



| API Overview | 2 |
|--|----|
| Device Communications / Message Format | 3 |
| Message Definitions / Error Codes | 4 |
| Message Details | 5 |
| Change History | 19 |

The content of this communication and / or document, including but not limited to images, specifications, designs, concepts, data and information in any format or medium is confidential and is not to be used for any purpose or disclosed to any third party without the express and written consent of Keymat Technology Ltd. Copyright Keymat Technology Ltd. 2022.

Storm, Storm Interface, Storm AXS, Storm ATP, Storm IXP, Storm Touchless-CX, AudioNav, AudioNav-EF and NavBar are trademarks of Keymat Technology Ltd. All other trademarks are the property of their respective owners

Storm Interface is a trading name of Keymat Technology Ltd Storm Interface products include technology protected by international patents and design registration. All rights reserved



API for controlling the Activation Sensor from the Host Computer

This document details how the Microphone Activation Sensor can be controlled from a host that has USB capabilities. The API incorporating this command set is downloadable from <u>www.storm-interface.com</u>.

Host API Library - Overview

The Host API Library is a middleware application between the host application and the Microphone Activation Sensor. This is available as a download together with the HIDAPI library.

- The API library allows for the host application to invoke the Microphone Activation Sensor functions as listed. It encapsulates all the communications to USB and provides a simple API for the host application developers.
- HIDAPI This is a third party library, which allows an application to interface with USB HID-Compliant devices on Windows, Linux, and Mac OS X. While it can be used to communicate with standard HID devices like keyboards, mice, and Joysticks, it is most useful with custom (Vendor-Defined) HID devices. This allows for host software to scan for the device using its VID/PID.

The AudioComm uses USB for communicating with the host. It includes an HID-compliant device One advantage of using this implementation, which uses only HID interfaces, is that no drivers are required on the host system.

The protocol for communicating with host is described fully in the following pages. The basic architecture is shown below.

| Microphone Activation Sensor Functions e.g. SetLedLevel etc., | | | | | | |
|--|-----|--|--|--|--|--|
| | ΑΡΙ | | | | | |
| HIDAPI | | | | | | |

With this approach the developer does not need to worry about the communication at low level.

The library can be ported to your specific platform if required.

Currently the library has been tested on Windows and Linux (Ubuntu) platform.



Device Communications and Message Format

This uses the ASCII/binary Message format described below. Every message that is sent from a host should be acknowledged with the control byte ACK (0x06). A retransmission should be initiated if an NAK (0x15) is received or if no acknowledge is received at all.

Message Formats

| A | Alpha character, 'A'-'Z' and 'a' - 'z' |
|---|--|
| С | Control character one byte in length. |
| Н | Hexadecimal characters, '0'-'9', 'A'-'F' |
| N | Numeric character, '0'-'9' |
| S | Special characters, entire character set 0x00 - 0xFF |

ASCII Message Format

| | Message Field | Туре | Length | Description | |
|---|---------------|------|--------|---|--|
| 1 | STX | С | 1 | Control character Start of Text = 0x02 | |
| 2 | Message Id | Н | 2 | Defines the type of message and format of the data field | |
| 3 | Data Length | Н | 2 | Hexadecimal value represented in ASCII defines the number of bytes in the data field. '00' to 'FF'. Maximum data field size is 256 bytes. | |
| 4 | Data Field | S | var | In binary format | |
| 5 | ЕТХ | С | 1 | Control character ETX = 0x03 | |
| 6 | LRC | С | 1 | Longitudinal Redundancy Check Digit, calculated on all previous data including STX | |



Controlling the Microphone Activation Sensor from the Host Computer

Message Definitions and Error Codes

Here is a general table describing the message lds, more detailed descriptions for each message ld follows. When a message is one way only, the Message ld. is the same for both the message and response.

| ID. | Message | Description |
|-----|-----------------------|---|
| 01 | GetDeviceStatus | Retrieves device status incl: Serial Number, Tamper Status, Firmware Version |
| 02 | SetSymbolStatus | Set the illumination, where 0 is off and 1-9 is on, $(1 - 9 = \text{LED brightness})$ |
| 03 | SetCurrentTable | Set the code table to be used : either default = 0 or customised = 1 |
| 04 | LoadCodeTable | Loads a new customised code table |
| 05 | WriteDefaultToFlash | Writes changed values to Flash |
| 06 | ResetToFactoryDefault | Resets Microphone Activation Sensor to factory defaults |

Error Code

Every response message contains one of the following error codes:

| 00 | No error |
|----|-----------------------------------|
| 01 | Command not recognized |
| 02 | Command not support at this stage |
| 03 | Parameter not supported |
| 04 | Hardware fault |



List of Messages

| ID | Name | Description |
|----|--------------------------|---|
| 01 | Device Status Request | Retrieves device status incl: Serial Number, Tamper Status, Firmware Version |
| 02 | Set Symbol Status | Set the illumination, where 0 is off and 1-9 is on, ($1 - 9 = LED$ brightness level) |
| 03 | Set Current Table | Set the code table to be used : either default = 0 or customised = 1 |
| 04 | Load Code Table | Loads a new customised code table |
| 05 | Write default to flash | Writes changed values to flash |
| 06 | Reset to factory default | Reset device back to factory default |



Device Status (01)

Host sends this message to request the status of the Activation Sensor

MAS Status Response

Secure device sends this message to Host in response to the Device Status message.

| | Data Field | Туре | Length | Description | | |
|----|--------------------------|------|--------|--|--|--|
| ec | Error Code | SH | 2 | | | |
| Lb | Symbol LED Brightness | SN | 1 | Value (0 – 9) | | |
| JI | Ambient configuration | SN | 1 | 0 – disable 1 - enable | | |
| Kt | Keypad Table | SN | 1 | 0 – Default Table 1 – Alternate Table 2 – Customised Table | | |
| Kc | Keycode | SH | 12 | Customised keycode for each key | | |
| fw | Firmware Version | ANS | 20 | Left justified, if Firmware Version is less than 20 then just add enough spaces after the Firmware Version until this field is completed, for instance, "123456" becomes: "123456 " | | |
| sn | Serial Number | ANS | 8 | Serial number of device. | | |

Host sends this message to request information from the AUDIOCOM

| Host Device | AUDIOCOM |
|-------------|-------------------------------|
| [01] | [01][ec][l][kt][8*kc][fw][sn] |



Set Symbol Status (02)

Host sends this message to control brightness of LEDs

| | Data Field | Туре | Length | th Description | |
|---|----------------|------|--------|-------------------------------|--|
| 1 | LED brightness | SN | 1 | 0 – off, 1-9 Brightness level | |
| | | | | | |

LED Brightness Command Response

| | Data Field | Туре | Length | Description |
|----|------------|------|--------|-------------|
| ec | Error Code | н | 2 | |

| Host Device | | AUDIOCOM |
|-------------|---------------------|----------|
| [02][lb] | $ \longrightarrow $ | |
| | | [02][ec] |

Note: LED brightness of 0 value indicates LEDs are off

LED brightness of 9 value indicates full brightness



Keypad Table Command (03)

Host sends this message to set code table to be used.

| | Data Field | Туре | Length | Description | |
|---|------------|------|--------|--|--|
| 1 | Code Table | SN | 1 | 0 – Default Table 1 – Alternate Table 2 – Customised Table | |

Keypad Command & Response

| | Data Field | Туре | Length | Description |
|----|------------|------|--------|-------------|
| ec | Error Code | н | 2 | |

| Host Device | AudioCom |
|-------------|----------|
| [07][bp] | |
| | [07][ec] |



Load New Key Code Table Command (04)

Host sends this message to Load New Code Table

| | Data Field | Туре | Length | Description |
|---|---------------------|------|--------|-----------------|
| 1 | Load New Code Table | SH | 12 | Key Code Table: |

Load New Table Command & Response

| | Data Field | Туре | Length | Description |
|----|------------|------|--------|-------------|
| ec | Error Code | Н | 2 | |

| Host Device | AudioCom |
|------------------------|----------|
| [05][It][8 scan codes] | |
| | [05][ec] |

Note: Length is always 8,

Format of table is as follows:

<modifier for key 1><code for Key 1><modifier for key 2><Code for Key 2>.....etc

The code table is specified in the user manual together with the modifier code. For example to program the following for 4 way :

Key 1 – A

Key 2 – a

Key 3 – 9

Key 4 - !

<0xE1><0x04><0x00><0x04><0x00><0x26><0xE5><0x1E>

Note: 8 bytes must be sent, for unused key code pad the values with 0x00.

Note: For shift modifiers there is a left and right modifiers value defined. So we can use 0xE1 – Left Shift and 0xE5 – Right shift. Similarly there is left and right Alt



Microphone Activation Sensor Using the API

Reserved (06)



Keypad Table Command (07)

Host sends this message to set code table to be used.

| | Data Field | Туре | Length | Description | |
|---|------------|------|--------|--|--|
| 1 | Code Table | SN | 1 | 0 – Default Table 1 – Alternate Table 2 – Customised Table | |

Keypad Command & Response

| | Data Field | Туре | Length | Description |
|----|------------|------|--------|-------------|
| ec | Error Code | н | 2 | |

| Host Device | AudioCom |
|-------------|----------|
| [07][bp] | |
| | [07][ec] |



Write Config Data To Flash command (05)

Host sends this command to request the A Activation Sensor to write the configuration data from RAM to FLASH.

This command has no data associated with it.

RAM to FLASH Command & Response

| | Data Field | Туре | Length | Description |
|----|------------|------|--------|-------------|
| ec | Error Code | Н | 2 | |

| Host Device | AUDIOCOM-VOIP |
|-------------|---------------|
| [09] | [09][ec] |



Reset To Factory Default command (06)

Host sends this command to request the Activation Sensor to reset parameters back to factory default.

This command has no data associated with it.

Reset To Factory Default Command & Response

| | Data Field | Туре | Length | Description |
|----|------------|------|--------|-------------|
| ec | Error Code | Н | 2 | |

| Host Device | | AUDIOCOM |
|-------------|---|----------|
| [10] | | [40][1 |
| | < | |

Change History

| Instructions for | Date | Version | <u>Details</u> |
|------------------|-----------|---------|----------------|
| API | 15 Aug 24 | 1.0 | First Release |
| | | | |

| API | Date | <u>Version</u> | Details |
|-----|----------|----------------|---------------|
| | 5 Apr 19 | 1.0 | First Release |