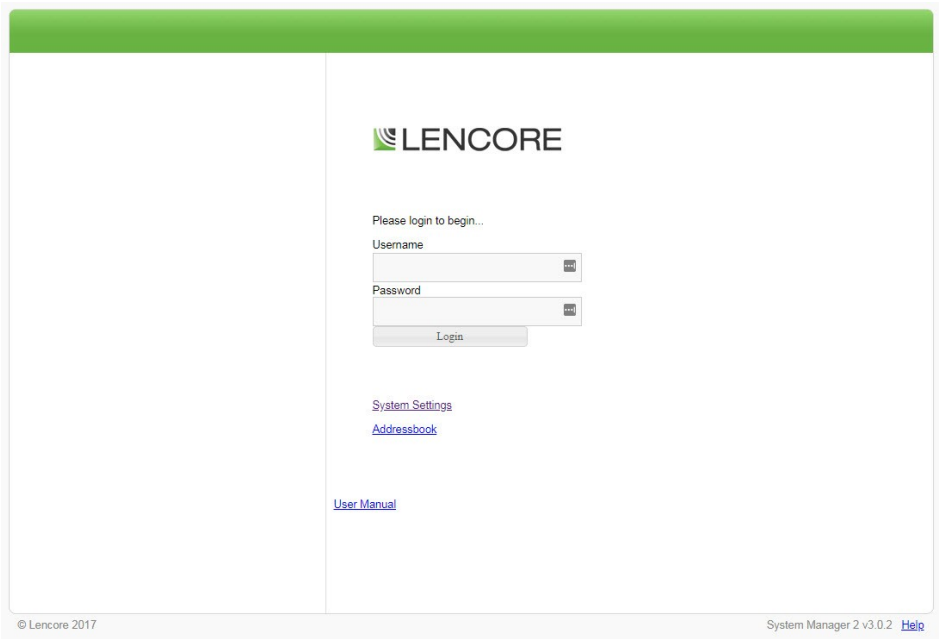
LENCORE

COMFORT. PRIVACY. PRODUCTIVITY.



APPLICATION NOTE: 3

System Manager 2
Third Party Interface
RESTful IP Communication

Rev B

APPLICATION NOTE

System Manager 2 Third Party Interface RESTful IP Communication

Contents

APPLICATION NOTE

System Manager 2 Third Party Interface RESTful IP Communication

Third Party Interface

While the iLON SmartServer based System Manager 2 came with a third-party interface (TPI) based on Sockets and TCP/IP , the Protocessor based version of System Manager 2 comes with a RESTful interface to allow more sophisticated access to the system.

TCP/IP

In order to connect to System Manager 2 and the Protocessor for purposes of third-party control via TCP/IP, you must establish a connection to the REST endpoint using a TCP/IP client to the IP address of the Protocessor using port 80. Once connected, you can control the Lencore hardware using the protocol that is listed later in the document.

TPI Protocol

The TPI supports a wide range of control and queries, where one can control an i.Net® OP directly.
It should be noted, that the TPI will NOT return a response, with data, when sending Zone control, as the messages for Zone control are one way. However, with direct control of the OP, you will receive responses, with data, from query message requests. However, you will always get confirmation REST responses for all commands.
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.
- The message is composed of 6 parameter bytes and up to 25 data bytes.
- See page 12 for examples of message string sequences.
- In this document, all parameters are shown as decimal numbers not hexadecimal numbers.

RESTful communication:

A typical REST call looks like the following:

```
// typical rest JSON request fields
parameters (object)

// example
{
        "parameters": { }
}
```

A typical RESTful response would look like this:

```
// typical rest JSON response payload fields
message (string)
data (object)
error (string)

// response
{
        "message": "",
        "data": { },
        "error": ""
}
```

Before any meaningful control can be done with the TPI interface, you must first login to the Protocessor and get a "token", as follows:

Resource endpoint//
http://ip.address/rest/login

// login parameters: username and password
Default username and password are "comfort"

// example ("HTTP GET" Request)
http://<ip.address>/rest/login?username=*somename*&password=*somepassword*

```
// example normal login response payload
{
        "message": "Logged in ok",
        "data": {
                "token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ"
        },
        "error": null
}
```

Now that we have a basic connection established, we can control various functions. A typical API call to control the volume of Masking Attenuation would look like this:

Example Payload of a request to a single OP:

// “OP” example request payload (MASK_ATTENUATION (set channel A volume))

{
“parameters”: { “OPAddress”: 1, “OPGroupSelect”: 0, “OPCommandCat”: 1, “OPSubCommand”: 1, “OPTransactionID”: 88, “OPDataCount”: 7, “OPData”:[49] }
}

// “OP” example response payload (MASK_ATTENUATION (set channel A volume))

{
  “message”: “Call successful”,
  “data”: {
        “OPAddress”: 1,
        “OPResultCode”: 1,
        “OPCommandCat”: 1,
        “OPSubCommand”: 1,
        “OPTransactionID”: 244,
        “OPDataCount”: 7,
        “OPData”: [
        49,
      ]
     },
     “error” : null
  }

The specific API call is implemented as below. Refer to the command listings at the end of this document for all the parameters.

// Control Resource endpoint

http://ip.address/rest/method/TPI/control

// control parameters
happn_token // Auth Token parameter- This is the Token from above login process

// control “parameters” JSON request payload
OPAddress (unsigned byte)
OPGroupSelect (unsigned byte)
OPCommandCat (unsigned byte)
OPSubCommand (unsigned byte)
OPTransactionID (unsigned byte) (Transaction ID is ignored)
OPDataCount (unsigned byte)
OPData (unsigned byte array)

// control “data” JSON response payload
OPAddress (unsigned byte)
OPResultCode (unsigned byte)
OPCommandCat (unsigned byte)
OPSubCommand (unsigned byte)
OPTransactionID (unsigned byte) (Transaction ID is actual used)
OPDataCount (unsigned byte)
OPData (unsigned byte array)

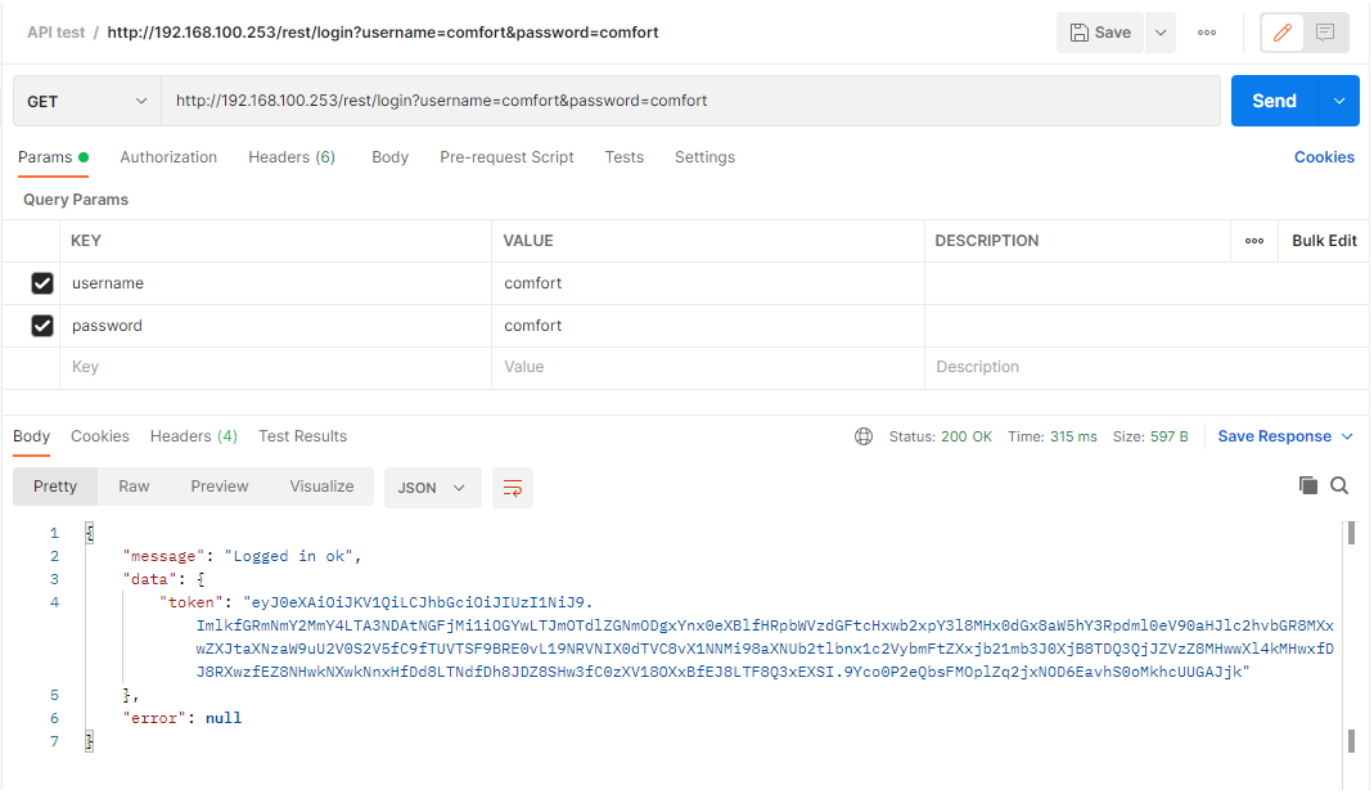
// example ("HTTP POST" Request)
http://<ip.address>/rest/method/TPI/control?happn_token=eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1

// "OP" example payload (set MASK_ATTENUATION (chA))

{
"parameters": { "OPAddress": 1, "OPGroupSelect": 0, "OPCommandCat": 1, "OPSubCommand": 1, "OPTransactionID": 0, "OPDataCount": 7, "OPData":[49] }
}

Example using Postman

GET token:
http://192.168.100.253/rest/login?username=comfort&password=comfort



POST masking volume change to channel A:
http://192.168.100.253/rest/method/TPI/control?happn_token= eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ

Parameters:

Params ● Authorization Headers (9) Body ● Pre-request Script Tests Settings

Query Params

	KEY	VALUE
☑	happn_token	eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.ImlkfGRmNm...
	Key	Value

Headers:

Params ● Authorization Headers (9) Body ● Pre-request Script Tests Settings

Headers 8 hidden

	KEY	VALUE
☑	Content-Type	application/json
	Key	Value

Body:

Params ● Authorization Headers (9) Body ● Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON

```
1 {
2 "parameters": { "OPAddress": 1, "OPGroupSelect": 0, "OPCommandCat": 1,
3 "OPSubCommand": 1, "OPTransactionID": 88, "OPDataCount": 7, "OPData":[49] }
4 }
```

API test / http://192.168.100.253/rest/login?username=comfort&password=comfort

Save

POST http://192.168.100.253/rest/method/TPI/control?happn_token=eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9.ImlkfGRmNmY2MmY4LTA3NDAtNGF... Send

Params Authorization Headers (9) Body Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary GraphQL JSON Beautify

```
{
"parameters": { "OPAddress": 1, "OPGroupSelect": 0, "OPCommandCat": 1,
"OPSubCommand": 1, "OPTransactionID": 88, "OPDataCount": 7, "OPData":[49] }
}
```

Body Cookies Headers (4) Test Results Status: 200 OK Time: 415 ms Size: 310 B Save Response

Pretty Raw Preview Visualize JSON

```
{
    "message": "Call successful",
    "data": {
        "OPAddress": 1,
        "OPResultCode": 1,
        "OPCommandCat": 1,
        "OPSubCommand": 1,
        "OPTransactionID": 244,
        "OPDataCount": 7,
        "OPData": [
            49
        ]
    },
    "error": null
}
```

Example of a change to all OP's:

// GLOBAL example payload (set MASK_ATTENUATION (chA))

{
“parameters”: { “OPAddress”: 0, “OPGroupSelect”: 127, “OPCommandCat”: 1,
“OPSubCommand”: 1, “OPTransactionID”: 88, “OPDataCount”: 7, “OPData”:[49] }
}

// example GLOBAL response payload example (set MASK_ATTENUATION (chA))

{
“message”: “Call successful”,
"data": "Success",
"error": null
}

Example of a change to a group of OPs by using a masking Zone:

// "MASK ZONE 1" example response payload example (set MASK_ATTENUATION (chA))

{
"parameters": { "OPAddress": 1, "OPGroupSelect": 1, "OPCommandCat": 1,
"OPSubCommand": 1, "OPTransactionID": 88, "OPDataCount": 7, "OPData":[49] }
}

// "MASK ZONE 1" example response payload example (set MASK_ATTENUATION (chA))

{
"message": "Call successful",
"data": "Success",
"error": null
}

// "OP" example response payload example (set MASK_ATTENUATION (chA))

{
"message": "Call successful",
"data": {
"OPAddress": 1,
"OPResultCode": 1,
"OPCommandCat": 1,
"OPSubCommand": 1,
"OPTransactionID": 19,
"OPDataCount": 7,
"OPData": [
49
]
},
"error": null
}

"data": "Success", "error":
null
}

In all the above examples, all the parameters come from the list of commands on page 10. Each command is 31 bytes.

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
```

List of commands expressed as TPI numerical values.

Note: Assume OP-1, all sample values set to **DEFAULT**, and Transaction ID set to 88.

The Transaction ID can be any parameter value 1 to 255, 0 can be used but not recommended. Refer to the System Manager (dB) to TPI comparison chart at the end of this document.

MASK_ATTENUATION ***TPI values*** = 0 to 97 (see dB to TPI comparison chart at the end of this document)

ch***1***	1	0	1	1	88	7	**49**	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	2	88	7	**49**	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	3	88	7	**49**	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	4	88	7	**49**	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	5	88	7	**49**	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	1	6	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
! Req. ch***2***	1	0	1	7	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
! Req. ch***3***	1	0	1	8	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
! Req. ch***4***	1	0	1	9	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
! Req. ch***all***	1	0	1	10	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

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

ch***1***	1	0	1	16	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
ch***2***	1	0	1	17	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
ch***3***	1	0	1	18	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
ch***4***	1	0	1	19	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
* ch***all***	1	0	1	20	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

Note: * When **MASK_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**.

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

ch***1***	1	0	10	1	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
ch***2***	1	0	10	2	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
ch***3***	1	0	10	3	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
ch***4***	1	0	10	4	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
* ch***all***	1	0	10	5	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
! Req. ch***1***	1	0	10	6	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
! Req. ch***2***	1	0	10	7	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
! Req. ch***3***	1	0	10	8	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
! Req. ch***4***	1	0	10	9	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
! Req. ch***all***	1	0	10	10	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

Mute T*PIpp values* = 1 to mute, 0 to unmute

ch***1***	1	0	10	16	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
ch***2***	1	0	10	17	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
ch***3***	1	0	10	18	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
ch***4***	1	0	10	19	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
* ch***all***	1	0	10	20	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

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**.

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

ch***1***	1	0	20	1	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
ch***2***	1	0	20	2	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
ch***3***	1	0	20	3	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
ch***4***	1	0	20	4	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
* ch***all***	1	0	20	5	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
! Req. ch***1***	1	0	20	6	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
! Req. ch***2***	1	0	20	7	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
! Req. ch***3***	1	0	20	8	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
! Req. ch***4***	1	0	20	9	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
! Req. ch***all***	1	0	20	10	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

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

ch***1***	1	0	20	16	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
ch***2***	1	0	20	17	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
ch***3***	1	0	20	18	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
ch***4***	1	0	20	19	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
* ch***all***	1	0	20	20	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

Note: * When **MUSIC_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**.

i.NET 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. See dB to TPI comparison chart at the end of this document.

Example 1 sets OP-3 channel-A masking attenuation to 58.5 dB (49).
Example command = “3 0 1 1 88 7 **49**"

Example 2 sets masking group 1 attenuation to 57 dB (46).
Example command = “1 1 1 5 89 7 **46**"

Example 3 sets paging group 3 attenuation to 55.5 dB (43).
Example command = “3 10 10 20 90 7 **43**"

Example 4 sets music group 2 attenuation to 48 dB (28).
Example command = “2 20_20 20 91 7 **28**"

Example response from **example 1** above.
Response_Msg = “3 1 1 1 88 7 49 0 0 0 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 1 1 5 89 7 46 0 0 0 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 1 10 20 90 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"

Example response from **example 4** above.
Response_Msg = “2 1 20 20 91 7 28 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0"

Mute/Unmute commands

The 31-byte string to mute/unmute a masking zone is:

<zone#> 1 1 20 88 7 <mute>

<zone #> = the masking zone # to control (must be assigned in System Manager first).
<mute> = 1-mute, 0-unmute
The “88” is the transaction ID. It can be any number between 1 and 255.

The 31-byte string to mute/unmute a paging zone is:

<zone#> 10 10 20 88 7 <mute>

<zone #> = the paging zone # to control (must be assigned in System Manager first).
<mute> = 1-mute, 0-unmute
The “88” is the transaction ID. It can be any number between 1 and 255

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.
```

Superflag byte 11 (MSB) and 12 (LSB) decoding:

From returned bytes in Req ch ***all*** **MASK_ATTENUATION** sub command 10
Or
From returned bytes in Req ch ***all*** **PAGE_ATTENUATION** sub command 10
Or
From returned bytes in Req ch ***all*** **MUSIC_ATTENUATION** sub command 10

A 0 in bit position means the channel is unmuted. (Default)
A 1 in bit position means the channel is muted.

```
Bit position:
15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1, 0
 |   |   |   |   |   |  |  |  |  |  |  |  |  |  |  |
 |   |   |   |   |   |  |  |  |  |  |  |  |  |  |  |
 |   |   |   |   |   |  |  |  |  |  |  |  |  |  |  |
 |   |   |   |   |   |  |  |  |  |  |  |  |  |  |  \------ MASK  ch-A
 |   |   |   |   |   |  |  |  |  |  |  |  |  |  \--------- PAGE  ch-A
 |   |   |   |   |   |  |  |  |  |  |  |  |  \------------ MUSIC ch-A
 |   |   |   |   |   |  |  |  |  |  |  |  |
 |   |   |   |   |   |  |  |  |  |  |  |  \--------------- MASK  ch-B
 |   |   |   |   |   |  |  |  |  |  |  \------------------ PAGE  ch-B
 |   |   |   |   |   |  |  |  |  |  \--------------------- MUSIC ch-B
 |   |   |   |   |   |  |  |  |  |
 |   |   |   |   |   |  |  |  |  \------------------------ MASK  ch-C
 |   |   |   |   |   |  |  |  \---------------------------- PAGE  ch-C
 |   |   |   |   |   |  |  \------------------------------ MUSIC ch-C
 |   |   |   |   |   |  |
 |   |   |   |   |   |  \--------------------------------- MASK  ch-D
 |   |   |   |   |   \------------------------------------ PAGE  ch-D
 |   |   |   |   \---------------------------------------- MUSIC ch-D
 |   |   |   |
 |   |   |   \------------------------------------------- Not used
 |   |   \----------------------------------------------- Not used
 |   \--------------------------------------------------- Not used
 \------------------------------------------------------- Not used
```

System Manager (dB) to TPI Comparison Chart

System Manager (dB) Value to TPI Value Comparison	
Mask Attenuation:	
System Manager (dB) Value	TPI Value
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

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

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:	
dB Value	**TPI Value**
82.5	49
81.5	48
80.5	47
79.5	46
78.5	45
77.5	44
76.5	43 (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

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 LLC
835 New York Ave. Suite 21
Huntington, NY 11743
516-682-9292
info@lencore.com
www.lencore.com