Interface

Sound Manager 2 comes equipped with two separate forms of third party interface (TPI), TCP/IP and RS232. Both TCP/IP and RS232 use the same protocol and no difference exists except for the transport mechanism.

TCP/IP

In order to connect to Sound Manager 2 and the iLon server for purposes of third party control via TCP/IP, you must open a connection using a TCP/IP client to the IP address of the iLon Server using port 61000. Once connected, you can control the Lencore hardware using the protocol that is listed later in the document. If you cannot connect via port 61000, it is more than likely the port is already in use, or the TPI FPM modules have not been installed. Please contact Lencore support in this case of the later.

RS232

If you would like to use RS232 for third party control, you must connect to the serial port on the iLon server. The DB9 on the iLon server is NOT used for this purposes, the DB9 is for setup and debugging of the iLon server only. The actual RS232 port that you need to use is available on the screw terminal strip on the top side of the iLon Server, see table below.

Terminal		
25	Ground	Baud Rate: 9600 Parity: None Start Bits: 8 Stop Bits: 1
24	Transmit	CTS/RTS: No connection needed.
23	Receive	

TPI Protocol

The TPI supports a wide range of control and queries, where one can control both an Emerald OP and an iNet OP directly. You can also control Zones of intermixed OPs. Emerald OPs support some features that the iNet OPs don't, in the case of Zone control, the iNet OPs will ignore messages that they do not support.

It should be noted, that the TPI will NOT return a response when sending Zone control, as the messages for Zone control are one way. However with direct control of the OP, you will receive responses from both control and queries message requests.

When sending direct OP messages, you should wait for a reply until you send the next message. With Zone control messages, since you won't get a response, you should wait 500ms between messages.

This document will not list all commands available, only the common commands that a TPI control system may use.

Protocol Definition

- All protocol values are listed in actual decimal values and should not be mistaken for ASCII encoded values.
- An OP message is a collection of parameters where each parameter is represented as a byte.
- There is no start or end-of-message byte.
- The message is composed of 6 parameter bytes and up to 25 data bytes.
- See pages 7 and 17 for examples of message string sequences.
- In this document, all parameters are shown as decimal numbers not hexadecimal numbers.

APPLICATION NOTE

THIRD PARTY INTERFACE- SOCKET IP COMMUNICATION

SPECTRA i.NET® OP Communication Messages

OP_Request Header.

Request header

Byte 1 = OP Address	= 1 to 199	if parameter 2 is INDIVIDUAL.
= Group number	= 1 to 255	if parameter 2 is MASK_GROUP, PAGE_GROUP or MUSIC_GROUP.
= Global	= 0	if parameter 2 is GLOBAL_SELECT.
Byte 2 = Group Select	= 0	INDIVIDUAL
	= 1	MASK_GROUP
	= 10	PAGE_GROUP
	= 20	MUSIC_GROUP
	= 127	GLOBAL_SELECT
Byte 3 = Command Category	= 1	MASK_ATTENUATION
	= 10	PAGE_ATTENUATION
	= 20	MUSIC_ATTENUATION
Byte 4 = Sub Command	= 1	Set value for channel A
(dependent on	= 2	Set value for channel B
command category)	= 3	Set value for channel C
	= 4	Set value for channel D
	= 5	Set value for all four channels
	= 6	Request current value for channel A
	= 7	Request current value for channel B
	= 8	Request current value for channel C
	= 9	Request current value for channel D
	= 10	Request current value for all four channels
	= 11 to 21	Dependent on command category
Byte 5 = Transaction ID	= 1 to 255	
Byte 6 = Parameter count	= 6 + any data bytes	
Bytes 7 to 31	= 25 available data bytes	

OP_Response Header.

Response header

Byte 1 = OP Address	= 1 to 199 or ...	
= Group number	= 1 to 255 or ...	
= Global	= 0	
Byte 2 = Result Code	= 1	COMMAND_SUCCESS
	= 2	COMMAND_FAIL
	= 3	NO_RESPONSE_FROM_DSP
Byte 3 = Command Category	= 1	MASK_ATTENUATION
	= 10	PAGE_ATTENUATION
	= 20	MUSIC_ATTENUATION
Byte 4 = Sub Command	= 1 to 21	Dependent on command category
Byte 5 = Transaction ID	= 1 to 255	
Byte 6 = Byte count	= 6 to 31	
Bytes 7 to 31	= Return data bytes if any	

APPLICATION NOTE

THIRD PARTY INTERFACE- SOCKET IP COMMUNICATION

List of required "Response" parameter count

MASK_ATTENUATION

SubCommand 1 to 9, Required byte count = 1 + header = 7
SubCommand 10, Required byte count = 6 + header = 12
See below for byte 11, 12 decoding.

PAGE_ATTENUATION

SubCommand 1 to 9, Required byte count = 1 + header = 7
SubCommand 10, Required byte count = 6 + header = 12
See below for byte 11, 12 decoding.

MUSIC_ATTENUATION

SubCommand 1 to 9, Required byte count = 1 + header = 7
SubCommand 10, Required byte count = 6 + header = 12
See below for byte 11, 12 decoding.

APPLICATION NOTE

THIRD PARTY INTERFACE- SOCKET IP COMMUNICATION

Emerald OP 24 Channel Communication Messages

OP_Request Header.

Request header

Byte 1 = OP Address	= 1 to 199	if parameter 2 is INDIVIDUAL.
= Group number	= 1 to 255	if parameter 2 is MASK_GROUP, PAGE_GROUP or MUSIC_GROUP.
= Global	= 0	if parameter 2 is GLOBAL_SELECT.
Byte 2 = Group Select	= 0	INDIVIDUAL
	= 1	MASK_GROUP
	= 10	PAGE_GROUP
	= 20	MUSIC_GROUP
	= 127	GLOBAL_SELECT
Byte 3 = Command Category	= 1	MASK_ATTENUATION
	= 10	PAGE_ATTENUATION
	= 20	MUSIC_ATTENUATION
Byte 4 = Sub Command	= 31	Set value for channel 1
(dependent on	= 32	Set value for channel 2
command category)	= 33	Set value for channel 3
	= 34	Set value for channel 4
	= 35	Set value for channel 5
	= 36	Set value for channel 6
	= 37	Set value for channel 7
	= 38	Set value for channel 8
	= 39	Set value for channel 9
	= 40	Set value for channel 10
	= 41	Set value for channel 11
	= 42	Set value for channel 12
	= 43	Set value for channel 13
	= 44	Set value for channel 14
	= 45	Set value for channel 15
	= 46	Set value for channel 16
	= 47	Set value for channel 17
	= 48	Set value for channel 18
	= 49	Set value for channel 19
	= 50	Set value for channel 20
	= 51	Set value for channel 21
	= 52	Set value for channel 22
	= 53	Set value for channel 23
	= 54	Set value for channel 24
	= 55	Set value for all 24 channels or zoned command
	= 61	Request current value for channel 1
	= 62	Request current value for channel 2
	= 63	Request current value for channel 3
	= 64	Request current value for channel 4
	= 65	Request current value for channel 5
	= 66	Request current value for channel 6
	= 67	Request current value for channel 7
	= 68	Request current value for channel 8
	= 69	Request current value for channel 9
	= 70	Request current value for channel 10
	= 71	Request current value for channel 11
	= 72	Request current value for channel 12
	= 73	Request current value for channel 13
	= 74	Request current value for channel 14
	= 75	Request current value for channel 15
	= 76	Request current value for channel 16
	= 77	Request current value for channel 17
	= 78	Request current value for channel 18
	= 79	Request current value for channel 19
	= 80	Request current value for channel 20
	= 81	Request current value for channel 21
	= 82	Request current value for channel 22

APPLICATION NOTE

THIRD PARTY INTERFACE- SOCKET IP COMMUNICATION

	= 83	Request current value for channel 23
	= 84	Request current value for channel 24
	= 85	Request current value for all 24 channels
Byte 5 = Transaction ID	= 1 to 255	
Byte 6 = Byte count	= 6 + any data bytes	
Bytes 7 to 31	= 25 available data bytes	

OP_Response Header.

Response header

Byte 1 = OP Address	= 1 to 199 or ...	
= Group number	= 1 to 255 or ...	
= Global	= 0	
Byte 2 = Result Code	= 1	COMMAND_SUCCESS
	= 2	COMMAND_FAIL
	= 3	NO_RESPONSE_FROM_DSP
Byte 3 = Command Category	= 1	MASK_ATTENUATION
	= 10	PAGE_ATTENUATION
	= 20	MUSIC_ATTENUATION
Byte 4 = Sub Command	= 1 to 85	Dependent on command category.
Byte 5 = Transaction ID	= 1 to 255	
Byte 6 = Parameter count	= 6 to 31	
Bytes 7 to 31	= Return bytes if any.	

APPLICATION NOTE

THIRD PARTY INTERFACE- SOCKET IP COMMUNICATION

Mute TPI values = 1 to mute, 0 to unmute

```

ch 1      1 0 1 91 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 2      1 0 1 92 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 3      1 0 1 93 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 4      1 0 1 94 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 5      1 0 1 95 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 6      1 0 1 96 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 7      1 0 1 97 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 8      1 0 1 98 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 9      1 0 1 99 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 10     1 0 1 100 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 11     1 0 1 101 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 12     1 0 1 102 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 13     1 0 1 103 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 14     1 0 1 104 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 15     1 0 1 105 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 16     1 0 1 106 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 17     1 0 1 107 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 18     1 0 1 108 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 19     1 0 1 109 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 20     1 0 1 110 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 21     1 0 1 111 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 22     1 0 1 112 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 23     1 0 1 113 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 24     1 0 1 114 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
* ch all  1 0 1 115 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

```

Note: The Req. ch all will return all masking volume levels.

1 0 1 85 88 12 <ch1 vol> <ch2 vol> ... <ch24 vol> 0

* When MASK_GROUP option is used in the header, only the "ch all" sub command can be used.

For MASK_GROUP MASK_ATTENUATION can use Sub Command 55 or 5 for all channel.

For MASK_GROUP MUTE: Can use Sub Command 115 or 20 for all channel mute.

! The Req.'s (request for current values), can only use INDIVIDUAL or GLOBAL_SELECT.

Mute request all channels

```
! Req. ch all 1 0 1 145 88 6 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

Returns status of channel - muted or not - for masking, paging, & music

```
1 0 1 145 88 6 <Byte1> <Byte2> <Byte3> <Byte4> <Byte5> <Byte6> <Byte7> <Byte8> <Byte9> <Byte10> <Byte11>
<Byte12> 0...
```

PAGE_ATTENUATION TPI values = 0 to 49 (see dB to TPI comparison chart at the end of this document)

Command: 1 0 10 <ch#> <txid> <parameters> <vol> 0

```

ch 1      1 0 10 31 88 7 43 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 2      1 0 10 32 88 7 43 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 3      1 0 10 33 88 7 43 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 4      1 0 10 34 88 7 43 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 5      1 0 10 35 88 7 43 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 6      1 0 10 36 88 7 43 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 7      1 0 10 37 88 7 43 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 8      1 0 10 38 88 7 43 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 9      1 0 10 39 88 7 43 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 10     1 0 10 40 88 7 43 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 11     1 0 10 41 88 7 43 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 12     1 0 10 42 88 7 43 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 13     1 0 10 43 88 7 43 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 14     1 0 10 44 88 7 43 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 15     1 0 10 45 88 7 43 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 16     1 0 10 46 88 7 43 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 17     1 0 10 47 88 7 43 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 18     1 0 10 48 88 7 43 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 19     1 0 10 49 88 7 43 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 20     1 0 10 50 88 7 43 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 21     1 0 10 51 88 7 43 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 22     1 0 10 52 88 7 43 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 23     1 0 10 53 88 7 43 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

```

APPLICATION NOTE

THIRD PARTY INTERFACE- SOCKET IP COMMUNICATION

```
ch 24      1 0 10 54 88 7 43 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
* ch all   1 0 10 55 88 7 43 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
! Req. ch 1 1 0 10 61 88 6 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
! Req. ch 2 1 0 10 62 88 6 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
! Req. ch 3 1 0 10 63 88 6 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
! Req. ch 4 1 0 10 64 88 6 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
! Req. ch 5 1 0 10 65 88 6 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
! Req. ch 6 1 0 10 66 88 6 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
! Req. ch 7 1 0 10 67 88 6 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
! Req. ch 8 1 0 10 68 88 6 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
! Req. ch 9 1 0 10 69 88 6 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
! Req. ch 10 1 0 10 70 88 6 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
! Req. ch 11 1 0 10 71 88 6 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
! Req. ch 12 1 0 10 72 88 6 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
! Req. ch 13 1 0 10 73 88 6 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
! Req. ch 14 1 0 10 74 88 6 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
! Req. ch 15 1 0 10 75 88 6 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
! Req. ch 16 1 0 10 76 88 6 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
! Req. ch 17 1 0 10 77 88 6 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
! Req. ch 18 1 0 10 78 88 6 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
! Req. ch 19 1 0 10 79 88 6 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
! Req. ch 20 1 0 10 80 88 6 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
! Req. ch 21 1 0 10 81 88 6 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
! Req. ch 22 1 0 10 82 88 6 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
! Req. ch 23 1 0 10 83 88 6 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
! Req. ch 24 1 0 10 84 88 6 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
! Req. ch all 1 0 10 85 88 6 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

Note: * When **PAGE_GROUP** option is used in the header, only the "ch **all**" sub command can be used.
! The Req.'s (request for current values), can only use **INDIVIDUAL** or **GLOBAL_SELECT**.

Mute **TPI** values = 1 to mute, 0 to unmute

```
ch 1      1 0 10 91 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 2      1 0 10 92 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 3      1 0 10 93 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 4      1 0 10 94 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 5      1 0 10 95 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 6      1 0 10 96 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 7      1 0 10 97 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 8      1 0 10 98 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 9      1 0 10 99 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 10     1 0 10 100 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 11     1 0 10 101 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 12     1 0 10 102 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 13     1 0 10 103 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 14     1 0 10 104 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 15     1 0 10 105 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 16     1 0 10 106 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 17     1 0 10 107 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 18     1 0 10 108 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 19     1 0 10 109 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 20     1 0 10 110 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 21     1 0 10 111 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 22     1 0 10 112 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 23     1 0 10 113 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
ch 24     1 0 10 114 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
* ch all 1 0 10 115 88 7 <mute> 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```


APPLICATION NOTE

THIRD PARTY INTERFACE- SOCKET IP COMMUNICATION

Emerald Examples

Note: Unused data bytes should not be sent. For example, if the byte count is 7, only 7 bytes should be sent. Trailing zeros should not be sent.

Example 1 sets OP-3 channel-1 masking attenuation to 0 dB (49).

```
Example command = "3 INDIVIDUAL_MASK_ATTENUATION 31 88 7 49 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0"
Actual message (decimal bytes) = 3,0,1,31,88,7,49
```

Example 2 sets masking group 1 attenuation to -1.5 dB (46).

```
Example command = "1 MASK_GROUP_MASK_ATTENUATION 55 88 7 46 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0"
Actual message (decimal bytes) = 1,1,1,55,88,7,46
```

Example 3 sets paging group 3 attenuation to +18 dB (43) (source 1).

```
Example command = "3 PAGE_GROUP_PAGE_ATTENUATION 55 88 8 43 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0"
Actual message (decimal bytes) = 3,10,10,55,88,8,43,1
```

Example 4 sets music group 2 attenuation to +3 dB (28) (source 3).

```
Example command = "2 MUSIC_GROUP_MUSIC_ATTENUATION 55 88 8 28 3 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0"
Actual message (decimal bytes) = 2,20,20,55,88,8,28,3
```

Example response from **example 1** above.

```
Response_Msg = "3 COMMAND_SUCCESS MASK_ATTENUATION 31 88 7 49 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0"
Actual message (decimal bytes) = 3,1,1,31,88,7,49,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
```

Example response from **example 2** above.

```
Response_Msg = "1 COMMAND_SUCCESS MASK_ATTENUATION 55 88 7 46 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0"
Actual message (decimal bytes) = 1,1,1,55,88,7,46,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
```

Example response from **example 3** above.

```
Response_Msg = "3 COMMAND_SUCCESS PAGE_ATTENUATION 55 88 7 43 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0"
Actual message (decimal bytes) = 3,1,10,55,88,7,43,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
```

Example response from **example 4** above.

```
Response_Msg = "2 COMMAND_SUCCESS MUSIC_ATTENUATION 55 88 7 28 3 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0"
Actual message (decimal bytes) = 2,1,20,55,88,7,28,3,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
```

APPLICATION NOTE

THIRD PARTY INTERFACE- SOCKET IP COMMUNICATION

List of required "Response" byte count

MASK_ATTENUATION

SubCommand 31 to 84, Required byte count = 1 + header = 7

SubCommand 85, Required byte count = 24 + header = 30

PAGE_ATTENUATION

SubCommand 31 to 84, Required byte count = 2 + header = 8

SubCommand 85, Required byte count = 24 + header = 30

MUSIC_ATTENUATION

SubCommand 31 to 84, Required byte count = 2 + header = 8

SubCommand 85, Required byte count = 24 + header = 30

APPLICATION NOTE

THIRD PARTY INTERFACE- SOCKET IP COMMUNICATION

Sound Manager (dB) to TPI Comparison Chart

<u>Sound Manager (dB) Value to TPI Value Comparison</u>	
Mask Attenuation:	
<u>Sound Manager (dB) Value</u>	<u>TPI Value</u>
82.5	97
82	96
81.5	95
81	94
80.5	93
80	92
79.5	91
79	90
78.5	89
78	88
77.5	87
77	86
76.5	85
76	84
75.5	83
75	82
74.5	81
74	80
73.5	79
73	78
72.5	77
72	76
71.5	75
71	74
70.5	73
70	72
69.5	71
69	70
68.5	69
68	68
67.5	67
67	66
66.5	65
66	64
65.5	63
65	62

APPLICATION NOTE

THIRD PARTY INTERFACE- SOCKET IP COMMUNICATION

64.5	61
64	60
63.5	59
63	58
62.5	57
62	56
61.5	55
61	54
60.5	53
60	52
59.5	51
59	50
58.5	49 (Default)
58	48
57.5	47
57	46
56.5	45
56	44
55.5	43
55	42
54.5	41
54	40
53.5	39
53	38
52.5	37
52	36
51.5	35
51	34
50.5	33
50	32
49.5	31
49	30
48.5	29
48	28
47.5	27
47	26
46.5	25
46	24
45.5	23
45	22
44.5	21
44	20

APPLICATION NOTE

THIRD PARTY INTERFACE- SOCKET IP COMMUNICATION

43.5	19
43	18
42.5	17
42	16
41.5	15
41	14
40.5	13
40	12
39.5	11
39	10
38.5	9
38	8
37.5	7
37	6
36.5	5
36	4
35.5	3
35	2
34.5	1
34	0 (off)

Page / Music Attenuation:	
<u>dB Value</u>	<u>TPI Value</u>
82.5	49
81.5	48
80.5	47
79.5	46
78.5	45
77.5	44
	43
76.5	(Default)
75.5	42
74.5	41
73.5	40
72.5	39
71.5	38
70.5	37
69.5	36
68.5	35
67.5	34
66.5	33

APPLICATION NOTE

THIRD PARTY INTERFACE- SOCKET IP COMMUNICATION

65.5	32
64.5	31
63.5	30
62.5	29
61.5	28
60.5	27
59.5	26
58.5	25
57.5	24
56.5	23
55.5	22
54.5	21
53.5	20
52.5	19
51.5	18
50.5	17
49.5	16
48.5	15
47.5	14
46.5	13
45.5	12
44.5	11
43.5	10
42.5	9
41.5	8
40.5	7
39.5	6
38.5	5
37.5	4
36.5	3
35.5	2
34.5	1
34	0 (off)

Lencore Acoustics Corp.
1 Crossways Park Drive West
Woodbury, NY 11797
516-682-9292
info@lencore.com
www.lencore.com

Revision History

Version 1.0 - Initial Release

Version 2.0- Added Communication Message (i.Net & Emerald)