



# WMI Interface for Intel<sup>®</sup> NUC Products

## WMI Specification

---

June 2020  
Version 1.0

## Revision History

Version	Description	Date
1.0	First release	04 June 2020

### Disclaimer

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL® PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER, AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT. UNLESS OTHERWISE AGREED IN WRITING BY INTEL, THE INTEL PRODUCTS ARE NOT DESIGNED NOR INTENDED FOR ANY APPLICATION IN WHICH THE FAILURE OF THE INTEL PRODUCT COULD CREATE A SITUATION WHERE PERSONAL INJURY OR DEATH MAY OCCUR.

All products referred to in this document are evaluated as Information Technology Equipment (I.T.E.) for installation in homes, offices, schools, computer rooms, and similar locations. The suitability of this product for other PC or embedded non-PC applications or other environments, such as medical, industrial, alarm systems, test equipment, etc. may not be supported without further evaluation by Intel.

Intel Corporation may have patents or pending patent applications, trademarks, copyrights, or other intellectual property rights that relate to the presented subject matter. The furnishing of documents and other materials and information does not provide any license, express or implied, by estoppel or otherwise, to any such patents, trademarks, copyrights, or other intellectual property rights.

Intel may make changes to specifications and product descriptions at any time, without notice.

Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined." Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them.

Contact your local Intel sales office or your distributor to obtain the latest specifications before placing your product order.

Intel, the Intel logo and Intel Core are trademarks of Intel Corporation in the U.S. and/or other countries.

\* Other names and brands may be claimed as the property of others.

Copyright © 2020 Intel Corporation. All rights reserved.

# Table of Contents

<b>Revision History .....</b>	<b>2</b>
<b>1. Introduction .....</b>	<b>4</b>
<b>2. Specification of HDMI CEC Control .....</b>	<b>4</b>
2.1 Get Physical/Logical address on the designated HDMI Port via CEC .....	4
2.2 Get EDID information Function on the designated HDMI Port .....	6
2.3 Get HPD_IN/HPD_OUT pin status on the designated HDMI Port .....	6
2.4 System HDMI Port Property .....	7
2.5 Set Display Device Power On/Off .....	8
2.6 Query Display Device Power status .....	8
2.7 Switch the Display Device HDMI input to whichever the Intel NUC is connected to .....	9
2.8 Display Device Audio Volume mute/unmute/increase/decrease .....	10
2.9 CEC Message Passing Through .....	10
<b>3. Specification of USB Port Power Control.....</b>	<b>13</b>
3.1 Get USB Power Status Function .....	13
3.2 Set USB Power Status Function.....	14
<b>4. Specification of Display Emulation Control.....</b>	<b>15</b>
4.1 Get Display Emulation Mode .....	15
4.2 Query EC EDID Data for the designated HDMI Port .....	15
4.3 System HDMI Port Property .....	17
<b>Appendix A – Examples of HDMI CEC command usage.....</b>	<b>18</b>
Switch the Display Device's HDMI input to the one the Intel NUC is connected to .....	18
Display Device Audio Volume mute/unmute/increase/decrease.....	18
CEC Message Passing Through .....	18
<b>Appendix C – Error Code Definition .....</b>	<b>21</b>

## 1. Introduction

This document provides a specification for the WMI interface calls in select Intel® NUC products facilitating the query and control of various features from within an operating system, such as HDMI\* CEC, USB port power and display emulation.

To locate WMI object through WMI explorer or programmed code, BIOS provides a MOF (Managed Object Format) in the ASL code.

1. WMI Object Name: "CISD\_WMI"
2. GUID: 8C5DA44C-CDC3-46b3-8619-4E26D34390B7

Reference:

More information about WMI and ACPI can be found at:

[https://msdn.microsoft.com/enus/library/windows/hardware/dn614028\(v=vs.85\).aspx](https://msdn.microsoft.com/enus/library/windows/hardware/dn614028(v=vs.85).aspx)

## 2. Specification of HDMI CEC Control

### 2.1 Get Physical/Logical address on the designated HDMI Port via CEC

		Description				
Control method		WMAA				
Arg 0		Instance				
Arg 1		Method ID (201h)				
Arg 2 Input Parameter	Byte 0	HDMI Port number  01h – HDMI Port 1 in the system 02h – HDMI Port 2 in the system  .....  0nh – HDMI Port n in the system				
	Byte 1	Reserved				
	Byte 2	Reserved				
	Byte 3	Reserved				
Return Value	Byte 0	Return Code – Refers to Appendix C				
	Byte 1	HDMI Physical address with the format of w.x.y.z.				
	Byte 2	<table><tr><td>Byte 1</td><td>Byte 2</td></tr><tr><td>wx</td><td>yz</td></tr></table>	Byte 1	Byte 2	wx	yz
	Byte 1	Byte 2				
wx	yz					
Byte 3	Return HDMI Logical address of the device connected on the HDMI port					

Logical Addresses

Address	Device
0	TV
1	Recording Device 1
2	Recording Device 2
3	Tuner 1
4	Playback Device 1
5	Audio System
6	Tuner 2
7	Tuner 3
8	Playback Device 2
9	Recording Device 3
10	Tuner 4
11	Playback Device 3
12	Reserved
13	Reserved
14	Specific Use
15	Unregistered (as Initiator address) Broadcast (as Destination address)

## 2.2 Get EDID information Function on the designated HDMI Port

		Description
Control Method		WMAA
Arg 0		Instance
Arg 1		Method ID (202h)
Arg 2 Input Parameter	Byte 0	HDMI Port number 01h – HDMI Port 1 in the system 02h – HDMI Port 2 in the system ..... 0nh – HDMI Port n in the system
	Byte 1	Reserved
	Byte 2	Reserved
	Byte 3	Reserved
Return Value	Byte 0 ~ Byte 255	If something error, EC does not return EDID Data Byte[0] = Return Code – Refers to Appendix C Byte[255:1] = 0x00 Else EDID table

## 2.3 Get HPD\_IN/HPD\_OUT pin status on the designated HDMI Port

		Description
Control Method		WMAA
Arg 0		Instance
Arg 1		Method ID (203h)
Arg 2 Input Parameter	Byte 0	HDMI Port number 01h – HDMI Port 1 in the system 02h – HDMI Port 2 in the system ..... 0nh – HDMI Port n in the system
	Byte 1	Reserved
	Byte 2	Reserved
	Byte 3	Reserved
Return Value	Byte 0	Return Code – Refers to Appendix C

		Description
	Byte 1	HPD_IN/HPD_OUT Status xxxxyyyyb xxxxb – HPD_IN (0000b → disabled; 0001b → enabled) yyyyb – HPD_OUT (0000b → disabled; 0001b → enabled)
	Byte 2	
	Byte 3	

## 2.4 System HDMI Port Property

		Description
Control Method		WMAA
Arg 0		Instance
Arg 1		Method ID (204h)
Arg 2 Input Parameter	Byte 0	Function Number 01h – Return the system HDMI Port Property
	Byte 1	Reserved
	Byte 2	Reserved
	Byte 3	Reserved
Return Value	Byte 0	Return Code – Refers to Appendix C
	Byte 1	The number of the HDMI ports on the system
	Byte 2	The bit map of the HDMI ports in the system that has display device connected 00000001b → HDMI Port 1 connected 00000010b → HDMI Port 2 connected 00000100b → HDMI Port 3 connected .... 10000000b → HDMI Port 8 connected
	Byte 3	The bit map of the HDMI ports in the system supporting CEC control 00000001b → HDMI Port 1 supports CEC control 00000010b → HDMI Port 2 supports CEC control 00000100b → HDMI Port 3 supports CEC control .... 10000000b → HDMI Port 8 supports CEC control

## 2.5 Set Display Device Power On/Off

		Description
Control Method		WMAA
Arg 0		Instance
Arg 1		Method ID (205h)
Arg 2 Input Parameter	Byte 0	HDMI Port Number 01h – HDMI Port 1 in the system 02h – HDMI Port 2 in the system ..... 0nh – HDMI Port n in the system
	Byte 1	Display Device Power Setting 01h – Display Device Power On 02h – Display Device Power Off
	Byte 2	Reserved
	Byte 3	Reserved
Return Value	Byte 0	Return Code – Refers to Appendix C
	Byte 1	Reserved
	Byte 2	Reserved
	Byte 3	Reserved

## 2.6 Query Display Device Power status

		Description
Control Method		WMAA
Arg 0		Instance
Arg 1		Method ID (206h)
Arg 2 Input Parameter	Byte 0	HDMI Port Number 01h – HDMI Port 1 02h – HDMI Port 2 ..... 0nh – HDMI Port n
	Byte 1	Reserved



		Description
Return Value	Byte 2	Reserved
	Byte 3	Reserved
	Byte 0	Return Code – Refers to Appendix C
	Byte 1	Target Display Device Power Status 00h – Power Off 01h – Power On
	Byte 2	Reserved
	Byte 3	Reserved

## 2.7 Switch the Display Device HDMI input to whichever the Intel NUC is connected to

		Description
Control Method		WMAA
Arg 0		Instance
Arg 1		Method ID (207h)
Arg 2 Input Parameter	Byte 0	HDMI Port Number 01h – HDMI Port 1 in the system 02h – HDMI Port 2 in the system ..... 0nh – HDMI Port n in the system
	Byte 1	Reserved
	Byte 2	Reserved
	Byte 3	Reserved
Return Value	Byte 0	Return Code – Refers to Appendix C
	Byte 1	Reserved
	Byte 2	Reserved
	Byte 3	Reserved

## 2.8 Display Device Audio Volume mute/unmute/increase/decrease

		Description
Control Method		WMAA
Arg 0		Instance
Arg 1		Method ID (208h)
Arg 2 Input Parameter	Byte 0	HDMI Port Number 01h – HDMI Port 0 in the system 02h – HDMI Port 1 in the system ..... 0nh – HDMI Port (n – 1) in the system
	Byte 1	Command 01h – Mute/Unmute Toggle 02h – Volume Up 03h – Volume Down
	Byte 2	Reserved
	Byte 3	Reserved
Return Value	Byte 0	Return Code – Refers to Appendix C
	Byte 1	Reserved
	Byte 2	Reserved
	Byte 3	Reserved

## 2.9 CEC Message Passing Through

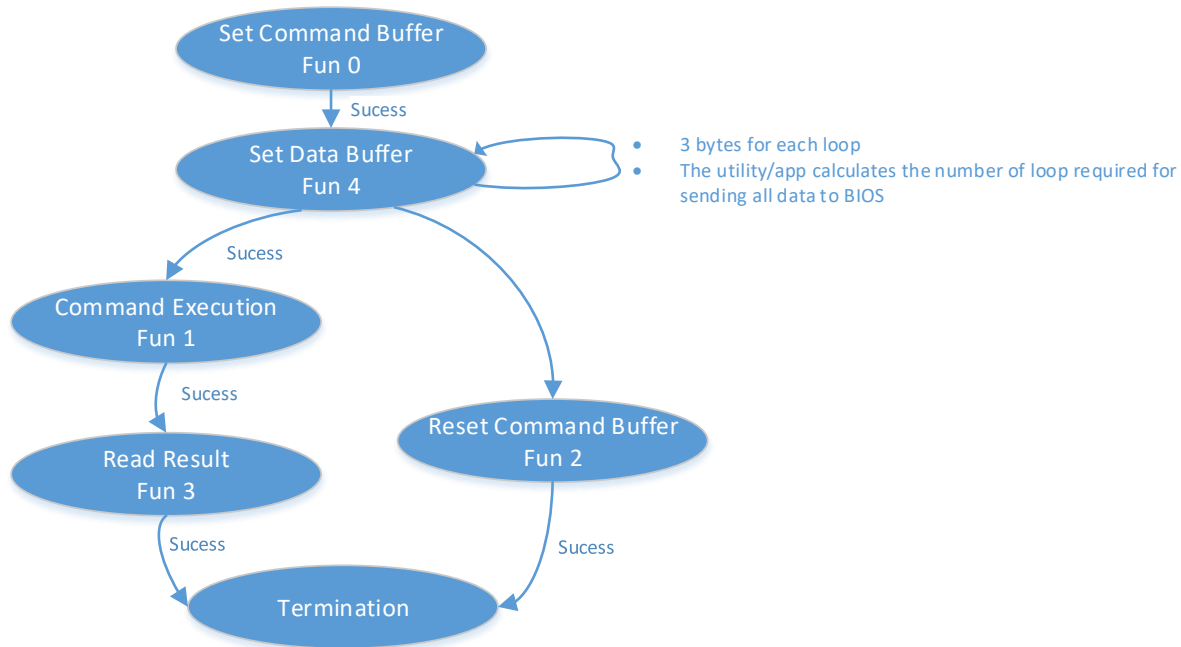
		Description
Control Method		WMAA
Arg 0		Instance
Arg 1		Method ID (209h)
Arg 2 Input Parameter	Byte 0	Fun(0): Set Command Buffer Fun(1): Execute Command Buffer Fun(2) <sup>1</sup> : Reset Command Buffer Fun(3): Read Result Fun(4): Set Data Buffer of the Command
	Byte 1	Fun(0): HDMI Port number

		Description
		01h – HDMI Port 1 in the system 02h – HDMI Port 2 in the system ..... 0nh – HDMI Port n in the system Fun(1/2/3): N/A Fun(4): Data Buffer Byte [ 3 x n + 0], n = the index of the iteration
	Byte 2	Fun(0): CEC Opcode Fun(1/2/3): N/A Fun(4): Data Buffer Byte [ 3 x n + 1], n = the index of the iteration
	Byte 3	Fun(0): Number of bytes of parameters for the command Fun(1/2/3): N/A Fun(4): Data Buffer Byte [ 3 x n + 2], n = the index of the iteration
Return Value	Byte 0 ~ 255	Fun(0/1/2/4): Byte[0] = Return Code – Refers to Appendix C Fun(3): Byte[0] = Return Code – Refers to Appendix C Byte[n:1] = Return Value

**Note:**

This function is used for resetting the command buffer in BIOS to the initial emptied state

Method ID 209h is a CEC message passing through function call which passes the CEC command along with the required parameters to the CEC callback function. The utility can use this function call to execute the CEC message that is not supported in the predefined function call in the WMI interface.



CEC message passing through function call is split into separate parts: Set Command Buffer, Set Data Buffer, and Read Result. The high-level flow chart is illustrated in the above diagram. The complete execution process must start with the Method ID 209h with Fun 0 for CMD buffer sending followed by Method ID 209h with Fun 4 for Input Data buffer sending. The times of Method ID 209h fun 4 execution depends on the number of bytes of the parameters of the command. Method Id 209h Fun 4 is able to send 3 bytes long data to BIOS at a time. For example, if the parameter of the CEC command are 17 bytes long, then the utility/app must execute Method ID 209h Fun 4 six times to send all 17 bytes long data to BIOS. Then, executes Method ID 209h Fun 3 to get the return value. An example in the Appendix demonstrates how this function call works. The example below shows the format of the Data buffer in bytes, using CEC - Vendor Command with ID as an example

Command: Vendor Command with ID, Opcode = 0xA0

Parameter 1: Vendor ID, 3 bytes long

Parameter 2: Vendor Specific Data: 14 bytes long

For example, parameter 1 = 0x123456, and parameter 2 = 0x00FF321A53649552467343817658

Parameter 1			Parameter 2													
12	34	56	00	FF	32	1A	53	64	95	52	46	73	43	81	76	58
method 209h fun 4 1st			method 209h fun 4 2nd			method 209h fun 4 3rd			method 209h fun 4 4th			method 209h fun 4 5th			method 209h fun 4 6th	
Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Byte 9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14	Byte 15	Byte 16

### 3. Specification of USB Port Power Control

#### 3.1 Get USB Power Status Function

		Description
Control Method		WMAA
Arg 0		Instance
Arg 1		Method ID (301h)
Arg 2 Input Parameter	Byte 0	Function Number 01h – Get power status of USB ports
	Byte 1	Reserved
	Byte 2	Reserved
	Byte 3	Reserved
Return Value	Byte 0	Return Code – Refers to Appendix C
	Byte 1	Current power state for each of external USB ports. Each of Bit presents the power on/off for each of USB port. "0" means power off and "1" means power on. Bit0: USB Port 0 Bit1: USB Port 1 Bit2: USB Port 2 Bit3: USB Port 3 Bit4: USB Port 4 ---
	Byte 2	Current power state for each of internal USB ports. Each of Bit present the power on/off for each of USB port. "0" means power off and "1" means power on. Bit0: Internal USB Port 1 Bit1: Internal USB Port 2 ---
	Byte 3	Current power state for each of USB charging ports. Each of Bit present the power on/off for each of USB port. "0" means power off and "1" means power on. Bit0: USB charging Port 1 ---

### 3.2 Set USB Power Status Function

		Description
Control Method		WMAA
Arg 0		Instance
Arg 1		Method ID (302h) Set USB Power Status Function
Arg 2 Input Parameter	Byte 0	Set power state for each of external USB ports <sup>1,2</sup> .  Each of Bit presents the power on/off for each of USB port. "0" means power off and "1" means power on.  Bit0: USB Port 0 Bit1: USB Port 1 Bit2: USB Port 2 Bit3: USB Port 3 Bit4: USB Port 4  ---
	Byte 1	Set power state for each of internal USB ports <sup>1,2</sup> .  Each of Bit present the power on/off for each of USB port. "0" means power off and "1" means power on.  Bit0: Internal USB Port 1 Bit1: Internal USB Port 2  ---
	Byte 2	Set power state for each of USB charging ports <sup>1,2</sup> .  Each of Bit present the power on/off for each of USB port. "0" means power off and "1" means power on.  Bit0: USB charging Port 1  ---
	Byte 3	Reserved
Return Value	Byte 0	Return Code – Refers to Appendix C
	Byte 1	Reserved
	Byte 2	Reserved
	Byte 3	Reserved

**Note:**

1. This function only applies the USB port power on/off change at runtime only. The USB port power on/off setting is not preserved across boot.
2. The USB port mapping for each bit is dependent upon the specific Intel NUC model being used.

## 4. Specification of Display Emulation Control

### 4.1 Get Display Emulation Mode

		Description
Control Method		WMAA
Arg 0		Instance
Arg 1		Method ID (501h)
Arg 2 Input Parameter	Byte 0	Function Number 01h – Query the Display Emulation mode configuration
	Byte 1	Reserved
	Byte 2	Reserved
	Byte 3	Reserved
Return Value	Byte 0	Return Code – Refers to Appendix C
	Byte 1	Display Emulation Mode 01h – Normal Mode 02h – Virtual Display Emulation 03h – Persistent Display Emulation
	Byte 2	Inconsistent Display Warning 01h: Block Boot 02h: Countdown Others: Reserved Note: This field is only valid when the byte 1 is 03h.
	Byte 3	Reserved

### 4.2 Query EC EDID Data for the designated HDMI Port

		Description
Control Method		WMAA
Arg 0		Instance
Arg 1		Method ID (503h)
Arg 2 Input Parameter	Byte 0	HDMI Port number 01h – HDMI Port 1 on the system 02h – HDMI Port 2 on the system .....

		Description
		Onh – HDMI Port n on the system
	Byte 1	Reserved
	Byte 2	Reserved
	Byte 3	Reserved
Return Value	Byte 0 ~ 255	<p>If something error, EC does not return EDID Data</p> <p>Byte[0] = Return Code – Refers to Appendix C</p> <p>Byte[255:1] = 0x00</p> <p>Else</p> <p>EC EDID data</p>



### 4.3 System HDMI Port Property

		Description
Control Method		WMAA
Arg 0		Instance
Arg 1		Method ID (504h)
Arg 2 Input Parameter	Byte 0	Function Number 01h – Return the system HDMI Port Property
	Byte 1	Reserved
	Byte 2	Reserved
	Byte 3	Reserved
Return Value	Byte 0	Return Code – Refers to Appendix C
	Byte 1	The number of the HDMI ports on the system
	Byte 2	The bit map of the HDMI ports in the system that has display device connected  00000001b → HDMI Port 1 connected 00000010b → HDMI Port 2 connected 00000100b → HDMI Port 3 connected .... 10000000b → HDMI Port 8 connected
	Byte 3	The bit map of the HDMI ports in the system supporting Display Emulation control  00000001b → HDMI Port 1 supports Display Emulation control 00000010b → HDMI Port 2 supports Display Emulation control 00000100b → HDMI Port 3 supports Display Emulation control .... 10000000b → HDMI Port 8 supports Display Emulation control

## Appendix A – Examples of HDMI CEC command usage

### *Switch the Display Device's HDMI input to the one the Intel NUC is connected to*

1. Executes Method ID 204h to get the HDMI port status on the system. Check section 2.4 for details.
2. Executes the Method ID 207h to switch the display device's HDMI input number.

ARG 1	ARG 2				Return Value			
	Byte 0	Byte 1	Byte 2	Byte 3	Byte 0	Byte 1	Byte 2	Byte 3
204h	01h	N/A	N/A	N/A	Return code	# of HDMI	Bitmap of connected HDMI Port	Bitmap of CEC capable Port
207h	HDMI Port #	N/A	N/A	N/A	Return code	N/A	N/A	N/A

### *Display Device Audio Volume mute/unmute/increase/decrease*

1. Executes Method ID 204h to get the HDMI port status on the system. Check section 2.4 details of the function
2. Method ID 208h is a predefined function call executing the display device's audio volume control. Based on the HDMI port check from Method ID 204h, and the user selection of the target display device via the UI, executes the Method ID 208h to execute the task.

ARG 1	ARG 2				Return Value			
	Byte 0	Byte 1	Byte 2	Byte 3	Byte 0	Byte 1	Byte 2	Byte 3
204h	N/A	N/A	N/A	N/A	Return code	# of HDMI	Bitmap of connected HDMI Port	Bitmap of CEC capable port
208h	HDMI Port #	Command 01h – Mute/Unmute Toggl 02h – Volume Up 03h – Volume Down	N/A	N/A	Return code	N/A	N/A	N/A

### *CEC Message Passing Through*

The below example does the same work as Method ID 209h does to achieve the result of the display device's audio volume control using the CEC Message Passing Through function instead of using the predefined function call.

1. Executes Method ID 204h to get the HDMI port status on the system. Check the section 2.4 to read the detail of the function

- Method ID 209h is a CEC Message Passing Through function call that simply passes the CEC command along with the required parameters to the EC CEC callback function. The utility can use this function call to execute the CEC message that is not supported in the predefined function call in the WMI interface.
- Be noted, the application itself needs to calculate the number of iterations of Method ID 209h function 4 must execute to send all data to BIOS. In this example, the input parameter only has one byte long. Therefore, it only needs Method 209h fun 4 executed once for data buffer sending, and function (0) for command buffer sending. Read Section 3.9 to learn the steps in detail of dealing with data buffer manipulation if the input data buffer of the HDMI CEC command is larger than 3 bytes.
- After sending command buffer and the required data buffer to BIOS, Method ID 209h fun 1 must be executed to inform BIOS to pass the data package to EC to start the HDMI CEC command execution
- Execute Method ID 209h fun 3 to read the result. Be noted, the application itself needs to know the number of bytes of the expected result. In this example, there is no return value.

ARG 1	ARG 2				Return Value			
	Byte 0	Byte 1	Byte 2	Byte 3	Byte 0	Byte 1	Byte 2	Byte 3
204h	N/A	N/A	N/A	N/A	Return code	# of HDMI	Bitmap of connected HDMI Port	Bitmap of CEC capable Port

ARG 1	ARG 2				Return Value	
	Byte 0	Byte 1	Byte 2	Byte 3	Byte 0	Byte 1 ~ 255
209h	Function	HDMI Port #	CEC Opcode	Number of bytes of parameters	Return code	Return Value
	00h	01h	44h	01h	00h	N/A

ARG 1	ARG 2				Return Value	
	Byte 0	Byte 1	Byte 2	Byte 3	Byte 0	Byte 1 ~ 255
209h	Function	Data Buffer [3 x n + 0]	Data Buffer [3 x n + 1]	Data Buffer [3 x n + 2]	Return code	Return Value
	04h	Command 01h – Mute 02h – Unmute 03h – Volume Up 04h – Volume Down	N/A	N/A	00h	N/A

ARG 1	ARG 2				Return Value	
	Byte 0	Byte 1	Byte 2	Byte 3	Byte 0	Byte 1 ~ 255
209h	Function				Return code	Return Value

	01h	N/A	N/A	N/A	00h	N/A
--	-----	-----	-----	-----	-----	-----

ARG 1	ARG 2				Return Value	
	Byte 0	Byte 1	Byte 2	Byte 3	Byte 0	Byte 1 ~ 255
209h	Function				Return code	Return Value
	03h	N/A	N/A	N/A	00h	N/A

## Appendix C – Error Code Definition

Error Code	Description
00h	No Error
E1h	Function not supported
E2h	Undefined device
E3h	EC no respond
E4h	Invalid Parameter
E5h	Node busy. Command could not be executed because command processing resources are temporarily unavailable
E6h	Command execution failure. Parameter is illegal because destination device has been disabled or is unavailable
E7h	Invalid CEC Opcode
E8h	Data Buffer size is not enough
EFh	Unexpected error
Others	Reserved