Intel® High Definition Audio Specification
Document Change Notification

Date:	June 3, 2008		
Company:	Intel Corporation		
Address:	1900 Prairie City Rd.		
City:	Folsom	State:	CA
Country:	USA	Zip:	95630

Change Identification:	**DCN No: HDA036-A**
Change Revision:	1.0
Document Revision:	Intel® High Definition Audio 1.0

Title: Display Port Support and HDMI Miscellaneous Corrections

Brief description of the functional changes:

This DCN provides a set of addition and changes to the HD Audio specification that defines how the Display Port audio controls are being exposed. It also provides miscellaneous correction and clarifications on the HDMI verbs and parameters. Note that this DCN is built on top of the DCN No: HDA034-A, HDA035-A, and HDA039-A.

Definition Text Formatting:

xxx	Original text in existing specification or DCN released earlier.
yyy	New text inserted by this new DCN.
~~zzz~~	Deleted text introduced by this new DCN.

New Definitions:

7.1 Codec Architecture

The High Definition Audio Specification defines a complete codec architecture that is fully discoverable and configurable so as to allow a software driver to control all typical operations of any codec. While this architectural objective is immediately intended for audio codecs, it is intended that such a standard software driver model not be precluded for modems and other codec types ~~(e.g., HDMI, etc.)~~. This goal of the architecture does not imply a limitation on product differentiation or innovative use of technology. It does not restrict the actual implementation of a given function but rather defines how that function is discovered and controlled by the software function driver.

...

7.2.2 Function Groups

A *function group* is a collection of widgets which are all common to a single application/purpose and which are controlled by a single "Function Driver." A codec contains one or more function group(s). While it is possible for a codec to contain more than one function group of a given type, this would not be typical. Currently defined function groups are:

- Audio Function Group
- Vendor Specific Modem Function Group
- ~~HDMI Function Group (to be defined at a future time)~~

...

7.2.3.1 Audio Output Converter Widget

The Audio Output Converter Widget is primarily a DAC for analog converters or a digital sample formatter (e.g., for S/PDIF) for digital converters. Its input is always connected to the High Definition Audio Link interface in the codec, and its output will be available in the connection list of other widget(s), such as a Pin Widget. This widget may contain an optional output amplifier, or a processing node, as defined by its parameters (Figure 50). Its parameters also provide information on the capabilities of the DAC and whether this is a mono or stereo (1- or 2-channel) converter, or more than 2 channels. In order to save parameter space in an Audio Function Group incorporating several Audio Output Converter Widgets, the function group node may optionally contain defaults for many of the DAC and amplifier parameters; however, these defaults may be over-ridden with local parameters if necessary.

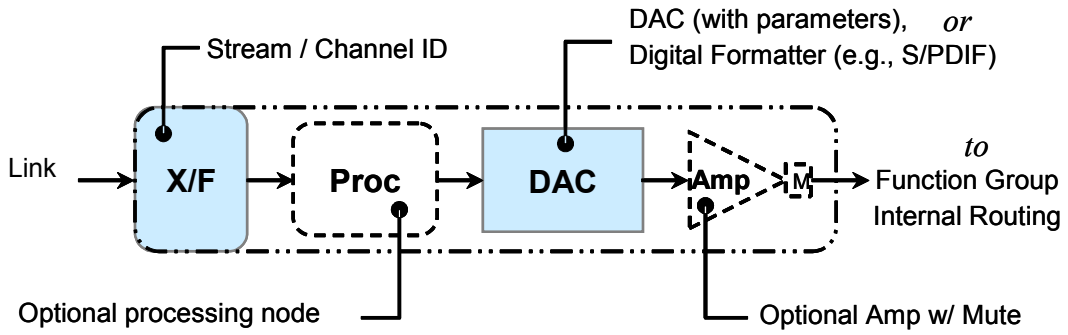

Figure 50. Audio Output Converter Widget

The Audio Output Converter Widget provides controls to access all its parametric configuration state, as well as to bind a stream and channel(s) on the Link to this converter. In the case of a 2-channel converter, only the "left" channel is specified; the "right" channel will automatically become the next larger channel number within the specified stream (see Section 7.3.3.11). In the case of a converter with more than 2 channels (e.g. HDMI or Display Port), the "first" channel is specified; with the total number of channels the converter should decode from the specified stream (see Section 7.3.3.35).

...

7.2.3.3 Pin Widget

The Pin Widget provides the external (analog or digital) connection for the audio and other function groups. A Pin Widget further includes those signals directly related to the external connections, such as jack sense and Vref control signals (Figure 52). However, GPIO pins are *not* identified as part of a Pin Widget but are a resource of the function group (see Sections 7.2.2.1 and 7.2.2.2). The Pin Widget's capabilities are highly parameterized defining optional support for:

- Input, output (or both), including the presence and capability of amplifier(s)
- Stereo or mono (1- or 2-channel), or more than 2 channels
- Plug (presence) detection
- Attached device impedance sensing
- VRef bias for microphone support

Every Pin Widget must contain a Configuration Default Register as defined in Section 7.3.3.31. In order to save parameter space in a function group incorporating several Pin Widgets, the function group node may optionally contain defaults for the amplifier parameters; however, these defaults may be over-ridden with local parameters if necessary.

The channel played on the external output pin (input to the Pin Widget) will be selected from its own input connection list; the channel on the external input pin will be available in the connection list of other widget(s), such as an Audio Input Converter Widget.

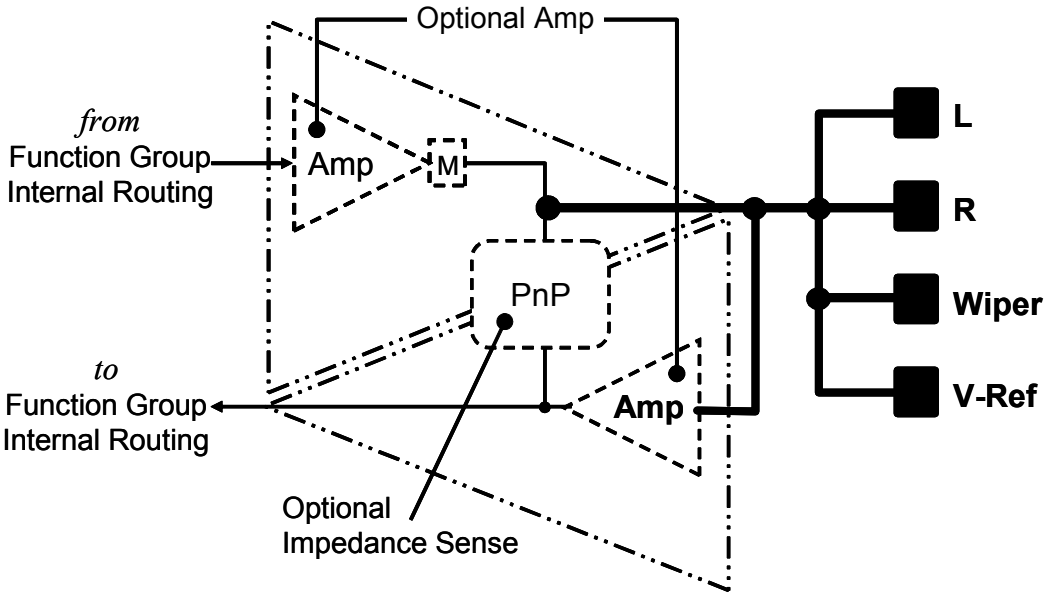

Figure 52. Pin Widget

The digital display connection which can render audio, such as HDMI or Display Port (DP), is classified under digital pin widget. They are more specifically called out as digital display pin widget in this specification.

▪ ▪ ▪

7.3.3.13 Pin Widget Control

Pin Widget Control controls several aspects of the Pin Widget.

Command Options:

Table 81. Enable VRef

	Verb ID	Payload (8 Bits)	Response (32 Bits)
Get	F07h	0	Bits 31:8 are 0 Bits 7:0 are PinCntl
Set	707h	Bits 7:0 are PinCntl	0

PinCntl format:

7	6	5	4:3	2	1:0
H-Phn Enable	Out Enable	In Enable	Rsvd	VRefEn[2]	VRefEn[1:0] / EPT

Figure 63. PinCntl Format

H-Phn Enable disables/enables a low impedance amplifier associated with the output. The value 1 enables the amp. Enabling a non-existent amp is ignored. For digital pin widgets, including digital display ~~HDMI~~, this control has no function.

Out Enable allows the output path of the Pin Widget to be shut off. The value 1 enables the path. Enabling a non-existent amp is ignored. For a digital display ~~HDMI~~ pin widget, disabling the output will cause the samples to no longer be sent to the digital display ~~HDMI S~~sink device, whilst o~~. O~~ther signaling may continue, such as clock recovery and other Info Frame packets.

In Enable allows the input path of the Pin Widget to be shut off. The value 1 enables the path.

VRefEn: Voltage Reference Enable controls the VRef signal(s) associated with the non digital Pin Widget. If more than one of the bits in the VRef[7:0] field of the Pin Capabilities parameter (Section 7.3.4.9) are non-zero, then this control allows the signal level to be selected.

The VRefEn field encoding selects one of the possible states for the VRef signal(s). If the value written to this control does not correspond to a supported value as defined in the Pin Capabilities parameter, the control must either retain the previous value or take the value of 000, which will put the control in a Hi-Z state and prevent damage to any attached components.

Table 82 enumerates the possible values for VRefEn which correlate to the values identified in the Pin Capabilities parameter (see Figure 85).

Table 82. VRefEn Values

VRefEn Encoding	VREF Signal Level
000b	Hi-Z
001b	50%

010b	Ground (0 V)
011b	*Reserved*
100b	80%
101b	100%
110b-111b	*Reserved*

EPT: Encoded Packet Type controls the packet type used to transmit the audio stream on the associated digital Pin Widget. Native audio packet type is always supported. If there are non native packet types supported as declared in the Pin Capabilities parameter (Section 7.3.4.9), then this control allows the encoded packet type to be selected.

The EPT field encoding selects one of the possible packet types for sending out on the digital Pin Widget. If the value written to this control does not correspond to a supported value as defined in the Pin Capabilities parameter, the control must either retain the previous value or take the value of 00, which will select the default native audio packet type.

Table 83 enumerates the possible values for EPT which correlates to the supported type identified in the Pin Capabilities parameter (see Figure 77).

Table 83. EPT Values

EPT Encoding	Description
00b	Native. For HDMI Pin Widget, it indicates Audio Sample Packet. For Display Port Pin Widget, it indicates Audio Stream Packet.
01b-10b	*Reserved*
11b	High Bit Rate. For HDMI Pin Widget, it indicates HBR Audio Stream Packet. For Display Port Widget, this is not supported.

Applies to:

- Pin Complex~~, both Digital (HDMI) and analog~~

▪ ▪ ▪

7.3.3.14.1 Intrinsic Unsolicited Responses

Intrinsic unsolicited responses are generated as a result of asynchronous hardware ~~H/W~~ events such as presence detect and hot plug (represented by ELD valid) events. These responses have a predefined sub tag of 0 and are enabled by the Unsolicited Responses Control verb.

The Unsolicited response for intrinsic events is defined as:

31:26	25:21	20:2	1	0
Tag	Sub Tag	*Reserved*	ELDV	PD

Figure 65. Intrinsic Unsolicited Response Format

Data Structure:

Table 85. Intrinsic Unsolicited Response Fields

Sub Tag	This field has a predefined value of 0 for intrinsic messages.
PD	Presence Detect: This bit reflects the present state of the Pin Sense – Presence Detect bit when the unsolicited response is triggered. ~~When this bit is set, sense measurement has changed on the pin widget and s~~Software can optionally use the pin sense control verb to determine the latest ~~current~~ pin sense data state. This bit implementation is only required for digital display pin widget. Non digital display pin widget is optional to implement this bit. ~~For analog pin widgets, this UR means that Presence Detect or Impedance has changed on the pin widget~~. ~~For digital pin widgets, including HDMI pin widgets, this means that presence detect (and optionally ELD valid bit) has changed.~~
ELDV	ELD Data valid: This bit reflects the present state of the Pin Sense – ELD Valid bit when the unsolicited response is triggered. Software can optionally use the pin sense control verb to determine the latest pin sense data state ~~When this bit is set, new ELD data has been written~~. This bit implementation is only required for digital display pin widget. ~~NOTE: Generation of this UR is gated by the Presence Detect bit, meaning that this UR can only be generated if the Presence Detect (PD) bit is already SET.~~

The codec adheres to the following rules for generating unsolicited responses triggered by intrinsic events:

- If a second UR is generated while the previous UR is still waiting to be sent then only the new UR is sent.
- In the case of a monitor plug-in event, the ELD content is populated before the presence detect bit has been set. Setting of the ELD valid bit in this case will not generate an (extra) UR because it is gated by PD. In such cases when the PD bit is also set, an Unsolicited Response is generated that contains both PD and ELDV bits.

▪ ▪ ▪

7.3.3.15 Pin Sense

The **Pin Sense** control returns the Presence Detect status, EDID-Like Data (ELD) Valid, and the impedance measurement of the device attached to the pin.

Some codecs may require that the impedance measurement be triggered by software; in that case, sending the Execute command will cause the impedance measurement to begin. The “Presence Detect” bit will always be accurate if that functionality is supported by the widget.

Note that the Pin Complex Widget may support the generation of an Unsolicited Response to indicate that the Sense Measurement (either the Presence Detect or the Impedance) value has changed, the generation of which implies that the measurement is complete.

Command Options:

Table 86. Pin Sense

	Verb ID	Payload (8 Bits)	Response (32 Bits)
Get	F09h	0	**For Analog pin widget:** Bit 31 is Presence Detect Bits 30:0 are Impedance Value

			For Digital pin widget: Bit 31 is Presence Detect Bit 30 is ELD Valid Bits 29:0 are 0
Execute	709h	**For Analog pin widget:** Right Chnl: bit 0 Rsvd: bits 1:7	0
		For Digital pin widget: Not Applicable	

Presence Detect is a bit indicating the state of the Presence Detect capability. A 1 indicates that there is "something" plugged into the jack associated with the Pin Complex. This bit will only be valid if the widget has Presence Detect capability as indicated by the "Presence Detect Capable" bit of the Pin Capabilities parameter (see Section 7.3.4.9).

EDID-Like Data (ELD) Valid is a bit indicating the state of the ELD memory. When the contents are valid ELD is set to 1 and cleared to zero when not valid.

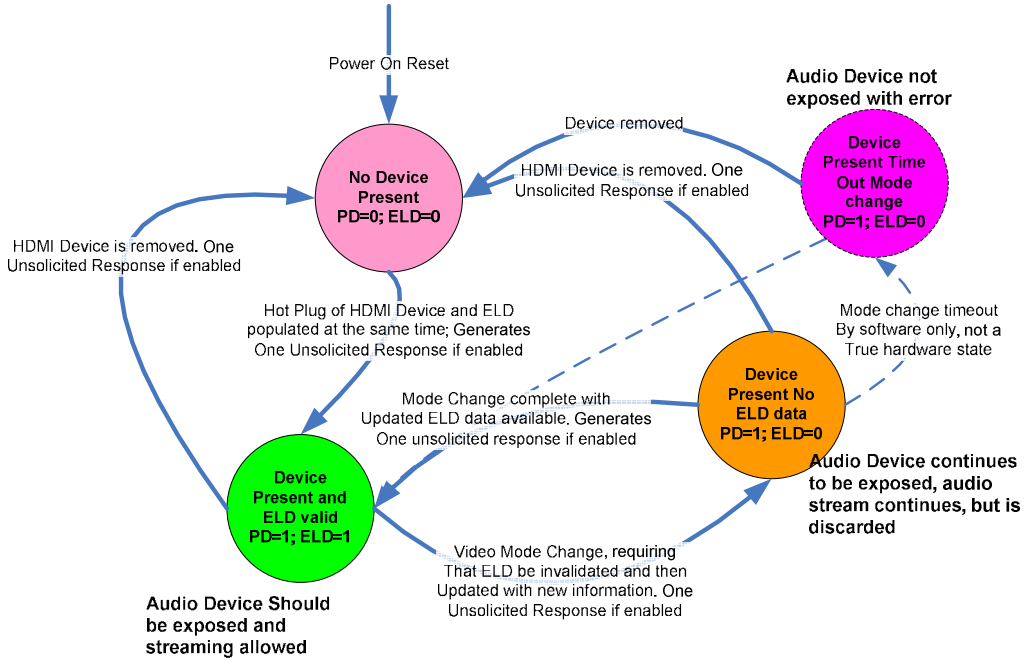

Figure 68. Presence Detect and ELD valid unsolicited responses flow for digital display ~~HDMI~~ codecs

Impedance returns the measured impedance of the widget. A returned value of 0x7FFF,FFFF (all 1's) indicates that a valid sense reading is not available, or the sense measurement is busy (refer to Section 7.3.1) if it has been recently triggered. This field is only valid if the widget has Sense capability as indicated by the "Impedance Sense Capable" bit of the Pin Capabilities parameter.

Right Chnl: Normally impedance sensing is done on the left channel or "tip" of the connector. However, Pin Widgets may optionally support sensing on the right channel or "ring" of the connector.

When this bit is 1, the impedance value is taken on the right channel if the Pin Widget supports this; if not supported, this bit is ignored. When this bit is 0, the left channel is sensed.

Applies to:

- Pin Complex ~~including Digital, HDMI and Analog~~

▪ ▪ ▪

7.3.3.31 Configuration Default

The **Configuration Default** is a 32-bit register required in each Pin Widget. It is used by software as an aid in determining the configuration of jacks and devices attached to the codec. At the time the codec is first powered on, this register is internally loaded with default values indicating the typical system use of this particular pin/jack. After this initial loading, it is completely codec opaque, and its state, including any software writes into the register, must be preserved across reset events. Its state need not be preserved across power level changes.

Command Options:

Table 102. Configuration Default

	Verb ID	Payload (8 Bits)	Response (32 Bits)
Get	F1Ch[1]	0	Config bits [31:0]
Set 1	71Ch	Config bits [7:0]	0
Set 2	71Dh	Config bits [15:8]	0
Set 3	71Eh	Config bits [23:16]	0
Set 4	71Fh	Config bits [31:24]	0

Data Structure:

The Configuration Default register is defined as shown in Figure 66.

31:30	29:24	23:20	19:16	15:12	11:8	7:4	3:0
Port Connectivity	Location	Default Device	Connection Type	Color	Misc	Default Association	Sequence

Figure 70. Configuration Data Structure

Port Connectivity[1:0] indicates the external connectivity of the Pin Complex. Software can use this value to know what Pin Complexes are connected to jacks, internal devices, or not connected at all. The encodings of the Port Connectivity field are defined in Table 101.

Location[5:0] indicates the physical location of the jack or device to which the pin complex is connected. This allows software to indicate, for instance, that the device is the "Front Panel Headphone Jack" as opposed to rear panel connections. The encodings of the Location field are defined in Table 102.

[1] The Verb Codes F1Dh, F1Eh, and F1Fh are reserved for the Configuration Default register and must not be reassigned to anything else. However, they need not be implemented since standard software drivers will not use them. If a codec elects to respond to these codes, the response must be identical in all respects to the response to Verb Code F1Ch.

The Location field is divided into two pieces, the upper bits [5:4] and the lower bits [3:0]. The upper bits [5:4] provide a gross location, such as “External” (on the primary system chassis, accessible to the user), “Internal” (on the motherboard, not accessible without opening the box), on a separate chassis (such as a mobile box), or other.

The lower bits [3:0] provide a geometric location, such as “Front,” “Left,” etc., or provide special encodings to indicate locations such as mobile lid mounted microphones. An “x” in Table 102 indicates a combination that software should support. While all combinations are permitted, they are not all likely or expected.

Default Device[3:0] indicates the intended use of the jack or device. This can indicate either the label on the jack or the device that is hardwired to the port, as with integrated speakers and the like. The encodings of the Default Device field are defined in Table 103.

Connection Type[3:0] indicates the type of physical connection, such as a 1/8-inch stereo jack or an optical digital connector, etc. Software can use this information to provide helpful user interface descriptions to the user or to modify reported codec capabilities based on the capabilities of the physical transport external to the codec. The encodings of the Connection Type field are defined in Table 104.

Color[3:0] indicates the color of the physical jack for use by software. Encodings for the Color field are defined in Table 105.

Misc[3:0] is a bit field used to indicate other information about the jack. Currently, only bit 0 is defined. If bit 0 is set, it indicates that the jack has no presence detect capability, so even if a Pin Complex indicates that the codec hardware supports the presence detect functionality on the jack, the external circuitry is not capable of supporting the functionality. The bit definitions for the Misc field are in Table 106.

Default Association and **Sequence** are used together by software to group Pin Complexes (and therefore jacks) together into functional blocks to support multichannel operation. Software may assume that all jacks with the same association number are intended to be grouped together, for instance to provide six channel analog output. The Default Association can also be used by software to prioritize resource allocation in constrained situations. Lower Default Association values would be higher in priority for resources such as processing nodes or Input and Output Converters. Note that this is the default association only, and software can override this value if required, in particular if the user provides additional information about the particular system configuration. A value of 0000b is reserved and should not be used. Software may interpret this value to indicate that the Pin Configuration data has not been properly initialized. A value of 1111b is a special value indicating that the Association has the lowest priority. Multiple different Pin Complexes may share this value, and each is intended to be exposed as independent devices.

Sequence indicates the order of the jacks in the association group. The lowest numbered jack in the association group should be assigned the lowest numbered channels in the stream, etc. The numbers need not be sequential within the group, only the order matters. Sequence numbers within a set of Default Associations must be unique.

Table 103. Port Connectivity

Value	Value
00b	The Port Complex is connected to a jack (1/8", ATAPI, etc.).
01b	No physical connection for Port.
10b	A fixed function device (integrated speaker, integrated mic, etc.) is attached.

Value	Value
11b	Both a jack and an internal device are attached. The Information provided in all other fields refers to the integrated device. The PD pin will reflect the status of the jack; the user will need to be queried to figure out what it is.

Table 104. Location

	Bits [3:0]										
Bits [5:4]	0h: N/A	1h: Rear	2h: Front	3h: Left	4h: Right	5h: Top	6h: Bottom	7h: Special	8h: Special	9h: Special	Ah-Fh: Reserved
00b: External on primary chassis	x	x	x	x	x	x	x	x (Rear panel)	x (Drive bay)		
01b: Internal	x							x (riser)	x (Digital Display ~~HDMI~~)	x (ATAPI)	
10b: Separate chassis	x	x	x	x	x	x	x				
11b: Other	x						x	x (Mobile Lid–Inside) (e.g., mic inside mobile lid)	x (Mobile Lid–Outside)		

▪ ▪ ▪

7.3.3.34 ~~HDMI~~ EDID-Like Data (ELD) Data

The audio software for the digital display ~~HDMI~~ codec will need information about the audio capabilities of an attached digital display ~~HDMI~~ sink device. This information is stored in the digital display ~~HDMI~~ sink device's EDID. Typically, the EDID flows through a graphics adapter to graphics software, so the graphics adapter HW will not have knowledge of the EDID contents.

To that end, a new mechanism is defined here for passing the digital display ~~HDMI~~ sink device's audio EDID information from the graphics software to the audio software. The data payload containing the audio information will be known as EDID-Like Data or ELD and will contain a subset of the digital display ~~HDMI~~ sink device's EDID information. The size and contents of the ELD buffer will be determined by the digital display ~~HDMI A~~audio codec manufacturer.

The ELD information will be valid if the digital display ~~HDMI~~ sink is attached and powered on and the ELD Valid bit is set. The Pin Widget that is associated with this digital display ~~HDMI~~ widget will report if the device is attached and that the ELD memory is populated and valid by reporting Presence Detect of 1 and ELD Valid of 1 to a Pin Sense control command. As with the Presence Detect bit, the changes to the ELD Valid bit can also result in the generation of unsolicited responses.

Command Options:

Table 111. ~~HDMI~~ ELD Data

	Verb ID	Payload (8 Bits)	Response (32 Bits)
Get	F2Fh	Byte offset into ELD memory	Bit 31 is ELD Valid indication Bits 30:8 are 0 Bits 7:0 are ELD data byte at specified offset into the ELD memory

Response Structure:

31	30:8	7:0
ELD Valid	*Reserved*	ELD Byte from memory

Figure 72. ELD Data Response Format

ELD Valid is a bit that indicates to software that the byte being returned is not valid.

ELD Byte [7:0] is the byte of configuration data specified by offset. For a non-existent ELD location, GET returns a value of 0. ~~Note that the byte index will auto-increment after a Get command is completed.~~

Applies to:

- Digital Display ~~HDMI~~ Pin Complex

7.3.3.34.1 ELD Memory Structure

The ELD memory structure is split into 3 blocks: header, baseline, and vendor defined. The header block contains the version and structure size information. The baseline block contains information about the digital display ~~HDMI~~ sink device standard features which OS class driver can understand. The vendor defined block contains information about any digital display ~~HDMI~~ sink device extended features that are specific to a particular vendor and may only be understood by a vendor specific driver.

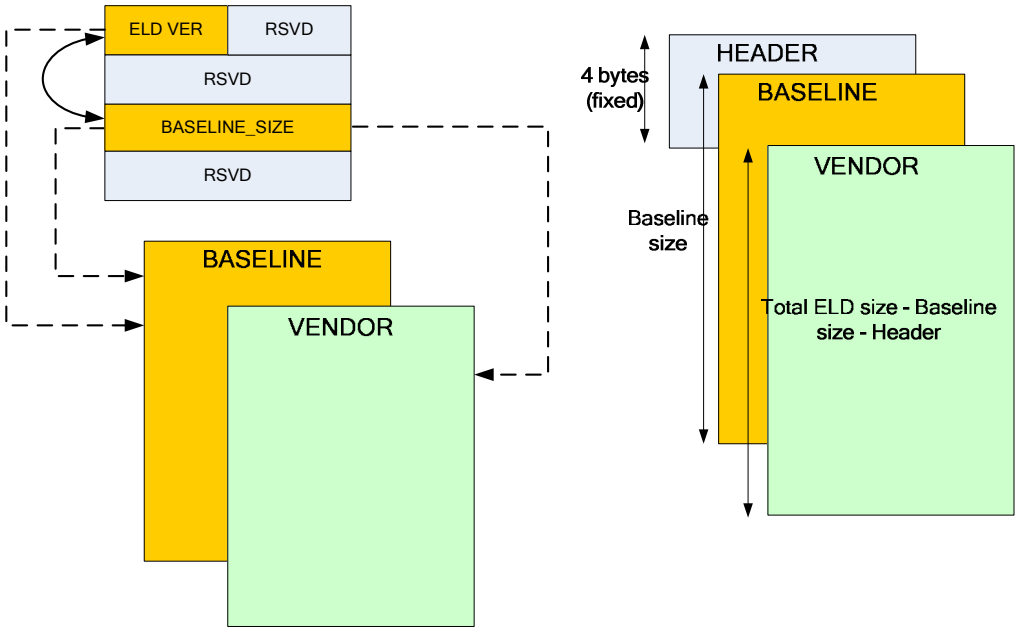

Figure 73. ELD Memory Structure

The size of the ELD memory structure is discovered by issuing ~~HDMI~~ DIP-Size command verb with ELD buffer size bit set. The header block is a fixed size of 4 bytes. The baseline block is variable size in multiple of 4 bytes, and its size is defined in the header block Baseline_ELD_Len field (in number of DWords). The vendor defined block is also variable size, and its size is the remaining bytes of the ELD memory structure, i.e. ELD buffer size – Header size of 4 bytes – (Baseline_ELD_Len * 4 bytes).

Header Block:

	Bit							
Byte offset into ELD memory	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	ELD_Ver					*Reserved*		
1	*Reserved*							
2	Baseline_ELD_Len							
3	*Reserved*							

Figure 74. Header Block of ELD Memory Structure

ELD_Ver[4:0] indicates the baseline ELD version number. Each version number has a fixed baseline ELD structure with a defined maximum number of bytes. It also indicates the CEA specification that the baseline ELD structure supports.

Table 112. ELD_Ver Encoding

Value	Description
00000b	*Reserved*

Value	Description
00001b	Indicates version 1, which is an obsolete ELD structure. Treated as reserved.
00010b	Indicates version 2, which supports CEA version 861-D or below. Maximum Baseline ELD size of 80 bytes (15 SAD count).
00011b – 11110b	*Reserved*
11111b	Indicates an ELD that has been partially populated through implementation specific mean of default programming before an external graphics driver is loaded. Only the field that is called out as "canned" field will be populated, and audio driver should ignore the non "canned" field.

Baseline_ELD_Len[7:0] indicates the length of the baseline structure in number of DW. There is a limit of the maximum length of the baseline structure supported per baseline ELD version number. It is required that the baseline structure length is equal or below the maximum number of bytes supported associated with the baseline ELD version.

Baseline Block:

	Bit							
Byte offset into ELD memory	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
4	CEA_EDID_Ver			MNL				
5	SAD_Count				Conn_Type		S_AI	HDCP
6	Aud_Synch_Delay							
7	*Rsvd*	RLRC	FLRC	RC	RLR	FC	LFE	FLR
8 to 15	Port_ID							
16 to 17	Manufacturer Name							
18 to 19	Product Code							
20 to 20 + MNL – 1	Monitor_Name_String							
20 + MNL to 20 + MNL + (3 * SAD_Count) – 1	CEA_SADs							
20 + MNL + (3 * SAD_Count) to 4 + Baseline_ELD_Len * 4 – 1	*Reserved*							

Figure 75. Baseline Block of ELD Memory Structure for ELD_Ver = 00010b

MNL[4:0] indicates the length of the monitor name string in number of bytes.

Table 113. MNL Encoding

Value	Description
00000b	Indicates the absence of a monitor name string.
00001b – 10000b	Indicates a monitor name string of length of 1 – 16 bytes.
10001b – 11111b	*Reserved*

CEA_EDID_Ver[2:0] indicates the CEA EDID Timing Extension version number of the digital display ~~HDMI~~ sink device supports. There is a limit of the latest CEA EDID Timing Extension version number supported per baseline ELD version number. It is required that the version number is equal to or less than the supported version associated with the baseline ELD version. OS class driver and other function driver shall support the highest version number and below for any given CEA EDID Timing Extension version number. Changes to CEA EDID Timing Extension version number may affect the content of the info frame.

Table 114. CEA_EDID_Ver Encoding

Value	Description
000b	Indicates no CEA EDID Timing Extension block present.
001b	Indicates CEA-861.
010b	Indicates CEA-861-A.
011b	Indicates CEA-861-B, C, or D.
100b – 111b	*Reserved*

HDCP indicates the support for HDCP. If set to 1, it indicates the receiver supports HDCP over the digital display ~~HDMI~~ link.

S_AI indicates the Supports_AI capability from the HDMI Vendor Specific Data Block. If set to 1, it indicates the sink supports at least one function that uses information carried by the ACP, ISRC1, or ISRC2 packets. A value of 0 indicates the sink does not support ACP, ISRC1, or ISRC2 packets. This bit is not applicable for Display Port, and should be 0.

Conn_Type[1:0] indicates the pin connection type of the device currently plugged in.

Table 115. Conn_Type Encoding

Value	Description
00b	Indicates a HDMI connection type.
01b	Indicates a Display Port connection type.
10~~01~~b – 11b	*Reserved*

▪ ▪ ▪

7.3.3.35 Converter Channel Count

The **Converter Channel Count** control is used by software to program the number of channels in the incoming stream that the converter must render, if the converter supports more than 2 channels.

Command Options:

Table 118. Converter Channel Count

	Verb ID	Payload (8 Bits)	Response (32 Bits)
Get	F2Dh	0	Bits 31:8 are 0 Bits 7:0 are Converter Channel Count ~~7:0~~
Set	72Dh	Converter Channel Count 7:0	

The Converter Channel Count control is used to specify the number of active channels in the audio stream. It is used in conjunction with the Channel value set in the Converter Stream, Channel control to specify which channels in an incoming audio stream are to be decoded by the codec. Converter Channel Count is 0-indexed as is Channel value in the Converter Stream, Channel control.

For the example below, allow S to be the Channel value in the Converter Stream, Channel control, and C to be the Converter Channel Count. Assume two channels intended for stereo playback are destined for a codec in a stream with greater than two total channels.

Assuming the first channel to be decoded is stream position 3:

S = 2

For Stereo, 2 streams are required:

C = 1

The codec would then output a stereo stream using the third and fourth channels in the incoming stream.

Note that when using the digital output converter pairing with HDMI Pin Widget configured for sending HBR packet type, the converter channel count must be programmed to 8. Per HDMI specification, these HBR audio streams will be packetized into 128 bit chunks when transmitted over HDMI. On the other hand HD Audio specification today only support up to 192 KHz frame rate, hence, these HBR audio streams will have to be transmitted as 8 channels of 16 bit data with 192 KHz frame rate (24.576 Mbps), or 8 channels of 16 bit data with 96 KHz frame rate (12.288 Mbps), over HD Audio Link. The mapping of the 8 channels of HBR audio stream over HD Audio Link to the 128 bit chunks HBR audio stream over HDMI will be carried out according to the sequence order of samples arriving.

Applies to:

- ~~HDMI~~ Output Converter

▪ ▪ ▪

7.3.3.36 ~~HDMI~~ Data Island Packet – Size info (~~HDMI~~ DIP-Size)

The **~~HDMI~~ DIP-Size** control is used to get the sizes of the various digital display ~~HDMI~~ packet buffers in the HW.

Command Options:

Table 119. ~~HDMI~~ DIP-Size

	Verb ID	Payload (8 Bits)	Response (32 Bits)
Get	F2Eh	Bits 7:4 are 0 Bit 3 is ELD buffer size Bits 2:0 are Packet Index (PI)	Bits 31:8 are 0 Bits 7:0 are 0 based size of buffer implemented for ~~HDMI~~ Data Island Packet PI or ELD

Data Structure:

When Bit 3 (ELD buffer size) is set to 0:

Table 120. DIP-Size Packet Index

PI Value	Definition
0h	Audio Infoframe
1h	GP1 – General Purpose 1
2h	GP2 – General Purpose 2
3h	GP3 – General Purpose 3
4h	GP4 – General Purpose 4
5h	GP5 – General Purpose 5
6h	GP6 – General Purpose 6
7h	GP7 – General Purpose 7

When Bit 3 (ELD Buffer size) is set to 1 then PI value is don't care and ELD buffer size is returned.

When audio data is transmitted over the digital display ~~HDMI~~ link like HDMI or Display Port, associated control information is transmitted as "Data Island Packets" over the link. Some of the packet types applicable to audio are ACP packets (for content protection), ISRC1/2 (for content info), Audio infoframe etc. For a detailed list of supported control packets and respective formats, please refer to the specific digital display ~~HDMI~~ specification. Please note that Audio Sample/Stream Packets are also DIPs like the others discussed above. This document uses DIP to refer to non-audio sample packets unless otherwise stated.

The HDMI specification defines a data island packet with a header of 4 bytes (3 bytes content + 1 byte ECC) and packet body of 32 bytes (28 bytes content and 4 bytes ECC). Display Port specification on the other hand defines a data island packet (secondary data packet) with header of 4 bytes protected by 4 bytes of parity, and data of theoretically up to 1024 bytes with each 16 bytes chunk of data protected by 4 bytes of parity. Note that the ECC or parity bytes are not present in the DIP content populated by software and are hardware generated.

The present definition of data island packets does not utilize the full packet size ~~all 28 bytes~~ in the body, giving hardware ~~h/w~~ designers an opportunity to optimize by implementing smaller buffers for each packet type.

The ~~**HDMI**~~ **DIP-Size** verbs allows software to query the size of the ~~HDMI~~ DIP buffers supported by the underlying hardware and make an informed decision based on the compatibility of software and hardware ~~h/w~~ capabilities.

Note that an HDMI transmitter must support at least 4 DIP packet buffers (one audio infoframe and three general purpose buffers). Display Port transmitters need only to support at least 1 DIP packet buffers (one audio infoframe).

Applies to:

- Digital Display ~~HDMI~~ Pin Complex

7.3.3.37 ~~HDMI~~ Data Island Packet – Index (~~HDMI~~ DIP-Index)

The ~~**HDMI**~~ **DIP-Index** control sets the packet buffer index and data byte index within that packet buffer.

Command Options:

Table 121. ~~HDMI~~ DIP-Index

	Verb ID	Payload (8 Bits)	Response (32 Bits)
Get	F30h	0	Bits 31:8 are 0 Bits 7:5 are currently set Packet Index Bits 4:0 are current Byte Index pointer in this packet
Set	730h	Bits 7:5 are Packet Index (PI) Bits 4:0 are Byte Index (0 based byte location)	0

Packet Index (PI) selects the target Data Island Packet buffer for subsequent **DIP-Data** and **DIP-XmitCtrl** control verbs.

Data Structure:

Table 122. DIP-Index Packet Index

PI Value	Definition
0h	Audio Infoframe
1h	GP1 – General Purpose 1
2h	GP2 – General Purpose 2
3h	GP3 – General Purpose 3
4h	GP4 – General Purpose 4
5h	GP5 – General Purpose 5
6h	GP6 – General Purpose 6
7h	GP7 – General Purpose 7

Byte Index sets the target byte offset within targeted packet buffer that is accessed in **DIP-Data** control verb.

Software **must** ensure that *packet buffer index* and *byte index* are set to correct values before attempting to access data in the buffers or change the transmit controls. Once the *packet index* and *byte index* are set, any subsequent accesses (read/write) automatically increment the byte-index by one byte after the operation. Upon reaching the maximum possible value (32 for data island packets), it wraps around to 0.

Applies to:

- Digital Display ~~HDMI~~ Pin Complex

7.3.3.38 ~~HDMI~~ Data Island Packet – Data (~~HDMI~~ DIP-Data)

The ~~**HDMI**~~ **DIP-Data** control accesses (Read/Write) a byte in the packet buffer previously set using **DIP-Index** control.

Command Options:

Table 123. ~~HDMI~~ DIP-Data

	Verb ID	Payload (8 Bits)	Response (32 Bits)
Get	F31h	0	Bits 31:8 are 0 Bits 7:0 are data byte located at target byte index in the target packet buffer
Set	731h	Data byte to be written in the target byte index in the target packet buffer	0

Index of byte accessed via this control is determined by current byte-index value for this packet buffer. This index is set to a new value upon receiving **DIP-Index** control verb and automatically incremented after a **DIP-Data** operation. Therefore first **DIP-Data** control after **DIP-Index** control verb targets the byte determined by the *byte-index* supplied as parameter in **DIP-Index** control verb and subsequent accesses (read/write) automatically target the next byte in that buffer.

Upon reaching the maximum value (32 for data island packets), the byte index automatically wraps around to 0.

It is software's responsibility to ensure that packet index and byte index are set to correct values before it starts sending **DIP-Data** control verbs for any packet buffer.

Note that the ECC or parity bytes are not present in the DIP content populated by **DIP-Data** verb, these bytes are hardware generated.

Applies to:

- Digital Display ~~HDMI~~ Pin Complex

7.3.3.39 ~~HDMI~~ Data Island Packet – Transmit Control (~~HDMI~~ DIP-XmitCtrl)

The ~~**HDMI**~~ **DIP – XmitCtrl** control sets the transmission controls for the currently indexed packet buffer.

It is software's responsibility to ensure that the infoframe index is set to point to appropriate buffer and the buffer content are set to correct values before it sets the transmission control. Once transmission control is set to anything but disabled, hardware ~~h/w~~ transmits the contents of indexed buffer at frequency determined by transmission control.

Command Options:

Table 124. ~~HDMI~~ DIP-XmitCtrl

	Verb ID	Payload (8 Bits)	Response (32 Bits)
Get	F32h	0	Bits 31:8 are 0 Bits 7:6 are transmit control for currently indexed packet buffer Bits 5:0 are 0
Set	732h	Bits 7:6 are transmit control for currently indexed packet buffer Bits 5:0 are 0	0

Data Structure:

Table 125. DIP-XmitCtrl Value

XmitControl Value	Definition
00	Disable Transmission
01	*Reserved*
10	Transmit once and then disable
11	Transmit at best effort

Applies to:

- Digital Display ~~HDMI~~ Pin Complex

7.3.3.40 Content Protection Control (CP_CONTROL)

Content Protection Control verb is used to set the state of content protection on the audio port. This control is only valid for pin widgets that have "CP Caps" bit set in the Audio widget capabilities response.

Command Options:

Table 126. Protection Control

	Verb ID	Payload (8 Bits)	Response (32 Bits)
Get	F33h	0	Bits 31:10 are 0 Bit 9 is CES - Current Encryption State Bit 8 is READY bit Bits 7:3 are UR sub tag for CP state Bit 2 is 0 Bits 1:0 are current CP request state
Set	733h	Bits 7:3 are UR sub tag for CP state Bit 2 is 0 Bits 1:0 are requested CP state	0

Table 127. Current Encryption State (CES)[2]

CES Value	Definition
0	Encryption OFF
1	Encryption ON

Table 128. Ready Indication

READY Value	Definition
0	Hardware ~~HAW~~ is NOT ready or in a state to accept CP control verbs. Audio driver should not send CP control verbs in this state.
1	Hardware ~~HAW~~ is in a state to accept CP control verbs to potentially change encryption state on the audio port.

[2] Note that Current Encryption State is always the true state of encryption on the hardware and is not buffered or software settable.

Table 129. Content Protection (CP) State

CP State	Definition
00	Don't care – State can be anything
01	*Reserved*
10	Protection OFF – Protection state must be NO-encryption to facilitate operations like recording
11	Protection ON – Protection state must be ON to facilitate rendering of protected content

Content Protection Control is used by audio driver to request setting of content protection state on the audio link. The CP state can be set to ON, OFF or DON'T CARE. Audio driver sets the CP state on the link as result of a request received from elements higher in the audio stack.

In case of HDCP, both video and audio share the same external link as well as the hardware ~~h/w~~ encryption logic. To arbitrate between audio and video side requests, the video driver acts as the master. Any request to change the CP state are actually sent to the video driver which makes decision to accept or deny the requests based on various factors such as current hardware ~~h/w~~ state, video stack state etc.

Typical flow to set CP state is as follows:

1. Video subsystem initializes the digital display ~~HDMI~~ link, including setting up the display hot plug registers as follows:
   a. Set the CP_READY bit – If the unsolicited responses are enabled, this causes an UR to be generated.
2. Audio software requests setting of a CP session and the request trickles down to the audio driver.
3. Audio software uses GET_CP_CONTROL verb to check if digital display ~~HDMI~~ codec HW (in this case video subsystem) is in a state to accept requests to change CP state or not. READY bit is the indicator of hardware's capability to accept CP state change commands.
4. If READY bit is SET in the response to GET_CP_Control verb, audio software sends a SET_CP_Control verb with intended CP state parameters to the audio codec.
   a. This verb also contains a sub tag field that would subsequently be used to identify the "notification unsolicited responses (UR)".
   b. Audio software may start its timeout for "hardware ~~H/W~~ acting on my request" event.
5. Audio hardware upon receiving this verb sends a default response (all 0's). This response identifies to audio driver that CP request was received successfully by the hardware ~~h/w~~.
   a. Audio hardware ~~H/W~~ clears the CP_READY bit – this is an indication to the audio software that hardware ~~h/w~~ CP state machine has accepted the request and has started working on it.
   b. While CP_READY = 0, the CES ~~HDCP status~~ bit is treated as invalid by audio software (as returned in the GET_CP_CONTROL verb).
   c. While CP_READY = 0, audio software doesn't send another CP command to the audio device.
6. CP_READY bit is also used by audio driver (in polling mode) to poll for the completion of request.
7. Video subsystem interacts with HDCP hardware ~~h/w~~ to start the process of changing the state of encryption (if needed).
8. Once HDCP hardware is done transitioning the state (or alternatively, video subsystem has determined that it doesn't need to change the CP state), video subsystem sets the READY bit.
   a. Note that when the CP state on the link changes, it doesn't generate any UR. The CP state change UR is gated by CP_READY bit.
9. When CP_READY bit gets set, audio hardware ~~h/w~~ sends a UR, if enabled (when SET_CP_CONTROL[sub tag] != 0 and UR are enabled).
   a. This UR is received by audio software.

10. Audio software would clear the timeout, if any set earlier.
11. Audio software can now use GET_CP_STATE verb to read the status of CP on the link.
12. The GET_CP_CONTROL[status] bit indicates the true "encryption" state of the HDCP hardware.

The same flow can be depicted graphically as follows:

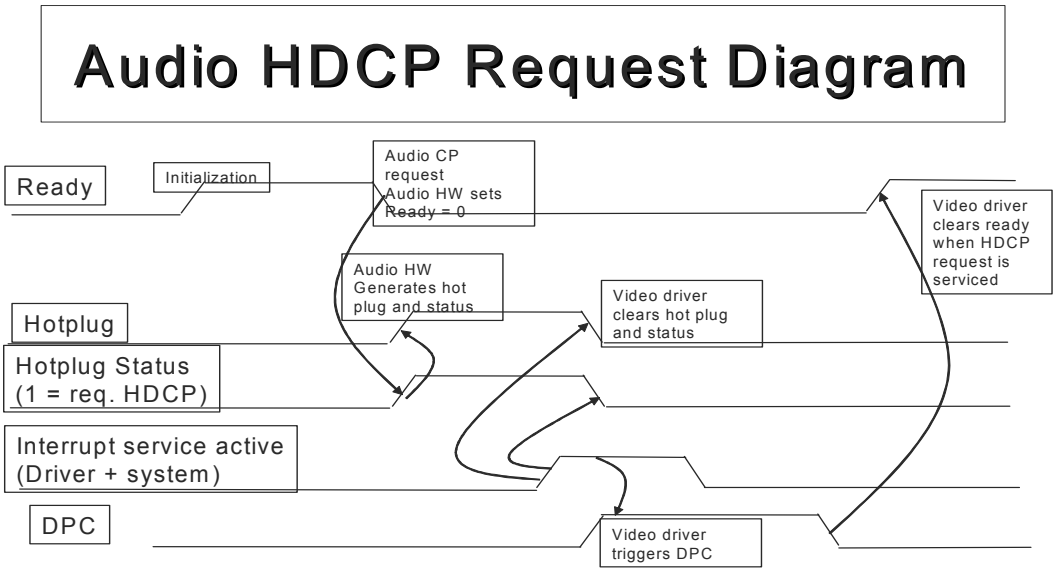

Figure 76. Audio HDCP Request Diagram

Applies to:

- Digital Display ~~HDMI~~ Pin Complex

7.3.3.41 Audio Sample Packet (ASP) ~~Converter~~ Channel ~~to HDMI Slot~~ Mapping

When audio data is transmitted through the audio software stack, the stream channels are placed consecutively (tightly packed) in memory in channel order defined by the operating system, e.g. for 5.1, 16-bit content, sample for all 6 channels are placed consecutively in memory without any holes; therefore each sample consumes 12 consecutive bytes in memory. When this content is transmitted on the HD Audio link, the stream maintains its format (all 12 bytes are transmitted together in the same order as in memory).

Also, during audio content's flows through the audio software stack, speaker masks define the relative location of channels in a given stream (e.g. first 2 bytes of a 12-byte set of samples in above example are always for Front Left channel).

When audio samples are transmitted on the digital display link such as HDMI or Display Port ~~link~~, they are packetized in the form of A~~a~~udio S~~s~~ample/Stream P~~p~~acket (ASP). HDMI and Display Port follow~~s~~ the definitions in CEA-861 spec for relative placement of channel data in the audio sample packet. For most cases, the mapping used in PC software is different than the mapping defined by the CEA-861 spec.

Note that in the PC environment, a common practice is to do speaker-fill – if rendered content's channel count is less than the number of speakers actually present in the system then software can map a given sample channel to multiple speakers to do "speaker fill". This allows all channels to play content. Therefore irrespective of the actual number of channels of audio content being rendered, all speakers play

some content. For example when a CD is played on a PC with 7.1 support, the audio output can appear on all 8 speakers.

In order to keep these options available for PC platforms that include these digital display ~~HDMI~~, ~~HDMI~~ the corresponding digital display pin widgets ~~converters~~ are required to support mapping of a given stream channel (converter output) to multiple ASP ~~HDMI~~ channels (slots) via this verb. Note that the power on reset and function reset values for the channel-slot mapping are as described below in the notes.

~~"Set HDMI – Channel to slot mapping"~~ ASP Channel Mapping verb is used to map an digital display audio channel ~~HDMI~~ slot in audio sample packet to corresponding sample channel coming on HD Audio link into an digital display ~~HDMI~~ converter.

Command Options:

Table 130. ~~Converter~~ ASP Channel ~~to HDMI Slot~~ Mapping

	Verb ID	Payload (8 Bits)	Response (32 Bits)
Get	F34h	Bits 7:4 are 0 Bits 3:0 are ASP ~~HDMI~~ slot number	Bits 31:8 are 0 Bits 7:4 are Converter Channel Number Bits 3:0 are ASP ~~HDMI~~ slot number
Set	734h	Bits 7:4 are Converter Channel Number Bits 3:0 are ASP ~~HDMI~~ slot number	0

Note:

- The power-on reset and function-reset values for the ASP slot ~~HDMI channel~~ to converter channel mapping are as follows:
  - Converter channel 0 mapped to ASP ~~HDMI~~ slot 0
  - Converter channel 1 mapped to ASP ~~HDMI~~ slot 1
  - Converter channel 2 mapped to ASP ~~HDMI~~ slot 3
  - Converter channel 3 mapped to ASP ~~HDMI~~ slot 2
  - Converter channel 4 mapped to ASP ~~HDMI~~ slot 4
  - Converter channel 5 mapped to ASP ~~HDMI~~ slot 5
  - Converter channel 6 mapped to ASP ~~HDMI~~ slot 6
  - Converter channel 7 mapped to ASP ~~HDMI~~ slot 7
- During setup of a stream, audio software is responsible for ensuring that all ASP ~~HDMI~~ channels (slots) are mapped to corresponding converter channels (outputs).
- Specifying a mapping value of '0xF' or a value greater than Converter Channel Count means that the ASP ~~HDMI~~ slot will not be driven with data (i.e. it is identified as unallocated in audio sample packet)
- The value for the ASP ~~HDMI~~ slot number field in this verb is 0-based, i.e. ASP ~~HDMI~~ slots are numbered from 0-7. This is different from the notation used in the CEA-861 spec where ASP ~~HDMI~~ slots are numbered from 1-8. CEA-861 ASP ~~HDMI~~ slot number 1 corresponds to ASP ~~HDMI~~ slot number 0 in this verb, slot number 2 in CEA-861 is noted as slot number 1 and so on.

Applies to:

- ~~HDMI Output Converter~~ Digital Display Pin Complex

▪ ▪ ▪

7.3.4.6 Audio Widget Capabilities

The Audio Capabilities control returns a set of bit fields describing the audio capabilities of the widget.

Parameter ID: 09h

Response Format:

31:24	23:20	19:16	15:13	12	11	10	9	8	7	6	5	4	3	2	1	0
Rsvd	Type	Delay	Chan Count Ext	CP Caps	L-R Swap	Power Cntrl	Digital	Conn List	Unsol Cap-able	Proc Wid-get	Stripe	Format Over-ride	Amp Param Over-ride	Out Amp Present	In Amp Present	Chan Count LSB (Stereo)

Figure 82. Audio Widget Capabilities Response Format

Type defines the functionality of the widget node. This is an enumerated list.

Table 132. Widget Type

Value	Type
0h	Audio Output
1h	Audio Input
2h	Audio Mixer
3h	Audio Selector
4h	Pin Complex~~; includes analog, digital and HDMI variants~~
5h	Power Widget
6h	Volume Knob Widget[3]
7h	Beep Generator Widget[4]
8h-Eh	*Reserved*
Fh	Vendor defined audio widget

▪ ▪ ▪

~~7.3.4.7.1 HDMI LPCM CAD~~

~~The Consolidated Audio Descriptor for LPCM content describes the current sample size/rate capability for the HDMI widget. This is potentially changed to reflect the dynamic configuration of the HDMI widget. This information will be valid if the HDMI sink is attached and powered on and the ELD Valid bit is set. Whenever the ELD bit toggles to 1, this field needs to be re-read by the software stack to figure out the updated LPCM capability of the HDMI widget.~~

[3] In the case of the Volume Knob Widget, none of the parameter bits [19:0] are used and may be omitted or set to 0. However, software assumes the capability of unsolicited responses and a connection list, as these are required by this widget type.

[4] In the case of the Beep Generator Widget, the only meaningful parameter bits are 2 ("Out Amp Present") and 3 ("Amp Param Override"). None of the other parameter bits are used and may be omitted or set to 0.

~~**Parameter ID: 20h**~~

~~**Response Format:**~~

~~31:30~~	~~29:28~~	~~27:24~~	~~23:20~~	~~19:18~~	~~17:14~~	~~13:10~~	~~9:8~~	~~7:4~~	~~3:0~~
~~44.1MS/ 44.1 Support~~	~~Bit Rate of 192 kHz~~			~~Bit Rate of 96 kHz~~			~~Bit Rate of 48 kHz~~		
	~~24b:20b Bit Rate Supported~~	~~Max Channels without CP ON~~	~~Max Channels with CP ON~~	~~24b:20b Bit Rate Supported~~	~~Max Channels without CP ON~~	~~Max Channels with CP ON~~	~~24b:20b Bit Rate Supported~~	~~Max Channels without CP ON~~	~~Max Channels with CP ON~~

~~**Figure 83. Audio Widget LPCM CAD Format**~~

~~**44.1 Support** bit indicate that the HDMI sink codec supports 44.1 kHz and~~

~~**44.1MS Support** bit indicates that the HDMI sink codec supports 44.1 kHz multiples~~

- ~~If this bit is 0 then 88.2 and 176.4 kHz are not supported by the sink~~
- ~~If this bit is 1 and 96 kHz fs is supported then 88.2 kHz is also supported~~
- ~~If this bit is 1 and 192 kHz fs is supported then 176.4 is also supported~~

...

7.3.4.9 Pin Capabilities

The Pin Capabilities parameter returns a bit field describing the capabilities of the Pin Complex Widget.

Parameter ID: 0Ch

Response Format:

31:28	27	26:25	24	~~26~~ 23:17	16	15:8	7	6	5	4	3	2	1	0
Rsvd	HBR	*Rsvd*	DP	*Rsvd*	EAPD Cap-able	VRef Control	HDMI	Balanced I/O Pins	Input Cap-able	Output Cap-able	Head-phone Drive Capable	Presence Detect Capable	Trigger Req'd	Impedance Sense Capable

Figure 85. Pin Capabilities Response Format

HBR (High Bit Rate) indicates the pin widget capability in sending out audio stream using High Bit Rate packet. Capable if set to 1. This bit is only applicable for HDMI pin widget.

DP (Display Port) indicates whether the Pin Complex Widget supports connection to a Display Port sink. Supported if set to 1. Note that it is possible for the pin widget to support more than one digital display connection type, e.g. HDMI and DP bit are both set to 1.

EAPD Capable indicates the codec has an EAPD pin and that this Pin Widget provides support for controlling that pin.

VRef Control[7:0] is a bit field used to indicate what voltages may be produced on the associated VRef pin(s). If all bits in the bit field are 0, then VRef generation is not supported by the Pin Complex. Also, if the Input Capable bit is a 0, then the VRef bit field has no meaning and all bits must be 0.

If the Output Capable bit and any bits in the VRef field are set, then bit 0 (Hi-Z) must also be set to indicate that the VRef signal can be turned off to support output devices.

Figure 87 describes the VRef bit field. A 1 in any position indicates that the associated signal level is supported. All values of VRef are specified as a percentage of the analog voltage rail, AVdd.

7:6	5	4	3	2	1	0
Rsvd	100%	80%	*Rsvd*	Ground	50%	Hi-Z

Figure 86. VRef Bit Field

HDMI indicates ~~that~~ whether the Pin Complex Widget supports connection to a HDMI s~~S~~ink. Supported if set to 1. Note that it is possible for the pin widget to support more than one digital display connection type, e.g. HDMI and DP bit are both set to 1.

▪ ▪ ▪

7.3.6 Required Parameter and Control Support

Table 134 specifies which parameters are required (R) for each specification-defined node. It also indicates optional (o) parameters which are used to declare the presence of optional features in the associated node. A shaded square in the table indicates that the subject parameter is not applicable to the subject node type. The squares marked with (a) indicate an alternative; the parameter is required in either the Audio Function Group (AFG), to be used as a default, or else in all of the indicated widgets. If these parameters are present in the AFG, they are only needed in the individual widgets that have non-default capabilities. Some parameters are marked with an asterisk (*) for the "Vendor_Specific_Audio_Widget indicating they are not required by the specification since a vendor specific node may largely define its own parameters. If, however, the vendor specific node implements features that can be defined by an existing parameter, then using the standard parameter is preferable to defining a new one.

Table 134. Required Support for Parameters

Required Parameter Support	Parameter ID	Root Node	Audio Function Group	Modem Function Group	Vendor Defined Function Group	Audio Output Converter	Audio Input Converter	Pin Complex Widget (non Digital Display)	~~HDMI~~ Pin Complex Widget (Digital Display)	Mixer (SumAmp)	Selector (Mux)	Power Widget	Volume Knob	Beep Generator	Vendor Defined Wdiget

Vendor ID	00	R													
Revision ID	02	R													
Subordinate Node Count	04	R	R	R	R										
Function Group Type	05		R	R	R										
Audio Function Group Capabilities	08		o												
Audio Widget Capabilities	09					R	R	R	R	R	R	R	R	R	R
Sample Size, Rate CAPs	0A		A			A	a								*
Stream Formats	0B		A			A	a								*
Pin Capabilities	0C							R	R						*
Input Amp Capabilities	0D		A				a	a		a	a				*
Output Amp Capabilities	12		A			a		a	a	a	a				*
Connection List Length	0E						R	R	R	R	R	R			*
Supported Power States	0F		R	R	o	o	o	o	o	o	o	R			*
Processing Capabilities	10					o	o	o	o		o				*
GPI/O Count	11		o	o	o										
Volume Knob Capabilities	13												R		
~~HDMI LPCM CAD~~	~~20~~		~~A~~			~~A~~	~~a~~		~~R~~						

Note that the Audio Function Group Capabilities parameter provides a default delay for the entire AFG to be used in lieu of adding specific delays listed for each widget in the Audio Widget Capabilities parameter. This is required if one or more widgets in the AFG opts to not report a correct delay in its Audio Widget Capabilities parameter; if all widgets do report an accurate delay number, the Audio Function Group Capabilities parameter is not required.

Table 135 specifies which verbs and controls are required (R) for each specification-defined node. It also indicates conditional (c) verbs which are required only if the respective optional capability is declared to be available. Another conditional verb (X) is required when the codec supports multiple **SDI** signals. A shaded square in the table indicates that the subject verb is not applicable to the subject node type. Some parameters are marked with an asterisk (*) for the "Vendor_Specific_Audio_Widget" indicating they are not required by the specification since a vendor specific node may largely define its own verbs. If, however, the vendor specific node implements controls that can be accessed with an existing verb, then using the standard verb is preferable to defining a new one.

Table 135. Required Support for Verbs

Required Verb Support	Get Code	Set Code	Root Node	Audio Function Group	Modem Function Group	Vendor Defined Function Group	Audio Output Converter	Audio Input Converter	Pin Complex Widget (non Digital Display)	~~HDMI~~ Pin Complex Widget (Digital Display)	Mixer (SumAmp)	Selector (Mux)	Power Widget	Volume Knob	Beep Generator	Vendor Defined Widget

Get Parameter	F00		R	R	R	R	R	R	R	R	R	R	R	R	R	R
Connection Select	F01	701						c	c	c		c				*
Get Connection List Entry	F02							R	R	R	R	R	R	R		*
Processing State	F03	##					c	c	c	c		c				*
Coefficient Index	D- -	5- -					c	c	c	c		c				*
Processing Coefficient	C- -	4- -					c	c	c	c		c				*
Amplifier Gain/Mute	B- -	3- -					c	c	c	c	c	c			c	*
Stream Format	A- -	2- -					R	R								*
Digital Converter 1	F0D	70D					c	c								*
Digital Converter 2	F0D	70E					c	c								*
Power State	F05	705		R	R	c	c	c	c	c	c	c	R			c
Channel/Stream ID	F06	706					R	R								*
SDI Select	F04	704					X	X								*
Pin Widget Control	F07	707							R	R						*
Unsolicited Enable	F08	708					c	c	c	c	c	c	c	c		*
Pin Sense	F09	709							c	c						*
EAPD/BTL Enable	F0C	70C							c							*
All GPI Controls	F10 thru F1A	710 thru 71A		c	c											
Beep Generation Control	F0A	70A													R	
Volume Knob Control	F0F	70F												R		
Implementation ID, Byte 0	F20	720		R	R	R										
Implementation ID, Byte 1	F20	721		R	R	R										
Implementation ID, Byte 2	F20	722		R	R	R										
Implementation ID, Byte 3	F20	723		R	R	R										
Config Default, Byte 0	F1C	71C							R	R						
Config Default, Byte 1	F1C	71D							R	R						
Config Default, Byte 2	F1C	71E							R	R						
Config Default, Byte 3	F1C	71F							R	R						
Stripe Control	F24	724					c									
Converter Channel Count	F2D	72D					c									
~~HDMI Data Island Pkt~~ DIP-Size	F2E	~~72E~~								R						
~~HDMI~~ ELD Data	F2F									R						
~~HDMI Data Island Pkt~~	F30	730								R						

Required Verb Support	Get Code	Set Code	Root Node	Audio Function Group	Modem Function Group	Vendor Defined Function Group	Audio Output Converter	Audio Input Converter	Pin Complex Widget (non Digital Display)	~~HDMI~~ Pin Complex Widget (Digital Display)	Mixer (SumAmp)	Selector (Mux)	Power Widget	Volume Knob	Beep Generator	Vendor Defined Wdiget
DIP-Index																
~~HDMI Data Island Pkt~~ DIP-Data	F31	731								R						
~~HDMI Data Island Pkt~~ ~~Cntrl~~ DIP-XmitCtrl	F32	732								R						
Content Protection Control	F33	733								c						
~~Cnv~~ ASP Channel ~~to~~ ~~HDMI Slot~~ Mapping	F34	734					~~c~~			R						
RESET		7FF		R	R	R										