High-Definition Multimedia Interface

Specification Version 2.0

September 4, 2013

CONFIDENTIAL

Preface

Notice

This Specification incorporates by reference the HDMI Specification Version 1.4b. However, this Specification incorporates text segments from the HDMI Specification Version 1.4b as explanatory text.

Text segments that are extracts or are otherwise derived from HDMI Specification Version 1.4b* form no part of this Specification but have been copied with permission of the owners, HDMI Consortium Founders and are included here purely for ease of comprehension to avoid excessive references to the earlier specification.

*HDMI Specification 1.4b is (c) Hitachi Consumer Electronics Co., Ltd.; Koninklijke Philips Electronics N.V.; Panasonic Corporation; Silicon Image, Inc.; Sony Corporation; Technicolor S.A.; and Toshiba Corporation.

When this Specification incorporates text segments from the HDMI Specification Version 1.4b, the reference includes a "double dagger" marker (‡) and the following statement at the beginning of the section: "This section incorporates text from the HDMI Specification 1.4b. See Notice for copyright information." Furthermore, such included text will be presented in blue text with the "Century" font, a font that has serifs.

Document Revision History

Version 2.0 The initial specification developed by the HDMI Forum.

Intellectual Property

The commercial use of This Specification in any product requires a separate license from the HDMI Licensing, LLC.

Contact Information

The URL for the HDMI Forum web site is: http://www.hdmiforum.org/

Table of Contents

Table of Figures

Table of Tables

Table of Equations

1 Introduction

This specification has been developed by the HDMI Forum.

2 Purpose and Scope

This document constitutes the Version 2.0 specification for the High-Definition Multimedia Interface (HDMI Specification Version 2.0). This Specification incorporates HDMI Specification Version 1.4b by reference and defines additional and improved functionality. Mechanical, electrical, behavioral, and protocol requirements necessary for compliance are described for Sources, Sinks, Repeaters, and Cables.

3 References

3.1 Normative References

The following standards contain provisions that, through reference in this text, constitute normative provisions of This Specification.

3.1.1 References Incorporated From HDMI 1.4b

(‡) This section incorporates text from the HDMI Specification 1.4b Section 2.2. See Notice for copyright information.

CEA, CEA-861-D, "A DTV Profile For Uncompressed High Speed Digital Interfaces"

IEC, IEC 60958-1, "Digital audio interface – Part 1: General", First edition 1999-12

IEC, IEC 60958-3, "Digital audio interface – Part 3: Consumer applications", Third edition 2006-05

ISO, ISO 639.2 Code for the representation of names of languages - Part 2: Alpha 3 code http://www.loc.gov/standards/iso639-2/langhome.html

ITU, ITU-R BT.709-5 Parameter values for the HDTV standards for production and international programme exchange (2002)

Royal Philips Electronics and SONY Corporation, "Super Audio CD System Description Version 2.0"

VESA, VESA E-EDID Standard, ENHANCED EXTENDED DISPLAY IDENTIFICATION DATA STANDARD Release A, Revision 1, February 9, 2000

3.1.2 References Introduced in This Specification

ATSC, A/52:2012, Digital Audio Compression (AC-3, E-AC-3)

CEA, CEA-861-F (2013-05), "A DTV Profile For Uncompressed High Speed Digital Interfaces"

CISPR 22:2008, 6th Edition, Information technology Equipment - Radio disturbance characteristics - Limits and methods of measurement

ETSI TS 101 547 V1.2.1 (published July 2012 as DVB Blue Book A154-1), Digital Video Broadcasting (DVB); Frame Compatible Plano-stereoscopic 3DTV; Technical Specification.
http://dvb.org/technology/standards/A154-1_DVB-3DTV_Intro.pdf

ETSI TS 101 154 V1.11.1 (2012-11), Digital Video Broadcasting (DVB); Specification for the use of Video and Audio Coding in Broadcasting Applications based on the MPEG-2 Transport Stream
http://www.etsi.org/deliver/etsi_ts/101100_101199/101154/01.11.01_60/ts_101154v011101p.pdf

ETSI EN 300 468 V1.13.1 (2012-04), Digital Video Broadcasting (DVB); Specification for Service Information (SI) in DVB systems.
http://www.etsi.org/deliver/etsi_en/300400_300499/300468/01.13.01_40/en_300468v011301o.pdf

HDMI, HDMI Licensing, LLC, “High-Definition Multimedia Interface Specification Version 1.4b”, October 11, 2011

IEC, IEC 60958-3-am1 Ed.3.0, “Amendment 1 to IEC 60958-3: Digital audio interface - Part 3: Consumer applications”, October, 2009

IEC, IEC 61937-1 Edition 2.0, "Digital audio - Interface for non-linear PCM encoded audio bitstreams applying IEC 60958 – Part 1: General"

IEC, IEC 61937-1 Amendment 1 Edition 2.0, "Digital audio - Interface for non-linear PCM encoded audio bitstreams applying IEC 60958 - Part 1: General"

IEC, IEC 61937-2 Edition 2.0, "Digital audio - Interface for non-linear PCM encoded audio bitstreams applying IEC 60958 – Part 2: Burst-info"

IEC, IEC 61937-2 Amendment 1 Edition 2.0, "Digital audio - Interface for non-linear PCM encoded audio bitstreams applying IEC 60958 - Part 2: Burst-info"

IEC, IEC 62574 ed 1.0, “Audio, video and multimedia systems – General channel assignment of multichannel audio, April 7, 2011

ITU-R BS.2159-4 (05/2012), Multichannel sound technology in home and broadcasting applications

ITU, Recommendation ITU-R BT.2020 (08/2012), Parameter values for ultra-high definition television systems for production and international programme exchange;
http://www.itu.int/rec/R-REC-BT.2020/en

NXP Semiconductors, I2C-bus specification and user manual UM10204, Rev. 5 - 9 October 2012

SMPTE, SMPTE 2036-2:2008, “UHDTV – Audio characteristics and audio channel mapping for program production”, 2008

VESA, VESA Enhanced Display Data Channel (E-DDC) Standard Version 1.2 December 26, 2007
http://www.vesa.org/vesa-standards/

3.2 Informative References

The following documents contain information that is useful in understanding This Specification.

ITU, Recommendation ITU-R BT.1359-1, Relative timing of sound and vision for broadcasting; http://www.itu.int/rec/R-REC-BT.1359/en

3.3 Usages and Conventions

(‡) This section incorporates text from the HDMI Specification 1.4b Section 1.5. See Notice for copyright information.

bit N	Bits are numbered in little-endian format, i.e. the least-significant bit of a byte or word is referred to as bit 0.
D[X:Y]	Bit field representation covering bit X to bit Y (inclusive) of value or field D.
0xNN	Hexadecimal representation of base-16 numbers are represented using 'C' language notation, preceded by '0x'.
0bNN	Binary (base-2) numbers are represented using 'C' language notation, preceded by '0b'.
NN	Decimal (base-10) numbers are represented using no additional prefixes or suffixes.

4 Definitions

4.1 Conformance Levels

(‡) This section incorporates text from the HDMI Specification 1.4b Section 2.1. See Notice for copyright information.

expected	A key word used to describe the behavior of the hardware or software in the design models assumed by this specification. Other hardware and software design models may also be implemented.
may	A key word that indicates flexibility of choice with no implied preference.
shall	A key word indicating a mandatory requirement. Designers are required to implement all such mandatory requirements.
should	A key word indicating flexibility of choice with a strongly preferred alternative. Equivalent to the phrase is recommended.
reserved fields	A set of bits within a data structure that are defined in this specification as reserved, and are not otherwise used. Implementations of this specification shall zero these fields. Future revisions of this specification, however, may define their usage.
reserved values	A set of values for a field that are defined in this specification as reserved, and are not otherwise used. Implementations of this specification shall not generate these values for the field. Future revisions of this specification, however, may define their usage.

4.2 Glossary of Terms

4.2.1 Terms Incorporated From HDMI 1.4b (Informative)

(‡) This section incorporates text from the HDMI Specification 1.4b Section 2.2. See Notice for copyright information.

HDMI Version 1.4b defines a number of terms that are utilized by This Specification. In order to enhance readability of this Specification, these terms are included in This Specification.

Audio Channel	Audio data intended to be delivered to a single audio speaker.
Audio System	A device, which is not a TV, that has the ability to render audio, e.g. an audio Amplifier.
Broadcast Message	This is a message, sent to Logical Address 15, which all devices are expected to receive.
Byte	Eight bits of data.
CEC Root Device	A device, generally a display (Sink) device, formally defined by the following rule: A device that has no HDMI output or, a device that has chosen to take the physical address 0.0.0.0 (see H14b Section 8.7).
Compressed (Audio)	All audio formats carried by HDMI other than L-PCM and One Bit Audio.
Data Stream Disparity	Integer indicating "DC-offset" level of link. A positive value represents the excess number of "1"s that have been transmitted. A negative value represents the excess number of "0"s that have been transmitted.
Deck	The part of a Recording Device or Playback Device that provides playback functionality e.g. from a media such as DVD or Hard Disk.
Destination	The target device for a CEC message.
Direct Stream Transport	An audio format which is a lossless compression of Direct Stream Digital (DSD), as used in SuperAudio CD. DST is described in ISO/IEC 14496, part 3, Amendment 6: Lossless coding of oversampled audio.
Downstream	In the direction of the primary audio and video data flow, i.e. towards the Sink (e.g. display).
Follower	A device that has just received a CEC message and is required to respond to it.
(HDMI) Source	A device with an HDMI output.
(HDMI) Sink	A device with an HDMI input.
(HDMI) Repeater	A device with one or more HDMI inputs and one or more HDMI outputs. Repeater devices shall simultaneously behave as both an HDMI Sink and an HDMI Source.
Initiator	The device that is sending, or has just sent, a CEC message and, if appropriate, is waiting for a Follower to respond.

Logical Address	A unique address assigned to each device (see H14b section CEC 10.2)
One Bit Audio	1-bit Delta-Sigma modulated signal stream such as that used by Super Audio CD
Playback device	A device that has the ability to play media, e.g. a DVD Player.
Pixel	Picture Element. Refers to the actual element of the picture and the data in the digital video stream representing such an element.
Receiver	A component that is responsible for receiving the four differential TMDS input pairs at the input to an HDMI Sink and converting those signals into a digital output indicating a 24 bit, 12 bit, or 6 bit TMDS decoded word and indicating the TMDS coding mode used to decode those bits. This digital output may be contained within a semiconductor device or may be output from a semiconductor device.
Recording device	A device that has the ability to record a source such as an internal tuner or an external connection.
Source Device	A device that is currently providing an AV stream via HDMI.
Stereo	2 channel audio.
Stream	A time-ordered set of digital data originating from one Source and terminating at zero or more Sinks. A stream is characterized by bounded bandwidth requirements.
Super Audio CD	Disk format of "Super Audio CD System Description", see http://www.licensing.philips.com.
T_{bit}	Time duration of a single bit carried across the TMDS data channels.
$T_{character}$	Time duration of a single TMDS character carried across the TMDS data channels. This is equal to $10*T_{bit}$.
Transmitter	A component that is responsible for driving the four differential TMDS output pairs into an HDMI output and for clocking the data driven into those four output pairs.
Tuner Device	A device that contains a tuner, e.g. an STB or a Recording Device.
TV	A device with HDMI input that has the ability to display the input HDMI signal. Generally it has no HDMI output.
Video Field	The period from one VSYNC active edge to the next VSYNC active edge.
Video Format	A video format is sufficiently defined such that when it is received at the monitor, the monitor has enough information to properly display the video to the user. The definition of each format includes a Video Format Timing, the picture aspect ratio, and a colorimetry space.
Video Format Timing	The waveform associated with a video format. Note that a specific Video Format Timing may be associated with more than one Video Format (e.g., 720X480p@4:3 and 720X480p@16:9).

YC_BC_R Digital representation of any video signal using one of several luminance/color-difference color spaces.

4.2.2 Terms Defined in This Specification

3D Audio
An audio system whose speakers are placed anywhere in 3D space. This is in contrast to 5.1 or 7.1 Audio which do not include an element of height and typically place speakers in a horizontal 2D plane. 3D Audio uses the channel layouts defined in ITU-R BS.2159-4 (Type B 10.2ch), SMPTE 2036-2 (22.2ch), or IEC 62574 (30.2ch). 3D Audio also includes the down-mixed audio streams defined in these standards, provided that the down-mixed audio stream includes 9 or more audio channels. For the avoidance of doubt in This Specification, 3D Audio refers to a finite number of discrete channels and not object-based audio.

Audio Description
An audio service that helps blind and visually impaired consumers to understand the action in a program.

Note – in some countries, this is referred to as “Video Description”.

CE Video Format
Any Video Format listed in CEA-861-F Table 1 except the VGA (640x480p) Video Format.

CEA Extension
A 128 byte EDID 1.3-compatible extension block defined in CEA-861-F, designed to allow declaration of audio formats, additional Video Formats (beyond those in the base EDID structure), and other characteristics of the Sink.

CEC 1.4b
CEC as defined in HDMI Specification Version 1.4b.

CEC 1.x
CEC as defined in HDMI Specification 1.4b or earlier.

CEC 2.0
CEC as defined in This Specification.

CEC 2.0+
CEC as defined in This Specification or a subsequent version.

CEC Switch
A Repeater which can be switched by CEC messages (see H14b Section CEC 11.1).

Device Type
Classification of a device’s capabilities; a device can have multiple device types. One of these (the most important one) is called the Primary Device Type, and is communicated in the operand [Primary Device Type]; the collection of all device types (including the primary device type) of a device is communicated in the operand [All Device Types]. The term “Device Type” (without “Primary” prefix) refers to the collection of all the capabilities.

Note: [Primary Device Type] in CEC 2.0 corresponds to [Device Type] in CEC 1.4b (and earlier) and hence should be chosen carefully for backward compatibility; [All Device Types] is new for CEC 2.0 and does not have an equivalent in earlier CEC 1.4b versions. [Primary Device Type] is communicated in <Report Physical Address> message, [All Device Types] is communicated in <Report Features> message, see Section 11.2.4.
Note: for some special cases of devices with multiple device types, see Section 11.3.2.

Generic Source
A Source Device which can become the Active Source.

H14b
References in the text of This Specification to sections, figures and tables in the Version 1.4b specification are prefixed here by “H14b” to clearly differentiate items only in the Version 1.4b specification from items introduced in This Specification.

H14b VSDB
The HDMI Vendor Specific Data Block defined by HDMI 1.4b.

H14b VSIF	The HDMI Vendor Specific InfoFrame packet defined by HDMI 1.4b.
HF-VSDB	The E-EDID Vendor Specific Data Block defined by This Specification.
HF-VSIF	The Vendor Specific InfoFrame packet defined by This Specification.
InfoFrame	A data structure defined in CEA-861-F that is designed to carry a variety of auxiliary data items regarding the audio or video streams or the Source Device and is carried from Source to Sink across HDMI.
Informative	Supplemental information such as additional guidance, supplemental recommendations, tutorials, commentary as well as background, history, development, and relationship with other elements. Informative data is not a requirement and does not compel compliance.
IT Video Format	Any Video Format that is not a CE Video Format. Note: the VGA (640x480p) Video Format is an IT Video Format.
Little-Endian	Defines the storage order of multi-byte values in a byte wide memory space. When multi-byte values are stored in Little-Endian order, the least significant byte is stored in the low order offset location. E.g. if a value X[15:0] is stored starting at offset N, X[7:0] will be stored to offset (N) and X[15:8] will be stored to offset (N+1).
Multi-Stream Audio	A collection of audio streams associated with one or more video streams.
Pixel Clock Period	1 / Pixel Clock Rate.
Pixel Clock Rate	The dot clock used for video timing. When Pixel replication is active, the rate includes the replicated Pixels. (e.g. Single Pixel Replication for 480p yields a Pixel Clock Rate = 54 MHz).
Processor	HDMI Repeater with characteristics as detailed in Table 11-7.
Pure CEC Switch	A device according to H14b Section CEC 11.1 which has no other functionality or Device Type (see Table 11-7).
R_{bit}	Synonymous with TMDS Bit Rate.
This Specification	When the words “This Specification” are included in this document, it is a reference to HDMI Specification Version 2.0.
TMDS Bit Period	1 / TMDS Bit Rate. Synonymous with T_{bit}.
TMDS Bit Rate	10x the TMDS Character Rate.
TMDS Character	A 10-bit TMDS-encoded value.
TMDS Character Clock	A clock which oscillates at the TMDS Character Rate.
TMDS Character Period	1 / TMDS Character Rate. Synonymous with $T_{character}$.

TMDS Character Rate The rate at which 10-bit TMDS characters are transmitted per data channel over the HDMI link. This rate is expressed in Mega-characters/second/channel (Mcsc).

0.5x the Pixel Clock Rate in MHz for 24-bit YC_BC_R 4:2:0 Pixel Encoding.
0.625x the Pixel Clock Rate in MHz for 30-bit YC_BC_R 4:2:0 Pixel Encoding.
0.75x the Pixel Clock Rate in MHz for 36-bit YC_BC_R 4:2:0 Pixel Encoding.
1x the Pixel Clock Rate in MHz for 4:2:2 Pixel Encoding,
24-bit 4:4:4 Pixel Encoding and 48-bit YC_BC_R 4:2:0 Pixel Encoding.
1.25x the Pixel Clock Rate in MHz for 30-bit 4:4:4 Pixel Encoding.
1.5x the Pixel Clock Rate in MHz for 36-bit 4:4:4 Pixel Encoding.
2x the Pixel Clock Rate in MHz for 48-bit 4:4:4 Pixel Encoding.

TMDS Clock Period 1 / TMDS Clock Rate.

TMDS Clock Rate The rate at which the clock channel oscillates on the HDMI cable.

1x the TMDS Character Rate for TMDS Character Rates ≤ 340 Mcsc.
0.25x the TMDS Character Rate for TMDS Character Rates > 340 Mcsc.

Worst Cable Emulator The cable emulator utilized by the compliance tests for Source Device eye compliance and Sink Device Jitter Tolerance testing of devices compliant with This Specification. The specification of the Worst Cable Emulator is available as a Companion Document to This Specification.

4.3 Acronyms

4.3.1 Acronyms Incorporated from HDMI 1.4b (Informative)

(‡) This section incorporates text from the HDMI Specification 1.4b Section 2.2. See Notice for copyright information.

ACR	Audio Clock Regeneration
ARC	Audio Return Channel
AVI	Auxiliary Video Information
CDC	Capability Discovery and Control (for HEAC)
CEA	Consumer Electronics Association
CEC	Consumer Electronics Control
CTS	Definition 1: Cycle Time Stamp Definition 2: Compliance Test Specification
DDC	Display Data Channel
DST	Direct Stream Transport
DTV	Digital Television
DVD	Digital Versatile Disc
DVI	Digital Visual Interface

E-DDC	Enhanced Display Data Channel
E-EDID	Enhanced Extended Display Identification Data
EDID	Extended Display Identification Data
HDCP	High-bandwidth Digital Content Protection
HDMI	High-Definition Multimedia Interface
HDTV	High-Definition Television
HEAC	HDMI Ethernet and Audio Return Channel
HPD	Hot Plug Detect
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronics Engineers
ITU	International Telecommunications Union
L-PCM	Linear Pulse-Code Modulation
LSb	least significant bit
MPEG	Moving Picture Experts Group
MSb	most significant bit
Rx	Receiver
SMPTE	Society of Motion Picture & Television Engineers
STB	Set-Top Box
SVD	Short Video Descriptor
TERC4	TMDS Error Reduction Coding – 4 bit
TMDS	Transition Minimized Differential Signaling
Tx	Transmitter
VESA	Video Electronics Standards Association
VSDB	Vendor-Specific Data Block

4.3.2 Acronyms that are introduced by This Specification

ACAT	Audio Channel Allocation Standard Type
ACK	Acknowledge
ALS	Auto Lipsync
AV	Audio/Video
BDP	Blu-ray Disk™ Player
DALS	Dynamic Auto Lipsync
DVB	Digital Video Broadcasting
EOM	End Of Message
FP	Frame Packing (Refers to the "Frame Packing" 3D transmission packing as defined in H14b Section 8.2.3.2)
Gbps	Giga bits per second (Giga = 10^9)
HbbTV	Hybrid Broadcast Broadband TV
ID	Identifier
LFSR	Linear Feedback Shift Register
Mcsc	Mega-characters/second/channel (Applies to TMDS Character Rate) (Mega = 10^6)
MHEG	Multimedia and Hypermedia Experts Group
MHP	Multimedia Home Platform
OSD	On Screen Display
OUI	Organizationally Unique Identifier
PC	Personal Computer
RC	Remote Control
Rsvd	A reserved value in a register or memory space
SbS	Side by Side (Refers to the "Side-by-Side (Half)" 3D transmission packing as defined in H14b Section 8.2.3.2)
SCART	Syndicat des Constructeurs d'Appareils Radiorécepteurs et Téléviseurs (standard audio video TV connector)
SCDC	Status and Control Data Channel
SSCP	Scrambler Synchronization Control Period

TaB	Top and Bottom (Refers to the "Top-and-Bottom" 3D transmission packing as defined in H14b Section 8.2.3.2)
TDR	Time Domain Reflectometry
TP	Test Point
UI	User Interface
VSIF	Vendor Specific InfoFrame

5 Overview

All features and functions of the HDMI Specification Version 2.0 (This Specification) are optional and if utilized shall be implemented according to the requirements specified for each respective feature or function. Note that there are minimum requirements for each feature or function included in the HDMI Specification Version 2.0.

A device that utilizes features and functions defined in This Specification shall be interoperable with other HDMI compliant devices including and not limited to devices that meet the minimum mandatory requirements of HDMI Specification Version 1.4b, for all HDMI features and functions that are implemented in both devices.

HDMI 1.4b defines TMDS signals at TMDS Character Rates of up to 340 Mcsc. This Specification adds TMDS signals at TMDS Character Rates from 340 to 600 Mcsc (Section 6.1.1), and adds scrambling for EMI/RFI reduction at all TMDS Character Rates (Section 6.1.2), and TMDS Character Error Detection (Section 6.2).

HDMI 1.4b defines several Pixel transport mechanisms. These define the transport of RGB and YC_BC_R 4:4:4 Pixels with Pixel sizes of 24, 30, 36, or 48 bits. HDMI 1.4b also defines a mechanism for transporting YC_BC_R 4:2:2 Pixels with Pixel sizes of 24, 30, or 36 bits. When the Video Format Timing being used is 2160p50 or 2160p60 as indicated in Section 7.1, This Specification adds a defined mechanism for transporting YC_BC_R 4:2:0 Pixels. YC_BC_R 4:2:0 Pixel Encoding is carried at a TMDS Character Rate equal to ½ the TMDS Character Rate for 8-bit 4:4:4 Pixel Encoding.

HDMI 1.4b defines several audio transport mechanisms. These include IEC 60958 L-PCM and IEC 61937 compressed audio that support audio sample rates up to 192kHz. In addition, HDMI 1.4b also defines transport mechanisms for One Bit Audio and DST audio. This Specification increases the number of compressed audio formats that may be transported via an IEC 61937 compressed stream. It also defines three new audio transport mechanisms. The following is a brief summary of the available Audio options:

- L-PCM.
- an IEC 61937 compressed (e.g. surround-sound) or DST audio stream at bit rates up to 49.152Mbps.
- from 2 to 32 channels of One Bit Audio.
- 3D Audio with support for 10.2, 22.2, and 30.2 speaker placement.
- Multi Stream Audio to support multiple video streams or multi-view video streaming (e.g. dual-view gaming with different audio for each view) or single-view video streaming (e.g. multi-lingual support). In this case, up to 4 audio streams can be transmitted simultaneously.

DDC is used in HDMI 1.4b for reading E-EDID and other purposes. This Specification adds a set of HDMI-specific DDC Registers in HDMI Sinks to exchange point-to-point dynamic data between the Source and the Sink (See Section 10.4, SCDC, Status and Control Data Channel).

This Specification extends the list of supported video and audio formats according to CEA-861-F (Sections 7.1 and 9.1), and extends Colorimetry as defined in H14b with the colorimetry defined in ITU-R BT.2020 (Section 7.2). This Specification also adds signaling features for 3D signals: 3D OSD Disparity, 3D Dual View and 3D Independent View (Sections 7.4.1, 7.4.2 and 7.4.3).

This Specification defines the Dynamic Auto Lipsync feature, which is an extension of H14b's Auto Lipsync feature to allow Sinks to dynamically modify and announce their latency information (Section 10.7).

Finally, This Specification defines CEC 2.0, an extension of CEC as defined in H14b with expanded sets of mandatory features to promote wider interoperability between all compliant devices (Section 11).

6 Link Layer

6.1 340 Mcsc to 600 Mcsc TMDS Character Rate Support

6.1.1 Electrical Characteristics for TMDS Character Rate above 340 Mcsc and up to 600 Mcsc

The operation of the TMDS link at TMDS Bit Rates ranging from 3.4 Gbps to 6.0 Gbps is defined in this section and extends the TMDS Specification in H14b Sections 4.2.3, 4.2.4, and 4.2.5. Any parameter that is not specified in this section is unchanged from HDMI 1.4b.

For TMDS Character Rates above 340 Mcsc, the TMDS Clock Rate shall be one fourth (1/4) of the TMDS Character Rate. The TMDS Bit Rate remains 10 times the TMDS Character Rate, and is therefore 40 times the TMDS Clock Rate. For TMDS Character Rates at or below 340 Mcsc, the TMDS Clock Rate is equal to the TMDS Character Rate, and the TMDS Bit Rate is equal to 10 times TMDS Clock Rate as specified in HDMI 1.4b. The Source shall inform the Sink of the relationship between TMDS Clock Rate and TMDS Character Rate using the control bit TMDS_Bit_Clock_Ratio, see Section 6.1.3.2.

The Sink Device shall indicate the maximum TMDS Character Rate that it supports in the Max_TMDS_Character_Rate field in the HF-VSDB (Section 10.3.2) if it supports TMDS Character Rate above 340 Mcsc and up to 600 Mcsc. The Source shall not transmit at TMDS Character Rates higher than the maximum rate supported by the Sink, as indicated by the Max_TMDS_Character_Rate field of the HF-VSDB.

6.1.1.1 TMDS Overview

The operation of the TMDS link at TMDS Bit Rates ranging from 3.4 Gbps to 6.0 Gbps is similar to that described by HDMI 1.4b. Test points TP1 and TP2 are system reference points for specifications and measurements and are connected by an HDMI Cable.

Specifications for TP1 and TP2 are provided in the following sections. An Eye Diagram is provided for TP2 only.

The Cable shall meet the Category 2 specifications defined in HDMI 1.4b.

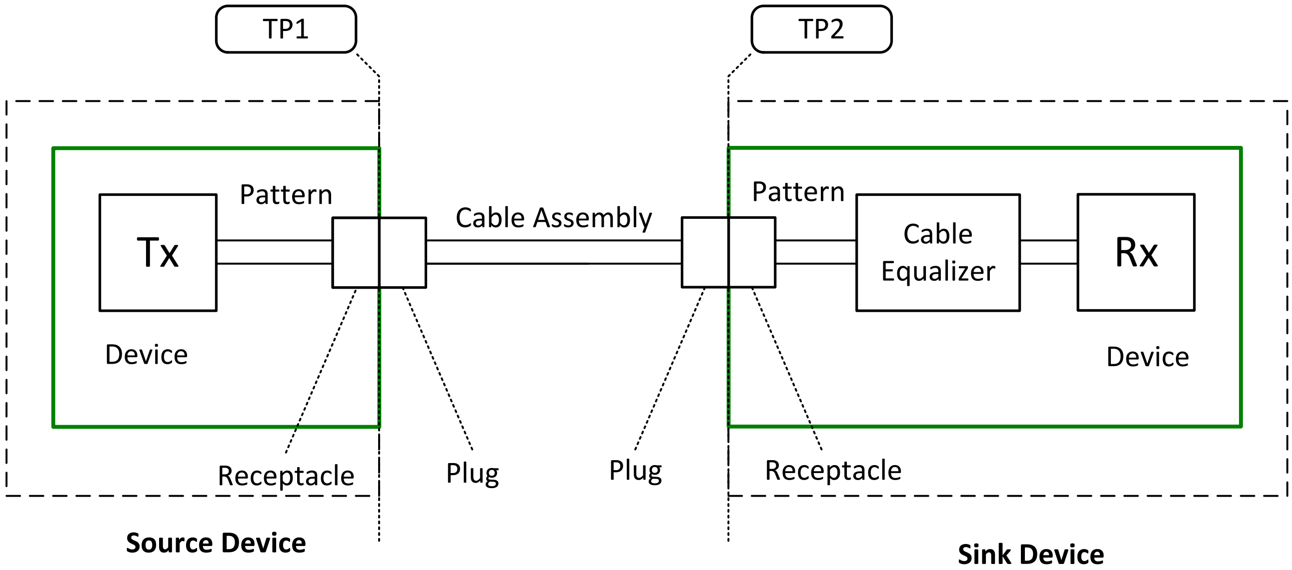


Figure 6-1: TMDS Link Test Points

6.1.1.2 Jitter and Eye Diagram Measurements

(‡) This section incorporates text from the HDMI Specification 1.4b Section 4.2.3.1. See Notice for copyright information.

The Jitter Transfer function provided in Equation 6-1 is unchanged from the HDMI 1.4b Specification (See H14b Equation 4-1) and applies to TMDS Bit Rates of up to 6.0 Gbps.

$$H(j\omega) = 1 / (1 + j\,\omega / \omega_0)$$

$$\text{Where } \omega_0 = 2\pi F_0,\ F_0 = 4.0 \text{ MHz}$$

Equation 6-1: Jitter Transfer Function of Ideal CRU for Ideal Recovery Clock Definition Defined in H14b Equation 4-1

6.1.1.3 Reference Cable Equalizer

The definition of the Reference Cable Equalizer is given in Equation 6-2 and illustrated in Figure 6-2 for TMDS Bit Rates ranging from 3.4 Gbps to 6.0 Gbps. In addition, a table with the Phase and Gain components of the Reference Cable Equalizer is included in the Companion Documentation package.

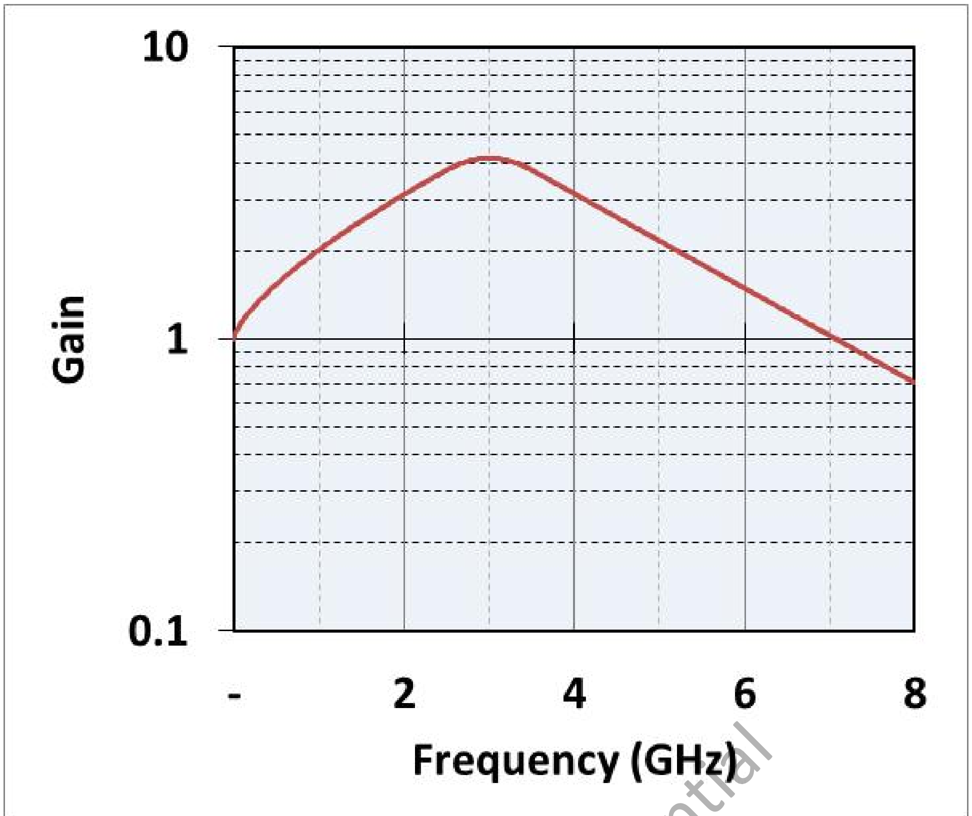


Figure 6-2: Reference Cable Equalizer for 3.4 Gbps < R_{bit} ≤ 6.0 Gbps

$$
\left|H(j\omega)\right| = \begin{cases} e^{A*\omega^{N}} & (\omega < \omega_0) \\ e^{-B*(\omega-1.2*\omega_0)^2+C} & (\omega_0 < \omega < 1.4*\omega_0) \\ e^{-D*\omega+E} & (1.4*\omega_0 < \omega) \end{cases}
$$

Where

$$N = 0.7$$

$$\omega_0 = 2\pi * 2.5GHz$$

$$A = 9.7E-8$$

$$B = \frac{7}{4} * A * \omega_0^{-1.3}$$

$$C = 1.07 * A * \omega_0^{0.7}$$

$$D = 0.7 * A * \omega_0^{-0.3}$$

$$E = 1.98 * A * \omega_0^{0.7}$$

Equation 6-2: Reference Equalizer Equations for 3.4 Gbps < R_{bit} ≤ 6.0 Gbps

6.1.1.4 HDMI Source TMDS Characteristics

No eye diagram measured at TP1 is specified for operation between 3.4 Gbps and 6.0 Gbps.

The Source shall meet the DC specifications in Table 6-1 and the AC specification in Table 6-2 across all operating conditions specified in H14b Table 4-22 when driving clock and data signals. Definitions for V_{swing}, rise time, fall time, intra pair skew, and inter pair skew parameters are provided in HDMI 1.4b.

Table 6-1: DC Characteristics for 3.4 Gbps < R_{bit} ≤ 6.0 Gbps at TP1

Item	Value
Single-Ended High Level Voltage Range: Data Channels 0,1,2	AVcc – 400 mV to AVcc + 10 mV
Single-Ended Low Level Voltage Range: Data Channels 0,1,2	AVcc – 1000 mV to AVcc - 400 mV
Single-Ended High Level Voltage Range: Clock Channel	AVcc – 400 mV to AVcc + 10 mV
Single-Ended Low Level Voltage Range: Clock Channel	AVcc – 1000 mV to AVcc - 200 mV
Single-Ended Swing Voltage: Data Channels 0,1,2	400 mV ≤ V_{swing} ≤ 600 mV
Single-Ended Swing Voltage: Clock Channel	200 mV ≤ V_{swing} ≤ 600 mV

Table 6-2: AC Characteristics for 3.4 Gbps < R_{bit} ≤ 6.0 Gbps at TP1

Item	Value
Rise/Fall time: Data (20% to 80%) Rise/Fall time: Clock (20% to 80%)	≥42.5 ps ≥75 ps
Intra-Pair Skew, Max	0.15 T_{bit}
Inter-Pair Skew, Max	0.20 $T_{character}$
Maximum Differential Voltage V_{high}	780 mV
Minimum Differential Voltage V_{low}	-780 mV
Clock Duty Cycle	40% to 60%
TMDS Clock Rate[Θ]	85 MHz to 150 MHz

[Θ] Ratio of TMDS Clock Period to TMDS Bit Period is 40:1.

The Source shall comply with the impedance characteristics at TP1 as specified in Table 6-3.

Table 6-3: Source Impedance Characteristics for 3.4 Gbps < R_{bit} ≤ 6.0 Gbps at TP1

Item	Value
TDR Rise Time at TP1 (10% - 90%)	≤200 ps
Through Connection Impedance[Δ]	100 Ω +/- 15%[◊]
Source Termination Impedance	75 to 150 Ω

[◊] A single excursion is permitted out to a max/min of 100 Ω +/- 25% and of a duration less than 250 ps.
[Δ] Impedance from TP1 to Source Termination

For all data channels across all operating conditions specified in H14b Table 4-22 and when configured as specified in Figure 6-3, the Source shall have output levels at TP2_EQ that meet the eye diagram requirements of Figure 6-4 after application of the Worst Cable Emulator for Category 2 cable and Reference Cable Equalizer in Equation 6-2. In Figure 6-3, TP2_EQ represents the connection point of the plug of Test Fixture (TPA-P). This requirement specifies minimum eye opening. The time axis is normalized to the bit time at the operating frequency.

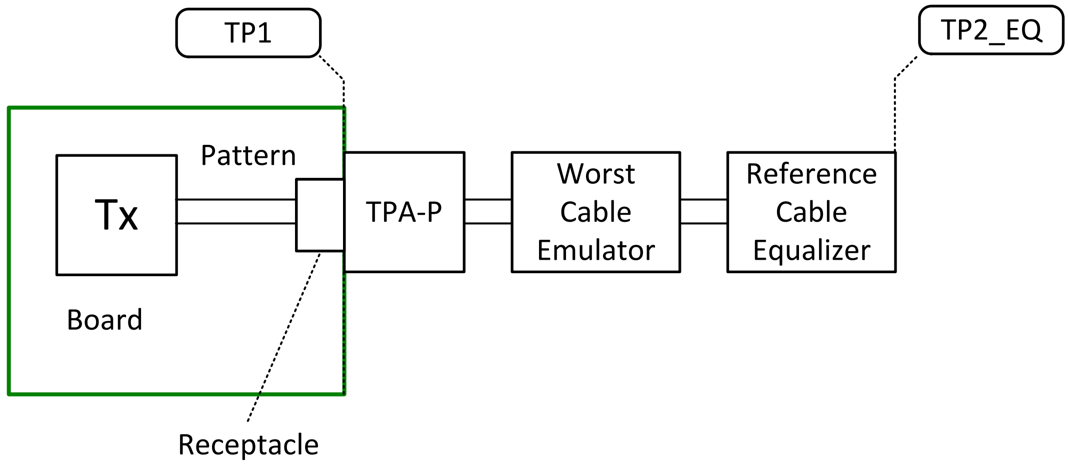


Figure 6-3: HDMI Source Test Point for Eye Diagram

For all clock and data channels, across all operating conditions specified in H14b Table 4-22, and when configured as specified in Figure 6-3, the Source shall meet the clock and data jitter requirements of Table 6-4 after application of the Worst Cable Emulator for Category 2 cable and Reference Cable Equalizer in Equation 6-2. In Figure 6-3, TP2_EQ represents the connection point of the plug of Test Fixture (TPA-P). This requirement specifies the maximum allowable jitter for clock and data channels. The time axis is normalized to the data bit time for both clock and data at the operating bit rate.

Table 6-4: HDMI Source Jitter Characteristics for 3.4 Gbps < R_{bit} ≤ 6.0 Gbps at TP2_EQ

Item	**Value**
Allowable Total Clock Jitter	0.3 T_{bit}
Allowable Total Data Jitter	(1-H) T_{bit} $^{\Theta}$

$^{\Theta}$H is defined in Figure 6-4 Eye Diagram at TP2_EQ

6.1.1.5 HDMI Sink TMDS Characteristics

Sink devices shall recover data on each data channel at a TMDS character error rate of 10^{-9} or better when operating at TMDS Bit Rates ranging from 3.4 Gbps to 6.0 Gbps and when presented with any signal that, following application of the Reference Cable Equalizer in Equation 6-2, is compliant with the eye mask diagram of Figure 6-4.

Functions defining the extent of the H and V variables represented in Figure 6-4 are provided in Table 6-5 and Figure 6-5.

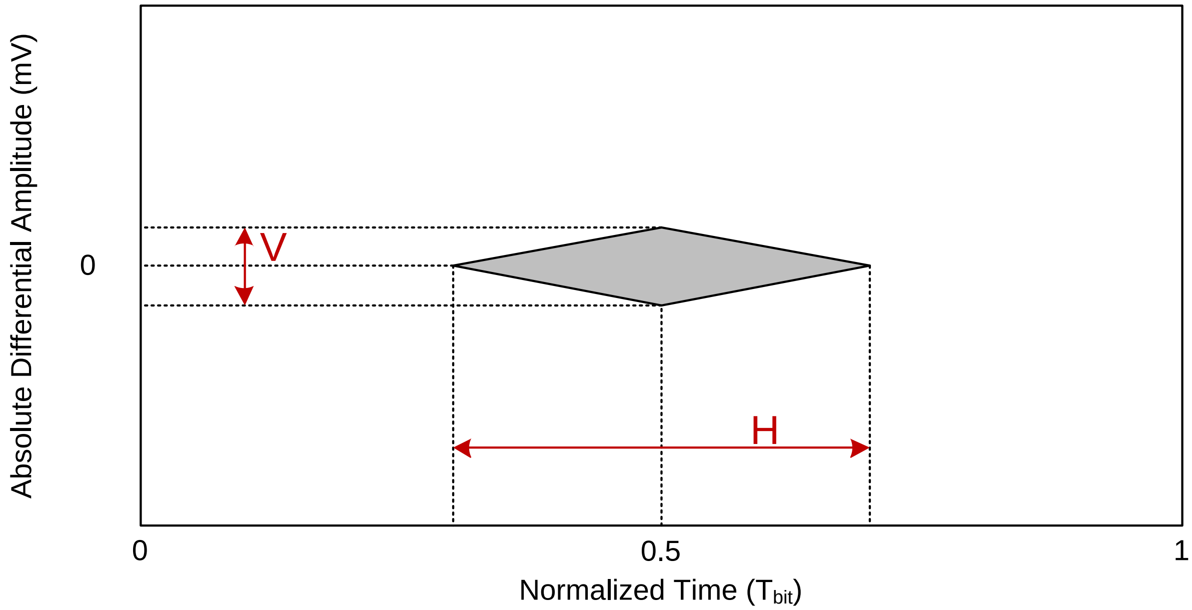


Figure 6-4: Eye diagram at TP2_EQ

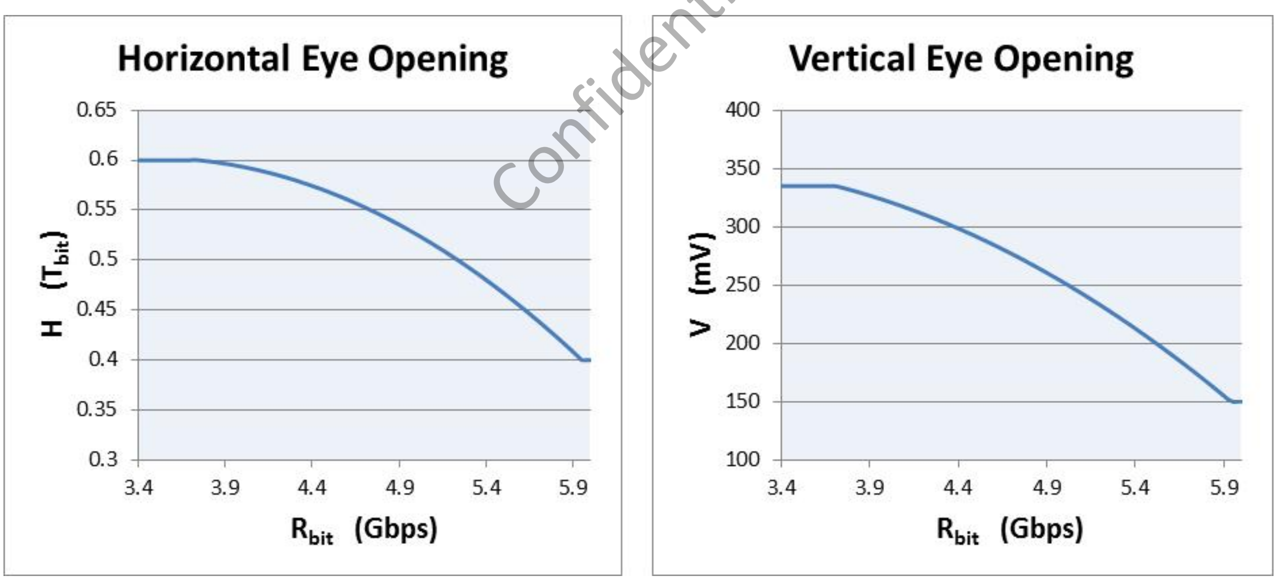


Figure 6-5: Plots of the Functions defining horizontal and vertical dimensions for the eye diagram at TP2_EQ

Table 6-5: Functions defining horizontal and vertical dimensions for the eye diagram at TP2_EQ

TMDS Bit Rate (Gbps)	H (T_{bit})	V (mV)
$3.4 < R_{bit} \leq 3.712$	0.6	335
$3.712 < R_{bit} < 5.94$	$-0.0332 R_{bit}^2 + 0.2312 R_{bit} + 0.1998$	$-19.66 R_{bit}^2 + 106.74 R_{bit} + 209.58$
$5.94 \leq R_{bit} \leq 6.0$	0.4	150

Sink Devices shall operate within the DC parameters specified in Table 6-6, and the AC parameters specified in Table 6-7 for TMDS Bit Rates ranging from 3.4 Gbps to 6.0 Gbps.

Table 6-6: Sink Operating DC Input Characteristics for devices supporting 3.4 Gbps < R_{bit} ≤ 6.0 Gbps at TP2

Item	Value
Input Differential Voltage Level, V_{idiff}	150 mV ≤ V_{idiff} ≤ 1200 mV
Input Common Mode Voltage, V_{icm}	
V_{icm1}	AVcc – 700 mV ≤ V_{icm1} ≤ AVcc – 37.5 mV
V_{icm2}	AVcc – 10 mV ≤ V_{icm2} ≤ AVcc + 10 mV

Table 6-7: Sink AC Input Characteristics for 3.4 Gbps < R_{bit} ≤ 6.0 Gbps at TP2

Item	Value
Minimum differential sensitivity (peak-to-peak) after the Reference Cable Equalizer	150 mV
Max Allowable Intra-Pair Skew at Sink Connector	0.15 T_{bit} + 112 ps
Max Allowable Inter-Pair Skew at Sink Connector	0.2 $T_{character}$ + 1.78 ns
TMDS Clock Jitter	0.30 T_{bit} (relative to Ideal Recovery Clock)

HDMI Sink Devices shall comply with the Impedance characteristics at TP2 as specified in Table 6-8.

Table 6-8: Sink Impedance Characteristics for 3.4 Gbps < R_{bit} ≤ 6.0 Gbps at TP2

Item	Value
TDR Rise Time at TP2 (10% - 90%)	≤ 200 ps
Through Connection Impedance[Δ]	100 Ω +/- 15%[◊]
Sink Termination Impedance	90 Ω to 110 Ω

[◊]A single excursion is permitted out to a max/min of 100 Ω ±25% and of duration less than 250 ps.
[Δ] Impedance from TP2 to Sink Termination

6.1.2 Scrambling for EMI/RFI Reduction

This Specification includes new scrambling techniques for reduction of Electo-Magnetic Interference (EMI) and Radio Frequency Interference (RFI) in the 3 data channels: TMDS Channels 0, 1, and 2. EMI/RFI reduction in the TMDS Clock channel is achieved by reducing the clock frequency as specified in Section 6.1.1 and by reducing the clock amplitude as permitted in Section 6.1.1.4. When transmitting above 3.4Gbps, Sources are recommended to reduce the clock amplitude to the minimum permitted in Section 6.1.1.4.

Devices capable of sending or receiving with TMDS Bit Rates ranging from 3.4 Gbps to 6.0 Gbps shall be capable of scrambling at rates above 3.4 Gbps and should be capable of scrambling at rates at or below 3.4 Gbps. Devices that are not capable of sending or receiving with TMDS Bit Rates greater than 3.4 Gbps may be capable of scrambling at all rates the device supports.

When the TMDS Bit Rate is greater than 3.4 Gbps, scrambling shall be enabled by the Source. When the TMDS Bit Rate is less than or equal to 3.4 Gbps, the Source shall enable scrambling if both the Source and Sink device support scrambling at that rate. For Scrambling Control mechanisms, refer to Section 6.1.3.1.

6.1.2.1 Operating Modes Overview

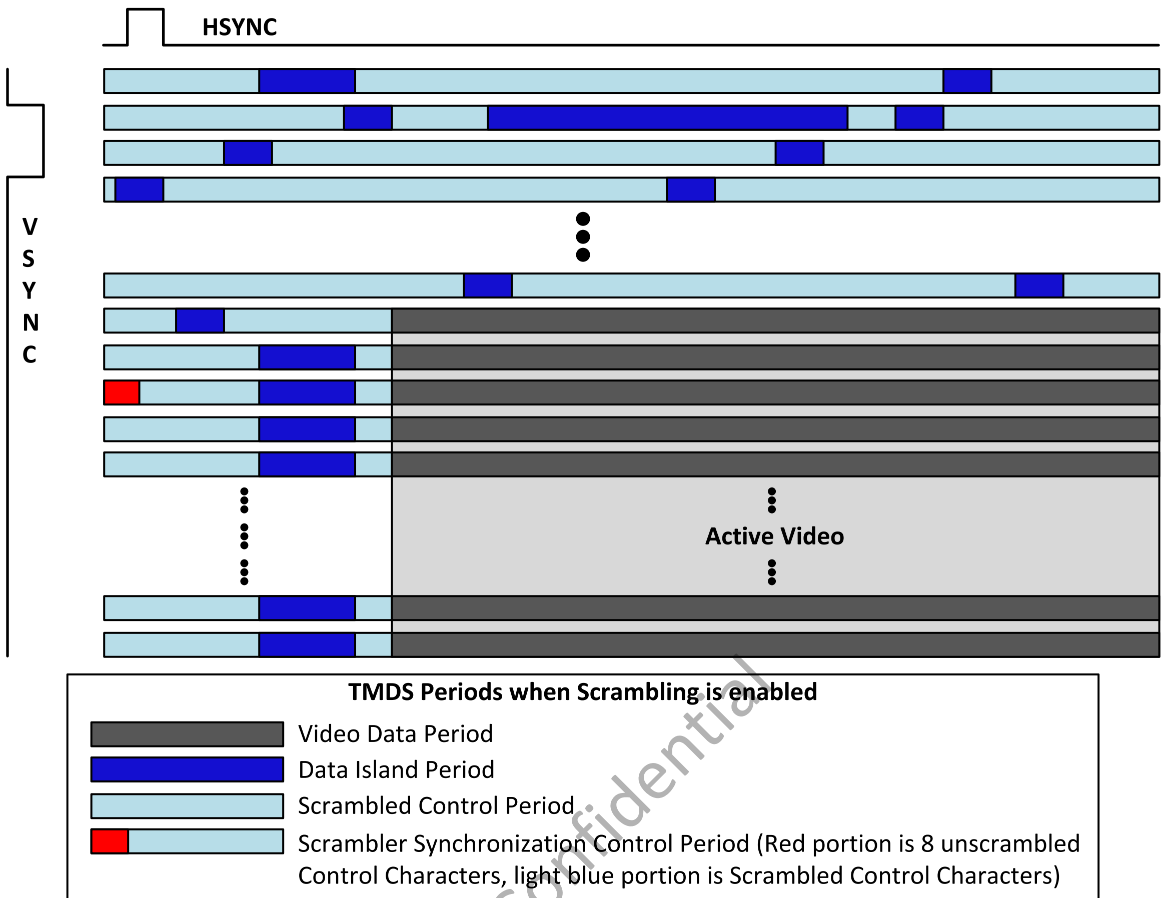


Figure 6-6: Overview of HDMI Data Transport periods

Figure 6-6 provides a simplified overview of the various periods that comprise an HDMI stream. Video Data Periods are dark gray, Data Island Periods are dark blue, and Scrambled Control Periods are light blue. A portion of one Control Period is shown in Red. This entire Control Period (with a light blue period and a red period) is defined as a Scrambler Synchronization Control Period (SSCP). The Red period represents an eight character period where unscrambled control codes are transmitted.

Scrambling shall be applied to Video Data Periods, Data Island Periods, Guard Bands, and Scrambled Control Periods. One SSCP per field shall be transmitted to maintain character synchronization; see Section 6.1.2.4 for options on where the unscrambled portion can be located. Control period characters are handled as a special case and are described in Sections 6.1.2.4 and 6.1.2.5.

Table 6-9: Summary of scrambling periods

Period	Transmitted Data (TMDS Channel)	Source Side Coding
Video Data Period	Active video Pixels or encrypted active video Pixels	8-bit values Scrambled, then TMDS Encoded
	Video Guard Bands	8-bit values from Table 6-13: Scrambled, then TMDS Encoded
Data Island Period	Channel 0 Data Island Guard Band	One of four 4-bit values (depending on VSYNC, HSYNC): Scrambled, then TERC4 Encoded
	Channel 1, 2 Data Island Guard Band	8-bit values from Table 6-14: Scrambled, then TMDS Encoded
	Packet Data or encrypted Packet Data	4-bit data: Scrambled , then TERC4 Encoded
Scrambled Control Period	Encoded HSYNC, VSYNC and CTL0:3	Transmitted Data encoded to a 4-bit vector, scrambled, then encoded to 10-bit Scrambled Control Period codes (Table 6-17)
Unscrambled Portion of the SSCP [Δ]	Encoded HSYNC, VSYNC and CTL0:3	Transmitted Data encoded to 10-bit TMDS Code per H14b Section 5.4.2

[Δ] 8 sequential Control Characters transmitted during 1 control period 1 time per field

6.1.2.2 Scrambling LFSR

The Linear Feedback Shift Register implements a pseudo-random number generator.

The LFSR generator polynomial is shown in Equation 6-3:

$$G(x) = 1 + x^{11} + x^{12} + x^{13} + x^{16}$$

Equation 6-3: LFSR Generator Polynomial

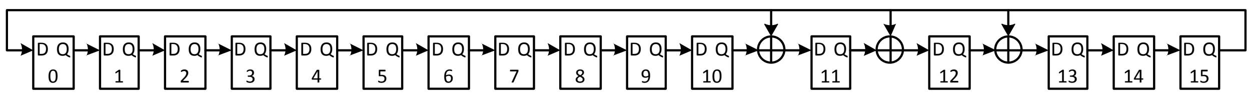


Figure 6-7: LFSR Diagram

The bit assignments shown in Table 6-11 and Table 6-12, Table 6-15, Table 6-16, and elsewhere in the text of this section as LQ[15] through LQ[8] correspond to the flip flop numbering in Figure 6-7.

One LFSR shall be used for each data channel for encoding on the Source side, and one LFSR shall be used for each data channel for decoding on the Sink side.

The Source shall initialize the LFSRs with the appropriate seed values when it transmits the 8-character sequence of Unscrambled Control Codes in the SSCP on the three data channels simultaneously. The Sink shall initialize the LFSRs with the appropriate seed values when it receives an 8-character sequence of Unscrambled Control Codes in the SSCP on the three data channels simultaneously. The SSCP is described in Section 6.1.2.4. LFSR outputs are not used and their states are undefined during the transmission of the Unscrambled Control Code portion of the SSCP. The LFSRs shall be initialized with seed values specified in Table 6-10 such that the LFSRs are set to a seed value during the first character period following the Unscrambled Control Code sequence. The seed values are used to scramble/descramble the first character on each channel following the Unscrambled Control Code sequence.

The LFSRs shall be advanced by eight states per character period during all subsequent character periods until the next sequence of Unscrambled Control Codes.

The first 32 outputs of the LFSRs for each of the 3 data channels are shown in Table 6-10. In Table 6-10, the seed value is the value loaded into each of the LFSRs on both Source and Sink sides for each of the 3 Data Channels, 0, 1, 2. The seed value is loaded into the LFSRs following transmission of a sequence of eight Unscrambled Control Characters in the SSCP such that the seed value becomes the LFSR Output Value and is used to scramble the first character that follows the transmission of the eight Unscrambled Control Codes.

LFSR Output Value 1 represents the 8th state of the LFSRs following the seed value. LFSR Output Value 1 is used to scramble the second character that follows the transmission of eight Unscrambled Control Characters. LFSR Output Value 2 represents the 8th state of the LFSR following LFSR Output Value 1, and so forth.

Table 6-10: First 32 LFSR Values for all Data Channels

LFSR Output Value	Data Channel 0 LFSR Value [15:0]	Data Channel 1 LFSR Value [15:0]	Data Channel 2 LFSR Value [15:0]
Seed Value	FFFF	FFFE	FFFD
1	77EB	76EB	75EB
2	4B7C	737D	3B7E
3	A445	3D78	AE3E
4	DDBD	3838	2EB6
5	EDCE	A03D	7628
6	FEFA	45B9	B07D
7	4AEA	094A	CDAA
8	0A44	8A08	32DD
9	CC0B	5095	CD36
10	ABDC	355C	AEDD
11	14B3	E431	CDB6
12	9B17	C1F2	2EDD
13	BF87	72D0	1D28
14	67A4	A879	C01F
15	CC6F	F9B0	A7D1
16	CFDC	A8ED	01BE
17	34DF	6DB0	8601
18	5F30	5064	4198
19	C052	C45C	C84E
20	EAD1	04D5	0ED8
21	49FD	3504	B00F
22	5547	BC31	BFAA
23	3F59	99A7	4AA4
24	693A	7F85	4444
25	3A60	DD75	CC4B
26	883F	25CE	EBDC
27	1797	BE22	7CFC
28	F714	FAA5	EC76
29	64E2	F5EE	7EFB
30	C26C	EEE0	9B74
31	A4D3	98F9	DC87

6.1.2.3 Scrambling Video Data, Data Island, and Guard Bands

On the Source side, all video data and Data Islands shall be scrambled by performing an XOR operation of the data and the specific bits of the LFSR shown in Table 6-11 and Table 6-12. The scrambled data is then encoded to TMDS for video data or TERC4 for Data Islands, as described in H14b Section 5.4.4 and H14b Section 5.4.3, respectively.

On the Sink side, scrambled video Pixels or Data Islands are extracted from TMDS or TERC4 codes. An XOR operation of the data and the specific bits of the Sink's LFSR shown in Table 6-11 and Table 6-12 shall be performed to reverse the scrambling operation and to yield the original data.

If both scrambling and encryption is enabled, the Source shall perform the encryption operation before the scrambling operation and the Sink shall perform the descrambling operation before the decryption operation.

Table 6-11: Bit assignments for XOR logic operation for 8-bit data

Data Output Bit	Logic Equation
SD[0]	D[0] **XOR** LQ[15]
SD[1]	D[1] **XOR** LQ[14]
SD[2]	D[2] **XOR** LQ[13]
SD[3]	D[3] **XOR** LQ[12]
SD[4]	D[4] **XOR** LQ[11]
SD[5]	D[5] **XOR** LQ[10]
SD[6]	D[6] **XOR** LQ[9]
SD[7]	D[7] **XOR** LQ[8]
D[x] is bit [x] of the Video or Guard Band Data[7..0], SD[x] is bit [x] of the output of the XOR operation, LQ[x] is the Q output of LFSR flip-flop [x].	

Table 6-12: Bit assignments for XOR logic operation for 4-bit data

Data Output Bit	Logic Equation
SD[0]	D[0] **XOR** LQ[15]
SD[1]	D[1] **XOR** LQ[14]
SD[2]	D[2] **XOR** LQ[13]
SD[3]	D[3] **XOR** LQ[12]
D[x] is bit [x] of the 4-bit Data Island data, SD[x] is bit [x] of the output of the XOR operation, LQ[x] is the Q output of LFSR flip-flop [x].	

Scrambling of Video Guard Bands is achieved by an XOR operation of the 8-bit data words shown in Table 6-13 with 8 bits of the LFSR as shown in Table 6-11, prior to TMDS Encoding.

Table 6-13: 8-bit values that map to the TMDS Video Guard Band Codes

TMDS Channel	8-bit input to XOR operation [7..0]
0	0b10101011
1	0b01010101
2	0b10101011

For Data Island Guard Bands, H14b Section 5.2.3.3 specifies that TMDS Channel 0 is encoded as one of 4 TERC4 values. For This Specification, the 4-bit data is scrambled by an XOR operation with 4 bits of the LFSR as shown in Table 6-12, prior to TERC4 encoding.

Data Island Guard Bands for Channels 1 and 2 are scrambled in a manner similar to the Video Guard Bands.

Scrambling of Data Island Guard Bands is achieved by an XOR operation of the 8-bit data words shown in Table 6-14 with 8 bits of the LFSR as shown in Table 6-11, prior to TMDS Encoding.

Table 6-14: 8-bit values that map to the TMDS Data Island Guard Band Codes

TMDS Channel	8-bit input to XOR operation [7..0]
0	N/A
1	0b01010101
2	0b01010101

When scrambling is enabled, the data stream disparity (Cnt) variable used in the TMDS Video Data Encode Algorithm described in H1.4b Figure 5-7, shall function as follows:

- Cnt shall be zero during Control Periods.
- Cnt shall track data stream disparity as described in H1.4b, Figure 5-7 during Data Island Periods and Video Data Periods.

Scrambling is applied to both Data Island Guard Bands, and to Data Island TERC4 codes. It is possible for a pair of 10-bit TMDS characters representing a scrambled trailing Data Island Guard Band to have the same values as a 10-bit TMDS character representing scrambled TERC4 data. Therefore, it is recommended that Sinks do not rely exclusively on decoding trailing Data Island Guard Bands to detect the end of a Data Island period.

6.1.2.4 Scrambler Synchronization Control Periods

Scrambler Synchronization Control Periods are special control periods in which both scrambled and unscrambled control codes are transmitted. Unscrambled control codes are used for character synchronization, inter-channel synchronization, and to reset the LFSRs. These codes are transmitted during the Scrambler Synchronization Control Period (SSCP). The Source shall transmit eight unscrambled control characters during one SSCP once per each Video Field on the 3 data channels, simultaneously. The Source may transmit the Unscrambled Control Character sequence at any point within an SSCP and may use any control period for the SSCP. The Source shall limit the number of unscrambled control codes to exactly eight characters. During an SSCP, the Source shall scramble control codes that are transmitted before or after the sequence of eight unscrambled control codes. H14b Section 5.2.1 and H14b Section 5.4.2 describe the generation of the eight unscrambled control codes transmitted during an SSCP.

In summary, the Source shall:

- Transmit unscrambled control codes on all three data channels simultaneously.
- Transmit one SSCP with one set of eight unscrambled control codes once per field.
- Limit the number of unscrambled control codes to exactly eight.
- Reinitialize the LFSRs on the Source side using the values shown in Table 6-10 when the unscrambled control codes are transmitted, see Section 6.1.2.2.

The Sink shall reinitialize the LFSRs on the Sink side using the values shown in Table 6-10 when the block of eight unscrambled control codes is received, see Section 6.1.2.2.

6.1.2.4.1 CTS Testing Considerations

In normal video transmission, one SSCP is transmitted per field. To accommodate CTS testing, a Sink shall also support reception of more frequent SSCPs in addition to supporting one SSCP per field. CTS test equipment may send the SSCP every one or more horizontal lines.

During CTS testing, the Sink might receive one SSCP per one or more horizontal lines. The Sink shall reinitialize the LFSRs for the three data channels whenever it determines that it has received unscrambled control characters simultaneously on the three data channels. This will ensure correct behavior both during normal operation and during CTS testing.

6.1.2.5 Scrambled Control Periods

Except for the control periods described in Section 6.1.2.4, all other control periods are scrambled.

To support scrambling of control periods, a controlVector and an IToggle value are defined in this section.

The 4-bit control Vector is defined as follows:

- controlVector[3:0] = {D1, D0, LN1, LN0}
- where D1 and D0 are encoded as shown in H14b Table 5-34.
- LN1 and LN0 encode one of 3 TMDS channels (Channel0 = 0b00, Channel1 = 0b01, Channel2 = 0b10).

Description of the generation of the IToggle bit is based on the variables summarized in Table 6-15.

Table 6-15: IToggle Bit Generation Variables

Value	Definition
ScrmblrCnt(n-1)	DC Disparity count of the previous access of the Scrambled Control Codes (Table 6-17)
ScrmblrCnt(n)	DC Disparity count of the current access of the Scrambled Control codes (Table 6-17). This value shall be 0 during the first character period following the Unscrambled Control Code sequence of the SSCP.
LQ[15]	LQ[15] is the Q output of LFSR flip-flop [15].

Utilizing the variables of Table 6-15, the IToggle bit generation method is described in the following pseudo code. Alternative implementations are possible but, given the same input data stream, they are required to generate the same output data stream as the pseudo code.

```
// Initial ScrmblrCnt() value is ScrmblrCnt()=0.
// Inputs are ScrmblrCnt(n-1), LQ[15], Output is IToggle

If     (ScrmblrCnt(n-1) ==  2)
   IToggle = 0
   ScrmblrCnt(n)  = 0
elseIf (ScrmblrCnt(n-1) == -2)
   IToggle = 1
   ScrmblrCnt(n)  = 0
elseIf (ScrmblrCnt(n-1) ==  0)

   IToggle = LQ[15]

   If LQ[15] =  0
      ScrmblrCnt(n) = -2
   else
      ScrmblrCnt(n) =  2
   End
End
```

Note that the Sink device does not need to track the value of IToggle since the two values that it is used to select between will decode to the same unscrambled value.

The 4-bit controlVector is scrambled by XOR operation with 4 bits from the LFSR as shown in Table 6-16. The scrambled output is combined with the IToggle bit to create a 5 bit Scrambled Control Vector (SCV) as shown in Table 6-16.

Table 6-16: Bit assignments for XOR logic operation for 4-bit data

Scrambled Control Vector	Logic Equation
SCV[0]	IToggle
SCV[1]	controlVector[0] **XOR** LQ[14]
SCV[2]	controlVector[1] **XOR** LQ[13]
SCV[3]	controlVector[2] **XOR** LQ[12]
SCV[4]	controlVector[3] **XOR** LQ[11]
controlVector[x] is bit [x] of the 4-bit controlVector. SCV[x] is the IToggle bit for x==0 and bit [x] of the output of the XOR operation, for x==1 to 4. LQ[x] is the Q output of LFSR flip-flop [x].	

The SCV determined according to Table 6-16 is the index or address used to select a scrambled control code from Table 6-17.

Table 6-17: 10-bit codes for scrambled control periods

SCV	10-bit Code [9:0]	SCV	10-bit Code [9:0]
0	0000010111	1	1111101000
2	0000011011	3	1111100100
4	0000011101	5	1111100010
6	0000011110	7	1111100001
8	0000100111	9	1111011000
10	0000110011	11	1111001100
12	0000111001	13	1111000110
14	0000111100	15	1111000011
16	0001000111	17	1110111000
18	0001100011	19	1110011100
20	0001110001	21	1110001110
22	0001111000	23	1110000111
24	0010000111	25	1101111000
26	0011000011	27	1100111100
28	0011100001	29	1100011110
30	0011110000	31	1100001111

6.1.3 Control

6.1.3.1 Scrambling Control

The scrambling capability of a Sink shall be indicated in the HF-VSDB LTE_340Mcsc_scramble bit, and (indirectly) the Max_TMDS_Character_Rate field. These bit fields are defined in Section 10.3.2. The Source shall control scrambling based on the setting of these bits and the conditions under which scrambling is required or optional, as defined in Section 6.1.2.

The Scrambling Control mechanisms described in this section shall be used by Sources and Sinks whenever scrambling is used, irrespective of the TMDS Character Rate. A Sink that has scrambling capability shall provide the following two control bits accessible by the Source through SCDC (Section 10.4): Scrambling_Enable (Section 10.4.1.3) shall be readable and writable from the Source's perspective; Scrambling_Status (Section 10.4.1.5) shall be Read-Only from the Source's perspective. The Sink shall clear the Scrambling_Status bit to a 0 anytime that the Scrambling_Enable bit is a 0.

The Source enables scrambling as part of a link initialization procedure. The Source shall write a 1 to the Scrambling_Enable bit in the Sink to enable scrambling in the Sink. The Source shall transmit a scrambled video signal following the write to the Scrambling_Enable bit.

The minimum time period between the write to the Scrambling_Enable bit, and the transmission of a scrambled video signal is not specified; however the Source shall not begin transmission of a scrambled video signal before writing a 1 to the Scrambling_Enable bit. The maximum time period between the write to the Scrambling_Enable bit and the transmission of a scrambled video signal shall be 100 ms.

The Sink shall detect a scrambled video signal following the Source's write of a 1 to the Scrambling_Enable bit and the detection process described in the following sentences. The Sink shall set the Scrambling_Status bit to a 1 if the Sink detects receipt of the scrambled control codes specified in Table 6-17, and two SSCPs, defined in Section 6.1.2.4. The Sink may set the Scrambling_Status bit to a 1 earlier (but after the receipt of scrambled control codes) if it determines by other means that it is receiving and properly decoding scrambled control codes. Thereafter, while scrambled control codes are detected, and for each received unscrambled Control Sequence, the Sink shall update the Scrambling_Status bit by setting it to a 1. If the Sink fails to detect scrambled control codes and the eight character long unscrambled control sequences for a period of time equal to 2 field periods of the currently transmitted Video Format, then the Sink shall clear the Scrambling_Status to a 0. The Sink shall apply scrambling when Scrambling_Status=1.

A Sink may elect to control audio and video muting independently of the scrambling enable procedure. In this case, the Sink may control muting of audio and/or video at any time and is not required to wait for the receipt of scrambled control codes and two sequential unscrambled control sequences, as described in the previous paragraph, before muting or unmuting audio and/or video.

The Source should poll the Scrambling_Status bit following the write of Scrambling_Enable to a 1 and following transmission of scrambled video. Polling should continue until reads of the Scrambling_Status bit return a 1. If the read of the Scrambling_Status bit returns a 1, the Source has verification that the TMDS link is functioning correctly with scrambling enabled. If the Scrambling_Status bit returns a 0, the Source should continue to poll the Scrambling_Status bit up to a maximum period of 200 ms from the start of scrambled video transmission. If the Scrambling_Status bit returns a 0 after 200 ms of polling, then the TMDS link is not functioning and an error condition is indicated.

6.1.3.2 Control for TMDS Bit Period/TMDS Clock-Period Ratio

The relationship between TMDS Bit Rate and TMDS Clock Rate is described in Section 6.1.1.

A sink that supports TMDS Bit Rates above 3.4 Gbps shall provide a read/write control bit, TMDS_Bit_Clock_Ratio (Section 10.4.1.3), accessible by the Source through SCDC (Section 10.4).

When the Source resets (=0) the TMDS_Bit_Clock_Ratio bit, it informs the Sink that the TMDS Bit Period/TMDS Clock Period ratio of the transmitted TMDS signal is 1/10. When the Source sets (=1) the TMDS_Bit_Clock_Ratio bit, it informs the Sink that the TMDS Bit Period/TMDS Clock Period ratio of the transmitted TMDS signal is 1/40.

When configuring the TMDS link for operation below 3.4 Gbps, the Source shall ensure that the TMDS_Bit_Clock_Ratio bit in the Sink is reset (=0), either by writing a 0, or by reading to the TMDS_Bit_Clock_Ratio bit to verify a 0 state, and the Source shall transmit TMDS clock and data signals that comply with HDMI 1.4b.

When configuring the TMDS link for operation with TMDS Bit Rates in the range from 3.4 Gbps to 6.0 Gbps, the Source shall set (=1) the TMDS_Bit_Clock_Ratio bit with an SCDC write, and the Source shall transmit TMDS clock and data signals that comply with parameters in Section 6.1.1.4 of This Specification. Following transmission of TMDS clock and data, a Source may read the Clock_Detected status flag in SCDC status register described in Section 10.4.1.7 to verify that the Sink has detected the transmitted clock signal.

Whenever the Source changes the TMDS_Bit_Clock_Ratio bit from a 0 to a 1, or from a 1 to a 0, the Source shall follow the following procedure:

1. The Source shall suspend transmission of the TMDS clock and data.
2. The Source shall write to the TMDS_Bit_Clock_Ratio bit to change it from a 0 to a 1 or from a 1 to a 0.
3. The Source shall allow a minimum of 1 ms and a maximum of 100 ms from the time the TMDS_Bit_Clock_Ratio bit is written until resuming transmission of TMDS clock and data at the updated data rate.
4. The Source may read the state of the Clock_Detected status bit via the SCDC to verify that the Sink is detecting the TMDS clock.

6.2 Character Error Detection

Character Error Detection provides a mechanism for the Sink device to report the number of Character Errors it has detected. This can be used by the Source as a check on link quality by sampling the Error Counters at periodic intervals.

6.2.1 Feature Overview

An HDMI receiver implements character error detection, together with an Error Counter for channels 0, 1, and 2. The receiver checks each incoming character as to whether it is valid in context. If the character is not valid, then the receiver increments an Error Counter associated with the channel on which the character was received. The transmitter may read the Error Counter at any time. The Error Counter is cleared when read.

Due to the nature of TMDS encoding, it is not possible to identify all single-bit errors that might occur in transmission. However, during reception of video data, in about 24% of the cases, a single bit error in an otherwise valid 10-bit TMDS character will have the result of converting the character into an invalid 10-bit character. In the remaining 76% of the cases, a single bit error will convert a valid 10-bit TMDS character into a different, but valid, 10-bit TMDS character. This is based on the assumption that single bit errors will occur at bit value transitions, due to accumulated ISI during the previous run length coupled with the impact of random jitter.

The introduction of a single bit error necessarily upsets the running disparity. This property is exploited to detect errors a statistical number of characters after the errored character.

The checking mechanism is described as a receiver state machine that exploits the properties of the HDMI protocol in the following way:

- During the Control reception state, the only valid characters that may be received are Control characters (which include Preamble characters) and leading Guard Band characters (a limited subset of TMDS Data characters). A leading Guard Band character causes a transition to the leading Guard Band reception state.
- During the leading Guard Band reception state, the only valid characters are Guard Band characters and TMDS or TERC4 characters. A TMDS character causes a transition to the Video Data reception state. A TERC4 character causes a transition to the Data Island Data reception state.
- During the Data Island Data reception state, the only valid characters are TERC4 and Control characters. A Control character causes a transition to the Control reception state.
- During the Video Data reception state, the only valid characters are TMDS Data characters (including the trailing Guard Band characters) and Control characters. A Control character causes a transition to the Control reception state. A trailing Guard Band character does not cause a state transition.

The essence of the technique is to apply checks during each state to detect if any characters that are not valid for that state are received.

When a TMDS character is transmitted in HDMI, the selection of the transmitted character is based on the 8-bit value being encoded and the current disparity. If the current disparity is negative, one group of 256 transmitted characters is possible. Likewise, if the current disparity is zero, a second group of 256 transmitted characters is possible. Finally, if the current disparity is positive, a third group of 256 transmitted characters is possible. Each transmitted character is required to be in one of these three groups. During reception of video data, the checking mechanism tracks the data stream disparity (denoted as Cnt as defined in H14b Table 5-35). The receiver checks that the received character is in the correct group of 256 valid characters given the value of Cnt(t-1).

For non-Video Data Periods, various optimizations are employed in Section 6.2.3 to simplify the implementation.

The encodings used for Control and Preamble characters form a very small set, and so no distinction is made between reception of Control characters and Preamble characters.

Likewise, the Guard Band lasts for a very short period of time (two character periods) and so, there is very little advantage in the explicit checking of Guard Band characters.

Furthermore, Data Islands occur relatively rarely. Distinguishing between Data Islands and Video Data Periods require interpretation of the Preamble across all three channels. Apart from this, each channel can be treated completely independently. So while checking explicitly for valid TERC4 (Data Island) characters is allowed, implementations may treat Data Islands and Video Data Periods identically. If Data Islands are treated separately, then it should be noted that, when scrambling is not in use, the Data Island Guard Band characters are not in the same subset of video data characters as Data Island data characters, and so the check needs to include these as well as the TERC4 character checks. When scrambling is used, the Guard Band characters can be any of the characters used for video data.

The result of the optimizations in Section 6.2.3 is to define the error checking function by means of a state machine with two main states: “Control Period” and “Data period”.

During the reception of video data, if an error occurs and it is immediately detected, the Error Counter is incremented. If an error occurs but is not immediately detected, the effect will be to create an error in the tracked value of the data stream disparity (i.e. Cnt(t-1)). This in turn may cause Cnt(t-1) to go from negative to zero, or from zero to positive, or vice versa. If Cnt(t-1) has been disrupted in this way, then there is a chance that a correct character will be checked against the wrong group of characters, resulting in a false error detection. The net effect is that the occurrence of the error is detected a few characters later. This effect is described as delayed detection. The latency to detection is, on average, slightly less than 5 characters. When this happens, the Error Counter is incremented (thus tracking the actual error that occurred).

When the Error Counter is incremented during the reception of video data, a variant state, called loose checking, is enabled. In this state, a character is considered valid if it would be valid for any of Cnt(t-1) < 0, Cnt(t-1) = 0, Cnt(t-1) > 0. Loose checking avoids further false detections, which otherwise would result in an erroneously high error rate being reported.

HDMI requires the character error rate to be extremely low (H14b Section 4.2.5 specifies 10^{-9} or lower), so errors should be extremely sparse. When loose checking is enabled, the probability of detecting an error reduces. However, loose checking only lasts for the duration of the scan line in which the error was detected. Given that errors are relatively sparse (e.g. 10^{-4} or lower), the impact on the accuracy of the technique is negligible (the technique is aimed at measuring error rates at 10^{-9} or lower). Note that during testing, character error rates of 10^{-4} or higher are readily detectable by visual inspection.

There is a small probability that an error occurs, but is not detected, sufficiently near the end of a scan line that the end of the line is reached before delayed detection is triggered. The net result is to slightly under-report the error rate, but again, the impact is negligible.

The character error detection mechanism can be used to provide an estimate of the Character Error Rate. A reasonable estimate of the HDMI Character Error Rate (required to be less 10^{-9}) can be obtained by measuring over $10*(1/10^{-9})$ characters. At a TMDS Character rate of 25 Mcsc, this translates to about 40 s of measuring time. At a TMDS Character Rate of 600 Mcsc, this translates to about 1.67 s. The Error Counters would be read at the start of the measurement period in order to clear them, and again at the end of the measurement period.

The character error detection mechanism can be used by the Source as a check on link quality by sampling the Error Counters at periodic intervals.

6.2.2 Error Counters

Separate Error Counters shall be maintained for each of the three channels.

Each Error Counter shall be 15 bits long, and shall be mapped into two bytes of the SCDC Source accessible registers as defined in Section 10.4.1.8. The lower addressed byte contains the least significant 8 bits of the Error Counter, and the higher addressed byte contains the most significant 7 bits of the Error Counter. The topmost bit of the Error Counter is a “Valid” flag.

Error checking shall start as soon as the receiver has achieved character lock with the incoming data stream on the appropriate channel. The Valid flag shall be set as soon as error checking starts, and shall not be cleared until the receiver detects that the +5V Power Signal on the HDMI cable is not asserted or the Sink is placed into standby. In particular, if the receiver loses sync with the incoming signal, then the Valid flag shall remain set and the Error Counter shall not be cleared. When the Valid flag is not set, the values contained in the Error Counters are undefined and shall be ignored by the Source.

The Error Counter shall be incremented whenever an error is detected, until it reaches its maximum value (0x7FFF). At that time, it shall not be incremented or “wrapped round”; it shall stay at its maximum value.

The Error Counter shall be cleared immediately after it is read by the Source. The Source is required to read both bytes of all three counters and the checksum in the same SCDC transaction, and the receiver shall provide a coherent result (it shall avoid the effects of a carry between the first byte and the second byte of each counter, adjacent counters, and the checksum due to an error detected during the read).

The Sink shall not clear the Error Counters under any other circumstances (e.g. the Sink will not clear the Error Counters on any access made by the Sink for internal purposes).

The Sink shall maintain a second set of three Error Counters, one counter for each data channel, for the purpose of determining the locked status of each channel in the receiver. The lock status for each channel is indicated by the SCDC Ch0_locked, Ch1_locked, and Ch2_locked bits (see Section 10.4.1.7). Lock is determined by utilizing the error detection methodology to verify that the character error rate is lower than 10^{-4}.

When the Sink determines that it is decoding a received stream, the Sink shall count the character errors detected for each channel over a period of 10 ms. A channel shall be deemed to be locked if it accumulates a maximum of one error per Mcsc over the 10 ms period. For example, in a 27Mcsc stream, a maximum of 27 errors on a channel may be accumulated over each 10 ms period if a sink is to declare a channel locked. Similarly, for a 597Mcsc channel, a maximum of 597 errors may be detected on a channel over a 10 ms period. For each channel, if the Error Count is less than or equal to the maximum allowed over this period, the Sink shall set the corresponding Ch0/1/2_locked bit to 1. If more than the maximum allowed errors are measured over this period the Sink shall clear the corresponding Ch0/1/2_locked bit to 0. This process shall repeat continuously such that the lock status is evaluated every 10 ms.

In addition, when the link is not active (i.e. +5V is not available, or the sink otherwise determines that no TMDS data is being transported) or when the Clock_Detected bit is set to zero, the Ch0_locked, Ch1_locked, and Ch2_locked bits shall all be set to 0.

6.2.3 Reference Implementation

The following is a description of the error checking algorithm used during the reception of video data. A detailed description of an error checker is given. Given an error rate that results in less than one error per active video line or vertical blank line, and randomly distributed errors, this algorithm detects at least 85% of errored characters. Other implementations are possible and are permitted but, given an error rate in the input stream of between of 10^{-6} and 10^{-9} and randomly distributed errors, they shall detect at least 85% of errored characters, measured over sufficient time for at least 1000 errors.

If Data Islands are treated separately (not shown in this Reference Implementation) then the check shall include these as well as the TERC4 character checks.

The Reference Implementation contains four 1-bit constant arrays, each of dimension 1024. Each offset in the arrays represents a potential received 10-bit codeword value. The values are received LSb first on the serial stream (as defined by H14b Section 5.4.1) and then formed into 10-bit words organized with the MSb in the leftmost position. The received 10-bit value is used as an index into the appropriate array. The array has the indexed value set (=1) if the 10-bit value is valid and set to 0 if the 10-bit value is invalid. One array is implemented for valid characters received during Control Periods and the remaining three arrays (corresponding to the groups described in Section 6.2.2) are implemented for valid characters received for each of the three possible values of Cnt(). Checking for one of the two possible 10-bit values for Guard Bands is performed explicitly.

The Reference Implementation is specified by the following C code[1]. First, the header file:

```
/*
 *  HDMI TMDS Symbol Error Checker
 */

void resetErrorChecker();
int readErrorCnt();
bool checkSymb(int D[10]);
```

Next, is the main code file:

```
/*
 *  HDMI TMDS Symbol Error Checker
 */

#include "SymbErrChck.h"

/*
  this routine implements the proposed symbol error checking function for HDMI Sink Devices
  this routine is implemented for each channel independently
*/

//  ctrl_period_valid (36 total)
//     Unscrambled Control Characters (4)
//     SVC Characters for scrambled control periods (32)
const int ctrl_period_valid[1024] = {
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,1,0,1,1,0,
  0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,1,0,0,1,0,0,0,
  0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,1,0,0,0,0,0,0,0,
  0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,
  0,0,0,0,0,0,0,1,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,
  0,0,0,1,0,0,1,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,
  0,1,1,0,1,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
};// 36 elements
```

[1] This material is included in electronic form in the Companion Documentation package.

```
//  data_period_cnt_neg_valid (256 total)
//      Data Characters generated if bias before encoding was negative (256)
//      Unscrambled Guard Band Characters (2, but replicate data characters so do not add to
//      the total)
//      TERC4 Characters (16, but replicate data characters so do not add to the total)
const int data_period_cnt_neg_valid[1024] = {
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,1,0,1,1,1,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,
  0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,1,0,0,0,1,0,1,0,1,1,1,1,1,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,1,1,
  0,0,0,0,0,0,0,1,0,0,0,0,0,0,1,1,0,0,0,0,0,0,0,0,0,0,0,0,1,0,1,1,
  0,0,0,1,0,0,1,1,0,0,0,0,1,0,1,1,0,1,1,1,1,0,1,1,1,1,1,1,1,1,1,1,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,1,0,1,1,1,
  0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,1,0,0,0,1,0,1,0,1,1,1,1,1,
  0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,
  0,0,0,1,0,0,0,1,0,0,0,0,0,0,0,1,0,1,0,1,0,0,0,1,1,1,0,1,1,1,1,1,
  0,0,0,0,0,0,0,1,0,0,0,0,0,0,1,1,0,0,0,0,0,0,0,0,0,0,0,0,1,0,1,1,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,1,0,1,1,
  0,0,0,1,0,0,1,1,0,0,0,0,1,0,1,1,0,0,0,0,0,0,0,0,1,0,0,0,1,0,1,1,
  0,1,1,1,1,0,1,1,1,0,0,0,1,0,1,1,1,1,1,1,1,0,1,1,1,1,1,1,1,1,1,1,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,1,0,1,1,1,
  0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,1,0,0,0,1,0,1,0,1,1,1,1,1,
  0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,
  0,0,0,1,0,0,0,1,0,0,0,0,0,0,0,1,0,1,0,1,0,0,0,1,1,1,0,1,1,1,1,1,
  0,0,0,0,0,0,0,1,0,0,0,0,0,0,1,1,0,0,0,0,0,0,0,0,0,0,0,0,1,0,1,1,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,1,0,1,1,
  0,0,0,1,0,0,1,1,0,0,0,0,1,0,1,1,0,0,0,0,0,0,0,0,1,0,0,0,1,0,1,1,
  0,1,1,1,1,0,1,1,1,0,0,0,1,0,1,1,1,1,1,1,1,0,1,1,1,1,1,1,1,1,1,1,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,1,0,1,1,1,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,
  0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,1,0,0,0,1,0,1,0,1,1,1,1,1,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,1,1,
  0,0,0,0,0,0,0,1,0,0,0,0,0,0,1,1,0,0,0,0,0,0,0,0,0,0,0,0,1,0,1,1,
  0,0,0,1,0,0,1,1,0,0,0,0,1,0,1,1,0,1,1,1,1,0,1,1,1,1,1,1,1,1,1,1
};// 256 elements
```

```
//  data_period_cnt_zero_valid (256 total)
//     Data Characters generated if bias before encoding was Zero (256)
//     Unscrambled Guard Band Characters (2, but replicate data characters so do not add to
//     the total)
//     TERC4 Characters (16, but replicate data characters so do not add to the total)
const int data_period_cnt_zero_valid[1024] = {
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  1,1,1,1,1,1,1,1,1,1,0,1,1,1,1,1,1,1,0,1,0,0,0,1,1,1,0,1,1,1,1,1,
  1,1,0,1,0,0,0,1,0,0,0,0,0,0,0,1,1,1,0,1,0,0,0,1,1,1,0,1,1,1,1,1,
  1,1,0,1,0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,
  1,1,0,1,0,0,0,1,0,0,0,0,0,0,0,1,1,1,0,1,0,0,0,1,1,1,0,1,1,1,1,1,
  1,1,1,1,1,0,1,1,1,0,0,0,1,0,1,1,1,0,0,0,0,0,0,0,1,0,0,0,1,0,1,1,
  1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,1,0,1,1,
  1,1,1,1,1,0,1,1,1,0,0,0,1,0,1,1,1,0,0,0,0,0,0,0,1,0,0,0,1,0,1,1,
  1,1,1,1,1,0,1,1,1,0,0,0,1,0,1,1,1,1,1,1,1,0,1,1,1,1,1,1,1,1,1,1,
  1,1,1,1,1,1,1,1,1,1,0,1,1,1,1,1,1,1,0,1,0,0,0,1,1,1,0,1,1,1,1,1,
  1,1,0,1,0,0,0,1,0,0,0,0,0,0,0,1,1,1,0,1,0,0,0,1,1,1,0,1,1,1,1,1,
  1,1,0,1,0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,
  1,1,0,1,0,0,0,1,0,0,0,0,0,0,0,1,1,1,0,1,0,0,0,1,1,1,0,1,1,1,1,1,
  1,1,1,1,1,0,1,1,1,0,0,0,1,0,1,1,1,0,0,0,0,0,0,0,1,0,0,0,1,0,1,1,
  1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,1,0,1,1,
  1,1,1,1,1,0,1,1,1,0,0,0,1,0,1,1,1,0,0,0,0,0,0,0,1,0,0,0,1,0,1,1,
  1,1,1,1,1,0,1,1,1,0,0,0,1,0,1,1,1,1,1,1,1,0,1,1,1,1,1,1,1,1,1,1,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
};// 256 elements
```

```
//  data_period_cnt_pos_valid (256 total)
//      Data Characters generated if bias before encoding was Positive (256)
//      Unscrambled Guard Band Characters (2, but replicate data characters so do not add to
//      the total)
//      TERC4 Characters (16, but replicate data characters so do not add to the total)
const int data_period_cnt_pos_valid[1024] = {
  1,1,1,1,1,1,1,1,1,1,0,1,1,1,1,0,1,1,0,1,0,0,0,0,1,1,0,0,1,0,0,0,
  1,1,0,1,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,0,0,1,0,0,0,0,0,0,0,
  1,1,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  1,1,1,1,1,0,1,0,1,0,0,0,1,0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,
  1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  1,1,1,0,1,0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  1,1,1,1,1,1,1,1,1,1,0,1,1,1,1,1,1,1,0,1,0,0,0,1,1,1,0,1,1,1,1,0,
  1,1,0,1,0,0,0,1,0,0,0,0,0,0,0,0,1,1,0,1,0,0,0,0,1,1,0,0,1,0,0,0,
  1,1,0,1,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  1,1,0,1,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,0,0,1,0,0,0,0,0,0,0,
  1,1,1,1,1,0,1,1,1,0,0,0,1,0,1,0,1,0,0,0,0,0,0,0,1,0,0,0,1,0,0,0,
  1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,
  1,1,1,1,1,0,1,0,1,0,0,0,1,0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,
  1,1,1,0,1,0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  1,1,1,1,1,1,1,1,1,1,0,1,1,1,1,1,1,1,0,1,0,0,0,1,1,1,0,1,1,1,1,0,
  1,1,0,1,0,0,0,1,0,0,0,0,0,0,0,0,1,1,0,1,0,0,0,0,1,1,0,0,1,0,0,0,
  1,1,0,1,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  1,1,0,1,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,0,0,1,0,0,0,0,0,0,0,
  1,1,1,1,1,0,1,1,1,0,0,0,1,0,1,0,1,0,0,0,0,0,0,0,1,0,0,0,1,0,0,0,
  1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,
  1,1,1,1,1,0,1,0,1,0,0,0,1,0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,
  1,1,1,0,1,0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  1,1,1,1,1,1,1,1,1,1,0,1,1,1,1,0,1,1,0,1,0,0,0,0,1,1,0,0,1,0,0,0,
  1,1,0,1,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,0,0,1,0,0,0,0,0,0,0,
  1,1,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  1,1,1,1,1,0,1,0,1,0,0,0,1,0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,
  1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  1,1,1,0,1,0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
  1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
};// 256 elements

/*
 Error Counter
 note that there is one for each channel, and each can be cleared independently
*/

static int errorCnt;

/*
 state variables
 the routine tracks whether the next symbol is expected to be a control symbol
 or is expected to be a data symbol (from a Data Island or from video data)
*/

bool inControlPeriod;

/*
 data islands are treated the same as video data periods -
 a straightforward extension can be made to check them separately
*/

/*
 state variable
 when in a data period, the routine tracks the current running disparity
*/

int Cnt;                 // running disparity count
bool looseChecking;
bool periodEnding;
```

```
void resetErrorChecker() {
  errorCnt = 0x8000;      // set the Valid flag when we start processing video data
  inControlPeriod = TRUE; // assumed starting state
  Cnt = 0;
  looseChecking = FALSE;
  periodEnding = FALSE;
}

int readErrorCnt () {
  int temp;
  temp = errorCnt;
  errorCnt = 0x8000;
  return temp;
}

int updateCnt(int D[10]) {
  int i, newCnt;
  newCnt = Cnt;  // value of Cnt after processing this symbol
  for (i=0; i<10; i++) {
    if (D[i] == 1) {
      newCnt++;
    } else {
      newCnt--;
    }
  }
  return newCnt;
}

bool checkSymb(int D[10]) {
  int i;
  int symVal;
  bool foundError = FALSE;

  // first convert to integer for convenience of lookup
  symVal = 0;
  for (i=0; i<10; i++)
  {
    symVal = (symVal<<1) + D[9-i];
  }

  if (inControlPeriod) {
    if (ctrl_period_valid[symVal] == 0) {               // not a valid control symbol
      if ((data_period_cnt_neg_valid[symVal] == 1) ||   // see if it is a valid data character
          (data_period_cnt_zero_valid[symVal] == 1) ||  // (i.e. Guard Band)
          (data_period_cnt_pos_valid[symVal] == 1)) {
        if (periodEnding) { // immediately previous character was a data character
          // end of control period
          inControlPeriod = FALSE;
          looseChecking = FALSE;
          periodEnding = FALSE;
        } else {
          periodEnding = TRUE; // next character might be another data character
        }
        Cnt = updateCnt(D);    // may already be in Data Period and processing TMDS data characters
      } else { // neither a control character nor a data character - definitely an error
        foundError = TRUE;
        periodEnding = FALSE;
      }
    } else { // valid control character
      if (periodEnding) { // last character was an isolated data character, so actually was an error
        foundError = TRUE;
        periodEnding = FALSE;
      } else {
        // all well, no alarm bells
      }
      Cnt = 0;  // reset to zero during Control Periods
    } // valid control character

  } else { // Data Period
```

```
    if (ctrl_period_valid[symVal] == 1) {
      if (periodEnding) { // immediately previous character was a control character
        // end of data period
        inControlPeriod = TRUE;
        periodEnding  = FALSE;
      } else {
        periodEnding = TRUE; // next character might be another control character
      }
    } else { // not a control character

      if (periodEnding) { // previous character was an isolated control character,
                          // so actually was an error
        foundError = TRUE;
        looseChecking = TRUE;
        periodEnding = FALSE;

      } else { // check for valid data character

        if (Cnt > 0) {
          if (!((looseChecking && (data_period_cnt_neg_valid[symVal] == 1)) ||
                (looseChecking && (data_period_cnt_zero_valid[symVal] == 1)) ||
                (data_period_cnt_pos_valid[symVal] == 1)))
          {
            foundError = TRUE;
            looseChecking = TRUE;
          }
        } else if (Cnt == 0) {
          if (!((looseChecking && (data_period_cnt_neg_valid[symVal] == 1)) ||
                (data_period_cnt_zero_valid[symVal] == 1) ||
                (looseChecking && (data_period_cnt_pos_valid[symVal] == 1))))
          {
            foundError = TRUE;
            looseChecking = TRUE;
          }
        } else {    // must be Cnt < 0
          if (!((data_period_cnt_neg_valid[symVal] == 1) ||
                (looseChecking && (data_period_cnt_zero_valid[symVal] == 1)) ||
                (looseChecking && (data_period_cnt_pos_valid[symVal] == 1))))
          {
            foundError = TRUE;
            looseChecking = TRUE;
          }
        } // end Cnt < 0
      } // end check for valid data cheracter
      Cnt = updateCnt(D);  // value of Cnt after processing this symbol
    } // end not a control character
  } // end inDataPeriod

  if (foundError) {
    if (errorCnt != 0xFFFF) {  // not maxed out
      errorCnt++;
    }
  }

  return foundError;
}
```

6.3 Auxiliary Channel Electrical Characteristics

Auxiliary channels are the non-TMDS interfaces defined in HDMI 1.4b. The basic requirements for these interfaces are described in HDMI 1.4b. The following sections provide additional information regarding these interfaces for devices that are compliant with This Specification.

6.3.1 CEC

The requirements specified in H14b Section 4.2.10 including H14b Table 4-40 shall be met. In addition the requirements specified in Table 6-18 shall be met.

Table 6-18: CEC line Electrical Specifications for all Configurations

Item	Rule / Description	Value
Standby and Power-On Characteristics	A device that does not implement the CEC protocol shall not load the CEC bus more than a device that implements CEC. This ensures communication between other CEC devices is not degraded. This may be accomplished by using a compliant CEC line driver that never actively pulls the bus low, or by keeping the CEC pin unconnected. The Adopter is cautioned that the 1.8 µA leakage current requirement of H14b Table 4-40 shall be observed, and that this may prevent the use of even a very weak pull-down resistor on the CEC pin. NOTE - Requirements for devices implementing the CEC Protocol are specified in Section 11.9.1 and H14b Section CEC 4.	see Section 11.9.1

7 Video Extensions

This Specification utilizes all of the Pixel encoding and transmission methods defined in HDMI Specification Version 1.4b. In addition, This Specification provides additional video transmission options and requirements.

7.1 YC_BC_R 4:2:0 Pixel Encoding

When transmitting a Video Format listed in Table 7-1, Source Devices may utilize the YC_BC_R 4:2:0 Pixel Encoding method defined in this section with the VIC set to the corresponding value. Table 7-1 may be supplemented with additional VICs in future versions of This Specification.

Table 7-1: Video Timings that may be used with YC_BC_R 4:2:0 Pixel Encoding

Resolution	Refresh Rate (Hz)	CEA-861-F VIC
3840 x 2160p	50	96, 106
3840 x 2160p	60	97, 107
4096 x 2160p	50	101
4096 x 2160p	60	102

This Specification adds a defined mechanism for transporting YC_BC_R 4:2:0 encoded Pixels. YC_BC_R 4:2:0 video is carried at a TMDS Character Rate equal to ½ the Pixel Clock Rate.

A Source shall not send a Video Format with YC_BC_R 4:2:0 Pixel Encoded data to a Sink that does not indicate support for such format in the Y420CMDB (YC_BC_R 4:2:0 Capability Map Data Block) or Y420VDB (YC_BC_R 4:2:0 Video Data Block), as defined in CEA-861-F Section 7.5.10 and 7.5.11.

When YC_BC_R 4:2:0 Pixel Encoding is active, Pixel repetition is not permitted. The Pixel Repeat (PR) field shall be set to 0 in the AVI InfoFrame when 4:2:0 encoded Pixels are being transmitted.

This Specification does not support the transport of 4:2:0 Pixels for interlaced Video Formats.

Figure 7-1 shows the signal mapping and timing for transferring YC_BC_R 4:2:0 Pixel Encoded progressive video data across HDMI. The two horizontally successive 8-bit Y components are transmitted in TMDS Channel 1 and 2, respectively in order. The 8-bit C_B or C_R components are alternately transmitted in TMDS Channel 0, line by line.

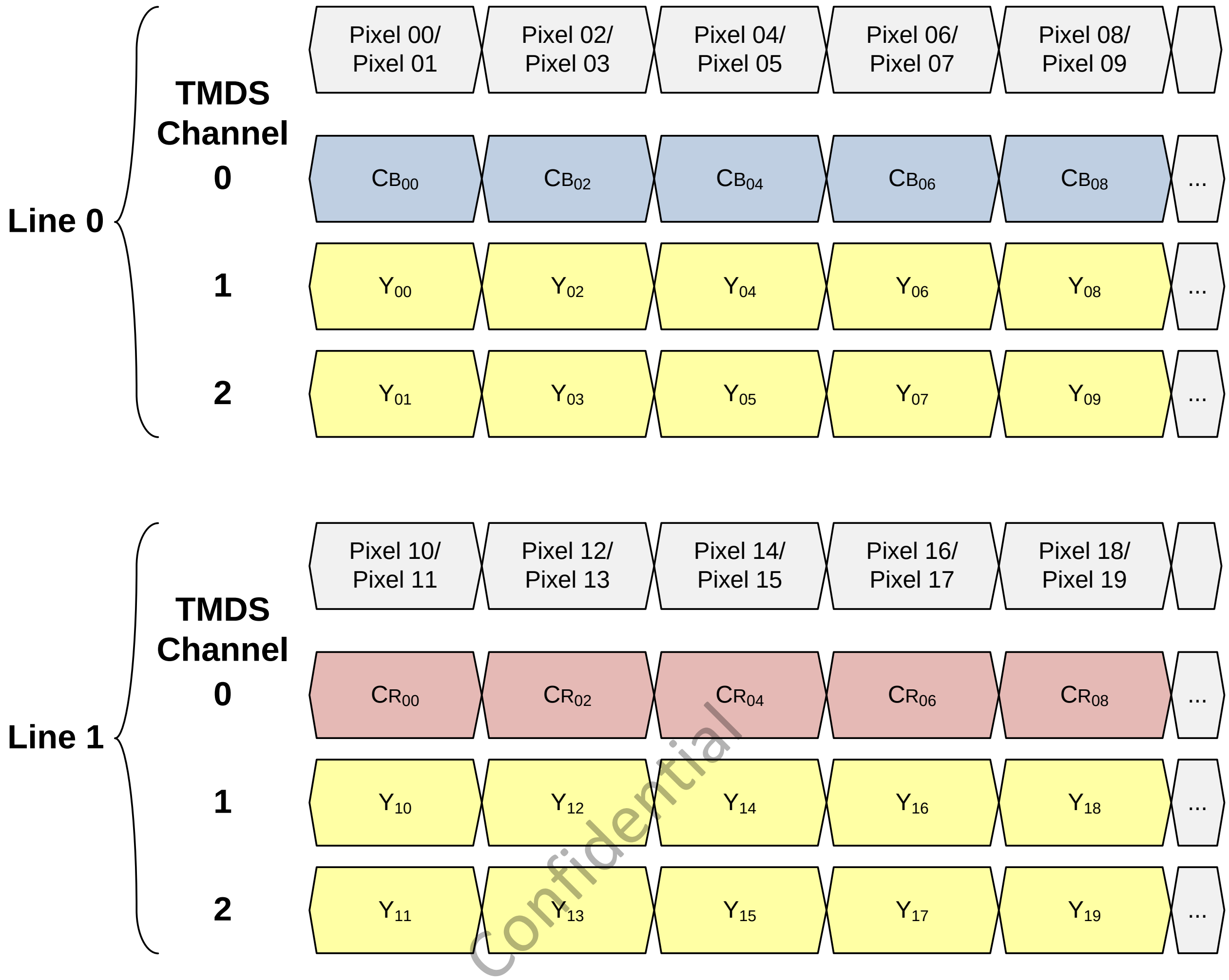


Figure 7-1: YC_BC_R 4:2:0 mapping for progressive Video Formats

The nominal sub-sampling position of YC_BC_R 4:2:0 Pixel Encoded progressive video data is shown in Figure 7-2. Y data is sampled for each Pixel. C_B/C_R data is sampled vertically centered and horizontally to the left of each 4 Pixel quartet.

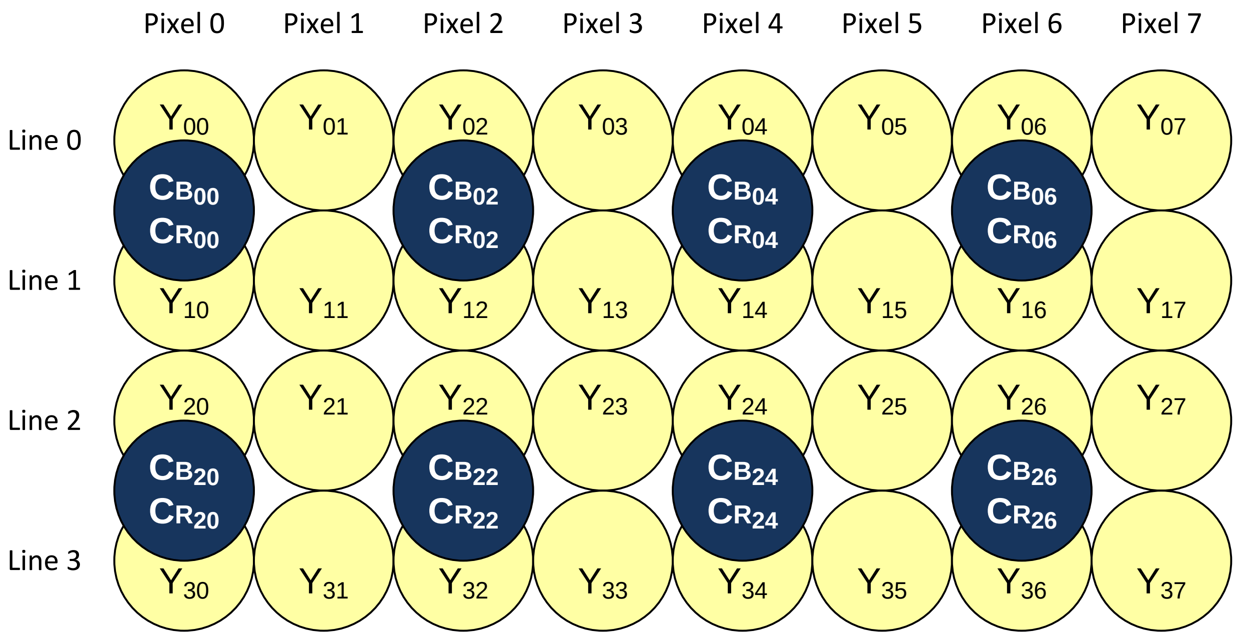


Figure 7-2: Sub-sampling position of YC_BC_R 4:2:0 for progressive Video Formats

7.1.1 Deep Color 4:2:0 Pixel Encoding and Packing

The prior section describes the transport of 4:2:0 encoded Pixels with 8-bits per color component. This Specification also supports Deep Color 4:2:0 Pixel Encoding with options for 10-, 12-, and/or 16-bits per color component. Source or Sink devices may support Deep Color 4:2:0 Pixel Encoding and shall not utilize Deep Color 4:2:0 Pixel Encoding on a particular Video Format that is not also supported by 4:2:0 Pixel Encoding with 8-bits per component.

A Sink capable of supporting Deep Color 4:2:0 Pixel Encoding shall set (=1) the appropriate DC_XXbit_420 bits of the HF-VSDB to indicate which color depths are supported. See section 10.3.2. Sink Devices may support any combination of DC_XXbit_420 bit settings.

A Sink that indicates support for Deep Color 4:2:0 Pixel Encoding, shall support it on all Video Formats indicated in the Y420CMDB (YC_BC_R 4:2:0 Capability Map Data Block) and Y420VDB (YC_BC_R 4:2:0 Video Data Block), as defined in CEA-861-F Section 7.5.10 and 7.5.11, unless that combination exceeds the Max_TMDS_Character_Rate indication in the HF-VSDB (See Section 10.3.2).

A Source shall not send a Deep Color 4:2:0 Pixel Encoded signal to a Sink that does not indicate its support in the HF-VSDB.

When transmitting video with Deep Color 4:2:0 Pixel Encoding, the CD bits of the General Control Packet shall be set accurately (See H14b Section 5.3.6 and Section 6.5.3).

Transmission of Deep Color 4:2:0 encoded Pixels is achieved by first mapping two 4:2:0 Pixels onto a single 4:4:4 Pixel. The mapping is described in Table 7-2, Table 7-3, Table 7-4, and Table 7-5 for 24-, 30-, 36-, and 48-bit Pixels respectively.

The mapped Pixels are then transported utilizing the packing methods described in H14b Section 6.5.2, H14b Section 6.5.3, and H14b Appendix D.

Table 7-2: Mapping Two 8-bit per component 4:2:0 Pixels to one 24-bit 4:4:4 Pixel prior to Deep Color Packing

			First eight 4:2:0 Pixels on each Line			
		Equivalent 4:4:4 Pixel	**4:2:0, Pixel 0/1**	**4:2:0, Pixel 2/3**	**4:2:0, Pixel 4/5**	**4:2:0, Pixel 6/7**
Line 0	**Channel 0**	CB[7:0]	CB_{00}[7:0]	CB_{02}[7:0]	CB_{04}[7:0]	CB_{06}[7:0]
	Channel 1	Y[7:0]	Y_{00}[7:0]	Y_{02}[7:0]	Y_{04}[7:0]	Y_{06}[7:0]
	Channel 2	CR[7:0]	Y_{01}[7:0]	Y_{03}[7:0]	Y_{05}[7:0]	Y_{07}[7:0]
Line 1	**Channel 0**	CB[7:0]	CR_{00}[7:0]	CR_{02}[7:0]	CR_{04}[7:0]	CR_{06}[7:0]
	Channel 1	Y[7:0]	Y_{10}[7:0]	Y_{12}[7:0]	Y_{14}[7:0]	Y_{16}[7:0]
	Channel 2	CR[7:0]	Y_{11}[7:0]	Y_{13}[7:0]	Y_{15}[7:0]	Y_{17}[7:0]
Line 2	**Channel 0**	CB[7:0]	CB_{20}[7:0]	CB_{22}[7:0]	CB_{24}[7:0]	CB_{26}[7:0]
	Channel 1	Y[7:0]	Y_{20}[7:0]	Y_{22}[7:0]	Y_{24}[7:0]	Y_{26}[7:0]
	Channel 2	CR[7:0]	Y_{21}[7:0]	Y_{23}[7:0]	Y_{25}[7:0]	Y_{27}[7:0]
Line 3	**Channel 0**	CB[7:0]	CR_{20}[7:0]	CR_{22}[7:0]	CR_{24}[7:0]	CR_{26}[7:0]
	Channel 1	Y[7:0]	Y_{30}[7:0]	Y_{32}[7:0]	Y_{34}[7:0]	Y_{36}[7:0]
	Channel 2	CR[7:0]	Y_{31}[7:0]	Y_{33}[7:0]	Y_{35}[7:0]	Y_{37}[7:0]

Table 7-3: Mapping Two 10-bit per component 4:2:0 Pixels to one 30-bit 4:4:4 Pixel prior to Deep Color Packing

			First eight 4:2:0 Pixels on each Line			
		Equivalent 4:4:4 Pixel	**4:2:0, Pixel 0/1**	**4:2:0, Pixel 2/3**	**4:2:0, Pixel 4/5**	**4:2:0, Pixel 6/7**
Line 0	**Channel 0**	CB[9:0]	CB_{00}[9:0]	CB_{02}[9:0]	CB_{04}[9:0]	CB_{06}[9:0]
	Channel 1	Y[9:0]	Y_{00}[9:0]	Y_{02}[9:0]	Y_{04}[9:0]	Y_{06}[9:0]
	Channel 2	CR[9:0]	Y_{01}[9:0]	Y_{03}[9:0]	Y_{05}[9:0]	Y_{07}[9:0]
Line 1	**Channel 0**	CB[9:0]	CR_{00}[9:0]	CR_{02}[9:0]	CR_{04}[9:0]	CR_{06}[9:0]
	Channel 1	Y[9:0]	Y_{10}[9:0]	Y_{12}[9:0]	Y_{14}[9:0]	Y_{16}[9:0]
	Channel 2	CR[9:0]	Y_{11}[9:0]	Y_{13}[9:0]	Y_{15}[9:0]	Y_{17}[9:0]
Line 2	**Channel 0**	CB[9:0]	CB_{20}[9:0]	CB_{22}[9:0]	CB_{24}[9:0]	CB_{26}[9:0]
	Channel 1	Y[9:0]	Y_{20}[9:0]	Y_{22}[9:0]	Y_{24}[9:0]	Y_{26}[9:0]
	Channel 2	CR[9:0]	Y_{21}[9:0]	Y_{23}[9:0]	Y_{25}[9:0]	Y_{27}[9:0]
Line 3	**Channel 0**	CB[9:0]	CR_{20}[9:0]	CR_{22}[9:0]	CR_{24}[9:0]	CR_{26}[9:0]
	Channel 1	Y[9:0]	Y_{30}[9:0]	Y_{32}[9:0]	Y_{34}[9:0]	Y_{36}[9:0]
	Channel 2	CR[9:0]	Y_{31}[9:0]	Y_{33}[9:0]	Y_{35}[9:0]	Y_{37}[9:0]

Table 7-4: Mapping Two 12-bit per component 4:2:0 Pixels to one 36-bit 4:4:4 Pixel prior to Deep Color Packing

			First eight 4:2:0 Pixels on each Line			
		Equivalent 4:4:4 Pixel	**4:2:0, Pixel 0/1**	**4:2:0, Pixel 2/3**	**4:2:0, Pixel 4/5**	**4:2:0, Pixel 6/7**
Line 0	**Channel 0**	CB[11:0]	CB_{00}[11:0]	CB_{02}[11:0]	CB_{04}[11:0]	CB_{06}[11:0]
	Channel 1	Y[11:0]	Y_{00}[11:0]	Y_{02}[11:0]	Y_{04}[11:0]	Y_{06}[11:0]
	Channel 2	CR[11:0]	Y_{01}[11:0]	Y_{03}[11:0]	Y_{05}[11:0]	Y_{07}[11:0]
Line 1	**Channel 0**	CB[11:0]	CR_{00}[11:0]	CR_{02}[11:0]	CR_{04}[11:0]	CR_{06}[11:0]
	Channel 1	Y[11:0]	Y_{10}[11:0]	Y_{12}[11:0]	Y_{14}[11:0]	Y_{16}[11:0]
	Channel 2	CR[11:0]	Y_{11}[11:0]	Y_{13}[11:0]	Y_{15}[11:0]	Y_{17}[11:0]
Line 2	**Channel 0**	CB[11:0]	CB_{20}[11:0]	CB_{22}[11:0]	CB_{24}[11:0]	CB_{26}[11:0]
	Channel 1	Y[11:0]	Y_{20}[11:0]	Y_{22}[11:0]	Y_{24}[11:0]	Y_{26}[11:0]
	Channel 2	CR[11:0]	Y_{21}[11:0]	Y_{23}[11:0]	Y_{25}[11:0]	Y_{27}[11:0]
Line 3	**Channel 0**	CB[11:0]	CR_{20}[11:0]	CR_{22}[11:0]	CR_{24}[11:0]	CR_{26}[11:0]
	Channel 1	Y[11:0]	Y_{30}[11:0]	Y_{32}[11:0]	Y_{34}[11:0]	Y_{36}[11:0]
	Channel 2	CR[11:0]	Y_{31}[11:0]	Y_{33}[11:0]	Y_{35}[11:0]	Y_{37}[11:0]

Table 7-5: Mapping Two 16-bit per component 4:2:0 Pixels to one 48-bit 4:4:4 Pixel prior to Deep Color Packing

			First eight 4:2:0 Pixels on each Line			
		Equivalent 4:4:4 Pixel	**4:2:0, Pixel 0/1**	**4:2:0, Pixel 2/3**	**4:2:0, Pixel 4/5**	**4:2:0, Pixel 6/7**
Line 0	**Channel 0**	CB[15:0]	CB_{00}[15:0]	CB_{02}[15:0]	CB_{04}[15:0]	CB_{06}[15:0]
	Channel 1	Y[15:0]	Y_{00}[15:0]	Y_{02}[15:0]	Y_{04}[15:0]	Y_{06}[15:0]
	Channel 2	CR[15:0]	Y_{01}[15:0]	Y_{03}[15:0]	Y_{05}[15:0]	Y_{07}[15:0]
Line 1	**Channel 0**	CB[15:0]	CR_{00}[15:0]	CR_{02}[15:0]	CR_{04}[15:0]	CR_{06}[15:0]
	Channel 1	Y[15:0]	Y_{10}[15:0]	Y_{12}[15:0]	Y_{14}[15:0]	Y_{16}[15:0]
	Channel 2	CR[15:0]	Y_{11}[15:0]	Y_{13}[15:0]	Y_{15}[15:0]	Y_{17}[15:0]
Line 2	**Channel 0**	CB[15:0]	CB_{20}[15:0]	CB_{22}[15:0]	CB_{24}[15:0]	CB_{26}[15:0]
	Channel 1	Y[15:0]	Y_{20}[15:0]	Y_{22}[15:0]	Y_{24}[15:0]	Y_{26}[15:0]
	Channel 2	CR[15:0]	Y_{21}[15:0]	Y_{23}[15:0]	Y_{25}[15:0]	Y_{27}[15:0]
Line 3	**Channel 0**	CB[15:0]	CR_{20}[15:0]	CR_{22}[15:0]	CR_{24}[15:0]	CR_{26}[15:0]
	Channel 1	Y[15:0]	Y_{30}[15:0]	Y_{32}[15:0]	Y_{34}[15:0]	Y_{36}[15:0]
	Channel 2	CR[15:0]	Y_{31}[15:0]	Y_{33}[15:0]	Y_{35}[15:0]	Y_{37}[15:0]

7.1.2 Signaling for YC_BC_R 4:2:0 Pixel Encoding

When a Source sends YC_BC_R 4:2:0 Pixel Encoded data across an HDMI cable, the Y2, Y1, and Y0 fields of the AVI InfoFrame shall be set Y2 = 0, Y1 = 1 and Y0 = 1, as defined in CEA-861-F Section 6.4 and CEA-861-F Table 9.

To indicate support for YC_BC_R 4:2:0 Pixel Encoding, an HDMI Sink shall use a Y420CMDB (YC_BC_R 4:2:0 Capability Map Data Block) and/or Y420VDB (YC_BC_R 4:2:0 Video Data Block), as defined in CEA-861-F Section 7.5.10 and 7.5.11, in its E-EDID, for all Video Formats for which it supports YC_BC_R 4:2:0 Pixel Encoding.

7.2 Colorimetry

7.2.1 Default Colorimetry

(‡) This section incorporates text from the HDMI Specification 1.4b Section 6.7.1. See Notice for copyright information.

H14b Section 6.7.1 is extended as follows:

Sources will typically use the specific default colorimetry for the video format being transmitted. If no colorimetry is indicated in the AVI InfoFrame's C field (C1, C0), then the colorimetry of the transmitted signal shall match the default colorimetry for the transmitted video format as specified in CEA-861-F Section 5.

For high-definition (720p, 1080i, 1080p, 2160p) CE Video Formats described in CEA-861-F Section 4, the default colorimetry is based on ITU-R BT.709-5.

For IT Video Formats, the default colorimetry is sRGB as is stated in CEA-861-F Section 5.1, and as in H14b Section 6.7.1.

7.2.2 BT.2020 Colorimetry

In addition to the colorimetry standards referenced in H14b Section 6.7, This Specification optionally supports colorimetry as defined in ITU-R BT.2020 for Pixel Encodings with 10 or more bits per component, with these corresponding updates to signaling:

- Sink Devices that are capable of, and wish to indicate support for receiving colorimetry as defined in ITU-R BT.2020, shall incorporate an EDID Colorimetry Data Block as defined in CEA-861-F Table 56 and 57, and should evaluate AVI InfoFrame bits C1..C0 and EC2..EC0 as defined in CEA-861-F Tables 10 and 12.
- The Source shall transmit an AVI InfoFrame with bits C1..C0 and EC2..EC0 set as defined in CEA-861-F Tables 10 and 12 to indicate ITU-R BT.2020 colorimetry. The Source shall not indicate ITU-R BT.2020 colorimetry in the AVI InfoFrame unless the Sink indicates support for colorimetry as defined in ITU-R BT.2020 in the Sink's Colorimetry Data Block (see CEA-861-F Table 56 and 57).

For any video categorized as ITU-R BT.2020, CEA-861-F Section 5.2.1 shall be used for any color space conversion needed in the course of processing.

ITU-R BT.2020 defines a set of 4320p and 2160p Video Formats, with several signal and coding formats. Table 7-6 lists those formats from this set that are supported by This Specification.

Note that the default colorimetry for 2160p Video Formats in This Specification and in CEA-861-F is different from the colorimetry defined in ITU-R BT.2020 (see Section 7.2.1 and this section).

Table 7-6: Video Formats defined in ITU-R BT.2020 that are supported by This Specification

Resolution	Colorimetry	Bits per Component	Frames per Second	Pixel Encoding
2160p	ITU-R BT.2020	10 or 12	24, 25 or 30	RGB or YC_BC_R 4:2:2 or YC_BC_R 4:4:4
			50 or 60	YC_BC_R 4:2:0 or YC_BC_R 4:2:2

7.3 Video Quantization Ranges

(‡) This section incorporates text from the HDMI Specification 1.4b Section 6.6. See Notice for copyright information.

H14b Section 6.6 is extended as follows:

Black and white levels for video components shall be either "Full Range" or "Limited Range." By default, Limited Range shall be used for CE Video Formats (i.e. all video formats defined in CEA-861-F Section 4, with the exception of VGA (640x480) format). For IT Video Formats (which include the VGA (640x480) Video Format) the default is Full Range video quantization, as described in H14b Section 6.6.

CEA-861-F defines various signaling features that allow to override the default quantization ranges defined in the previous paragraph. These signaling features are summarized in the following two subsections, and Table 7-7 and Table 7-8.

7.3.1 Video Quantization Ranges Signaling (RGB)

For RGB pixel encoding, the quantization bits Q1,Q0 in AVI InfoFrame Data Byte 3 (see CEA-861-F Section 6.4) allow the Source to override the default RGB Quantization Range (see Section 7.3 above) and to explicitly indicate the RGB Quantization Range. The value Q=0 (Q1=0, Q0=0) indicates that the Quantization Range corresponds to the default RGB Quantization Range. A Source shall not send a non-zero Q value that does not correspond to the default RGB Quantization Range for the transmitted Picture unless the Sink indicates support for the Q1,Q0 bits using QS=1 (AVI Q support) bit in a Video Capabilities Data Block (see CEA-861-F Section 7.5.6 and Table 60). This bit allows a Sink to declare that it supports the reception of either type of RGB Quantization Range, under the direction of AVI InfoFrame Q1,Q0 bits.

If the Sink declares a selectable RGB Quantization Range (QS=1) then it shall expect Limited Range pixel values if it receives Q=1 and it shall expect Full Range pixel values if it receives Q=2 (see CEA-861-F Section 6.4). For other values of Q, the Sink shall expect pixel values with the default range for the transmitted Video Format.

A Sink should set QS=1, and interpret and use Q1,Q0 as received in the AVI InfoFrame; a Source should use non-zero Q1,Q0, as detailed in Table 7-7.

Table 7-7: Video Quantization signaling (values for Q1, Q0) for RGB encoding

Sink's capability declaration in VSDB	Quantization range of Source's material	When Source is sending a CE Video Format	When Source is sending an IT Video Format
QS=0	Limited Range	Q1, Q0 = 0, 1*	not allowed
	Full Range	not allowed	Q1, Q0 = 1, 0*
QS=1	Limited Range	Q1, Q0 = 0, 1*	Q1, Q0 = 0, 1
	Full Range	Q1, Q0 = 1, 0	Q1, Q0 = 1, 0*

* recommended value; value 0,0 is also allowed but not recommended

7.3.2 Video Quantization Ranges Signaling (YCC)

This section applies when content is encoded in sYCC601 or AdobeYCC601. For these YCC content encodings, the quantization bits YQ1 and YQ0 in AVI InfoFrame Data Byte 5 (see CEA-861-F Section 6.4 and Table 15) allow the Source to override the default YCC Quantization Range (see Section 7.3 above) and to explicitly indicate the YCC Quantization Range. The YQ-field only applies when transmitting YCC colorimetry. A Source shall not send a YQ value that does not correspond to the default YCC Quantization Range specified for the colorimetry transmitted, unless the Sink indicates support for the YQ1 and YQ0 bits using QY=1 (AVI YQ support) in a Video Capabilities Data Block (see CEA-861-F

Section 7.5.6 and Table 60). This bit allows a Sink to declare that it supports the reception of either type of YCC Quantization Range, under the direction of AVI InfoFrame YQ data.

When transmitting any RGB colorimetry, the Source should set the YQ-field to match the RGB Quantization Range being transmitted (e.g., when Limited Range RGB, set YQ=0 or when Full Range RGB, set YQ=1) and the Sink shall ignore the YQ-field.

If the sink's EDID declares a selectable YCC Quantization Range (QY=1), then it shall expect Limited Range pixel values if it receives AVI YQ=0 and it shall expect Full Range pixel values if it receives AVI YQ=1. For other values of YQ, the sink shall expect pixel values with the default range for the transmitted Video Format.

When a Sink supports reception of any YCC content encoding, it should set QY=1, and interpret and use YQ1 and YQ0 as received in the AVI InfoFrame; a Source should use YQ1 and YQ0 as detailed in Table 7-8.

Table 7-8: Video Quantization signaling (values for YQ1 and YQ0) for YC_BC_R Pixel Encoding

Sink's capability declaration in VSDB	Quantization range of Source's material	When Source is sending a CE Video Format	When Source is sending an IT Video Format
QY=0	Limited Range	YQ1, YQ0 = 0, 0	not allowed
	Full Range	not allowed	YQ1, YQ0 = 0, 1
QY=1	Limited Range	YQ1, YQ0 = 0, 0	YQ1, YQ0 = 0, 0
	Full Range	YQ1, YQ0 = 0, 1	YQ1, YQ0 = 0, 1

7.4 3D Video Extension

7.4.1 3D OSD Disparity Indication

When a Source is sending a 3D Video Format which is not a "dual view" signal (see Section 7.4.2), and it has read an HF-VSDB with 3D_OSD_Disparity=1, the Source may insert signaling 3D_DisparityData in the HF-VSIF to convey depth information in the form of disparity values so as to enable the Sink to overlay additional information (graphics, menus, etc.) such that a depth violation between the 3D video and graphics is avoided. In all other circumstances, if the Source is sending an HF-VSIF, it shall clear field 3D_DisparityData_Present (=0) and not include the 3D_DisparityData in the HF-VSIF.

A Sink which is capable of receiving 3D OSD Disparity Indication and wishes to indicate its support for receiving such signaling, shall set the field 3D_OSD_Disparity (=1) in the HF-VSDB (see Section 10.3.2), and should use the received 3D_DisparityData in the received HF-VSIF to adapt its processing.

For placement and definition of the relevant fields, 3D_DisparityData_present (which determines if the block with 3D_DisparityData is present in the HF-VSIF), 3D_DisparityData_version, and 3D_DisparityData_length, see Section 10.2.

The block 3D_DisparityData is preceded with a byte containing version and length (Note – this byte is NOT counted in the length). Sinks that do not recognize the version (or do not need the contents of the block in their current state) shall skip over the block using the length indication.

The structure of the contents of the block 3D_DisparityData depends on the value of field 3D_DisparityData_version; 3D_DisparityData_length will indicate the size in bytes of block 3D_DisparityData, this size being dependent on the 3D_DisparityData_version:

- 3D_DisparityData_version = 0b000: no block 3D_DisparityData is inserted following this byte:
    - o 3D_DisparityData_length shall contain 0x00.

    - This indication may be used by a Source to indicate no reliable Disparity Data is available, and that Disparity Data sent previously is no longer valid.
- 3D_DisparityData_version = 0b001: 3D_DisparityData contains min/max disparity info using a method allowing indication of minimum and maximum disparity for the entire video frame:
    - 3D_DisparityData_length shall contain 0x03.
    - 3D_DisparityData shall be filled with production_disparity_hint_info as defined in Section 5.1.1 of ETSI TS 101 547 and Section 6.4.13.1 of ETSI EN 300 468 : 2x 12-bit values, containing the disparity values (min & max). See referred spec for details on coding of these fields, and Table 7-9 for distribution of these values over the 3 bytes in 3D_Disparity_Data.

      These disparity values (production_disparity_hint_info) shall be coded according to Table 7-9; the disparity of most of the content is expected to be within these values for most of the time. These values are generally constant for longer periods (e.g. production parameters for a broadcast event). For a mechanism to allow more dynamic disparity updates, see 3D_DisparityData_version = 0b010.

Table 7-9: 3D_Disparity_Data for 3D_DisparityData_version=001

3D_DisparityData	**Bit**							
byte	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
1	video_max_disparity_hint (bits 11..4)							
2	video_max_disparity_hint (bits 3..0)				video_min_disparity_hint (bits 11..8)			
3	video_min_disparity_hint (bits 7..0)							

- 3D_DisparityData_version = 0b010: 3D_DisparityData shall contain min/max disparity information for multiple regions of the video frame (see Section 5.1.2 of ETSI TS 101 547); this is used for dynamic indication (actual min/max disparity values in a video frame, or region thereof) and can vary at frame level. The Sink designer should be aware of possible variations in the value(s) that is (are) signaled.
    - 3D_DisparityData contains the contents from field multi_region_disparity as defined in Section B.11 of ETSI TS 101 154 and in Table 7-10**.** In Table 7-10 below, please also refer to Section B.11 of ETSI TS 101 154 for the definition of regions, max_disparity_in_picture, and min_disparity_in_region[i].

Table 7-10: 3D_Disparity_Data for 3D_DisparityData_version=010

3D_DisparityData byte	**Contents**
1	multi_region_disparity_length (can be 0x00, 0x02, 0x03, 0x04, 0x05, 0x0A or 0x11; value 0x01 is prohibited; other values are reserved for future use)
2	(if multi_region_disparity_length > 1) max_disparity_in_picture
3 to 3+(N-1)	(if multi_region_disparity_length > 1) min_disparity_in_region[i] With i =0 to i =N-1 where N = multi_region_disparity_length-1

Table 7-11 below is extracted from Section B.11 of ETSI TS 101 154 and extended to the HDMI use case to highlight the definition and values of multi_region_disparity_length and corresponding 3D_DisparityData_length.

Table 7-11: Definition and values of multi_region_disparity_length and 3D_DisparityData_length

multi_ region_ disparity_ length	Meaning of the value of "multi_region_disparity_length"	3D_DisparityData_length		"N" (see Table 7-10)
		If 3D_ DisparityData_ version=010	If 3D_ DisparityData_ version=011	
0	no disparity information is to be delivered	1	4	N/A
1	Prohibited	N/A	N/A	N/A
2	one minimum_disparity_in_region is coded as representing the minimum value in overall picture	3	6	1
3	two vertical minimum_disparity_in_regions are coded	4	7	2
4	three vertical minimum_disparity_in_regions are coded	5	8	3
5	four minimum_disparity_in_regions are coded	6	9	4
6 to 9	reserved for future use	reserved	reserved	
10	nine minimum_disparity_in_regions are coded	11[a]	14[a]	9
11 to 16	reserved for future use	reserved	reserved	
17	sixteen minimum_disparity_in_regions are coded	18[a]	reserved	16
18 to 255	reserved for future use	reserved	reserved	

[a] This combination of parameters might lead to a situation where the disparity data will not fit in the HF-VSIF, depending on values of 3D_Meta_Present, 3D_Structure and potential other data carried in this HF-VSIF. The Source can prevent such situation by sending less disparity data (e.g. sending 3D_DisparityData_version = 001 or 010 instead of 011, or by mapping the data for the regions to a smaller number of regions).

- 3D_DisparityData_version = 0b011: 3D_DisparityData contains both production_disparity_hint_info (in first three bytes) as well as multi_region_disparity (in remaining bytes); in this case 3D_DisparityData_Length = multi_region_disparity_length +4
- Other values for 3D_DisparityData_version and 3D_DisparityData_length are reserved.

Note: although the definition for 3D_DisparityData is in a DVB (broadcast) specification, it can also be used for non-broadcast use cases, e.g. 3D content generated by a game Source, that can calculate the 3D_DisparityData along with the 3D game content, and send this 3D_DisparityData along with the video to the Sink, which can employ such data irrespective of the type of Source (broadcast, game or otherwise).

Note that the general rules on updates to HF-VSIFs (see Section 10.2.1) also apply here, e.g. when a Source stops sending the HF-VSIF, or stops sending 3D_DisparityData within a HF-VSIF, the Sink shall no longer use the previous 3D_DisparityData.

Also note that the dynamic behavior defined by multi_region_disparity needs a swifter reaction than the maximum of 1 second mentioned in Section 10.2.1 since this general limit does not guarantee a smooth operation of the Sink’s graphics overlay without depth violation between the 3D video and graphics. Therefore, the Sink should align reaction to the content of the 3D_DisparityData field (especially when multi_region_disparity is present) with the associated video frames. A Source inserting 3D_DisparityData with multi_region_disparity should align this disparity data with the associated video frames.

7.4.2 3D Dual-View Signaling

A Source which supports 3D transmission may use the "Dual View" transmission mode. This uses two 2D video signals (of same Video Format) combined in a single 3D Video Format (one video signal in the "left" image and the other video signal in the "right" image). A Source which is using this "Dual View" transmission mode, and which has read an HF-VSDB with Dual_View=1 shall set the field 3D_DualView (=1) in the HF-VSIF (see Section 10.2; this requires the inclusion of the byte 3D_AdditionalInfo and so, 3D_AdditionalInfo_Present shall be set (=1)). In all other circumstances, if the Source is sending an HF-VSIF, and if the byte 3D_AdditionalInfo_present is included in the transmitted HF-VSIF, it shall clear field 3D_DualView (=0).

A Sink which is capable of receiving 3D Video Formats, is capable of receiving a “Dual View” signal, and which wishes to indicate its support for such signals, shall set the field Dual_View (=1) in the HF-VSDB (see Section 10.3.2), and should use the field 3D_DualView in the received HF-VSIF to adapt processing and/or instruct the user(s)/3D-glasses accordingly.

7.4.3 3D Independent View Signaling

When a Source is sending a 3D Video Format, and it has read an HF-VSDB with Independent_View=1, it may set the signaling fields 3D_ViewDependency and 3D_Preferred2DView in the HF-VSIF (see Section 10.2; this requires the inclusion of the byte 3D_AdditionalInfo and so, 3D_AdditionalInfo_Present shall be set (=1)) to indicate the coding relationship (if any) between the left and right view, and whether one of the views (if any) is preferred for 2D viewing. In all other circumstances, if the Source is sending an HF-VSIF, and if the byte 3D_AdditionalInfo_present is included in the transmitted HF-VSIF, it shall clear fields 3D_ViewDependency (=00) and 3D_Preferred2DView (=00).

A Sink which is capable of receiving 3D Video Formats, which wishes to indicate its support for such "Independent View" signaling, shall set the field Independent_View (=1) in the HF-VSDB (see Section 10.3.2).

7.4.3.1 3D_ViewDepedency

Depending on how the 3D signal has been created, one of the two views ('left" and "right") could have been derived from the other or not. An example is the MVC Stereo High profile where one of the views (e.g. left) is encoded directly, and used as a predictor for the other view (typically leading to a lower bit rate). This particular example would be coded as 0b10, i.e. "The left view originates from an independently coded view"; in cases where no such potential quality difference between the two views is present, this would be coded as 0b11, i.e. "Both views are from (substantially) independently coded views".

This signaling could be used by a 3D Sink that needs to do further processing on the received 3D signal. An example would be an auto-stereoscopic 3D display, that needs to derive multiple (e.g. 9) views out of the received "left" and "right" views. Using these signaling bits, it can determine which of the two views to preferably use as basis "2D" signal for its processing.

7.4.3.2 3D_Preferred2DView

When the Sink is displaying the received 3D signal as 2D (e.g. because the user prefers to watch the content in 2D), it can choose which of the two received views ('left" or "right") to display on the screen. The content creator may want to indicate which of the two views is most suitable for such 2D viewing (since the content on both views may be slightly different due to e.g. parallax); this indication is possible using 3D_Preferred2DView.

7.5 Additional Video Formats

This Specification refers to CEA-861-F for Video Format definitions whereas HDMI 1.4b makes references to Video Formats as defined by CEA-861-D. Compared to CEA-861-D, CEA-861-F defines additional Video Formats with associated VICs (see Section 10.1); those are Video Formats with a picture aspect ratio of “21:9” (64:27) as well as 2160p Video Formats.

8 Packet Definitions

HDMI 1.4b defines a number of packet types. These are summarized in H14b Table 5-8. In addition, This Specification also defines several additional packets. These are summarized in Table 8-1.

Table 8-1: Packet Types

Packet Type Value	Packet Type	Described in Section
0x0B	3D Audio Sample Packet (L-PCM format only)	8.1
0x0C	One Bit 3D Audio Sample Packet	8.2
0x0D	Audio Metadata Packet	8.3
0x0E	Multi-Stream Audio Sample Packet	8.4
0x0F	One Bit Multi-Stream Audio Sample Packet	8.5

8.1 3D Audio Sample Packet

The L-PCM audio format with 3D Audio is carried using 3D Audio Sample Packets. Audio formats other than L-PCM shall not be transmitted using 3D Audio Sample Packets. The 3D Audio stream contains 9 to 32 audio channels and is transmitted with consecutive packets such that, for a given sample, no packets of a different type interrupt the transfer of the channels. Each packet contains up to 8 audio channels. The packet header includes a sample_start and sample_present bit to denote the position of the packet within the 3D Audio sample. This is described in detail in Section 9.3.3.

When an audio stream is being transported via 3D Audio Sample Packets, other Audio Sample packet types (Packet Types 0x02, 0x07, 0x08, 0x09, 0x0C, 0x0E, and 0x0F) shall not be transmitted.

Table 8-2: 3D Audio Sample Packet Header

Byte \ Bit #	7	6	5	4	3	2	1	0
HB0	0	0	0	0	1	0	1	1
HB1	Rsvd (0)	Rsvd (0)	Rsvd (0)	sample_start	sample_present.sp3	sample_present.sp2	sample_present.sp1	sample_present.sp0
HB2	B.3	B.2	B.1	B.0	sample_flat.sp3	sample_flat.sp2	sample_flat.sp1	sample_flat.sp0

- sample_start [1 bit] If sample_start=1 then the current packet is the first packet of a 3D Audio sample. See Section 9.3.3 for details.
- sample_present.spX [4 fields, 1 bit each] If set (=1), indicates that Subpacket X contains an audio sample(s).
- sample_flat.spX [4 fields, 1 bit each] If set (=1), indicates that Subpacket X represents a “flatline” sample. A flatline audio sample is one in which no valid audio data is presented, but the timing remains accurate. The Sink device shall ignore the content of flatline samples and render the audio at zero level. Flatline samples are provided to maintain audio clock synchronization; they are only valid if “sample_present.spX” is set. All sample_flat.spX bits (with their corresponding sample_present.spX bits set) shall be set to the same value.
- B.X [4 fields, 1 bit each] If B.X is set (=1), Subpacket X contains the first frame in a 192 frame IEC 60958 Channel Status block; B.X is cleared (=0) otherwise.

The 3D Audio Sample Packet includes four Subpackets which are defined identically to the Audio Sample Subpacket defined in H14b Table 5-13. From 0 to 4 of these Subpackets may carry valid data as indicated by the sample_present and sample_flat bits.

8.2 One Bit 3D Audio Sample Packet

The One Bit Audio format with 3D Audio is carried using One Bit 3D Audio Sample Packets. Audio formats other than One Bit Audio shall not be transmitted using One Bit 3D Audio Sample Packets. The One Bit 3D Audio stream contains 9 to 32 audio channels and is transmitted across consecutive packets such that for a given sample, no packets of a different type interrupt the transfer of the channels. Each packet contains up to 8 audio channels. The packet header includes a sample_start and sample_present bit to denote the position of the packet within the One Bit 3D Audio sample. This is described in detail in Section 9.3.4.

When an audio stream is being transported via One Bit 3D Audio Sample Packets, other Audio Sample packet types (Packet Types 0x02, 0x07, 0x08, 0x09, 0x0B, 0x0E, and 0x0F) shall not be transmitted.

Table 8-3: One Bit 3D Audio Packet Header

Byte \ Bit #	7	6	5	4	3	2	1	0
HB0	0	0	0	0	1	1	0	0
HB1	Rsvd (0)	Rsvd (0)	Rsvd (0)	sample_start	samples_present.sp3	samples_present.sp2	samples_present.sp1	samples_present.sp0
HB2	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	samples_invalid.sp3	samples_invalid.sp2	samples_invalid.sp1	samples_invalid.sp0

- sample_start [1 bit] If sample_start=1 then the current packet is the first packet of a One Bit 3D Audio sample. See Section 9.3.4 for details.
- sample_present.spX [4 fields, 1 bit each] If set (=1), indicates that Subpacket X contains an audio sample(s).
- samples_invalid.spX [4 fields, 1 bit each] If set (=1), indicates that Subpacket X represents invalid samples. If cleared (=0), the samples in Subpacket X are valid. This bit is only valid if the relevant “sample_present.spX” is set. All samples_invalid.spX bits (with corresponding samples_present.spX bits set) shall be set to the same value.

Sample frequency information for One Bit 3D Audio shall be carried in the Audio InfoFrame (see H14b Section 8.2.2).

The One Bit 3D Audio Sample Packet includes four Subpackets which are defined identically to the One Bit Audio Sample Subpacket defined in H14b Table 5-25. From 0 to 4 of these Subpackets may carry valid data as indicated by the sample_present and samples_invalid bits.

8.3 Audio Metadata Packet

Additional information related to 3D Audio and Multi-Stream Audio is carried using the Audio Metadata Packet. The Audio Metadata Packet shall be transmitted if (and only if) (L-PCM Encoded) 3D Audio, One Bit 3D Audio, (L-PCM or IEC 61937 compressed) Multi-Stream Audio, or One Bit Multi-Stream Audio Sample packets are being transmitted. A Source shall always transmit an Audio Metadata Packet at least once per two Video Fields when 3D Audio or Multi-Stream Audio packets are being transmitted. This is described in detail in Section 9.3.5 and Section 9.4.3.

When transmitting 3D Audio, the Audio Metadata Packet describes the number of channels, Audio Channel Allocation Standard Type (ACAT), and channel/speaker allocation of the 3D Audio stream.

When transmitting 3D Audio Sample Packets or One Bit 3D Audio Sample Packets, the Source shall insert, and the Sink shall extract the Audio channel count and channel/speaker allocation information from the Audio Metadata Packet. The channel count and channel/speaker allocation information from the Audio InfoFrame shall not be used.

When transmitting Multi-Stream Audio, the Audio Metadata Packet describes the mapping information for multi-view video streaming, language code, and supplementary audio type (i.e., audio for visually/hearing impaired).

When transmitting Multi-Stream Sample Packets or One Bit Multi-Stream Audio Sample Packets, the Source shall insert, and the Sink shall extract the number of views, the number of audio streams and the Audio Metadata Descriptors from the Audio Metadata Packet.

Table 8-4: Audio Metadata Packet Header

Byte \ Bit #	7	6	5	4	3	2	1	0
HB0	0	0	0	0	1	1	0	1
HB1	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	3D_ AUDIO
HB2	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	NUM_AUDIO_STR		NUM_VIEWS	

- 3D_AUDIO [1 bit]

  The Source shall set this bit (=1) when transmitting 3D Audio packets.
  The Source shall reset this bit (=0) when transmitting Multi-Stream Audio packets.

  If 3D_AUDIO=1, the Audio Metadata Packet shall include the 3D Audio channel count and channel/speaker allocation information.

- NUM_VIEWS [2 bits]

  Indicates the number of views.

  If NUM_VIEWS=0b00, single-view video streaming is active.

  If NUM_VIEWS=0b01, dual-view video streaming is active. This mode shall be used only if 3D_DualView=1 (in HF-VSIF) and Source is sending 3D content with Dual View feature.

  Other values: Reserved for future use.

- NUM_AUDIO_STR [2 bits] Indicates the number of audio streams - 1.

  If NUM_AUDIO_STR=0b01, 0b10, or 0b11, Audio Metadata Packet shall contain NUM_AUDIO_STR+1 Audio Metadata Descriptors, one for each corresponding audio stream. Unused Audio Metadata Descriptors shall be reset (=0) by the Source, and shall be ignored by the Sink.

There are seven valid configurations of 3D_AUDIO, NUM_VIEWS, and NUM_AUDIO_STR bits. They are shown in Table 8-5. Other combinations shall not be used.

Table 8-5: Valid Bit Configurations for Audio Metadata Header

3D_AUDIO	NUM_VIEWS		NUM_AUDIO_STR		Description
1	0	0	0	0	3D Audio
0	0	0	0	1	Multi-Stream Audio (single-view, 2 audio streams)
0	0	0	1	0	Multi-Stream Audio (single-view, 3 audio streams)
0	0	0	1	1	Multi-Stream Audio (single-view, 4 audio streams)
0	0	1	0	1	Multi-Stream Audio (dual-view, 2 audio streams)
0	0	1	1	0	Multi-Stream Audio (dual-view, 3 audio streams)
0	0	1	1	1	Multi-Stream Audio (dual-view, 4 audio streams)
otherwise					Reserved

If 3D_AUDIO=1, the Audio Metadata Packet body shall be defined as in Table 8-6.

If 3D_AUDIO=0, the Audio Metadata Packet body shall be defined as in Table 8-12.

8.3.1 Audio Metadata Packet for 3D Audio

Table 8-6, Table 8-7, Table 8-8, Table 8-9, Table 8-10, and Table 8-11 provide descriptions of the Audio Metadata Packet contents that shall be used when 3D Audio is being transmitted.

Table 8-6: Audio Metadata Packet contents for 3D_AUDIO=1

Byte \ Bit #	7	6	5	4	3	2	1	0
PB0	Rsvd (0)	Rsvd (0)	Rsvd (0)	3D_CC4	3D_CC3	3D_CC2	3D_CC1	3D_CC0
PB1	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	ACAT3	ACAT2	ACAT1	ACAT0
PB2	3D_CA7	3D_CA6	3D_CA5	3D_CA4	3D_CA3	3D_CA2	3D_CA1	3D_CA0
PB3 … PB27	Reserved (0)							

- 3D_CC [5 bits] Indicates the channel count of the transmitted 3D Audio. If the audio channel count (CC0…CC2) in the Audio InfoFrame does not agree with the 3D Audio channel count (3D_CC0…3D_CC4), the channel count in the Audio InfoFrame shall be ignored. See Table 8-7 for details.
- ACAT [4 bits] Indicates the audio channel allocation standard referred to by the Source. Table 8-8 below shows the values of the ACAT field. Table 8-9 describes the allocation of speaker locations when ACAT is set to 0x01 (10.2 channels). Similarly, Table 8-10 and Table 8-11 describe the allocation of speaker locations when ACAT is set to 0x02 (22.2 channels) and 0x03 (30.2 channels), respectively.
- 3D_CA [1 byte] Channel/speaker allocation for 3D Audio. See Table 8-9, Table 8-10, and Table 8-11 for details. The 3D_CA fields are not valid and shall not be used for IEC 61937 compressed audio streams.

Table 8-7: 3D_CC field

3D_CC4	3D_CC3	3D_CC2	3D_CC1	3D_CC0	Audio Channel Count
0	0	0	0	0	Refer to Stream Header
0	0	0	0	1	Reserved
…	…	…	…	…	Reserved
0	0	1	1	1	Reserved
0	1	0	0	0	9 channels
0	1	0	0	1	10 channels
…	…	…	…	…	…
1	1	1	1	1	32 channels

NOTE: The rows marked "…" in Table 8-7 indicate the intervening bit settings. Thus the third row of this table (excluding the header row) indicates that bit settings of 0b00001 to 0b00111 are all Reserved and the seventh row indicates that bit settings of 0b01001 to 0b11111 refer to Audio Channel Counts of 10 to 32 channels respectively.

Table 8-8: Audio Channel Allocation Standard Type field

ACAT3	ACAT2	ACAT1	ACAT0	Description
0	0	0	0	Reserved
0	0	0	1	Up to 10.2 channels Refer to Table 8-9 Based on ITU-R BS.2159-4 (Type B 10.2ch)
0	0	1	0	Up to 22.2 channels Refer to Table 8-10 Based on SMPTE 2036-2
0	0	1	1	Up to 30.2 channels Refer to Table 8-11 Based on IEC 62574 ed 1.0
0	1	0	0	Reserved
…				
1	1	1	1	

Table 8-9: 3D_CA field for 10.2 Channels[△] (ACAT = 0x01)

3D_CA									**Channel Number**											
Binary								**Hex**												
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**		**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**
0	0	0	0	0	0	0	0	0x00	Reserved											
0	0	0	0	0	0	0	1	0x01	TpFC	LFE2	TpFR	TpFL	BR	BL	RS	LS	FC	LFE1	FR	FL
0	0	0	0	0	0	1	0	0x02	Reserved											
…								…												
1	1	1	1	1	1	1	1	0xFF												

[△]Refer to Appendix B and ITU-R BS.2159-4 Section 4.3

Table 8-10: 3D_CA field for 22.2 Channels[◊](ACAT = 0x02) (Part 1 of 2)

3D_CA									**Channel Number**											
Binary								**Hex**												
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**		**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**
0	0	0	0	0	0	0	0	0x00	Reserved											
0	0	0	0	0	0	0	1	0x01	TpFC	LFE2	TpFR	TpFL	BR	BL	SiR	SiL	FC	LFE1	FR	FL
0	0	0	0	0	0	1	0	0x02	TpFC	LFE2	TpFR	TpFL	BR	BL	SiR	SiL	FC	LFE1	FR	FL
0	0	0	0	0	0	1	1	0x03	Reserved											
…								…												
1	1	1	1	1	1	1	1	0xFF												

[◊]Refer to Appendix B and SMPTE 2036-2 Section 6
(Continued)

Table 8-10: 3D_CA field for 22.2 Channels[ϴ] (ACAT = 0x02) (Continued, Part 2 of 2)

3D_CA									Channel Number											
Binary								Hex												
7	6	5	4	3	2	1	0		24	23	22	21	20	19	18	17	16	15	14	13
0	0	0	0	0	0	0	0	0x00	Reserved											
0	0	0	0	0	0	0	1	0x01	-	-	-	-	-	-	-	-	-	-	-	-
0	0	0	0	0	0	1	0	0x02	BtFC	BtFR	BtFL	TpC	TpSiR	TpSiL	TpBC	TpBR	TpBL	BC	FRC	FLC
0	0	0	0	0	0	1	1	0x03	Reserved											
…								…												
1	1	1	1	1	1	1	1	0xFF												

Table 8-11: 3D_CA field for 30.2 Channels[ϴ] (ACAT = 0x03) (Part 1 of 3)

3D_CA									Channel Number											
Binary								Hex												
7	6	5	4	3	2	1	0		12	11	10	9	8	7	6	5	4	3	2	1
0	0	0	0	0	0	0	0	0x00	Reserved											
0	0	0	0	0	0	0	1	0x01	TpFC	LFE2	TpFR	TpFL	BR	BL	SiR	SiL	FC	LFE1	FR	FL
0	0	0	0	0	0	1	0	0x02	TpFC	LFE2	TpFR	TpFL	BR	BL	SiR	SiL	FC	LFE1	FR	FL
0	0	0	0	0	0	1	1	0x03	TpFC	LFE2	TpFR	TpFL	BR	BL	SiR	SiL	FC	LFE1	FR	FL
0	0	0	0	0	1	0	0	0x04	Reserved											
…								…												
1	1	1	1	1	1	1	1	0xFF												

[ϴ]Refer to Appendix B and IEC 62574 Section 3 (Continued Below)

Table 8-11: 3D_CA field for 30.2 Channels[Θ] (ACAT = 0x03) (Continued, part 2 of 3)

3D_CA									Channel Number											
Binary								Hex												
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**		**24**	**23**	**22**	**21**	**20**	**19**	**18**	**17**	**16**	**15**	**14**	**13**
0	0	0	0	0	0	0	0	0x00	Reserved											
0	0	0	0	0	0	0	1	0x01	-	-	-	-	-	-	-	-	-	-	-	-
0	0	0	0	0	0	1	0	0x02	BtFC	BtFR	BtFL	TpC	TpSiR	TpSiL	TpBC	TpBR	TpBL	BC	FRC	FLC
0	0	0	0	0	0	1	1	0x03	BtFC	BtFR	BtFL	TpC	TpSiR	TpSiL	TpBC	TpBR	TpBL	BC	FRC	FLC
0	0	0	0	0	1	0	0	0x04	Reserved											
…								…												
1	1	1	1	1	1	1	1	0xFF												

(Continued Below)

Table 8-11: 3D_CA field for 30.2 Channels[Θ] (ACAT = 0x03) (Continued, Part 3 of 3)

3D_CA									Channel Number							
Binary								Hex								
7	**6**	**5**	**4**	**3**	**2**	**1**	**0**		**32**	**31**	**30**	**29**	**28**	**27**	**26**	**25**
0	0	0	0	0	0	0	0	0x00	Reserved							
0	0	0	0	0	0	0	1	0x01	-	-	-	-	-	-	-	-
0	0	0	0	0	0	1	0	0x02	-	-	-	-	-	-	-	-
0	0	0	0	0	0	1	1	0x03	TpRS	TpLS	RSd	LSd	RS	LS	FRW	FLW
0	0	0	0	0	1	0	0	0x04	Reserved							
…								…								
1	1	1	1	1	1	1	1	0xFF								

8.3.2 Audio Metadata Packet for Multi-Stream Audio

Table 8-12, Table 8-13, and Table 8-14 provide descriptions of the Audio Metadata Packet contents when Multi-Stream Audio is being transmitted.

Table 8-12: Audio Metadata Packet contents when 3D_Audio=0

Byte \ Bit #	7	6	5	4	3	2	1	0
PB0 … PB4	Audio_Metadata_Descriptor_0							
PB5 … PB9	Audio_Metadata_Descriptor_1							
PB10 … PB14	Audio_Metadata_Descriptor_2							
PB15 … PB19	Audio_Metadata_Descriptor_3							
PB20 … PB27	Reserved (0)							

- Audio_Metadata_Descriptor_X — [4 fields, 5 Bytes each] — Describes the audio metadata for Subpacket X in Multi-Stream Audio Sample Packet or One Bit Multi-Stream Audio Sample Packet. See Table 8-13 for details.

Table 8-13: Audio Metadata Descriptor

Byte \ Bit #	7	6	5	4	3	2	1	0
PB(5*X+0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Multiview _Right	Multiview _Left
PB(5*X+1)	LC_Valid	Rsvd (0)	Rsvd (0)	Suppl_A_ Valid	Suppl_A_ Mixed	Suppl_A_Type		
PB(5*X+2)	Language_Code (3 Bytes)							
PB(5*X+3)								
PB(5*X+4)								

- Multiview_Left — [1 bit] — If Multiview_Left=1, the corresponding audio stream shall be mapped to the Left stereoscopic picture in the 3D Video Format; if Multiview_Left=0, the corresponding audio stream is not relevant for the Left stereoscopic picture. This bit is valid only if NUM_VIEWS=01 and shall be set to 0 when NUM_VIEWS=00.
- Multiview_Right — [1 bit] — If Multiview_Right=1, the corresponding audio stream shall be mapped to the Right stereoscopic picture in the 3D Video Format; if Multiview_Right=0, the corresponding audio stream is not relevant for the Right stereoscopic picture. This bit is valid only if NUM_VIEWS=01 and shall be set to 0 when NUM_VIEWS=00.
- LC_Valid — [1 bit] — If LC_Valid=1, the Language_Code is valid and correctly identifies the language of the corresponding audio stream; otherwise the language of the corresponding audio stream is unspecified and the 3 bytes for Language_Code shall be filled with 0.

- Suppl_A_Valid [1 bit] If Suppl_A_Valid=1, the corresponding audio stream contains a supplementary audio track as indicated by Suppl_A_Type. If Suppl_A_Valid=0, the corresponding audio stream contains a main (standalone) audio track, and Suppl_A_Mixed and Suppl_A_Type shall be set to 0.
- Suppl_A_Mixed [1 bit] If Suppl_A_Mixed=1, the corresponding audio stream contains a mix of main audio components and a supplementary audio track as indicated by Suppl_A_Type. If Suppl_A_Mixed=0, the corresponding audio stream contains a supplementary audio track as indicated by Suppl_A_Type that will need to be mixed with the main audio components from another stream (which has Suppl_A_Valid=0). This bit is valid only if Suppl_A_Valid=1.
- Suppl_A_Type [3 bits] Indicates the supplementary audio type as defined in Table 8-14. This field is valid only if Suppl_A_Valid=1.
- Language_Code [3 Bytes] Identifies the language of the corresponding audio stream. The language code is defined by ISO 639.2 (alpha-3, bibliographic codes); the first byte of the 3 character codes is placed at PB(5*X+2) in Audio_Metadata_Descriptor_X. This field is valid only if LC_Valid=1.

An STB receiving content from a digital broadcast with multiple audio streams may use these fields to identify the characteristics of those audio streams when it forwards them over HDMI using this mechanism; see DVB's Supplementary audio descriptor (EN 300 468, section 6.4.9) and ATSC's fields full_service_flag, audio_service_type and bsmod (A/52:2012) for corresponding broadcast signaling.

Appendix A.3 lists various example use cases and their associated signaling.

Table 8-14: Supplementary Audio Type

Suppl_A_Type			Description
0	0	0	Reserved
0	0	1	Audio for visually impaired (contains narrative description of content)
0	1	0	Audio for visually impaired (spoken subtitles)
0	1	1	Audio for hearing impaired (enhanced intelligibility of dialogue)
1	0	0	Additional audio (needs to be mixed with "main" audio)
otherwise			Reserved

8.4 Multi-Stream Audio Sample Packet

L-PCM and some IEC 61937 compressed audio formats with multiple (up to 4) audio streams at the same sample/frame rate are carried using Multi-Stream Audio Sample Packets. When employing L-PCM, each audio stream within a Multi-Stream Audio sample contains up to 2 audio channels. When employing IEC 61937, the maximum stream bit rate is 6.144 Mbps, and the channel count is specified by the relevant audio compression standard and is not limited by This Specification. The Subpacket configuration is determined by the stream_present bits in the packet header. This is described in detail in Section 9.3.5.

When an audio stream is being transported via Multi-Stream Audio Sample Packets, other Audio Sample packet types (Packet Types 0x02, 0x07, 0x08, 0x09, 0x0B, 0x0C, and 0x0F) shall not be transmitted.

Table 8-15: Multi-Stream Audio Sample Packet Header

Byte \ Bit #	7	6	5	4	3	2	1	0
HB0	0	0	0	0	1	1	1	0
HB1	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	stream_ present. sp3	stream_ present. sp2	stream_ present. sp1	stream_ present. sp0
HB2	B.3	B.2	B.1	B.0	stream_ flat.sp3	stream_ flat.sp2	stream_ flat.sp1	stream_ flat.sp0

- stream_present.spX [4 fields, 1 bit each] If Set (=1), indicates that Subpacket X contains an audio sample(s) of stream X.
- stream_flat.spX [4 fields, 1 bit each] If set (=1), indicates that Subpacket X represents a "flatline" sample of stream X. A flatline audio sample is one in which no valid audio data is presented, but the timing remains accurate. The Sink device shall ignore the content of flatline samples, and render the audio at zero level. Flatline samples are provided to maintain audio clock synchronization; they are only valid if "stream_present.spX" is set.
- B.X [4 fields, 1 bit each] If B.X is set (=1), Subpacket X contains the first frame in a 192 frame IEC 60958 Channel Status block; B.X is cleared (=0) otherwise.

The Multi-Stream Audio Sample Packet includes four Subpackets which are identical to the Audio Sample Subpacket shown in H14b Table 5-13. 0, 2, 3, or 4 (but not 1) of these Subpackets may carry valid data as indicated by the stream_present and stream_flat bits.

8.5 One Bit Multi-Stream Audio Sample Packet

The One Bit Audio format with multiple (up to 4) audio streams at the same sample/frame rate is carried using One Bit Multi-Stream Audio Sample Packets. Each audio stream within a One Bit Multi-Stream Audio sample contains 2 audio channels. The Subpacket configuration is determined by the stream_present bits in the packet header. This is described in detail in Section 9.4.2.

When an audio stream is being transported via One Bit Multi-Stream Audio Sample Packets, other Audio Sample packet types (Packet Types 0x02, 0x07, 0x08, 0x09, 0x0B, 0x0C, and 0x0E) shall not be transmitted.

Table 8-16: One Bit Multi-Stream Audio Packet Header

Byte \ Bit #	7	6	5	4	3	2	1	0
HB0	0	0	0	0	1	1	1	1
HB1	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	stream_ present. sp3	stream_ present. sp2	stream_ present. sp1	stream_ present. sp0
HB2	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	stream_ invalid. sp3	stream_ invalid. sp2	stream_ invalid. sp1	stream_ invalid. sp0

- stream_present.spX [4 fields, 1 bit each] If set (=1), indicates that Subpacket X contains an audio sample(s) of stream X.
- stream_invalid.spX [4 fields, 1 bit each] Indicates if Subpacket X represents invalid samples of stream X. Stream_invalid = 1 if the samples in Subpacket X are invalid; else = 0. This bit is only valid if the relevant "stream_present.spX" is set.

Sample frequency information for One Bit Multi-Stream Audio shall be carried in the Audio InfoFrame (see H14b Section 8.2.2).

The One Bit Multi-Stream Audio Sample Packet includes four Subpackets which are identical to the One Bit Audio Subpacket shown in H14b Table 5-25. 0, 2, 3, or 4 (but not 1) of these Subpackets may carry valid data as indicated by the stream_present and stream_invalid bits.

8.6 Audio InfoFrame

The Audio InfoFrame is defined in CEA-861-F Section 6.6. In addition to the rules described in H14b Section 8.2.2, when multiple audio streams are transmitted using Multi-Stream Audio Sample Packets or One Bit Multi-Stream Audio Sample Packets, an accurate Audio InfoFrame shall be transmitted at least once per two Video Fields. The Audio InfoFrame is used to describe the audio characteristics of all the audio streams.

H14b Table 8-7 summarizes the contents of the Audio InfoFrame. When transmitting 3D Audio streams as defined in This Specification, the Audio InfoFrame Channel Count cannot be used, and an alternative mechanism is required. With respect to the Audio InfoFrame:

- CC0...CC2 When a Source transmits 3D Audio streams, these fields shall be set to 0 and the 3D_CC fields defined in the Audio Metadata Packet shall be used to indicate the number of channels instead of these fields.
- CA0...CA7 When a Source transmits 3D Audio streams, these fields shall be set to 0 and the 3D_CA fields defined in the Audio Metadata Packet shall be used to indicate the channel/speaker allocation information instead of these fields.

9 Audio Extensions

9.1 Supported Audio Formats

(‡) This section incorporates text from the HDMI Specification 1.4b CEC Table 29. See Notice for copyright information.

H14b Section 7 defines the mechanisms to transport a variety of audio formats over the HDMI link. For many of the compressed formats, HDMI 1.4b indicates that an HDMI Source or Sink utilize an Audio Format Code as specified per CEA-861-D, table 37.

In addition to the Audio Coding Types supported by CEA-861-D Table 37 (or equivalently CEA-861-F Table 27), devices compliant with This Specification may support an IEC 61937-compliant compressed audio format that has an Audio Coding Extension Type defined in CEA-861-F, table 29 (with Audio Coding Type = 0x0F per CEA-861-F, Table 27). For these formats, the CEA Short Audio Descriptor is defined in CEA-861-F Tables 53 and 54.

H14b Section CEC 13.15.3 defines a mechanism which the TV can use to discover the audio formats supported by the Amplifier. In order to allow discovery of these additional audio formats, the operand [Audio Format ID and Code] is extended (relative to the definition in H14b CEC Table 29) as indicated in Table 9-1.

Table 9-1: Operand Description of [Audio Format ID and Code]

Name	Range Description		Length	Purpose
[Audio Format ID and Code]	[Audio Format ID] [Audio Format Code]		1 byte	Used to indicate the audio format that TV wants to inquire about
[Audio Format ID]	0	Bits 7-6	2 bits	Indicates that [Audio Format Code] and [Short Audio Descriptor] are as defined in CEA-861-F.
	1			Indicates that [Audio Format Code] denotes the CXT value as defined in CEA-861-F Table 29, and [Short Audio Descriptor] is as defined in CEA-861-F Tables 53 and 54.
[Audio Format Code]	If [Audio Format ID] = 0, 0x01≤N≤0x0F	Bits 5-0	6 bits	If [Audio Format ID]=0 then [Audio Format Code] is as defined in CEA-861-F for CEA Short Audio Descriptor.
	If [Audio Format ID] = 1, 0x00≤N≤0x1F			If [Audio Format ID]=1 then [Audio Format Code] field denotes the CXT value as defined in CEA-861-F, table 29.

Devices compliant with This Specification shall transmit audio utilizing only a single audio packet type at a time. If it is desired to transmit multiple audio streams, the Multi-Stream Audio or One Bit Multi-Stream audio packet type shall be utilized.

9.2 Supported Audio Rates

(‡) This section and its subsections incorporate text from the HDMI Specification 1.4b Section 7.2.2, Table 7-1, Table 7-2, Table 7-3, Table 7-4, and Figure 7-1. See Notice for copyright information.

H14b Table 7-4 defines the sample rates that are supported by HDMI 1.4b. In addition to these, in certain instances, This Specification supports additional audio sample rates. When these are being sent, the channel Status Bits indicate the current audio rate with the settings indicated in Table 9-3.

Table 9-2: Allowed Values for Channel Status bits 24 to 27, 30, and 31

Channel Status Bit Number						Sample Frequency or Frame Rate
24	25	26	27	30	31	
1	1	0	0	-	-	32 kHz
1	1	0	1	0	0	64kHz
1	1	0	1	0	1	128kHz
1	1	0	1	1	0	256kHz
1	1	0	1	1	1	512kHz
1	0	1	0	1	1	1024kHz
0	0	0	0	-	-	44.1 kHz
0	0	0	1	-	-	88.2 kHz
0	0	1	1	-	-	176.4 kHz
1	0	1	1	0	0	352.8kHz
1	0	1	1	0	1	705.6kHz
1	0	1	1	1	0	1411.2kHz
0	1	0	0	-	-	48 kHz
0	1	0	1	-	-	96 kHz
0	1	1	1	-	-	192 kHz
1	0	1	0	0	0	384kHz
1	0	0	1	-	-	768 kHz
1	0	1	0	1	0	1536kHz

Note that Channel Status bits 30 and 31 in Table 9-2 are “Don’t care” where those values are indicated by “-“. In those cases, Channel Status bits 24 through 27 uniquely specify the Sample Frequency or Frame Rate and, Channel Status bits 30 and 31 shall be set to 0 by the Source and shall be ignored by the Sink. The frequencies listed in the “Sample Frequency or Frame Rate” column of Table 9-2 are not supported by all HDMI audio packet types. Table 9-3 summarizes which audio packet types support which Sample Frequency or Frame Rate.

Table 9-3: Supported Sample Frequency or Frame Rate for each Packet Type

Packet Type (value) [o]	Sample Frequency or Frame Rate (From Table 9-2) [Δ]		
	32.0, 44.1, 48.0, 88.2, 96.0, 176.4, and 192.0 kHz	256.0, 352.8, 384.0, 512.0, 705.6, 768.0, 1024.0, 1411.2, and 1536.0 kHz	64.0 and 128.0 kHz
(0x02) Audio Sample (L-PCM and IEC 61937 compressed formats)	Y	N	Y
(0x07) One Bit Audio Sample Packet	Y	N	N
(0x08) DST Audio Packet	Y	N	N
(0x09) High Bitrate (HBR) Audio Stream Packet (IEC 61937)	N	Y	N
(0x0B) 3D Audio Sample Packet (L-PCM format only)	Y	N	N
(0x0C) One Bit 3D Audio Sample Packet	Y	N	N
(0x0E) Multi-Stream Audio Sample Packet	Y	N	N
(0x0F) One Bit Multi-Stream Audio Sample Packet	Y	N	N

[Δ] See Table 9-2 for the Channel Status bits for each sample frequency or frame rate. The Sample Frequency or Frame Rate that each Audio Packet type may support is indicated with a "Y". Where indicated with an "N", the Audio Packet Type shall not support the corresponding audio rates.

[o] See Section 8 and H14b Section 5.3 for a detailed description of each packet type.

9.2.1 Recommended N and Expected CTS Values

This Specification defines several new audio rates and formats. In order to facilitate interoperable systems, This Specification recommends several permutations for the ACR Packet CTS and N values. These are provided in Table 9-4, Table 9-5, and Table 9-6.

The CTS computation is based on the TMDS Clock Rate and the TMDS Character Rate. CTS shall be an integer number that satisfies the following:

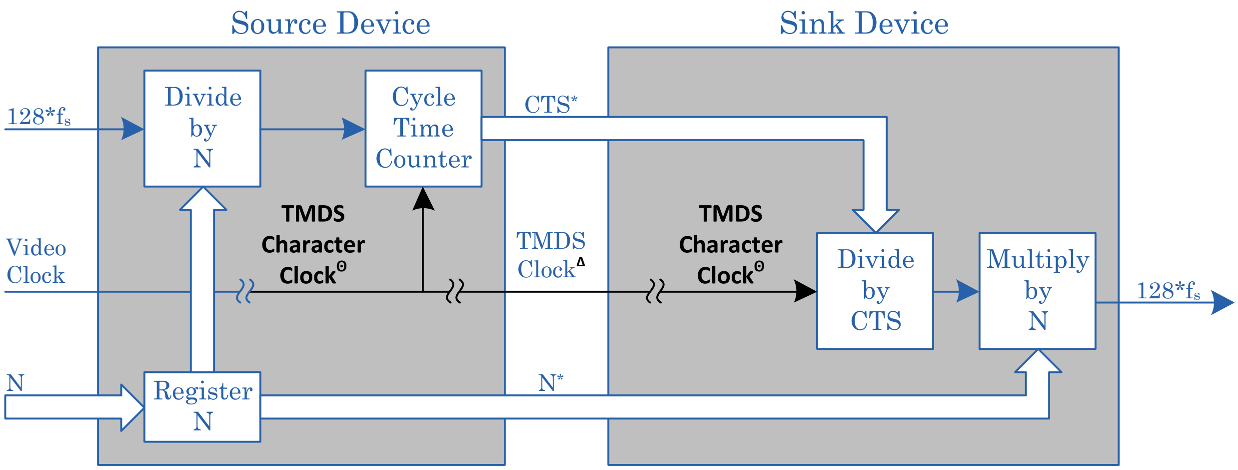
For TMDS Character Rates <= 340 Mcsc

$$(\text{Average CTS value}) = (f_{TMDS\_clock} * N) / (128 * f_S)$$

For TMDS Character Rates >340 Mcsc,

$$(\text{Average CTS value}) = (4 * f_{TMDS\_clock} * N) / (128 * f_S)$$

H14b Figure 7-1 depicts the Audio Clock Regeneration model for operation below 340 Mcsc. Figure 9-1 in This Specification provides the model that also considers operation above 340 Mcsc.



*Note: N and CTS values are transmitted using the "Audio Clock Regeneration" Packet.

⊖Note: The TMDS Character Clock is dependent on the Color Depth and the Video Clock (which oscillates at the Pixel Clock Rate)

∆Note: The TMDS Clock oscillates at

- the TMDS Character Rate when the TMDS Character Rate ≤ 340Mcsc
- ¼ the TMDS Character Rate when the TMDS Character Rate > 340Mcsc

Figure 9-1: Audio Clock Regeneration Model

Table 9-4: Recommended N and expected CTS for Audio Sample Frequency or Frame Rate of 32 kHz and Multiples thereof

TMDS Character Rate (Mcsc)	32 kHz N	32 kHz CTS	64kHz N	64kHz CTS	128kHz N	128kHz CTS	256 kHz N	256 kHz CTS
25.2/1.001	4576	28125	9152	28125	18304	28125	36608	28125
25.2	4096	25200	8192	25200	16384	25200	32768	25200
27	4096	27000	8192	27000	16384	27000	32768	27000
27 * 1.001	4096	27027	8192	27027	16384	27027	32768	27027
54	4096	54000	8192	54000	16384	54000	32768	54000
54 * 1.001	4096	54054	8192	54054	16384	54054	32768	54054
74.25/1.001	11648	210937-210938*	23296	210937–210938*	46592	210937–210938*	93184	210937–210938*
74.25	4096	74250	8192	74250	16384	74250	32768	74250
148.5/1.001	11648	421875	23296	421875	46592	421875	93184	421875
148.5	4096	148500	8192	148500	16384	148500	32768	148500
297/1.001	5824	421875	11648	421875	23296	421875	46592	421875
297	3072	222750	8192	297000	16384	297000	32768	297000
594/1.001	5824	843750	11648	843750	23296	843750	46592	843750
594	3072	445500	8192	594000	16384	594000	32768	594000
Other	4096	measured	8192	measured	16384	measured	32768	measured

* This value will alternate because of restriction on N.

Table 9-5: Recommended N and expected CTS for Audio Sample Frequency or Frame Rate of 44.1 kHz and multiples thereof

TMDS Character Rate	44.1 kHz		88.2 kHz		176.4 kHz		352.8 kHz	
(Mcsc)	N	CTS	N	CTS	N	CTS	N	CTS
25.2/1.001	7007	31250	14014	31250	28028	31250	56056	31250
25.2	6272	28000	12544	28000	25088	28000	50176	28000
27	6272	30000	12544	30000	25088	30000	50176	30000
27 * 1.001	6272	30030	12544	30030	25088	30030	50176	30030
54	6272	60000	12544	60000	25088	60000	50176	60000
54 * 1.001	6272	60060	12544	60060	25088	60060	50176	60060
74.25/1.001	17836	234375	35672	234375	71344	234375	142688	234375
74.25	6272	82500	12544	82500	25088	82500	50176	82500
148.5/1.001	8918	234375	17836	234375	35672	234375	71344	234375
148.5	6272	165000	12544	165000	25088	165000	50176	165000
297/1.001	4459	234375	8918	234375	17836	234375	35672	234375
297	4704	247500	9408	247500	18816	247500	37632	247500
594/1.001	8918	937500	17836	937500	35672	937500	71344	937500
594	9408	990000	18816	990000	37632	990000	75264	990000
Other	6272	measured	12544	measured	25088	measured	50176	measured

Table 9-6: Recommended N and expected CTS for Audio Sample Frequency or Frame Rate of 48 kHz and multiples thereof

TMDS Character Rate	48 kHz		96 kHz		192 kHz		384 kHz	
(Mcsc)	N	CTS	N	CTS	N	CTS	N	CTS
25.2/1.001	6864	28125	13728	28125	27456	28125	54912	28125
25.2	6144	25200	12288	25200	24576	25200	49152	25200
27	6144	27000	12288	27000	24576	27000	49152	27000
27 * 1.001	6144	27027	12288	27027	24576	27027	49152	27027
54	6144	54000	12288	54000	24576	54000	49152	54000
54 * 1.001	6144	54054	12288	54054	24576	54054	49152	54054
74.25/1.001	11648	140625	23296	140625	46592	140625	93184	140625
74.25	6144	74250	12288	74250	24576	74250	49152	74250
148.5/1.001	5824	140625	11648	140625	23296	140625	46592	140625
148.5	6144	148500	12288	148500	24576	148500	49152	148500
297/1.001	5824	281250	11648	281250	23296	281250	46592	281250
297	5120	247500	10240	247500	20480	247500	40960	247500
594/1.001	5824	562500	11648	562500	23296	562500	46592	562500
594	6144	594000	12288	594000	24576	594000	49152	594000
Other	6144	measured	12288	measured	24576	measured	49152	measured

9.3 3D Audio

9.3.1 Maximum 3D Audio Transport Capacity

This Specification defines new audio transport options. However, the capability to carry new audio streams is highly dependent on the video stream being transported.

Table 9-7 and Table 9-8 show the available audio sample rates for 3D Audio transmission at the various Video Format timings specified in CEA-861-F Section 4, assuming that 58 TMDS clock periods of the horizontal blanking interval are required for content protection re-synchronization. 3D Audio transmission shall be accomplished using 3D Audio packets or One Bit 3D Audio packets. (See Section 9.3.3 and Section 9.3.4).

Table 9-7: Max 3D Audio Sampling Frequencies for 24-bit Video Format Timings (Informative)

Description	Format Timing	Pixel Repetition	Vertical Freq (Hz)	Max fs 10.2ch (kHz)	Max fs 22.2ch (kHz)	Max fs 30.2ch (kHz)	Max frame rate 2ch, comp*
60/120/240Hz Formats							
VGA	640x480p	none	59.94/60	X	X	X	192
480i	1440x480i	2	59.94/60	44.1	X	X	352.8
480i	2880x480i	4	59.94/60	96	48	48	768
240p	1440x240p	2	59.94/60	44.1	X	X	352.8
240p	2880x240p	4	59.94/60	96	48	48	768
480p	720x480p	none	59.94/60	X	X	X	192
480p	1440x480p	2	59.94/60	88.2	48	44.1	705.6
480p	2880x480p	4	59.94/60	192	96	96	1536
720p	1280x720p	none	59.94/60	192	96	96	1536
1080i	1920x1080i	none	59.94/60	96	48	48	768
1080p	1920x1080p	none	59.94/60	192	96	96	1536
2160p	3840x2160p	none	59.94/60	192	192	192	1536
2160p(SMPTE)	4096x2160p	none	59.94/60	192	192	192	1536
480i/120	1440x480i	2	119.88/120	88.2	48	44.1	705.6
480p/120	720x480p	none	119.88/120	48	32	X	384
720p/120	1280x720p	none	119.88/120	192	192	192	1536
1080i/120	1920x1080i	none	119.88/120	192	96	96	1536
1080p/120	1920x1080p	none	119.88/120	192	192	192	1536
480i/240	1440x480i	2	239.76/240	176.4	96	88.2	1411.2
480p/240	720x480p	none	239.76/240	96	48	48	768
50/100/200Hz Formats							
576i	1440x576i	2	50	44.1	X	X	352.8
576i	2880x576i	4	50	96	48	48	768
288p	1440x288p	2	50	44.1	X	X	352.8
288p	2880x288p	4	50	96	48	48	768
576p	720x576p	none	50	X	X	X	192
576p	1440x576p	2	50	88.2	48	44.1	705.6
576p	2880x576p	4	50	192	96	96	1536
720p/50	1280x720p	none	50	192	192	176.4	1536
1080i/50	1920x1080i	none	50	192	176.4	96	1536

Description	Format Timing	Pixel Repetition	Vertical Freq (Hz)	Max fs 10.2ch (kHz)	Max fs 22.2ch (kHz)	Max fs 30.2ch (kHz)	Max frame rate 2ch, comp*
1080p/50	1920x1080p	none	50	192	192	192	1536
2160p	3840x2160p	none	50	192	192	192	1536
2160p(SMPTE)	4096x2160p	none	50	192	192	192	1536
1080i, 1250 total	1920x1080i	none	50	96	88.2	48	1024
576i /100Hz	1440x576i	2	100	88.2	48	44.1	705.6
576p/100Hz	720x576p	none	100	48	32	X	384
720p/100	1280x720p	none	100	192	192	192	1536
1080i/100	1920x1080i	none	100	192	192	192	1536
1080p/100	1920x1080p	none	100	192	192	192	1536
576i/200	1440x576i	2	200	176.4	96	88.2	1411.2
576p/200	720x576p	none	200	96	48	48	768
24/25/30Hz Formats							
720p	1280x720p	none	24	192	192	192	1536
720p	1280x720p	none	25	192	192	192	1536
720p	1280x720p	none	29.97/30	192	192	192	1536
1080p	1920x1080p	None	24	192	192	96	1536
1080p	1920x1080p	None	25	192	176.4	96	1536
1080p	1920x1080p	None	29.97/30	96	48	48	768
2160p	3840x2160p	none	24	192	192	192	1536
2160p	3840x2160p	none	25	192	192	192	1536
2160p	3840x2160p	none	29.97/30	192	192	192	1536
2160p(SMPTE)	4096x2160p	none	24	192	192	192	1536
2160p(SMPTE)	4096x2160p	none	25	192	192	192	1536
2160p(SMPTE)	4096x2160p	none	29.97/30	192	96	96	1536

* This Specification extends the maximum audio sample frequency or frame rates up to 1536kHz.

Table 9-8: Max 3D Audio Sampling Frequencies for 24-bit 4:2:0 Video Format Timings (Informative)

Description	Format Timing	Pixel Repetition	Vertical Freq (Hz)	Max fs 10.2ch (kHz)	Max fs 22.2ch (kHz)	Max fs 30.2ch (kHz)	Max frame rate 2ch, comp*
2160p	3840x2160p	none	50	192	192	192	1536
2160p(SMPTE)	4096x2160p	none	50	192	192	192	1536
2160p	3840x2160p	none	59.94/60	192	192	192	1536
2160p(SMPTE)	4096x2160p	none	59.94/60	96	88.2	48	1024

9.3.2 3D Audio Channel/Speaker Assignment

In cases where a Sink is capable of receiving 3D Audio, the HDMI 3D Speaker Allocation Descriptor described in Table 10-11, Table 10-12, and Table 10-13 (Section 10.3.3) shall be used to indicate the configuration of attached speakers. The current speaker assignment for 3D Audio shall be indicated in the 3D_CA field of the Audio Metadata Packet (Section 8.3).

9.3.3 3D Audio Data Packetization

Each Subpacket of a 3D Audio Sample Packet shall contain zero or one IEC 60958-defined "frame". If a Source needs to down mix the 3D Audio stream, the down mixed audio streams shall also be carried using 3D Audio Sample Packets. If a Sink does not support 3D Audio, a Source shall not transmit 3D Audio Sample Packets. Converting 3D Audio into the other audio formats is beyond the scope of This Specification.

Depending on the number of channels, a number of different Subpacket layouts are defined. Table 9-9, Table 9-10, and Table 9-11 show the channel mapping for 12, 24, and 32 channel 3D Audio Sample Packets, respectively.

Table 9-9: Channel Mapping for 12 Channel 3D Audio Sample Packet

Packet #	Sample_ start Value	Num Channels	Samples	Subpacket 0	Subpacket 1	Subpacket 2	Subpacket 3
0	1	12	1	Chnl 1,2 Sample 0	Chnl 3,4 Sample 0	Chnl 5,6 Sample 0	Chnl 7,8 Sample 0
1	0			Chnl 9,10 Sample 0	Chnl 11,12 Sample 0	Empty	Empty

Table 9-10: Channel Mapping for 24 Channel 3D Audio Sample Packet

Packet #	Sample_ start Value	Num Channels	Samples	Subpacket 0	Subpacket 1	Subpacket 2	Subpacket 3
0	1	24	1	Chnl 1,2 Sample 0	Chnl 3,4 Sample 0	Chnl 5,6 Sample 0	Chnl 7,8 Sample 0
1	0			Chnl 9,10 Sample 0	Chnl 11,12 Sample 0	Chnl 13,14 Sample 0	Chnl 15,16 Sample 0
2	0			Chnl 17,18 Sample 0	Chnl 19,20 Sample 0	Chnl 21,22 Sample 0	Chnl 23,24 Sample 0

Table 9-11: Channel Mapping for 32-Channel 3D Audio Sample Packet

Packet #	Sample_ start Value	Num Channels	Samples	Subpacket 0	Subpacket 1	Subpacket 2	Subpacket 3
0	1	32	1	Chnl 1,2 Sample 0	Chnl 3,4 Sample 0	Chnl 5,6 Sample 0	Chnl 7,8 Sample 0
1	0			Chnl 9,10 Sample 0	Chnl 11,12 Sample 0	Chnl 13,14 Sample 0	Chnl 15,16 Sample 0
2	0			Chnl 17,18 Sample 0	Chnl 19,20 Sample 0	Chnl 21,22 Sample 0	Chnl 23,24 Sample 0
3	0			Chnl 25,26 Sample 0	Chnl 27,28 Sample 0	Chnl 29,30 Sample 0	Chnl 31,32 Sample 0

There are four sample_present bits in the 3D Audio Sample Packet Header, one for each Subpacket. Each sample_present bit indicates whether the corresponding Subpacket contains a part of the 3D Audio sample or not.

In addition, there are four sample_flat.spX bits which are set (=1) if no useful audio data is available at the Source. This may occur during sample rate changes or temporary stream interruptions. When sample_flat.spX is set, Subpacket X continues to represent a sample period but does not contain useful audio data. The sample_flat.spX bit is only valid when the corresponding sample_present.spX bit is set. Noting that 3D Audio requires the use of multiple 3D Audio Sample Packets to transport each sample, for each sample, all sample_flat.spX bits (with their corresponding sample_present.spX bits set) shall be set to the same value.

Sequential 3D Audio Sample Packets shall carry one 3D Audio sample which contains 12, 24, or 32 channels of L-PCM audio (i.e. 6, 12, or 16 IEC 60958 frames). The Source shall begin transmitting one or more 3D Audio Sample Packets, as the Video Blanking Period permits. If all 3D Audio Sample Packets carrying the 3D Audio Sample cannot be transmitted during one Video Blanking Period, any remaining packets shall be transmitted during the following Video Blanking Period.

When the field sample_start = "1", it indicates that the current 3D Audio Sample Packet is the first packet of a 3D Audio sample and is fully packed with 8 audio channels. When the field sample_start = "0", it indicates that the current 3D Audio Sample Packet is an intermediate or final packet of a 3D Audio sample and contains 8 or fewer audio channels.

There are five valid configurations of sample_present bits for the 3D Audio Sample Packet. They are shown in Table 9-12.

Table 9-12: Valid Sample_Present Bit Configurations for 3D Audio

SP0	SP1	SP2	SP3	Description
0	0	0	0	No Subpackets contain parts of the audio sample.
1	0	0	0	Only Subpacket 0 contains the audio sample.
1	1	0	0	Subpackets 0 and 1 contain two contiguous parts of the audio sample.
1	1	1	0	Subpackets 0, 1, and 2 contain three contiguous parts of the audio sample.
1	1	1	1	Subpackets 0, 1, 2, and 3 contain four contiguous parts of the audio sample.

Figure 9-2, Figure 9-3, and Figure 9-4 depict audio transport on active video lines. During the Vertical Blanking period, audio samples should be transported as soon as possible (i.e. as soon as they become available for transport).

Figure 9-2 depicts an example of how the channels within 2 audio samples may be transmitted on an arbitrary active video line. In the figure, the video transported is Pixel doubled 8-bit 4:4:4 or 12-bit 4:2:2 480p / 576p video. This provides a total of 276 TMDS Character Periods which can transport up to 6 packets. A similar example is provided in Figure 9-3, but with 1080p video. Here, the total available TMDS character periods is 280, also providing room for up to 6 packets. Finally, Figure 9-4 depicts how samples would be transported on the same 1080p link as in Figure 9-3, but the sample becomes available during the video blanking period. In this example, the first 8 channels are transported when they become available, and the remaining 16 channels are transported during the next blanking period.

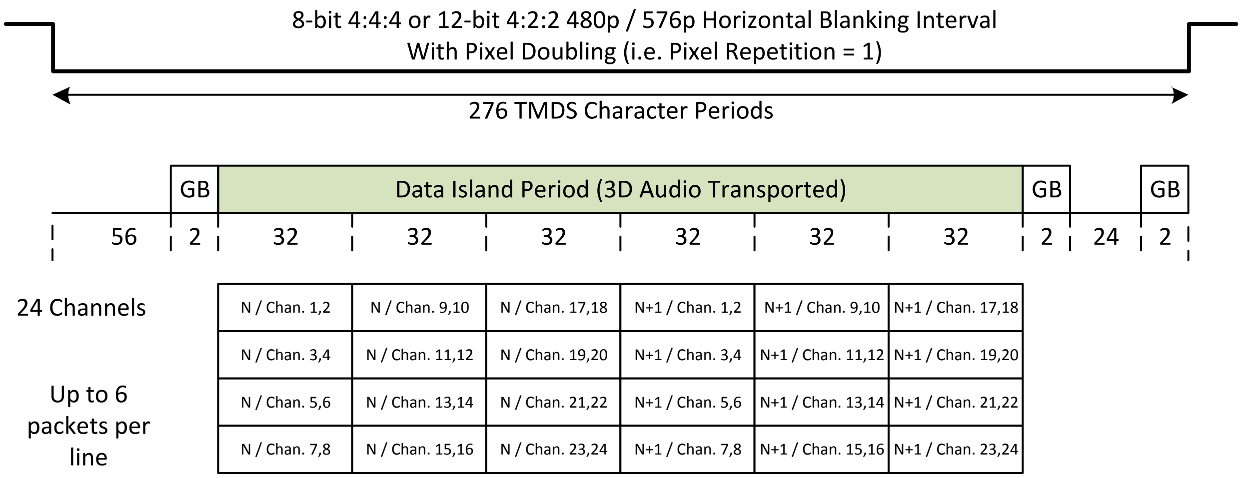


Figure 9-2: Example Audio Sample Timing for 3D Audio transmission with 480p/576p Video (Informative)

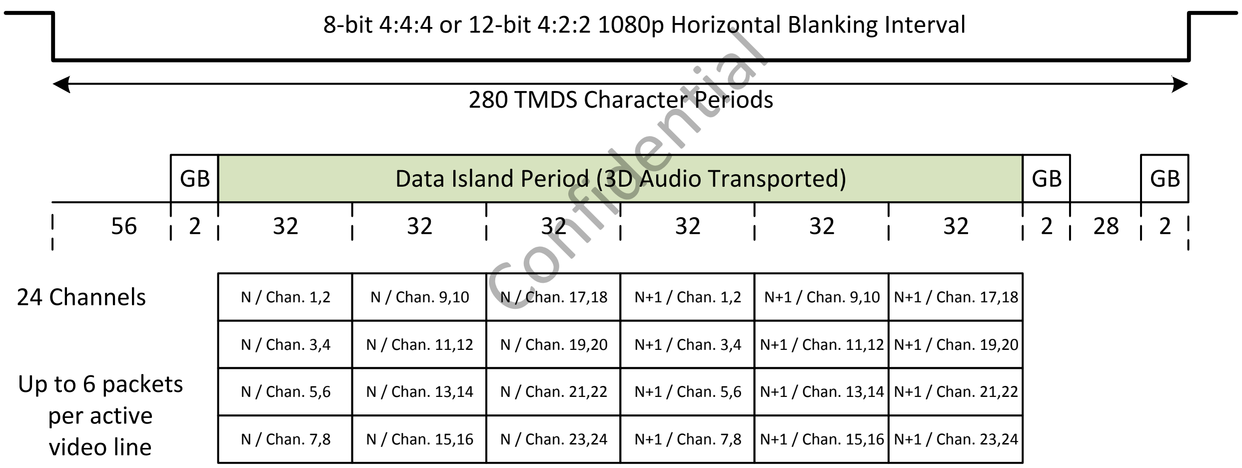


Figure 9-3: Example Audio Sample Timing for 3D Audio transmission with 1080p Video (Informative)

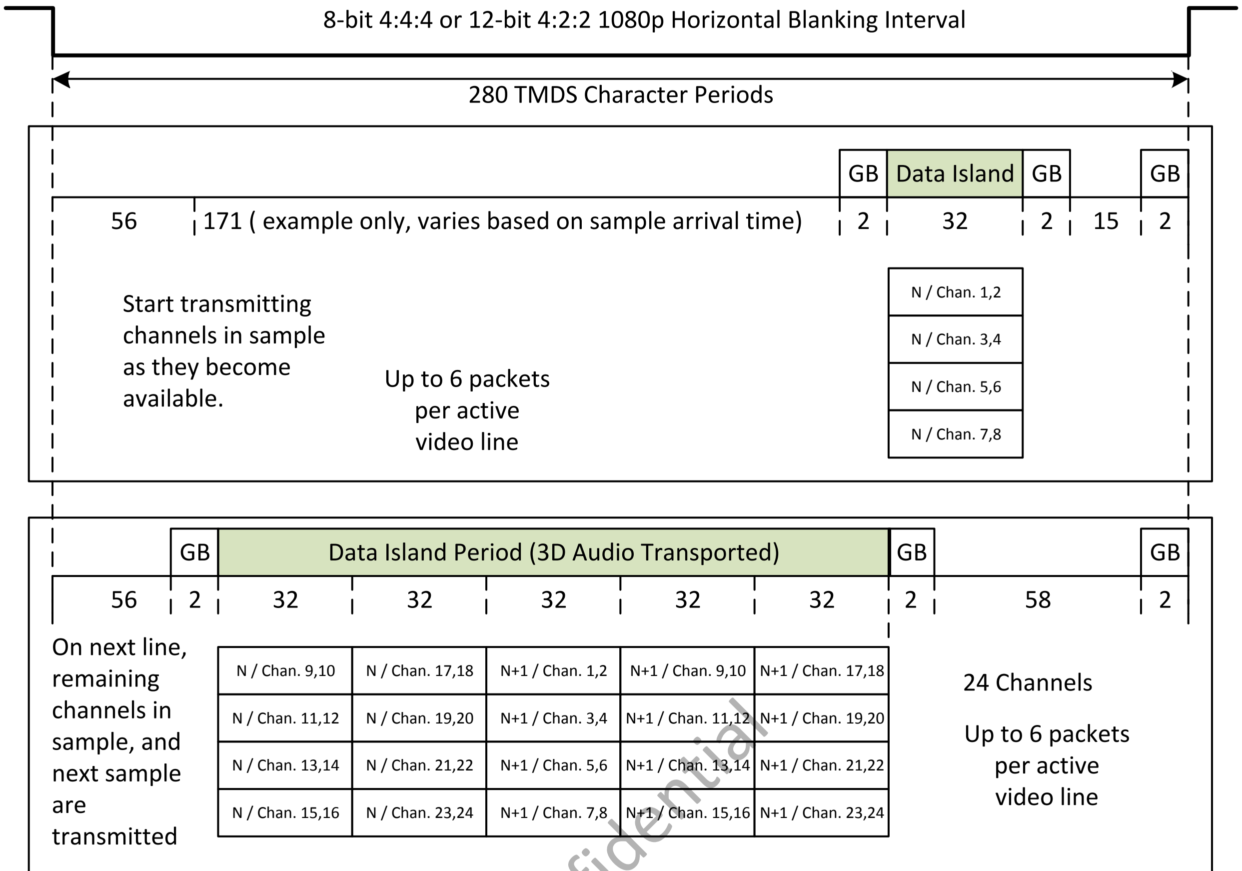


Figure 9-4: Example Audio Sample Timing for 3D Audio transmission with 1080p Video, Samples split across two video lines (Informative)

Additionally, refer to Appendix A.1 for an example on how to transmit (L-PCM encoded) 3D Audio samples.

9.3.4 One Bit 3D Audio Packetization

When transmitting One Bit 3D Audio, each Subpacket shall contain One Bit Audio bits for zero, one or two audio channels.

There are four sample_present bits in the One Bit 3D Audio Sample Packet Header, one for each Subpacket. The corresponding bit is set if that Subpacket contains audio samples. There are four samples_invalid.spX bits which are set (=1) if no useful audio data is available at the Source. When samples_invalid.spX is set, Subpacket X continues to represent a sample period but does not contain any useful data. Noting that 3D One Bit Audio requires the use of multiple 3D One Bit Audio Sample Packets to transport each sample, for each sample, all samples_invalid.spX bits (with corresponding sample_present.spX bits set) shall be set to the same value.

Contiguous One Bit 3D Audio Sample Packets can be used to carry between 9 and 32 audio channels of a One Bit 3D Audio sample.

Valid combinations of sample_present bits for One Bit 3D Audio Sample Packets are defined as in Table 9-12.

9.3.5 Audio Metadata Packetization for 3D Audio

Whenever a 3D Audio stream is being transmitted, an accurate Audio Metadata Packet shall be transmitted at least once per two Video Fields.

At the start of a new L-PCM encoded 3D Audio stream, or upon any change in such L-PCM encoded 3D Audio stream, an accurate Audio Metadata Packet should be transmitted immediately prior to transmission of the first affected audio sample packet with the sample_flat bit set to 0. If that does not occur, an accurate Audio Metadata packet shall be transmitted no later than one Video Field following the first affected audio sample with the sample_flat bit set to 0.

At the start of a new One Bit 3D Audio stream, or upon any change in such One Bit 3D Audio stream, an accurate Audio Metadata Packet shall be transmitted before the first affected sample.

The Audio Metadata Packet transmission may occur at any time within the Data Island period, including any horizontal or vertical blanking periods. When the contents of the Audio Metadata Packet are not updating, it should not be transmitted more than once per Video Field.

When 3D Audio is being received, the Sink shall ignore CC and CA fields in the Audio InfoFrame and instead refer to 3D_CC and 3D_CA in the Audio Metadata Packets.

9.4 Multi-Stream Audio

This Specification defines a mechanism to concurrently transmit 2, 3, or 4 audio streams with same sample/frame rate when supporting multi-view video streaming (e.g. dual-view gaming with different audio for each view) or single-view video streaming (e.g. multi-lingual support).

9.4.1 Multi-Stream Audio Data Packetization

Each Subpacket of a Multi-Stream Audio Sample Packet shall contain zero or one IEC 60958-defined “frames” of an IEC 60958 or IEC 61937 “block”. Three Subpacket layouts are defined. Table 9-13, Table 9-14, and Table 9-15 show the Multi-Stream Audio Packet Layout for 2, 3, and 4 audio streams, respectively. In the tables, the sample number a, b, c, and d refer to a sample in Stream 0, 1, 2, and 3 respectively.

Table 9-13: Mapping for Multi-Stream Audio Sample Packet with 2 audio streams

Packet #	Subpacket 0	Subpacket 1	Subpacket 2	Subpacket 3
0	Stream 0 Sample a	Stream 1 Sample b	Empty	Empty
1	Stream 0 Sample a+1	Stream 1 Sample b+1	Empty	Empty
…	-	-	-	-
N	Stream 0 Sample a+N	Stream 1 Sample b+N	Empty	Empty

Table 9-14: mapping for Multi-Stream Audio Sample Packet with 3 audio streams

Packet #	Subpacket 0	Subpacket 1	Subpacket 2	Subpacket 3
0	Stream 0 Sample a	Stream 1 Sample b	Stream 2 Sample c	Empty
1	Stream 0 Sample a+1	Stream 1 Sample b+1	Stream 2 Sample c+1	Empty
…	-	-	-	-
N	Stream 0 Sample a+N	Stream 1 Sample b+N	Stream 2 Sample c+N	Empty

Table 9-15: Mapping for Multi-Stream Audio Sample Packet with 4 audio streams

Packet #	Subpacket 0	Subpacket 1	Subpacket 2	Subpacket 3
0	Stream 0 Sample a	Stream 1 Sample b	Stream 2 Sample c	Stream 3 Sample d
1	Stream 0 Sample a+1	Stream 1 Sample b+1	Stream 2 Sample c+1	Stream 3 Sample d+1
…	-	-	-	-
N	Stream 0 Sample a+N	Stream 1 Sample b+N	Stream 2 Sample c+N	Stream 3 Sample d+N

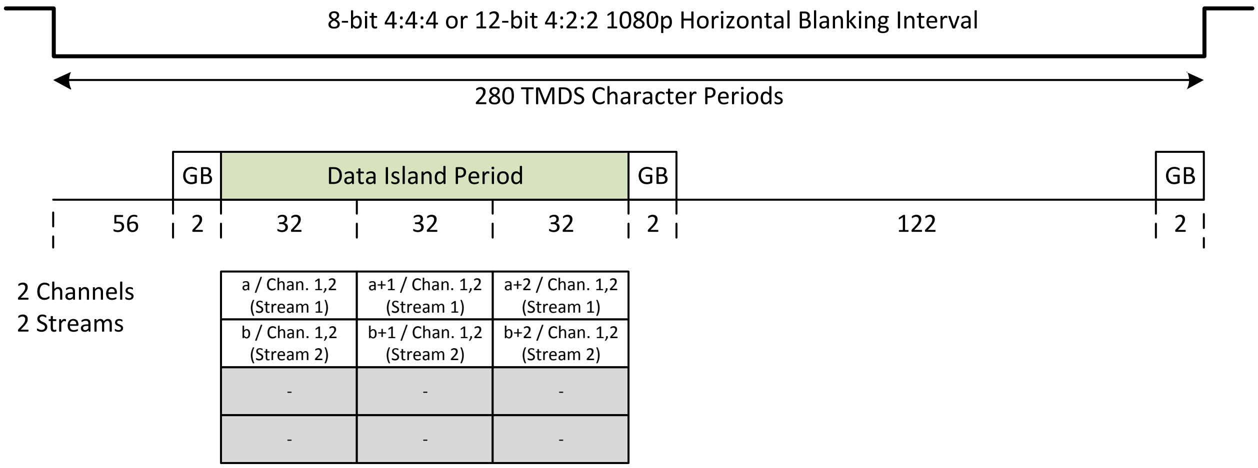
There are four stream_present bits in the Multi-Stream Audio Sample Packet Header, one for each Subpacket. The stream_present bit indicates whether the corresponding Subpacket contains an audio stream. In addition, there are four stream_flat.spX bits which are set to “1” if no useful audio data were available at the Source. This may occur during sample rate changes or temporary stream interruptions. When stream_flat.spX is set, Subpacket X continues to represent a sample period but does not contain useful audio data. The stream_flat.spX bit is only valid when the corresponding stream_present.spX bit is set.

A Multi-Stream Audio Sample Packet carries up to four audio samples where each sample corresponds to an independent audio stream. For example, if an HDMI Source is transmitting two separate audio streams, Subpacket 0 shall be used to carry an audio sample of stream 0 and Subpacket 1 shall be used to carry an audio sample of stream 1.

There are five valid configurations of stream_present bits for a Multi-Stream Audio Sample Packet. They are shown in Table 9-16

Table 9-16: Valid Stream_Present Bit Configurations for Multi-Stream Audio transmission

SP0	SP1	SP2	SP3	Description
0	0	0	0	No Subpackets contain audio samples.
1	0	0	0	Reserved.
1	1	0	0	Subpackets 0 and 1 contain audio samples for stream 0 and 1, respectively
1	1	1	0	Subpackets 0, 1, and 2 contain audio samples for stream 0, 1, and 2, respectively.
1	1	1	1	All Subpackets contain audio samples.



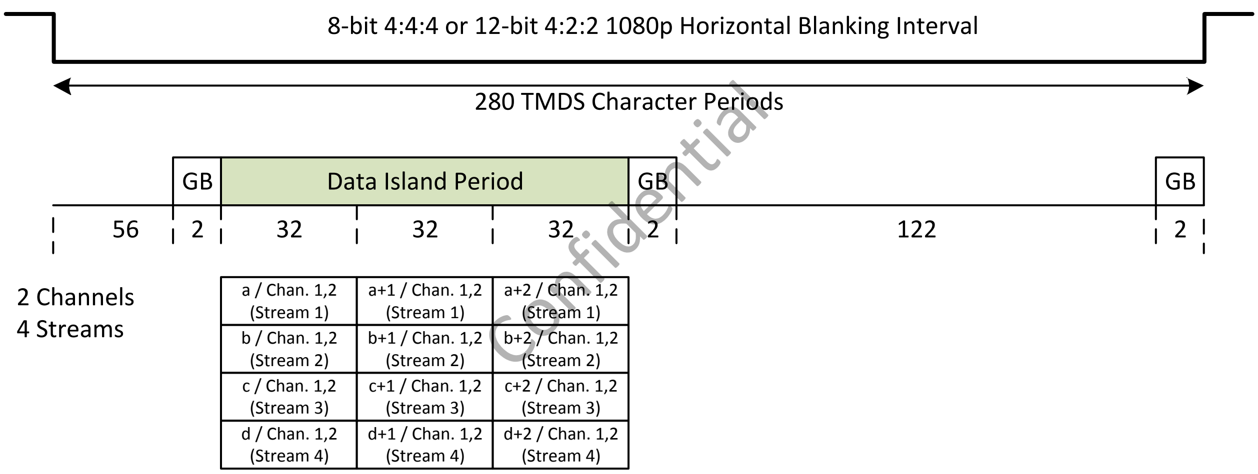


Figure 9-5: Example Audio Sample Timings for 2 Stream and 4 Stream Multi-Stream Audio transmission (Informative)

Additionally, refer to Appendix A.2 for an example on how to transmit Multi-Stream Audio samples for dual-view video streaming.

9.4.2 One Bit Multi-Stream Audio Packetization

When transmitting One Bit Multi-Stream Audio, each One Bit Multi-Stream Audio Sample Packet shall contain One Bit Audio bits for zero, two, three, or four audio streams.

There are four stream_present bits in the One Bit Multi-Stream Audio Sample Packet Header, one for each Subpacket. The corresponding bit is set if that Subpacket contains an audio sample of each individual stream. There are four stream_invalid.spX bits which are set to "1" if no useful audio data were available at the Source. When stream_invalid.spX is set, Subpacket X continues to represent a sample period but does not contain any useful data.

A One Bit Multi-Stream Audio Sample Packet carries up to four stereo One Bit Audio samples where each sample corresponds to an independent audio stream.

Valid combinations of stream_present bits for One Bit Multi-Stream Audio Sample Packets are defined as in Table 9-16.

9.4.3 Audio Metadata Packetization for Multi-Stream Audio

Whenever a Multi-Stream Audio stream is being transmitted, an accurate Audio Metadata Packet shall be transmitted at least once per two Video Fields.

At the start of a new (L-PCM or IEC 61937 compressed) Multi-Stream Audio stream or upon any change in such (L-PCM encoded or IEC 61937 compressed) Multi-Stream Audio stream, an accurate Audio Metadata Packet should be transmitted immediately prior to transmission of the first affected audio sample with the stream_flat bit set to 0. If that does not occur, an accurate Audio Metadata Packet shall be transmitted no later than one Video Field following the first affected audio sample with the stream_flat bit set to 0.

At the start of a new One Bit Multi-Stream Audio stream or upon any change in such One Bit Multi-Stream Audio stream, an accurate Audio Metadata Packet shall be transmitted before the first affected sample.

The Audio Metadata Packet transmission may occur at any time within the Data Island period, including any horizontal or vertical blanking periods. When the contents of the Audio Metadata Packet are not updating, it should not be transmitted more than once per Video Field.

10 Control and Configuration

HDMI 1.4b defines a number of Control and Configuration options. Many of these involve the DDC (i.e. I^2C) channel. In addition, for This Specification, the DDC is also used to exchange point-to-point dynamic data between the Source and the Sink using a new DDC address for the HDMI Status and Control Data Channel (SCDC).

10.1 Use of the AVI InfoFrame in This Specification

The AVI InfoFrame as defined in H14b Section 8.2.1 as used in This Specification has the following extensions as defined in CEA-861-F Section 6.4:

- VIC-field has been extended from 7 bits to 8 bits (allowing for more VIC codes), i.e.
  - VIC0...VIC7 Video Format Identification Code.
    - When transmitting any additional Video Format for which a VIC value has been defined in CEA-861-F tables 1, 2, and 3, an HDMI Source shall set the VIC field to the Video Code for that format.
- Y-field has been extended from 2 bits to 3 bits:
  - Y2,Y1,Y0=0,1,1 is used for 4:2:0 signaling (See Section 7.1)
  - CEA-861-F defines Y2, Y1, Y0=1, 1, 1 as "IDO-defined"; semantics of this value is not defined in This Specification and shall not be used by Devices built according to This Specification.
- Note: CEA-861-F Section 6.4 defines a "version 3" AVI InfoFrame, which is used instead of a "version 2" AVI InfoFrame in certain cases (Y2=1 or VIC7=1). In This Specification, it is always the case that Y2=0 (see previous bullet). Also, no VIC codes above 107 have been defined in CEA-861-F (thus VIC7=0). Therefore, the "version 3" AVI InfoFrame is not applicable for Source Devices built according to This Specification, and Source Devices shall always transmit a "version 2" AVI InfoFrame.

10.1.1 Signaling of 3D Video Formats

(‡) This section incorporates text from the HDMI Specification 1.4b Section 8.2.3.2. See Notice for copyright information.

The first paragraph of H14b Section 8.2.3.2 is extended as follows:

The 3D video format is indicated using the VIC (Video Identification Code) in the AVI InfoFrame (indicating the video format of one of the 2D pictures, as defined in CEA-861-F tables 1, 2 and 3) in conjunction with the 3D_Structure field in the HDMI Vendor Specific InfoFrame (indicating the 3D structure). In cases where the HF-VSIF needs to be used instead of H14b VSIF (see Section 10.2), the 3D_Structure field and other 3D-related signaling (e.g. 3D_Ext_Data and 3D_Metadata) is carried in the HF-VSIF, with 3D_Valid=1, see Section 10.2.

10.2 HDMI Forum Vendor Specific InfoFrame

The HDMI Forum Vendor Specific InfoFrame (HF-VSIF) Packet is provided to support features that require ancillary information to fully identify the stream content. The basic structure for a Vendor Specific InfoFrame is defined in H14b Section 5.3.5. The structure of the HF-VSIF shall conform to the definition provided in CEA-861-F, Section 6.1, for a version 1 Vendor Specific InfoFrame.

Transmission of the HF-VSIF by Source Devices is optional unless one (or more) of the features listed in Table 10-1 is active[1]. If such features are active, transmission of the HF-VSIF is mandatory. Whenever this packet is required, an accurate HF-VSIF shall be transmitted at least once per two Video Fields but no more than once per Video Field.

If it is necessary to transmit either an H14b VSIF or an HF-VSIF to support the current video stream, Source devices shall utilize the H14b VSIF whenever the signaling capabilities of the H14b VSIF allow this. Source Devices shall transmit the HF-VSIF when sending a video stream which uses one (or more) of the features listed in Table 10-1, and may transmit the HF-VSIF after a 3D-2D transition as described in Section 10.2.1. Sources shall not transmit the HF-VSIF at any other time. When the HF-VSIF is being transmitted, the Source Device shall not transmit the H14b VSIF. When the H14b VSIF is being transmitted, the Source Device shall not transmit the HF-VSIF.

Note that enabling Deep Color as defined in H14b Section 6.5.2 or Deep Color 4:2:0 as defined in Section 7.1.1 of This Specification, does not by itself require the use of the HF-VSIF.

Table 10-1: List of features that require transmission of the HF-VSIF

3D OSD Disparity Indication (Section 7.4.1)
3D Dual-View Signaling (Section 7.4.2)
3D Independent View Signaling (Section 7.4.3)

When the Video Format being transmitted corresponds to one of the Video Formats in Table 10-2, the H14b VSIF (with HDMI_VICs = 1, 2, 3, or 4) shall be utilized unless 3D Video is being transmitted or features listed in Table 10-1 are active. In the event 3D video is being transmitted in conjunction with the Video Formats listed in Table 10-2 and no features in Table 10-1 are active, the Source shall set AVI InfoFrame VIC with the corresponding "Equivalent CEA-861-F VIC" from Table 10-2 and the H14b VSIF shall be utilized to indicate the 3D Signaling. In the event 3D video is being transmitted in conjunction with the Video Formats listed in Table 10-2 and one or more features in Table 10-1 are active, the Source shall set AVI InfoFrame VIC with the corresponding "Equivalent CEA-861-F VIC" from Table 10-2 and the HF-VSIF shall be utilized to indicate the 3D Signaling.

When 3D Video is being transmitted and no features listed in Table 10-1 are active, the H14b VSIF (indicating the 3D Structure) shall be utilized in conjunction with the VICs defined in CEA-861-F for all Video Formats, including those listed in Table 10-2.

Table 10-2: H14b HDMI_VIC to CEA-861-F VIC Cross Reference

Video Format	Aspect Ratio	H14b HDMI_VIC	Equivalent CEA-861-F VIC
3840x2160p 29.97, 30 Hz	16:9	1	95
3840x2160p 25Hz	16:9	2	94
3840x2160p 23.98, 24 Hz	16:9	3	93
4096x2160p 23.98[Δ], 24 Hz	256:135	4	98

[Δ] HDMI 1.4b did not define a 23.98 Hz version of this timing, but the CEA-861-F VIC includes this rate.

Appendix E gives an overview of the signaling in the AVI InfoFrame, H14b VSIF and HF-VSIF for various Video Formats and modes (4:2:0, 2D, 3D, etc.).

[1] A feature is considered to be active when the Source is transmitting a video signal utilizing the feature.

The contents of the HF-VSIF are defined in Table 10-3, Table 10-4, and in the subsequent text. The first payload byte is the Checksum. This is followed by the IEEE Organizationally Unique Identifier (OUI) of C4-5D-D8 assigned to the HDMI Forum. This OUI shall be used by Devices compliant with This Specification to identify the VSIF as the HF-VSIF described in this section.

Table 10-3: HDMI Forum Vendor Specific InfoFrame Packet Header

Byte \ Bit #	7	6	5	4	3	2	1	0
HB0	Packet Type = 0x81							
HB1	Version = 1							
HB2	0	0	0	Length = Nv				

- Length [5 bits] This field indicates the number of bytes contained within the HF-VSIF Packet payload. It does not include the Packet Header bytes or the Checksum. Its maximum value is 27 (0x1B).

Table 10-4: HDMI Forum Vendor Specific InfoFrame Packet Contents

Byte \ Bit #	7	6	5	4	3	2	1	0
PB0	Checksum							
PB1	IEEE OUI, Third Octet (0xD8)							
PB2	IEEE OUI, Second Octet (0x5D)							
PB3	IEEE OUI, First Octet (0xC4)							
PB4	Version (=1)							
PB5	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	3D_Valid
(PB6)*	If(3D_Valid is set(=1)) then 3D_F_Structure				3D_ Additional Info_ present	3D_ Disparity Data_ Present	3D_ Meta_ present	Rsvd (0)
(PB7)*	If((3D_Valid is set(=1)) and (3D_F_Structure == 0b1000..0b1111)) then 3D_F_Ext_Data				Rsvd (0)			
(PB8)*	If(3D_Additionalinfo_present is set(=1)) then this byte contains 3D_AdditionalInfo: Rsvd (0)	Rsvd (0)	Rsvd (0)	3D_ DualView	3D_ ViewDependency		3D_ Preferred2DView	
(PB9)*	If(3D_DisparityData_present is set(=1)) then 3D_DisparityData_version			3D_DisparityData_length(J)				
(PB9+1)*	3D_DisparityData_1							
…	…							
(PB9+J)*	3D_DisparityData_J							
(PBm)*	If(3D_Meta_present is set(=1)) then 3D_Metadata_type			3D_Metadata_Length(K)				
(PBm+1)*	3D_Metadata_1							
	…							
(PBm+K)*	3D_Metadata_K							
…PB(Nv)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)

* Presence and location of these fields may vary depending on the value of “3D_Valid” in PB5, the “present” bits in PB6 (if these flags are present), and the value of 3D_F_Structure (if present)

Checksum	[1 Byte]	Checksum of the InfoFrame. The checksum shall be calculated such that a byte-wide sum of all three bytes of the Packet Header and all valid bytes of the HF-VSIF Packet contents (determined by Length), plus the Checksum itself, equals zero.
IEEE OUI	[3 Bytes]	The IEEE Organizationally Unique Identifier (OUI) of C4-5D-D8 assigned to the HDMI Forum.
Version	[1 Byte]	Version number associated with the contents of the HF-VSIF. Source Devices compliant with This Specification shall set this value to 1.
3D_Valid	[1 bit]	If set (=1), 3D_F_Structure, 3D_AdditionalInfo_present, 3D_DisparityData_present, 3D_Meta_present, and 3D_F_Ext_Data field (if 3D_F_Structure = 0b1000..0b1111) shall be present and valid. If clear (=0), 3D_F_Structure, 3D_AdditionalInfo_present, 3D_DisparityData_present, 3D_Meta_present, and 3D_F_Ext_Data field (if 3D_F_Structure = 0b1000..0b1111) shall not be present, resulting in a HF-VSIF (in This Specification) with bytes PB6..PB27 filled with 0.
3D_F_Structure	[4 bits]	The definition of this parameter is identical to the definition of the 3D_Structure parameter in H14b Section 8.2.3 and H14b Appendix H.1.
3D_F_Ext_Data	[4 bits]	The definition of this parameter is identical to the definition of the 3D_Ext_Data parameter in H14b Section 8.2.3 and H14b Appendix H.1.
3D_Meta_present	[1 bit]	If set (=1), 3D_Metadata_type, 3D_Metadata_Length(K) and 3D_Metadata_1 through _K shall be present and valid. 3D_Metadata_type, 3D_Metadata_Length (K) and 3D_Metadata_1 through _K are defined in H14b Appendix H.1.
3D_AdditionalInfo_present	[1 bit]	if set (=1), a 3D_AdditionalInfo byte shall be present, if reset(=0), 3D_AdditionalInfo shall not be present. The 3D_AdditionalInfo byte contains the 3D_DualView, 3D_ViewDependency, and 3D_Preferred2DView fields.
3D_DualView	[1 bit]	This value differentiates between ‘normal’ 3D video transmission and the use of the 3D transport mechanism for dual-view use cases (See Section 7.4.2). If reset (=0), the video being transmitted is ‘normal’ 3D video, as defined in HDMI 1.4b. If set (=1), Dual View mode is enabled. Here, the Source sends a 3D signal (as defined in HDMI 1.4b), and utilizes the “left” image for one view, and the “right” for the other view. This signaling is used by the TV to adapt processing and/or instruct the user(s)/3D-glasses accordingly.

- 3D_ViewDependency [2 bits]

  Indicates view dependency; used by the Source to indicate which of the views has or have been independently coded. (See Sections 7.4.3 and 7.4.3.1)

  0b00 No indication.

  0b01 The right view originates from an independently coded view.

  0b10 The left view originates from an independently coded view.

  0b11 Both views are from (substantially) independently coded views.

- 3D_Preferred2DView [2 bits]

  Indicates which view (if any) is preferred for 2D viewing. This is used by the Source to indicate which of the two views should be used for 2D viewing (e.g. if the Sink is displaying in 2D when the incoming signal is 3D). (See Sections 7.4.3 and 7.4.3.2)

  0b00 No indication.

  0b01 Use the right 3D view for 2D viewing.

  0b10 Use left 3D view for 2D viewing.

  0b11 Don't care.

  If Dual View mode is signaled by the Source (3D_DualView is set (=1)), 3D_ViewDependency shall be set to 0b00 or 0b11, and 3D_Preferred2DView shall be set to 0b00 or 0b11.

  If 3D_ViewDependency is set to 0b01 or 0b10, 3D_DualView shall be set to 0.

  If 3D_Preferred2DView is set to 0b01 or 0b10, 3D_DualView shall be set to 0.

- 3D_DisparityData_present [1 bit]

  If set (=1), 3D_DisparityData_version and 3D_DisparityData_length and 3D_DisparityData shall be present and valid. See more information below and in Section 7.4.1.

  A Source shall not set (=1) 3D_DisparityData_present unless it is sending a 3D Video Format, 3D_DualView is reset (=0), and Disparity information is available.

- 3D_DisparityData_version [3 bits]

  3D_DisparityData_version indicates the structure of the depth information. See Section 7.4.1.

- 3D_DisparityData_length [5 bits] 3D_DisparityData_length indicates the size in bytes of the 3D_DisparityData block; this size is dependent on the 3D_DisparityData_version (see Section 7.4.1). (Note – the byte with 3D_DisparityData_version and 3D_DisparityData_length is NOT counted in the length). Sinks that do not recognize the version (or do not need the contents of the block in their current state) shall skip over the block using the length indication.
- 3D_DisparityData [J byte] 3D_DisparityData is used by the Source to convey depth information. See Section 7.4.1.

10.2.1 HF-VSIF Transitions

This Section clarifies and defines the use of the HF-VSIF, specifically with respect to changes in the content of the VSIF; such changes include the beginning and end of HF-VSIF transmission, and also include switching from transmission of HF-VSIF to H14b VSIF and vice versa.

When there is any change in the HF-VSIF, the Sink shall begin to adapt its display processing in an appropriate manner within 1 second. This includes changes from “no HF-VSIF is being transmitted” to “HF-VSIF is being transmitted” and vice versa, as well as changes from H14b VSIF to HF-VSIF and vice versa.

When a Source is utilizing the HF-VSIF for 3D Video signaling and changes its transmission from 3D to 2D, the Source shall signal the end of 3D transmission by sending an appropriate HF-VSIF (with 3D_Valid=0) after the change from 3D to 2D, for at least 2 seconds or until re-start of HDMI video is necessary.

When the Source stops transmitting the HF-VSIF, the Sink shall interpret this as an indication that transmission of features described in this InfoFrame has stopped (e.g. transition from 3D as previously signaled in the InfoFrame to (default) 2D).

For recommendations regarding H14b VSIF transitions, see Appendix F.

10.3 E-EDID

10.3.1 Signaling of supported Video Formats

(‡) This section incorporates text from the HDMI Specification 1.4b Section 8.3.6. See Notice for copyright information.

The 4th paragraph of H14b Section 8.3.6 is extended as follows:

To indicate support for any Video Format in CEA-861-F Tables 1, 2 and 3, an HDMI Sink shall use a Short Video Descriptor (SVD) containing the Video Code for that format and may also use a Detailed Timing Descriptor (DTD).

Sinks that support one or more formats listed in Table 10-2 shall declare these both in the H14b VSDB (with HDMI_VIC = 01..04) as well as in the Video Data Block (with VIC codes from Table 10-2).

For requirements on Sinks that support YC_BC_R 4:2:0 Pixel Encoding, see Section 7.1.2.

10.3.2 HDMI Forum Vendor Specific Data Block

The HDMI Forum VSDB (HF-VSDB) is utilized by Sink Devices to indicate support features that have been defined by This Specification. This is a Vendor Specific Data Block (VSDB) as defined in CEA-861-F, Section 7.5.4. An H14b VSDB shall always be included, regardless of the inclusion of an HF-VSDB, to ensure correct functioning of DVI/HDMI discrimination as described in H14b Section 8.3.5.

The HF-VSDB must not be confused with the H14b VSDB defined in H14b Section 8.3.2. The HF-VSDB does not replace the H14b VSDB. Sinks compliant with This Specification must include the H14b VSDB as required by HDMI 1.4b, even if they include an HF-VSDB.

Inclusion of the HF-VSDB in E-EDID by Sink Devices is optional unless the Sink supports one (or more) of the features listed in Table 10-5. If one (or more) of the features listed in Table 10-5 are supported, the inclusion of the HF-VSDB in E-EDID is mandatory.

Table 10-5: List of features that require inclusion of the HF-VSDB

Deep Color 4:2:0 Indication (Section 7.1.1)
3D OSD Disparity Indication (Section 7.4.1)
3D Dual-View Signaling (Section 7.4.2)
3D Independent View Signaling (Section 7.4.3)
Status and Data Control Channel (Section 10.4)
340Mcsc to 600Mcsc TMDS Character Rate (Section 6.1.1)[Δ]
Scrambling for EMI/RFI Reduction (Section 6.1.2)[Δ]
Character Error Detection (Section 6.2)[Δ]

[Δ]This feature requires SCDC and hence requires inclusion of the HF-VSDB

If included, the HF-VSDB shall be located in a CEA Extension version 3 in the E-EDID of a Sink Device, immediately following the H14b VSDB (defined by HDMI 1.4b) in the E-EDID. CEA Extension version 3 details are described in CEA-861-F Section 7.5. Further details on the requirements of the data structures in the E-EDID are given in CEA-861-F Section 7.5 and implementation examples are given in CEA-861-F Annex A.

Source Devices that support one or more of the features in Table 10-5 shall be capable of parsing an HF-VSDB of any valid length. In future specifications, new fields may be defined. These additional fields will be defined such that a zero value indicates the same characteristics as is indicated if the field was not present. Sources compliant with This Specification shall use the length field to determine which fields are present, and shall process the HF-VSDB without considering non-zero values in fields defined as Reserved by This Specification.

The structure of the HF-VSDB is depicted in Table 10-6.

The first byte of the block indicates that the block is a VSDB and also indicates the length of the VSDB. The second, third, and fourth bytes contain the IEEE Organizationally Unique Identifier (OUI) of C4-5D-D8 assigned to the HDMI Forum. This OUI shall be used by Devices compliant with This Specification to identify the VSDB as the HF-VSDB described in this section.

Table 10-6: HDMI Forum Vendor Specific Data Block

Byte \ Bit #	7	6	5	4	3	2	1	0
0	Vendor Specific Tag Code (=3)			Length (=N)				
1	IEEE OUI, Third Octet (0xD8)							
2	IEEE OUI, Second Octet (0x5D)							
3	IEEE OUI, First Octet (0xC4)							
4	Version (=1)							
5	Max_TMDS_Character_Rate							
6	SCDC_ Present	RR_ Capable	Rsvd(0)	Rsvd(0)	LTE_340Mcsc _scramble	Independent _view	Dual_View	3D_OSD_ Disparity
7	Rsvd(0)	Rsvd(0)	Rsvd(0)	Rsvd(0)	Rsvd(0)	DC_48bit _420	DC_36bit _420	DC_30bit _420
…N	Reserved(0)*							

* No additional bytes are necessary but if present, they shall be zero.

- Length [5 bits] Total length of data block, not including this byte. The minimum value is 7 and the maximum value is 31.
- IEEE OUI [3 Bytes] The IEEE Organizationally Unique Identifier (OUI) of C4-5D-D8 assigned to the HDMI Forum.
- Version [1 Byte] Version number associated with the contents of the HF-VSDB. Sink Devices compliant with This Specification shall set this value to 1.
- Max_TMDS_ Character_Rate [1 byte] Indicates the maximum TMDS Character Rate supported.
  - The maximum Rate = Max_TMDS_Character_Rate * 5 MHz.
  - If the Sink does not support TMDS Character Rates > 340 Mcsc, then the Sink shall set this field to 0.
  - If the Sink supports TMDS Character Rates > 340 Mcsc, the Sink shall set Max_TMDS_Character_Rate appropriately and non-zero.

  This field may be set by the Sink to a value below the TMDS Character Rate corresponding to the maximum Pixel clock rate at the maximum color depth. This allows the Sink to support higher color depths at lower resolutions than it can support at higher resolutions.

  This Specification does not change the requirement to set the Max_TMDS_Clock field according to the definition of the H14b VSDB in H14b Section 8.3.2, reflecting the maximum supported TMDS Clock Rate for TMDS Character Rates ≤ 340 Mcsc.
- 3D_OSD_Disparity [1 bit] When set (=1), the Sink supports receiving 3D_OSD_Disparity Indication in the HF-VSIF.
  When reset (=0), the Sink does not support receiving 3D_OSD_Disparity Indication in the HF-VSIF.
- Dual_View [1 bit] When set (=1), the Sink supports receiving 3D Dual View signaling in the HF-VSIF.
  When reset (=0), the Sink does not support receiving 3D Dual View signaling in the HF-VSIF.
- Independent_View [1 bit] When set (=1), the Sink supports receiving 3D Independent View signaling in the HF-VSIF.
  When reset (=0), the Sink does not support receiving 3D Independent View signaling in the HF-VSIF.

- LTE_340Mcsc_scramble [1 bit] Less than or equal to 340 Mcsc scrambling support indication
  When set (=1), the Sink supports scrambling for TMDS Character Rates at or below 340 Mcsc.
  When reset (=0), the Sink does not support scrambling for TMDS Character Rates at or below 340 Mcsc.
  When SCDC_Present =0, this field shall also be=0.
  Note: Scrambling is always required when the transmitted TMDS Character Rate is greater than 340 Mcsc.
- RR_Capable [1 bit] When set (=1), the Sink is capable of initiating an SCDC Read Request.
  When reset (=0), the Sink is not capable of initiating an SCDC Read Request.
  When SCDC_Present =0, this field shall also be =0..
- SCDC_Present [1 bit] When set (=1), the Sink supports SCDC functionality.
  When reset (=0), the Sink does not support SCDC functionality.
- DC_30bit_420 [1 bit] When set (=1), the Sink supports 10-bits/component Deep Color 4:2:0 Pixel Encoding.
  When reset (=0), the Sink does not support 10-bits/component Deep Color 4:2:0 Pixel Encoding.
- DC_36bit_420 [1 bit] When set (=1), the Sink supports 12-bits/component Deep Color 4:2:0 Pixel Encoding.
  When reset (=0), the Sink does not support 12-bits/component Deep Color 4:2:0 Pixel Encoding.
- DC_48bit_420 [1 bit] When set (=1), the Sink supports 16-bits/component Deep Color 4:2:0 Pixel Encoding.
  When reset (=0), the Sink does not support 16-bits/component Deep Color 4:2:0 Pixel Encoding.

10.3.3 HDMI Audio Data Block

The HDMI Audio Data Block (HDMI ADB) allows a Sink to declare Multi-Stream Audio, 3D Audio and 3D speaker allocation capabilities.

In order to ensure backwards compatibility, the Source shall have the ability to process an HDMI ADB of any length up to 32 bytes. In future revisions of the specification, new fields may be defined. These additional fields will be defined such that a zero value indicates non-support of new features or characteristics. Sources shall use the length fields to determine which extension fields are present, and shall process the HDMI ADB with no regard to non-zero values in fields defined as Reserved in This Specification.

When a Sink supports Multi-Stream Audio and/or 3D Audio transmission, an HDMI ADB with an Extended Tag Code 18 shall be used to indicate support for Multi-Stream Audio,3D Audio characteristics, and 3D speaker allocation information.

- If a Sink supports Multi-Stream Audio transmission, it shall set the appropriate fields in Byte 3 of the HDMI ADB.
- If a Sink supports 3D Audio transmission, the HDMI ADB shall include one or more HDMI 3D Audio Descriptors (HDMI_3D_AD) following Byte 4.
- If a Sink supports 3D Audio transmission, the HDMI ADB shall include one HDMI 3D Speaker Allocation Descriptor (HDMI_3D_SAD) following the last HDMI 3D Audio Descriptor.

For 3D Audio transmission, audio characteristics and speaker allocation support shall be indicated in the HDMI Audio Block and not in the Short Audio Descriptors in the CEA-defined Audio Data Block and Speaker Allocation Data Block. In those CEA-defined blocks, audio characteristics and speaker allocation support (for 8 or less speakers) is indicated as defined in HDMI 1.4b. When transmitting the 3D Audio packets and the Audio Metadata Packet, the Source shall only send 3D Audio formats and channel/speaker configurations that the Sink supports as indicated in the HDMI ADB. See Table 10-7 for details.

The HDMI 3D Audio Descriptors described below indicate support for audio encodings listed in CEA-861-F Table 27. HDMI devices shall only support 3D Audio formats that conform to either ITU-R BS.2159-4 (Type B 10.2ch), SMPTE 2036-2 (22.2ch), or IEC 62574 (30.2ch). See Table 10-9, and Table 10-10 for details. These tables are classified according to the Audio Format Code given in Table 27 of CEA-861-F. For 3D Audio, This Specification supports Audio Format Codes 01 and 09.

As described above, an HDMI 3D Speaker Allocation Descriptor may also be included in the HDMI ADB and is required for Sinks supporting 3D Audio. A Sink shall indicate its audio capability by indicating which speaker, or pair of speakers, is present by setting the corresponding flag to one. The speaker designations are the same as those used in the Audio Metadata Packet. The HDMI 3D Speaker Allocation Descriptor shall also contain the 4-bit ACAT field to indicate the type of audio channel allocation standard. See Table 10-11, Table 10-12, and Table 10-13 for details. Refer to Appendix B for additional information concerning the relationship between the channel allocation in CEA-861-F Section 7.5.3 and audio channel allocation standards for 3D Audio.

For Multi-Stream Audio transmission, audio characteristics and speaker allocation support shall be indicated in the Short Audio Descriptors in CEA-defined Audio Data Block and Speaker Allocation Data Block. The HDMI ADB includes fields that allow the Sink to indicate support for Multi-Stream Audio transmission. When transmitting the Multi-Stream Audio packets and the Audio Metadata Packet, the Source shall only send Multi-Stream Audio configurations that the Sink supports as indicated in the HDMI ADB.

Table 10-7: HDMI Audio Data Block

Byte \ Bit #	7	6	5	4	3	2	1	0
1	Tag code=7 (Use Extended Tag)			L = Length of following data block payload (in bytes)Δ				
2	Extended Tag Code = 18 (0x12)							
3	Rsvd (0)					Supports_ MS_ NonMixed	Max_Stream_Count	
4	Rsvd (0)					NUM_HDMI_3D_AD (=X)Θ		
(5) - (8) ◊	(if NUM_HDMI_3D_AD > 0)					HDMI_3D_AD_1		
…	…							
(4*X+1) to (4*X+4) ◊	(if NUM_HDMI_3D_AD > 0)					HDMI_3D_AD_X		
(4*X+5) to (4*X+8) ◊	(if NUM_HDMI_3D_AD > 0)					HDMI_3D_SAD		

Δ If X>0 then L=4*X+7, otherwise L=3
Θ X shall be limited to X ≤ 6
◊ Assumes X>0

- Max_Stream_Count [2 bits] Indicates the maximum number of audio streams that the Sink is capable of receiving with Multi-Stream Audio packets. Refer to Table 10-8
- Supports_MS_NonMixed [1 bit] If set (=1), when receiving Multi-Stream Audio samples, the Sink supports mixing of a supplementary audio stream for visually/hearing impaired (Suppl_A_Valid=1 and Suppl_A_Mixed=0) with a main audio stream (Suppl_A_Valid=0) and rendering the combined audio. If cleared (=0), the Sink does not support this mixing capability. However, it can still receive and render pre-mixed audio streams (Suppl_A_Valid=1 and Suppl_A_Mixed=1).

  Suppl_A_Valid and Suppl_A_Mixed are transmitted in the Audio Metadata Packet, and their definitions, as they apply to Multi-Stream Audio, are in Section 8.3.2.

- NUM_HDMI_3D_AD [3 bits] indicates the number of HDMI 3D Audio Descriptors
- HDMI_3D_AD [4 bytes] HDMI 3D Audio Descriptor
- HDMI_3D_SAD [4 bytes] HDMI 3D Speaker Allocation Descriptor

Table 10-8: Max_Stream_Count field

Max_Stream_Count	Description
0b00	Sink does not support Multi-Stream Audio
0b01	Sink supports Multi-Stream Audio with 2 audio streams
0b10	Sink supports Multi-Stream Audio with 2 or 3 audio streams
0b11	Sink supports Multi-Stream Audio with 2, 3, or 4 audio streams

Table 10-9: HDMI 3D Audio Descriptor for Audio Format Code = 1 (L-PCM)

Byte \ Bit #	7	6	5	4	3	2	1	0
1	0	0	0	0	Audio Format Code = 0b0001			
2	0	0	0	Max Number of channels - 1				
3	0	192 kHz	176.4 kHz	96 kHz	88.2 kHz	48 kHz	44.1 kHz	32 kHz
4	0	0	0	0	0	24 bit	20 bit	16 bit

Table 10-10: HDMI 3D Audio Descriptor for Audio Format Code 9

Byte \ Bit #	7	6	5	4	3	2	1	0
1	0	0	0	0	Audio Format Code = 0b1001			
2	0	0	0	Max Number of channels - 1				
3	0	192 kHz	176.4 kHz	96 kHz	88.2 kHz	48 kHz	44.1 kHz	32 kHz
4	Audio Format Code dependent value (see CEA-861-F, Table 51)							

Table 10-11: HDMI 3D Speaker Allocation Descriptor for 10.2 channels (ITU-R BS.2159-4 (Type B 10.2ch))

Byte \ Bit #	7	6	5	4	3	2	1	0
PB1	FLW/FRW	Rsvd(0)	FLC/FRC	BC	BL/BR	FC	LFE1	FL/FR
PB2	TpSiL/TpSiR	SiL/SiR	TpBC	LFE2	LS/RS	TpFC	TpC	TpFL/TpFR
PB3	0	0	0	LSd/LRd	TpLS/TpRS	BtFL/BtFR	BtFC	TpBL/TpBR
PB4	ACAT (=0x01)				0	0	0	0

Shaded cells indicate the designated speakers associated with 10.2 channels.

Table 10-12: HDMI 3D Speaker Allocation Descriptor for 22.2 channels (SMPTE 2036-2)

Byte \ Bit #	7	6	5	4	3	2	1	0
PB1	FLW/FRW	Rsvd(0)	FLC/FRC	BC	BL/BR	FC	LFE1	FL/FR
PB2	TpSiL/TpSiR	SiL/SiR	TpBC	LFE2	LS/RS	TpFC	TpC	TpFL/TpFR
PB3	0	0	0	LSd/LRd	TpLS/TpRS	BtFL/BtFR	BtFC	TpBL/TpBR
PB4	ACAT (=0x02)				0	0	0	0

Shaded cells indicate the designated speakers associated with 22.2 channels.

Table 10-13: HDMI 3D Speaker Allocation Descriptor for 30.2 channels (IEC 62574 ed 1.0)

Byte \ Bit #	7	6	5	4	3	2	1	0
PB1	FLW/FRW	Rsvd(0)	FLC/FRC	BC	BL/BR	FC	LFE1	FL/FR
PB2	TpSiL/TpSiR	SiL/SiR	TpBC	LFE2	LS/RS	TpFC	TpC	TpFL/TpFR
PB3	0	0	0	LSd/LRd	TpLS/TpRS	BtFL/BtFR	BtFC	TpBL/TpBR
PB4	ACAT (=0x03)				0	0	0	0

Shaded cells indicate the designated speakers associated with 30.2 channels.

Table 10-14: Audio Channel Allocation Type (ACAT) field

ACAT.3	ACAT.2	ACAT.1	ACAT.0	Description
0	0	0	0	Reserved
0	0	0	1	Refer to 10.2 channels (ITU-R BS.2159-4 (Type B 10.2ch))
0	0	1	0	Refer to 22.2 channels (SMPTE 2036-2)
0	0	1	1	Refer to 30.2 channels (IEC 62574 ed 1.0)
0	1	0	0	Reserved
…				
1	1	1	1	

10.4 Status and Control Data Channel

The Status and Control Data Channel (SCDC) is an I^2C based system and is described in this section. It is a point-to-point communication protocol enabling the exchange of data between an HDMI Source and an attached HDMI Sink. The SCDC makes use of the same I^2C interface used in HDMI 1.4b for reading E-EDID and other purposes. This protocol extends the I^2C standard by providing a mechanism for the Sink Device (I^2C Slave) to request a Source Device (I^2C Master) to initiate a status check read.

Sink Devices that include the SCDC shall incorporate a valid HF-VSDB in the E-EDID and set (=1) the SCDC_Present bit. Prior to accessing the SCDC, Source Devices shall verify that the attached Sink Device incorporates a valid HF-VSDB in the E-EDID in which the SCDC_Present bit is set (=1). Source devices shall not attempt to access the SCDC unless the SCDC_Present bit is set (=1).

Sink Devices in Repeaters that do not implement SCDC shall ensure that, if the HF-VSDB is included in the E-EDID, the SCDC_Present bit is reset (=0). When the Sink is implemented in an HDMI Repeater, the SCDC registers reflect the properties, status and behavior of the Repeater, and do not reflect the properties, status and behavior of any downstream device except where indicated otherwise.

10.4.1 Status and Control Data Channel Structure

10.4.1.1 Format Overview

Table 10-15 provides a top level memory map for the Status and Control Data Channel Structure (SCDCS). Multi-byte values are stored in Little-Endian format. Reserved fields shall be equal to 0x00.

Table 10-15: Status and Control Data Channel Structure

Offset	R/W	Name	Description
0x01	R	Sink Version	See Section 10.4.1.2
0x02	R/W	Source Version	See Section 10.4.1.2
0x10	R/W	Update_0	See Section 10.4.1.3
0x11	R/W	Update_1	See Section 10.4.1.3
0x12-0x1F	R	Reserved for Update Related Uses	Sink Returns 0
0x20	R/W	TMDS_Config	See Section 10.4.1.4
0x21	R	Scrambler_Status	See Section 10.4.1.5
0x30	R/W	Config_0	See Section 10.4.1.6
0x31-0x3F	R	Reserved for Configuration	Sink Returns 0
0x40	R	Status_Flags_0	See Section 10.4.1.7
0x41	R	Status_Flags_1	See Section 10.4.1.7
0x42-0x4F	R	Reserved for Status Related Uses	Sink Returns 0
0x50	R	Err_Det_0_L	See Sections 6.2 and 10.4.1.8
0x51	R	Err_Det_0_H	See Sections 6.2 and 10.4.1.8
0x52	R	Err_Det_1_L	See Sections 6.2 and 10.4.1.8
0x53	R	Err_Det_1_H	See Sections 6.2 and 10.4.1.8
0x54	R	Err_Det_2_L	See Sections 6.2 and 10.4.1.8
0x55	R	Err_Det_2_H	See Sections 6.2 and 10.4.1.8
0x56	R	Err_Det_Checksum	See Sections 6.2 and 10.4.1.8
0xC0	R/W	Test_Config_0	See Section 10.4.1.9
0xC1-0xCF	R	Reserved for test features	Sink Returns 0
0xD0	R	Manufacturer IEEE OUI, Third Octet	See Section 10.4.1.10
0xD1	R	Manufacturer IEEE OUI, Second Octet	See Section 10.4.1.10
0xD2	R	Manufacturer IEEE OUI, First Octet	See Section 10.4.1.10
0xD3-0xDD	R	Device ID	See Section 10.4.1.10
0xDE-0xFF	R/W	ManufacturerSpecific	See Section 10.4.1.10
All Remaining Offsets	R	Reserved	Sink Returns 0

10.4.1.2 Version

The Sink shall set the accurate values for the Sink in the Sink Version field in the SCDCS. Sink Devices compliant with this version of the specification shall set Sink Version = 1. Source Devices can determine the Sink Device Version by reading this field.

Table 10-16: SCDCS - Sink Version, Read Only

Offset	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x01	Sink Version (1)							

After the Source Device has read a valid HF-VSDB from the Sink Device E-EDID and has determined that SCDC_Present is set (=1), the Source Device should write the accurate Version of the Source Device to the Source Version field in the SCDCS. Source Devices compliant with This Specification should set the Source Version. If a Source Device compliant with This Specification writes the Source Version field, then it shall set the Source Version = 1. The Sink may interpret either the value 0 or the value 1 as indicating a Source Device that implements the version of the SCDC registers defined in This Specification.

Whenever the Hot Plug Detect pin has voltage = low for 100 ms or more, the Sink shall reset the Source Version field to 0.

Table 10-17: SCDCS - Source Version, Read/Write

Offset	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x02	Source Version							

It is anticipated that future versions of This Specification will be developed. When those versions are developed, additional rules regarding the allowable settings of Sink and Source Version number may be implemented.

10.4.1.3 Update Flags

Several features in This Specification utilize Update Flags that are accessed via the SCDC. The Update Flags apply when both the Source and Sink devices support one or more features that rely on the Update flags.

The purpose of these flags is to provide a mechanism for the Sink Device to efficiently inform the Source Device that additional Source Device action may be required. In practice, the Source Device will read these flags to determine if any have been set (=1) by the Sink Device. If so, and if the Source Device is currently supporting a feature that utilizes the Update Flag (or flags) that have been set (=1), the Source will respond by taking the actions required by that feature.

The Sink Device shall set (=1) the Update Flags according to the requirements of the feature related to each of the Update flags (see bullet list following Table 10-18).

For each bit in the Update Flags register, the Sink Device shall reset (=0) the Update Flag under the following conditions:

- when the Source Device writes a “1” to the bit location of a particular flag, or
- when the +5V Power Signal from the Source is not present, or
- when the Hot Plug Detect pin has voltage = low for 100 ms or more, or
- for features that are not supported by the Sink Device.

When the Read Request is active, Source Devices shall read the update flags based on the rules and requirements of Section 10.4.4, Section 10.4.5, and Section 10.4.6. Otherwise, the Source Device shall periodically read the Update Flags based on the requirements of Section 10.4.1.3.1.

The positions of the Update Flag fields are summarized in Table 10-18. All update flags are readable and writable.

Table 10-18: SCDCS - Update Flags

Offset	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x10	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	RR_Test	CED_ Update	Status_ Update
0x11	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)

- Status_Update [1 bit] The Sink shall set (=1) this bit when a value is changed in the Status Flags register defined in Section 10.4.1.7. The Source may write this bit to 1 to clear it. The Sink shall reset the bit to 0 when the Source writes a 1 value to this bit.
- CED_Update [1 bit] The Sink shall set (=1) this bit as described in Section 10.4.1.8. The Source may write this bit to 1 to clear it. The Sink shall reset the bit to 0 when the Source writes a 1 value to this bit. The Sink shall not change this bit when the Source reads a Character Error Detection Register.
- RR_Test [1 bit] The Sink shall set (=1) this bit when a test Read Request is generated in response to the setting (=1) of the TestReadRequest bit in the Test Configuration register defined in Section 10.4.1.9.

When the Source Device detects that an Update Flag is set (=1) for a feature that is currently active, the Source Device shall clear Update Flags which are set (=1) in a given register offset by writing back the value read from the same offset. It is recommended that the Source Device does this write operation before handling the function(s) for which the update flag was found to be set. This will allow detection of further updates that may occur during the management of the function(s). Note that it is possible a Sink Device may immediately set (=1) the bit again when a new update occurs.

When SCDC Read Request is enabled, and when any update flag transitions from 0 to 1, the Sink shall notify the Source with a Read Request (See Section 10.4.4).

10.4.1.3.1 Update Flag Polling

If the Sink Device does not set (=1) the RR_Capable bit in the HF-VSDB, or if the Source Device does not set (=1) the RR_Enable bit in the SCDCS, Source Devices use polling to implement features that require support of the SCDCS Update Flags.

Source Devices implementing Update Flag Polling should read the Update Flag Registers once every Video Field when video is active. More frequent reading is permitted. Source Devices implementing Update Flag Polling shall read the Update Flag Registers at least once per 250 ms when video is active.

10.4.1.4 TMDS Configuration

The positions of the Configuration fields are summarized in Table 10-19. All Configuration fields are readable and writable.

For each bit in the TMDS Configuration register, the Sink Device shall reset (=0) the bit:

- when the +5V Power Signal from the Source is not present, or
- when the Hot Plug Detect pin has voltage = low for 100 ms or more

Table 10-19: SCDCS – TMDS Configuration

Offset	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x20	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	TMDS_Bit_Clock_Ratio	Scrambling_Enable

- Scrambling_Enable [1 bit] The Source shall set (=1) this bit to enable scrambling in the Sink
  The Source shall reset (=0) this bit to disable scrambling in the Sink
  See Section 6.1.3.1.
- TMDS_Bit_Clock_Ratio [1 bit] 0 = (TMDS Bit Period)/(TMDS Clock Period) ratio is 1/10
  1 = (TMDS Bit Period)/(TMDS Clock Period) ratio is 1/40
  See Section 6.1.3.2.

10.4.1.5 Scrambler Status

The Status Flags are all Read Only.

For each bit in the Scrambler Status register, the Sink Device shall reset (=0) the bit:

- when the +5V Power Signal from the Source is not present, or
- when the Hot Plug Detect pin has voltage = low for 100 ms or more

Table 10-20: SCDCS - Status Flags

Offset	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x21	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Scrambling_Status

- Scrambling_Status [1 bit] This bit shall be set (=1) by the Sink when the Sink device detects scrambled control code sequences and reset (=0) when the sink does not detect scrambled control code sequences.
  See Section 6.1.3.1.

10.4.1.6 Configuration

The positions of the Configuration fields are summarized in Table 10-19. All Configuration fields are readable and writable.

For each bit in the Configuration register, the Sink Device shall reset (=0) the bit:

- when the +5V Power Signal from the Source is not present, or
- when the Hot Plug Detect pin has voltage = low for 100 ms or more

Table 10-21: SCDCS – Config_0

Offset	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x30	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	RR_Enable

- RR_Enable [1 bit] The Source shall set this bit (=1) when the Source supports Read Request.

  The Source shall reset this bit (=0) when the Source only supports polling of the update flags.

  The Sink shall reset (=0) this bit when the SCDC of the Sink goes from the disabled to the enabled state.

10.4.1.7 Status Flags

The Status Flags are all Read Only.

For each bit in the Status Flags registers, the Sink Device shall reset (=0) the bit:

- when the +5V Power Signal from the Source is not present, or
- when the Hot Plug Detect pin has voltage = low for 100 ms or more

Table 10-22: SCDCS - Status Flags

Offset	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x40	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Ch2_ Locked	Ch1_ Locked	Ch0_ Locked	Clock_ Detected
0x41	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)

- Clock_Detected [1 bit] This bit shall be set (=1) by the Sink when the Sink device detects a valid clock signal and reset (=0) by the Sink if this condition no longer exists.
- Ch0_Locked [1 bit] This bit shall be set (=1) by the Sink when the Sink device is successfully decoding data (as described in Section 6.2.2) on HDMI Channel 0 and reset (=0) by the Sink if this condition no longer exists.
- Ch1_Locked [1 bit] This bit shall be set (=1) by the Sink when the Sink device is successfully decoding data (as described in Section 6.2.2) on HDMI Channel 1 and reset (=0) by the Sink if this condition no longer exists.
- Ch2_Locked [1 bit] This bit shall be set (=1) by the Sink when the Sink device is successfully decoding data (as described in Section 6.2.2) on HDMI Channel 2 and reset (=0) by the Sink if this condition no longer exists.

10.4.1.8 Character Error Detection

The Character Error Detection counters are not writable by the Source and are cleared on read by the Source. The checksum is read only by the Source.

For each field in the Character Error Detection registers, the Sink Device shall reset (=0) the field when the +5V Power Signal from the Source is not present.

Sink Devices should NOT reset (=0) these fields if the Sink de-asserts the Hot Plug Detect pin (voltage = low).

Table 10-23: SCDCS - Character Error Detection

Offset	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x50	Channel 0 Error Count bits 7 -> 0							
0x51	Ch0_Valid	Channel 0 Error Count bits 14 -> 8						
0x52	Channel 1 Error Count bits 7 -> 0							
0x53	Ch1_Valid	Channel 1 Error Count bits 14 -> 8						
0x54	Channel 2 Error Count bits 7 -> 0							
0x55	Ch2_Valid	Channel 2 Error Count bits 14 -> 8						
0x56	Checksum of Character Error Detection							

See Section 6.2 for definition of Error Counter values and Valid flags.

The Checksum shall be implemented such that a one-byte sum of the 7 registers of Character Error Detection including the Checksum itself (offset 0x50 to 0x56) is equal to zero.

If RR_Enable is set (=1), the Sink shall issue a Read Request when it sets the CED_Update Flag (i.e. the CED_Update Flag transitions from a 0 to a 1). The Sink shall ensure that the CED_Update flag is set if any Error Counter value increments by more than 4 in one second, or if any Error Counter transitions to its maximum value. This indicates that the affected channel is not achieving the required character error rate.

10.4.1.9 Test Configuration

The Test Configuration registers are provided to facilitate compliance testing. For each field in the Test Configuration register, the Sink Device shall reset (=0) all fields:

- when the +5V Power Signal from the Source is not present, or
- when the Hot Plug Detect pin has voltage = low for 100 ms or more

Table 10-24: SCDCS - Test Read Request, Read/Write

Offset	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0xC0	TestRead Request	TestReadRequestDelay						

- TestReadRequest [1 bit] After setting the RR_Enable bit, Source Devices may set this bit to 1. Sink devices shall delay TestReadRequestDelay milliseconds before issuing a Read Request. After the Read Request has been issued, the Sink Device shall automatically clear (=0) the TestReadRequest field. In addition, immediately prior to issuing the Read Request, the Sink shall set (=1) the RR_Test Update Flag.
- TestReadRequestDelay [7 bits] Used in conjunction with the TestReadRequest field. This value should be written with the same DDC transaction that is used to set (=1) TestReadRequest.

10.4.1.10 Manufacturer Specific Registers

The final 48 offsets in the SCDC register space are allocated for manufacturer specific usages.

The Sink shall set the first three offsets in the Manufacturer Specific Registers (offsets 0xD0 .. 0xD2) to an IEEE Organizationally Unique Identifier (OUI) assigned to the manufacturer. The following 11 offsets are populated with the Device ID. The remaining offsets in the Manufacturer Specific Registers may be used for manufacturer defined applications.

The Manufacturer Specific Register offsets are described in more detail in Table 10-25.

Table 10-25: SCDCS – ManufacturerSpecific

Offset	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0xD0	Manufacturer_OUI_1							
0xD1	Manufacturer_OUI_2							
0xD2	Manufacturer_OUI_3							
0xD3-0xDA	Device ID: Device_ID_String							
0xDB	Device ID: Hardware_Major_Rev				Device ID: Hardware_Minor_Rev			
0xDC	Device ID: Software_Major_Rev							
0xDD	Device ID: Software_Minor_Rev							
0xDE-0xFF	ManufacturerSpecific							

- Manufacturer_OUI_1 [1 Byte] Read Only.
  Manufacturer's IEEE_OUI, Third Octet.
  Example: for IEEE OUI AB-CD-EF, this field is set to 0xEF
- Manufacturer_OUI_2 [1 Byte] Read Only.
  Manufacturer's IEEE_OUI, Second Octet.
  Example: for IEEE OUI AB-CD-EF, this field is set to 0xCD
- Manufacturer_OUI_3 [1 Byte] Read Only.
  Manufacturer's IEEE_OUI First Octet
  Example: for IEEE OUI AB-CD-EF, this field is set to 0xAB
- Device_ID_String [8 bytes] Read Only.
  Device Identification String. Identifies the Sink Device.
  Up to eight ASCII characters starting at offset 0xD3.
  If less than eight characters are used, the unused bytes shall be set to 0x00.
- Hardware_Major_Rev [4 bits] Read Only.
  Hardware major revision. Integer, typically incremented on a major silicon or board revision
- Hardware_Minor_Rev [4 bits] Read Only.
  Hardware minor revision. Integer, reset to 0 when major revision increments, typically incremented on a minor silicon revision (e.g. metal mask change) or minor board revision.
- Software_Major_Rev [1 byte] Read Only.
  Firmware/software major revision. Integer, typically incremented on new functionality.
- Software_Minor_Rev [1 byte] Read Only.
  Firmware/software minor revision. Integer, reset to 0 when firmware/software major revision increments, typically incremented on bug fixes.

- ManufacturerSpecific [34 Bytes] Read Only or Read/Writable as defined by the Manufacturer. Usage Defined by Manufacturer.

10.4.2 Timing

(‡) This section incorporates text from the HDMI Specification 1.4b Section 8.4.1. See Notice for copyright information.

H14b Section 8.4.1 describes the following timing requirements:

Data is synchronized with the SCL signal and timing shall comply with the Standard Mode of the I^2C specification (100 kHz maximum clock rate).

I^2C Bus is a standard two-wire (clock and data) serial data bus protocol. Refer to the I^2C specification for details.

An HDMI Sink may pause transactions by holding the SCL line Low during the SCL-low period following the Acknowledge bit as permitted by the I^2C specification. This is commonly referred to as clock stretching. All HDMI Source Devices shall delay the DDC transaction while the SCL line is being held low by the Sink Device.

10.4.3 Data Transfer Protocols

The 8-bit I^2C slave addresses[1] for the SCDC are 0xA8/0xA9. The LSb indicates the access type, 1 for read and 0 for write. A Sink which does not support SCDC shall not acknowledge any read or write to these addresses.

In order to minimize the length of the message when the Source reads the update flags (e.g. during a Read Request), Sink Devices shall support a fast read mode called "Update Read". In this mode, the 8-bit I^2C address 0xA9 is used without a repeated Start and reads commence from offset 0x10, corresponding of the UPDATE_0 sub-address. A complete "Update Read" is depicted in Figure 10-1.

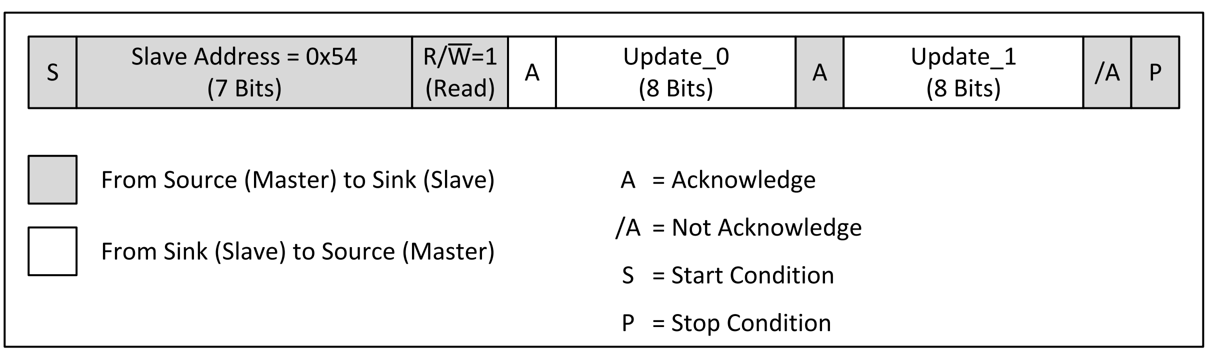


Figure 10-1: SCDC Update Read

[1] I^2C addresses are comprised of a 7-bit address followed by a single bit indicating a read or a write access (1 or 0 respectively). This Specification defines the 7-bit address as 0x54. When the R/W bit is added, the 8-bit address is 0xA9 for read accesses and 0xA8 for write accesses.

To read any other byte, the Source shall use a Combined Format read consisting of a one-byte write to indicate the offset followed by a repeated START condition and the read of the value(s). The timing of a two byte SCDC Combined Format Read is depicted in Figure 10-2. All HDMI Sinks that implement SCDC shall support multi-byte reads with auto-increment of the offset.

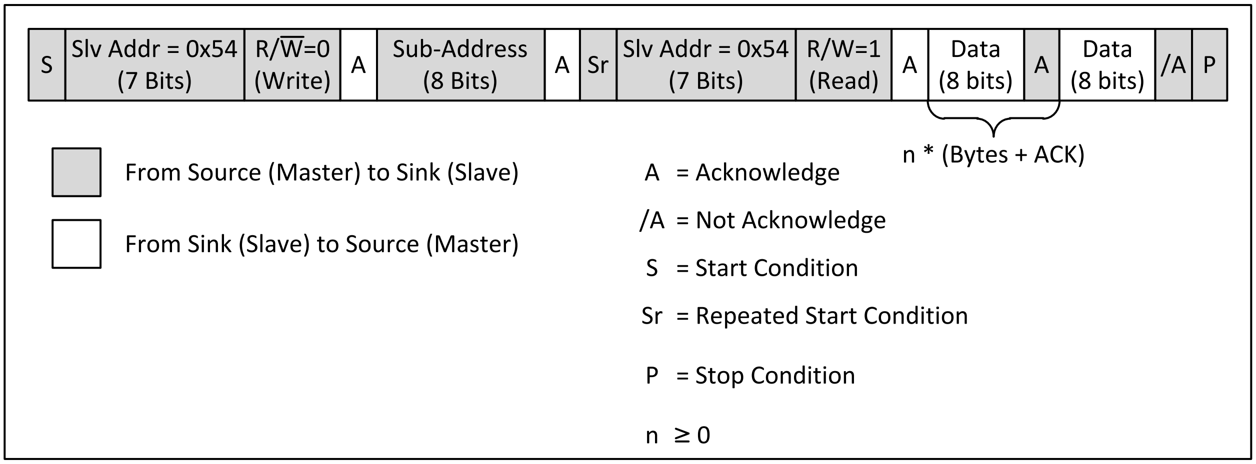


Figure 10-2: SCDC Combined Format Read

To write any byte, the Source shall transmit the offset followed by the value(s) to be written. The SCDC Write timing is depicted in Figure 10-3. All HDMI Sink Devices that implement SCDC shall support multi-byte write with auto-increment of the offset:

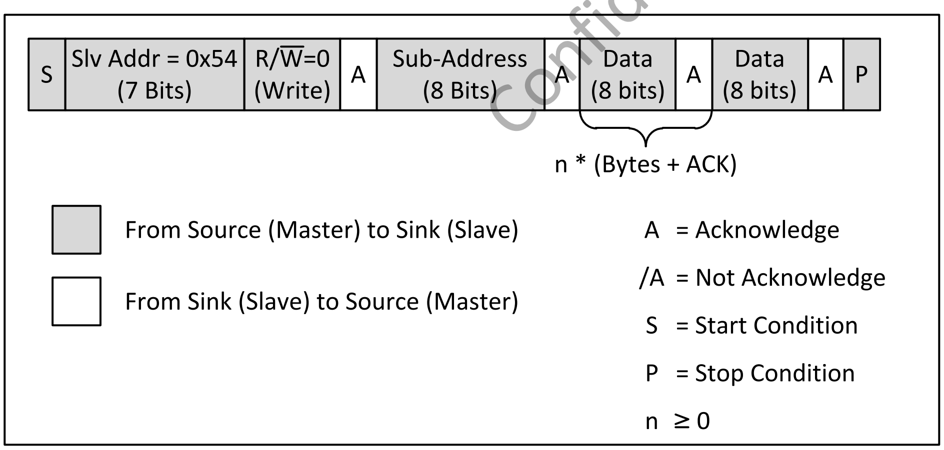


Figure 10-3: SCDC Write

10.4.4 SCDC Read Request

The SCDC Read Request feature provides a mechanism whereby the Sink device can notify the Source device that it is requesting the Source device to read the Update Flags. A Sink that supports the SCDC Read Request feature shall set (=1) the RR_Capable bit in the E-EDID HF-VSDB (see Section 10.3.2). Prior to enabling the SCDC Read Request feature, the Source shall first verify that the RR_Capable bit in the E-EDID HF-VSDB is set. If RR_Capable is set, then the Source may enable the feature by setting (=1) the RR_Enable bit in the SCDC registers (see Section 10.4.1.6). If HPD is de-asserted to a low voltage, the Sink will reset (=0) the RR_Enable bit. If HPD is then asserted to a high voltage level, the Source shall re-read E-EDID to determine if the read request feature is still supported before re-enabling Read Request feature in accordance with the above procedure.

When SCDC Read Request is enabled (as indicated by the state of the RR_Enable bit), and when any Update Flag (see Section 10.4.1.3) transitions from 0 to 1, the Sink shall notify the Source by generating a Read Request. The Read Request consists of the Sink device initiating a START condition by driving the SDA signal low any time when the I^2C bus is free (SCL and SDA are both high, and the bus has met the minimum bus free time (t_{BUF}) specified by the I^2C Specification). If the I^2C bus is busy, the Sink shall postpone generating the SCDC Read Request until the I^2C bus becomes free for the minimum bus free time (t_{BUF}).

When generating a Read Request on the bus and the SCL transitions to low, the Sink shall release the SDA signal within the maximum data valid acknowledge time ($t_{VD;ACK}$) as required by the I^2C Specification.

Regardless of whether a START condition on the bus is initiated by the Source or initiated by the Sink (i.e., by generating a Read Request), the Sink shall respond in accordance with the I^2C specification. The Sink must be capable of responding to an I^2C transaction immediately preceded by the START condition. If it is unable to respond immediately, the Sink is permitted to stretch the clock as specified in both the I^2C Specification and in H14b Section 8.4.1.

When the Source has enabled Read Request, and a Sink-initiated START condition occurs, one of four things shall occur:

1. The Source completes a valid I^2C transaction to read the SCDC Update Flags registers to discover which function or functions have new values. This sequence is depicted in Figure 10-4.

2. The Source generates a STOP condition. Specifically, the Source drives SDA low, then generates a valid LOW period on SCL (by driving SCL low and then releasing SCL after the minimum LOW period, t_{LOW}), then releases SDA. This sequence is depicted in Figure 10-5.

3. The Source completes a valid I^2C transaction that does not result in a read of the SCDC Update Flags registers. This sequence is also depicted in Figure 10-4. The sink can distinguish this case from case (1) by observing the offset of the register(s) which are read.

4. None of the above occurs within a time-out period of 1 ms. In this case the sink shall then initiate a STOP condition by releasing SDA as depicted in Figure 10-6.

If (1) or (2) occur, then the Sink should interpret this as a Read Request Acknowledge.

If (3) occurs, then the Sink should interpret this as a Read Request Not-Acknowledge. If the Read Request is not acknowledged the Sink shall retry the Read Request after the bus is free.

If (4) occurs, then the Sink should interpret this as a Read Request Not-Acknowledge. For the flag transition that generated the Read Request, the Sink shall retry Read Request after a minimum hold-off of 10 ms. The Sink may generate a new Read Request at any time without waiting 10 ms if a different flag transitions from 0 to 1.

When the Source has enabled the Read Request feature, and it responds to a Read Request by generating a STOP condition (case 2 above), which is interpreted as a Read Request Acknowledge, the Source shall subsequently read the SCDC Update Flags register(s) to discover which function(s) have new values.

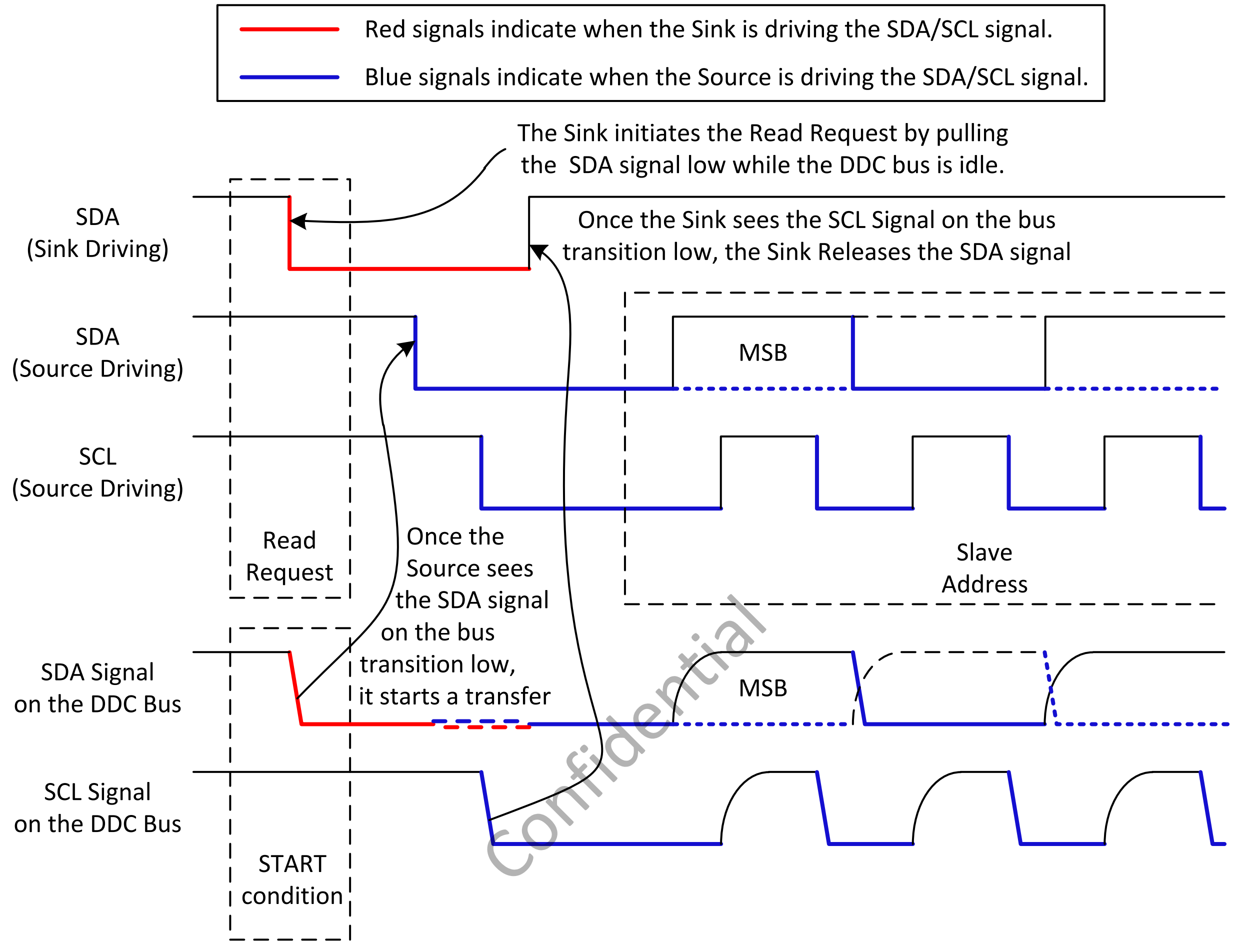


Figure 10-4: Read Request signal

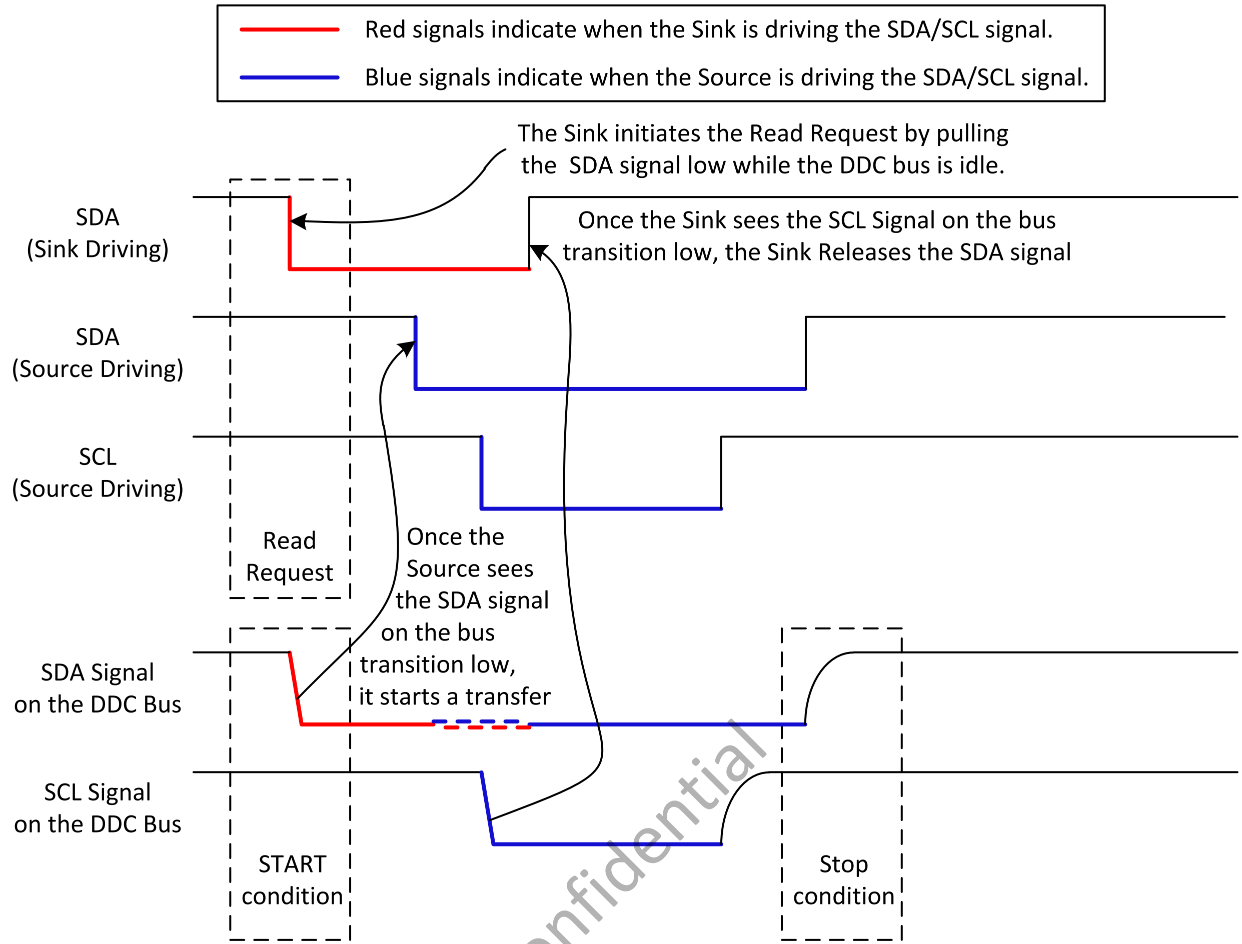


Figure 10-5: Read Request signal with STOP condition

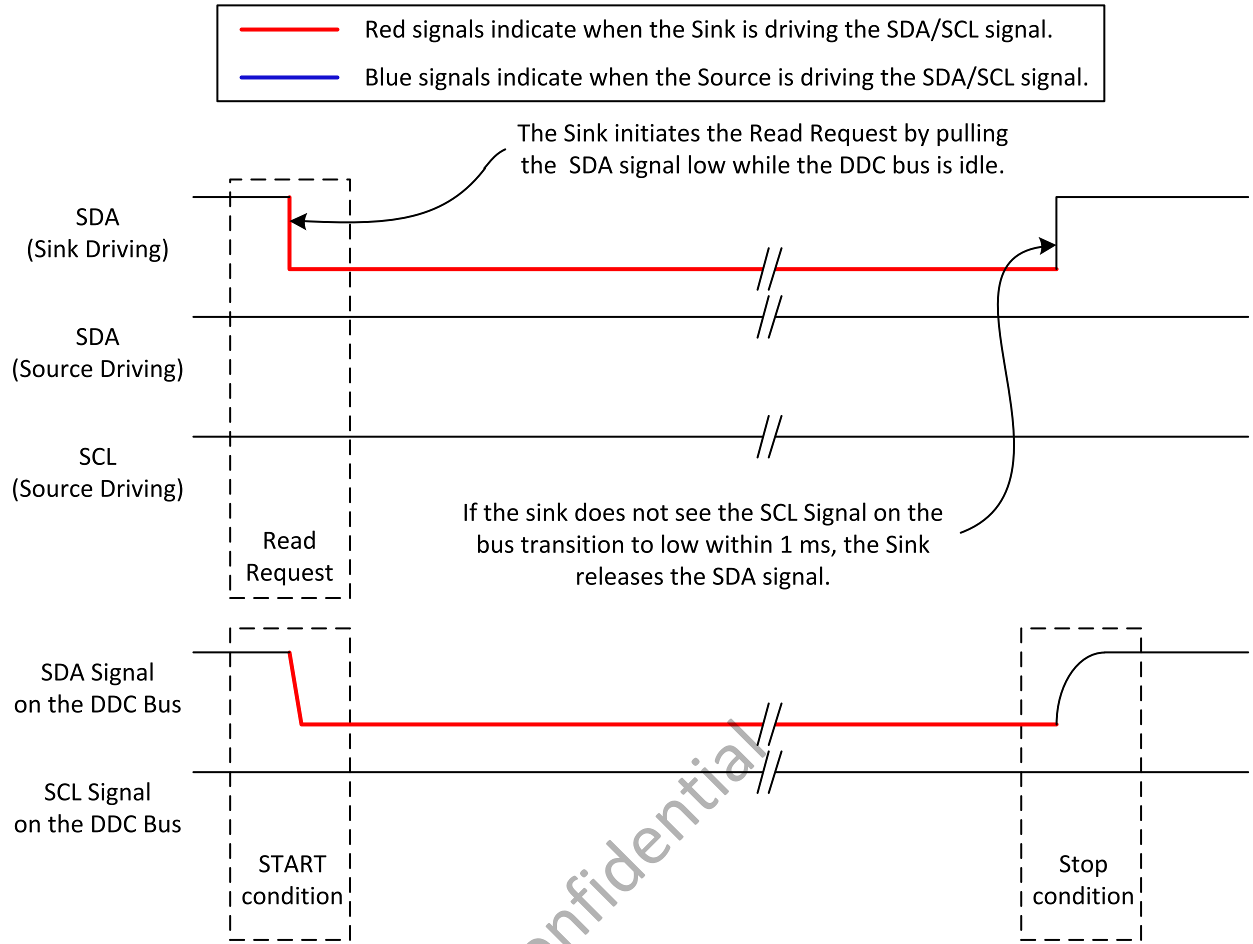


Figure 10-6: Read Request Timeout

The Sink may initiate a Read Request only if the RR_Enable bit is set (=1) (i.e. Read Request is enabled), and if the I^2C minimum bus free time has been satisfied.

Read Request remains enabled as long as the +5V Power signal is provided. If for any reason the +5 Power signal is removed, the Read Request feature shall be disabled by the Sink, the bit RR_Enable shall be cleared to 0, and the SDA line shall be released.

To facilitate compliance testing, a Sink that supports the Read Request feature shall support the SCDC Read Request test register (see Section 10.4.1.9) which contains two fields: TestReadRequestDelay (7 bits), and TestReadRequest (1 bit). When the value of TestReadRequest transitions from 0 to 1, the Sink shall generate a Read Request after a delay specified by the value of TestReadRequestDelay (specified in milliseconds).

10.4.5 SCDC Sink

The Sink shall enable the Read Request feature only after the RR_Enable bit is set (=1) by the Source. Read Request remains active as long as the +5V Power signal is provided. When the +5 Power signal is removed, the Read Request is disabled by the Sink, the bit RR_Enable is cleared to 0 and the SDA line is released.

10.4.6 SCDC Source

The Source shall provide continuous uninterrupted power to the Sink via the +5V power signal when the Read Request feature is enabled.

10.5 Relative Audio / Video Latency

This section extends H14b Section 7.5.

If a Source or Repeater is connected to a downstream device which does not indicate Video_Latency and Audio_Latency fields in the H14b VSDB (part of EDID, see H14b Section 8.3.2 and Section 10.6 of This Specification), or has equal values in those fields, it shall transmit audio and video data streams over HDMI with no more than ±2 ms of audio delay relative to the video.

If a Source or Repeater is connected to a downstream device which indicates Video_Latency and Audio_Latency fields in the H14b VSDB, and the Video_Latency is larger than the Audio_Latency, it shall either transmit audio and video data streams with no more than ±2 ms of audio delay relative to the video, or internally delay the audio by (Video_Latency-Audio_Latency) before transmitting over HDMI (thus compensating for the latency mismatch in the downstream device).

If a Source or Repeater is connected to a downstream device which indicates Video_Latency and Audio_Latency fields in the H14b VSDB, and Video_Latency is smaller than the Audio_Latency (a case which is strongly discouraged, see Table 10-26 in Section 10.6.1.1, case 3), it shall transmit audio and video data streams over HDMI with no more than ±2 ms of audio delay relative to the video.

When the downstream device's H14b VSDB includes Interlaced_Video_Latency and Interlaced_Audio_Latency fields, and interlaced content is being transmitted, the Source or Repeater shall apply the above rules using these Interlaced_Video/Audio_Latency field indications.

Note that for audio-only outputs, the rules and guidance in this section do not apply.

10.6 Auto Lipsync Correction Feature

(‡) This section and its subsections incorporate text from the HDMI Specification 1.4b Section 8.9 and some of its subsections. See Notice for copyright information.

Implementation of this Auto Lipsync Correction Feature is optional. If this feature is implemented, it shall be implemented according to the requirements specified in this Section (10.6) and its subsections.

Some common home theater device configurations render the audio in a device other than the TV. In these configurations, the video processing latency of the TV may cause perceptible lipsync issues to the user unless compensation is applied. These issues can be prevented by delaying the audio to compensate for the video processing latency. The HDMI Auto Lipsync Correction feature allows a Source or Repeater to automatically determine and apply the necessary amount of audio delay before presentation of that audio signal.

10.6.1 EDID Latency Info

HDMI Sinks and Repeaters shall declare accurate audio and video latency information in the EDID, allowing an upstream HDMI Source or Repeater to determine how best to maintain synchronization between the rendered audio and video. These fields and other lipsync-related fields are located in the H14b VSDB (see H14b Section 8.3.2). The latency values within these fields indicate the amount of time between the video or audio entering the HDMI input to the actual presentation to the user (on a display or speakers), whether that presentation is performed by the device itself or by another device downstream.

Note - for Sink Devices where the latency depends on the video processing mode, it is recommended that the latency values stored in the EDID reflect the latency values for the video processing mode which was active when the Sink started; also see Section 10.6.1.3.

The rules and guidance on filling the values differs slightly for Sink Devices without HDMI output (e.g. TV) and Devices with HDMI input and HDMI output (e.g. Repeater), as detailed in the following subsections.

10.6.1.1 Devices without HDMI output

For devices without HDMI output (e.g. TV), various cases of the relationship of VL and AL (the Video Latency and Audio Latency as indicated in H14b VSDB) are described in Table 10-26 and the following text. Also see the mechanism described in Section 10.7.3 (using flag Audio_Output_Compensated).

Table 10-26: Video Latency (VL) and Audio Latency (AL) in EDID of Device without HDMI output

case	relationship of VL and AL	Description
1	AL < VL	strongly discouraged
2	AL = VL	recommended case
3	VL < AL ≤ VL + 20 ms	strongly discouraged
4	VL + 20 ms < AL	forbidden

These devices should internally compensate for their own video processing latency by adding a delay to the audio stream that corresponds to the video latency. In this case, the EDID-indicated audio (AL) and video latencies (VL) will be equal as in Case 2 of Table 10-26. This delay should be applied for audio sent to internal speakers as well as for audio sent to ARC, external S/PDIF, or other audio outputs so that upstream amplifiers connected to such audio outputs will also be in-sync.

If the strong recommendation in the preceding paragraph is not followed, the EDID-indicated audio latency and video latency will not be identical (cases 1, 3 and 4 in Table 10-26).

In case 1 (AL < VL) , the device would be relying on the upstream device to compensate for the device's inability to do a proper lipsync operation. Since this compensation ability in the upstream device is not mandatory, lipsync for such devices cannot be guaranteed. Hence this case (declaring audio latency AL < video latency VL in the EDID) is strongly discouraged.

Case 3 (VL < AL ≤ VL + 20 ms) is strongly discouraged, since this would mean that an upstream device would have to delay the video (compared to the audio) which is cumbersome or impossible. This case would lead to a small lipsync issue: the audio will be rendered slightly too late (up to 20 ms compared to video).

Case 4 (AL > VL + 20 ms) is forbidden.

10.6.1.2 Devices with HDMI output

A Repeater shall calculate the latency fields for its upstream EDID to indicate the overall video and audio latency from the reception by the Repeater to the eventual rendering by the Repeater or by downstream device(s). The Repeater shall indicate a video latency in the upstream EDID equal to the video latency found in the downstream device's EDID plus the Repeater's own internal video processing latency, and shall indicate an audio latency in the upstream EDID equal to the audio latency found in the downstream device's EDID plus the Repeater's own internal audio processing latency.

If the Repeater is a video processor, the video data will be delayed by its internal processing before being passed downstream; for this case, the Repeater should include delay compensation (equal to the delay in the video path inside the Repeater) in its audio path. See Figure 10-7 for an example where the TV indicates a video latency of 80 ms in its EDID, and the video processor has an additional video latency of 40 ms. The TV includes an audio delay matching its own video latency (80 ms), in the audio path to its speakers. The video processor also includes an audio delay matching its own video path latency (40 ms), in the path to its HDMI output; consequently, video and audio are in sync on its HDMI input and on its HDMI output. In the upstream EDID, the video processor indicates the sum (120 ms) of the 80 ms downstream video/audio latency and its own 40 ms video/audio latency.

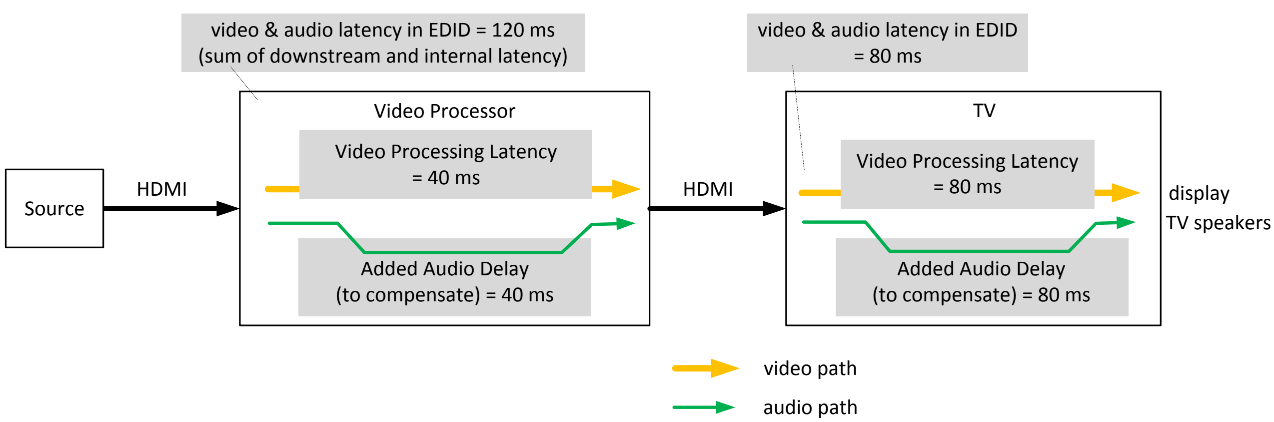


Figure 10-7: EDID Latency Handling for a Repeater with Video Processing

If the Repeater is an audio amplifier with no latency in its video and audio paths, which passes video through unmodified but which renders (amplifies) the audio directly, then the upstream audio latency will be equal to the Repeater's audio processing latency (including the audio delay). In this case, the amplifier typically adds an audio delay sufficient to compensate for the video latency of the downstream device, so that the upstream audio and video latencies will be equal, whether that Repeater forwards the audio downstream or renders the audio directly. See Figure 10-8 for an example where the TV indicates a video latency of 80 ms in its EDID, and includes a matching audio delay (80 ms) in its audio path to its speakers. This video latency indicated in the EDID is used by the Amplifier to delay the audio towards its speakers for compensation; the Amplifier does not add this delay in its audio path from its HDMI input to its HDMI output. This accomplishes lipsync between video (from the TV) and audio (from the Amplifier's speakers). In the Amplifier's EDID for the upstream Source, it replicates the downstream video latency (80 ms).

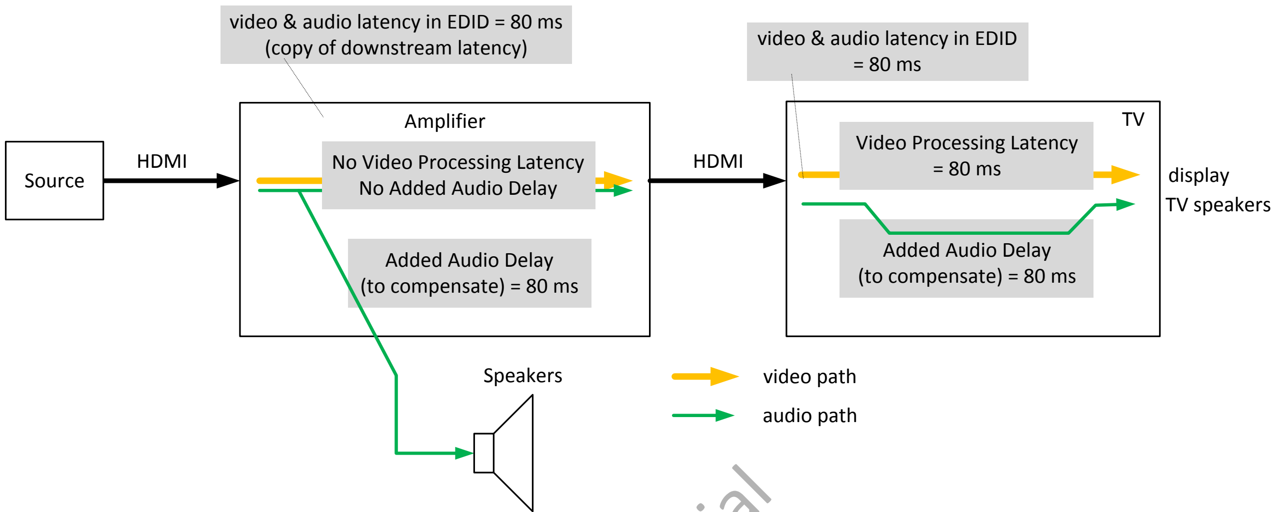


Figure 10-8: EDID Latency Handling for an Amplifier

A Repeater may also combine the video processor in Figure 10-7 and the Amplifier in Figure 10-8, as illustrated in Figure 10-9. Such a device will delay the video due to its internal processing (40 ms in this example) before passing it downstream, and includes a 40 ms audio delay compensation on the audio path to its HDMI output equal to its own video processing latency.

On the audio path to its speakers, the device applies audio delay compensation equal to the total delay in the video path (120 ms), i.e. the sum of its own video latency (40 ms) and the downstream video latency indicated by the TV in its EDID (80 ms). In the upstream EDID, the device indicates the sum (120 ms) of the downstream video/audio latency (80 ms) and its own video/audio latency (40 ms).

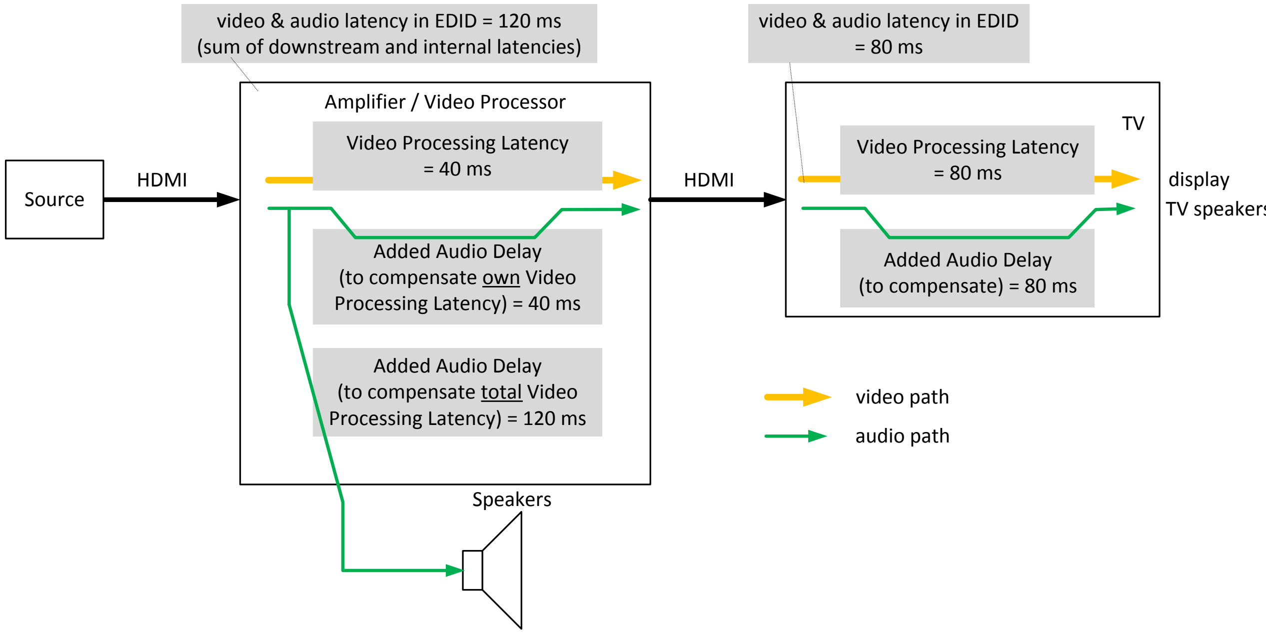


Figure 10-9: EDID Latency Handling for an Amplifier with Video Processing

10.6.1.3 Supporting a Range of Latency Values

If the video latency of a device differs significantly depending upon the video format (e.g. when processing SD versus HD formats) or other factors (e.g. multiple processing modes including a low-latency mode for games or other applications), it is recommended that the video latency field indicates a latency that is between the extremes but skewed toward the longer latency. An audio/video mismatch is more perceptible if the audio leads the video than if the video leads the audio by a similar amount (see Figure 2 in ITU-R BT.1359-1). Because of this effect, indicating a value that is closer to the maximum video delay may result in better overall user experience. For example, a value of roughly (2 * max_latency + min_latency)/3 may be used. The same is true for the audio latency but in this case, the indicated value should be skewed towards the minimum latency. For example, a value of roughly (max_latency + 2*min_latency)/3 may be used.

If the optimum indication for the video latency for interlaced video formats is significantly different than the optimum indication of latency for progressive formats, then the I_Latency_Fields_Present flag should be set, allowing the EDID to indicate separate latencies for these two categories of video formats. This approach may be used anytime but it is recommended in case the difference between the two latencies is more than 30 msecs.

See Section 10.7 for the Dynamic Auto Lipsync feature, which provides a more effective way of communicating the actual video latency.

10.6.2 Compensation

If a Source or Repeater is connected to a downstream device which has Video_Latency and Audio_Latency fields in the H14b VSDB with Video_Latency > Audio_Latency, it should delay the audio towards the HDMI output to compensate for the video latency of the downstream device(s), by an amount equal to the video_latency minus the audio_latency of the downstream (or rendering) devices. If these latency values are identical, no compensation shall be applied. If Video_Latency is smaller than Audio_Latency, the above formula (Video_Latency minus Audio_Latency) will yield a negative value, and compensation by delaying the audio is not possible (See Table 10-26 in Section 10.6.1.1, cases 3 and 4).

It may not be possible to determine the audio latency of non-HDMI audio outputs (e.g. S/PDIF or analog outputs). For uncompressed audio formats, typically the value will be close to zero and so the device can simply delay the audio by the amount of video latency in the downstream EDID. For compressed audio formats, the device may assume that the audio latency is near the standard decompression latency specified in the relevant IEC 61937-x standard or in the codec vendor's documentation, and subtract this decompression latency from the audio delay determined in the previous paragraph.

It is expected that an audio delay capability of 100 ms will support full compensation for almost all of the TV and video processor products on the market today. Note that at the time of writing of This Specification, latencies observed in the market indicate that 100 ms delay compensation might not be sufficient, and that a longer delay compensation (e.g. 150-200 ms) might be needed.

If transmitting a progressive video format, the Video_Latency and Audio_Latency fields shall be used for the calculations in this section. If transmitting an interlaced video format and if I_Latency_Fields_Present == 0, the same fields shall be used; if I_Latency_Fields_Present == 1, then the Interlaced_Video_Latency and Interlaced_Audio_Latency fields shall be used.

10.7 Supporting Dynamic Latency Changes: Dynamic Auto Lipsync

The basic Auto Lipsync feature described in Section 10.6 (using values in EDID) does not provide an easy way for Sinks that change their latency, to inform upstream devices of the correct latency. Therefore, it is extended with the Dynamic Auto Lipsync (DALS) feature which defines a mechanism for Sinks to dynamically modify and announce their latency information. It allows a Sink to indicate its current video latency, especially if it differs from the video latency information in the EDID, so that the connected Source or Repeater is aware of the correct video latency.

This dynamic latency could, for example, be caused by the user changing the video processing mode; a change of video resolution leading to a change in video latency; or a Source starting or stopping to indicate Content Type="game".

To ensure correct lipsync for the user, particularly in a configuration of devices with variable video latency, this feature is valuable and should be implemented in the following device categories:

- Sink Devices should implement Dynamic Auto Lipsync if the device has a certain video latency, "A", in one mode of operation and at least one other different video latency, "B", in another mode of operation and a mode switch between the two modes is possible during normal operation of the device. Sink Devices with only one video latency should also implement Dynamic Auto Lipsync, if they prefer to communicate the video latency accurately and unambiguously.
- All Amplifiers should implement Dynamic Auto Lipsync in order to match their audio delay (internal, on the path towards their speakers only) to the actual video latency of the downstream device(s).
- All Repeaters should implement Dynamic Auto Lipsync in order to propagate the actual video latency information from the downstream device(s) to the upstream device(s) – including their own latency information.
- In some cases, the Dynamic Auto Lipsync feature is also relevant for Source Devices, see Appendix D.

The roles of Amplifier and Repeater are not mutually exclusive: an Amplifier can be either a Repeater (if it has one or more HDMI inputs) or a Source (if it has no HDMI inputs). Hence the combination Amplifier/Repeater (i.e. Amplifier with HDMI input(s) has to fulfill both the requirements for an Amplifier as well as the requirements for a Repeater.

All devices implementing the Dynamic Auto Lipsync feature, shall also implement the Auto Lipsync Correction feature as defined in Section 10.6 and its subsections.

When Dynamic Auto Lipsync is implemented, a Sink shall use it to provide updates whenever the video latency of the Sink changes, due to Video Format changes (e.g. when processing SD versus HD formats), processing mode changes (e.g. entering or leaving a low-latency mode for game applications – based on user choice, Content Type indication or otherwise) or for any other reason. If a device has used this mechanism, it shall also use the mechanism when it returns to a state where the video latency is same as what is indicated in EDID, to make sure the connected Source or Repeater is aware of the change.

Figure 10-10 depicts an example of basic Dynamic Auto Lipsync operation: initially, the TV has its 'normal' latency of 80 ms. Then, the user switches the TV to low latency mode. The TV communicates the updated video latency value via CEC messages to the Amplifier. The Amplifier uses the updated video latency, and communicates this via CEC messages to upstream device(s).

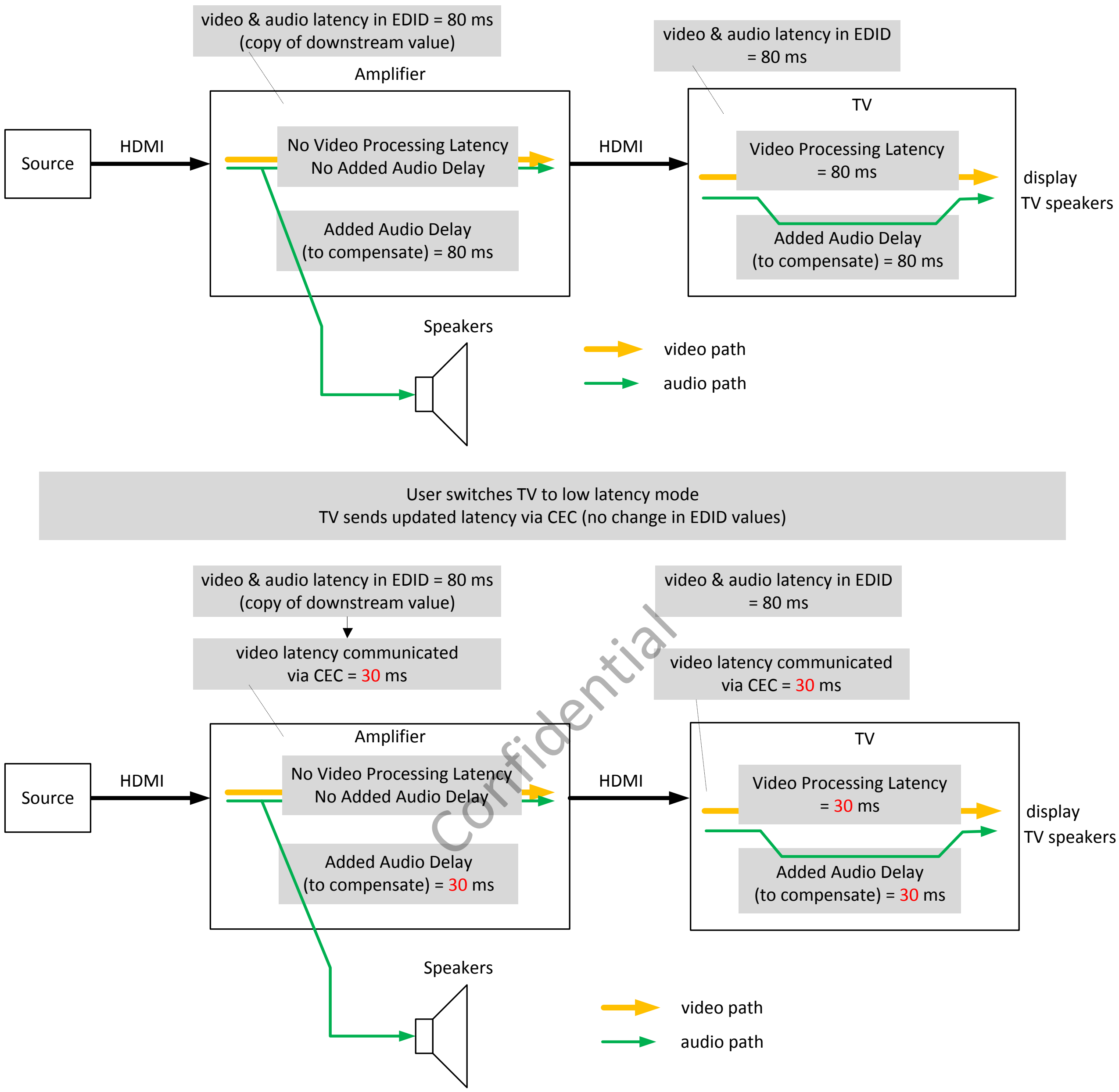


Figure 10-10: Dynamic Auto Lipsync Example of Operation

10.7.1 CEC transport for Dynamic Auto Lipsync

All devices that support Dynamic Auto Lipsync shall implement CEC messaging as described in this section. The CEC-based transport for Dynamic Auto Lipsync uses two messages: <Report Current Latency> and <Request Current Latency>. For details of the messages and the operands see Table 10-27 and Table 10-28. For message dependencies, see Section 11.11.

Since this feature relates to the topology of the cluster (see H14b Figure 8-10), CEC broadcast messages including the [Physical Address] parameter are used. This allows all devices to use the feature without having to do a bus scan to determine the mapping of Logical Addresses to Physical Addresses in the cluster.

The CEC transport used by Dynamic Auto Lipsync is not coupled to a specific version of CEC. This means a device with Dynamic Auto Lipsync can be one of three following variants:

- device which implements CEC 2.0 as defined in Section 11 of This Specification
- device which implements CEC as defined in H14b Supplement 1
- device which does not implement either of those CEC versions

In the first two cases, the device shall use its own Logical Address as Initiator for the CEC messages sent for the Dynamic Auto Lipsync feature.

In the third case, i.e. a device implementing Dynamic Auto Lipsync which does not implement CEC 2.0 or CEC 1.4b, the device

- shall implement H14b Sections CEC 4 through CEC 10 related to low level CEC characteristics and mechanisms, and
- shall implement H14b Section CEC 12, and
- shall implement Sections 11.9.1 through 11.9.5 in This Specification, and
- shall use Logical Address 15 ('Unregistered') as Initiator when sending CEC messages, and
- shall not send any other CEC messages apart from those listed in Table 10-27. This implies the device shall not initiate or respond to polling messages, and shall send neither <Report Physical Address> nor <Report Features>

Table 10-27: Message Descriptions for the Dynamic Auto Lipsync feature

Opcode	value	Description	Parameters	Parameter description	Response	Addressing		Mandatory for	
						Direct	Broadcast	Initiator	Follower
<Request Current Latency>	0xA7	Used by Amplifier (or other device) to request current latency values	[Physical Address]		The device at target Physical Address sends <Report Current Latency> with current values		•	Amplifier Repeater	TV Repeater
<Report Current Latency>	0xA8	Used by TV (or other Initiator) to report updates of latency	[Physical Address] [Video Latency] [Latency Flags] [Audio Output Delay]*	Current video latency and related flags	The Amplifier (or other device) uses the reported values		•	TV Repeater	Amplifier Repeater
* Operand [Audio Output Delay] is only present when [Audio Output Compensated] (part of [Latency Flags])=3									

Table 10-28: Operand Descriptions for Dynamic LipSync

Name	Range Description		Length	Purpose
[Audio Output Delay]	1..251; indicates the amount of audio delay in TV towards audio output (e.g. ARC, SPDIF) currently valid; coded same as latency value defined in EDID (see H14b Section 8.3.2): Value is (number of milliseconds / 2) + 1 with a minimum allowed value of 1 (indicating 0 ms) and a maximum allowed value of 251 (indicating 500 ms). Values 0 and 252..255 are reserved and shall not be used.		1 byte	Delay of audio in TV towards audio output (e.g. ARC, SPDIF). Only present and used in conjunction with [Audio Output Compensated]=3
[Latency Flags]	reserved	Bit 7..3	5 bits	Flags for Latency
	[Low Latency Mode]	Bit 2	1 bit	
	[Audio Output Compensated]	Bits 1-0	2 bits	
[Low Latency Mode]	0=normal latency mode 1=low latency mode		1 bit	Flag to indicate if device is in a low latency mode
[Audio Output Compensated]	0=N/A (not applicable) – used when sent by non-TV 1=TV's audio output is delay compensated 2=TV's audio output is NOT delay compensated 3=TV's audio output is partially delayed		2 bits	Flag to indicate if TV's audio output is delay compensated
[Video Latency]	1..251; indicates the amount of video latency currently valid; coded same as latency value defined in EDID (see H14b Section 8.3.2): Value is (number of milliseconds / 2) + 1 with a minimum allowed value of 1 (indicating 0 ms) and a maximum allowed value of 251 (indicating 500 ms). Values 0 and 252..255 are reserved and shall not be used.		1 byte	Current video latency

The Initiator shall set [Audio Output Compensated] and [Audio Output Delay] as described in Section 10.7.3.

The Initiator shall set [Video Latency] to indicate the current video latency from the Initiator's HDMI input to the display. Note: the EDID mechanism for reporting video latency has separate fields for progressive and interlaced Video Formats. For the Dynamic Auto Lipsync mechanism, such a distinction is not needed, since [Video Latency] always refers to the current latency for the current Video Format. In turn, this means that changing from progressive to interlaced format or vice versa can result in an update of the reported value.

The Initiator shall set [Low Latency Mode] to 1 when in "low latency" mode (e.g. "game mode") and set it to 0 when in normal latency mode. This low latency mode can be selected by Content Type or user choice or by other means. The main purpose of the [Low Latency Mode] flag is the system configuration where the Source Device and Amplifier are connected on different HDMI inputs of the TV as depicted in Figure 10-11 below. If the Source requests the TV to go to low latency mode (e.g. via Content Type = "game") or the user switches the TV to low latency mode, the TV shall send the TV's new video latency value to the Amplifier. Some Amplifiers may adapt processing further if they are aware that the TV's low latency mode has been activated. They may use the [Low Latency Mode] flag for this purpose.

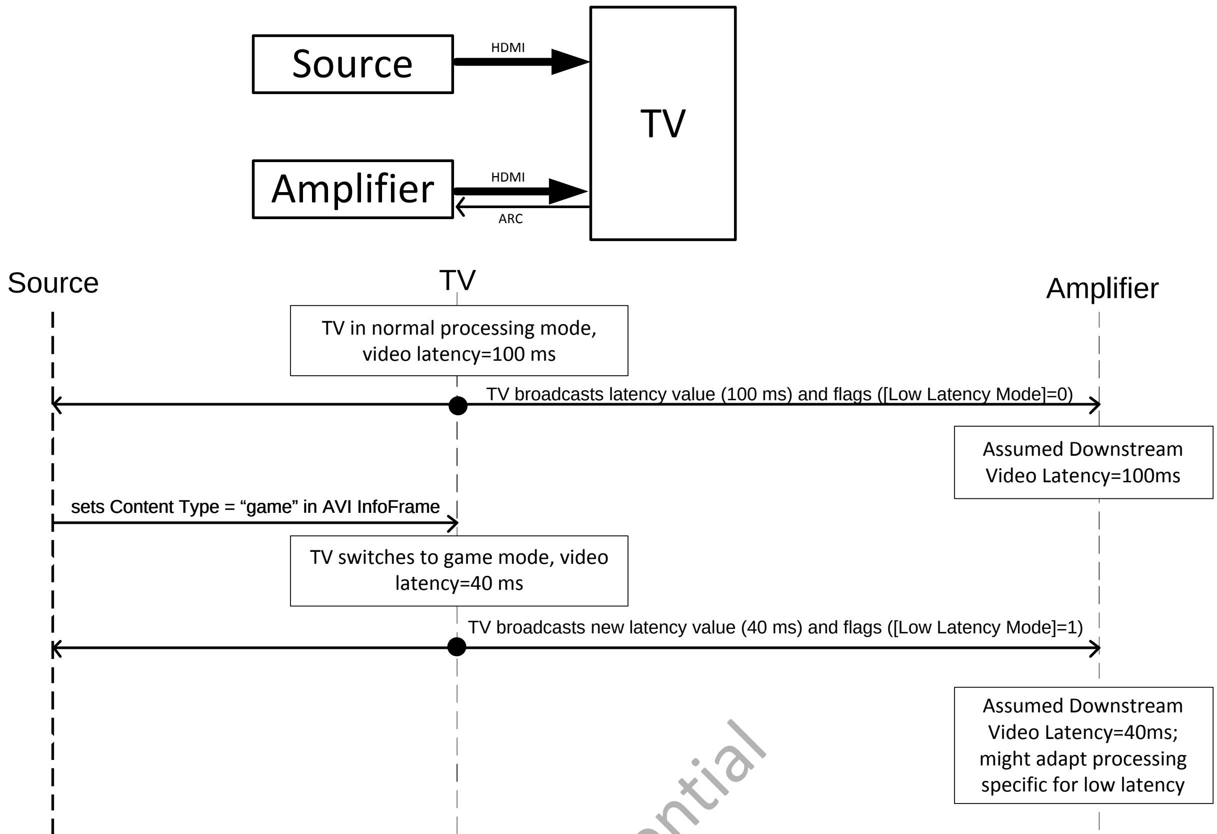


Figure 10-11: Example of the use of [Low Latency Mode] flag

10.7.2 Dynamic Auto Lipsync operation

Whenever a TV or Repeater implementing Dynamic Auto Lipsync starts up, and whenever this Device's processing mode changes such that it leads to a change in one or more of the entities reported in the operands related to DALS (See Table 10-28), the Device shall communicate the latency value(s) and associated flags using the (broadcast) message <Report Current Latency> including the Physical Address of the Initiator. See example in Figure 10-12.

When transitioning from one Video Format to another, or switching from one input to another, a device may go through one or more intermediate states. In this case it shall only send a <Report Current Latency> message once it has reached a stable configuration and has started rendering video. It shall not send a <Report Current Latency> for any intermediate states.

A device shall not send a <Report Current Latency> message if the latency (and flags) have not changed compared to the latency (and flags) in its previous <Report Current Latency> message – unless in response to a <Request Current Latency> message targeted to the device (see below).

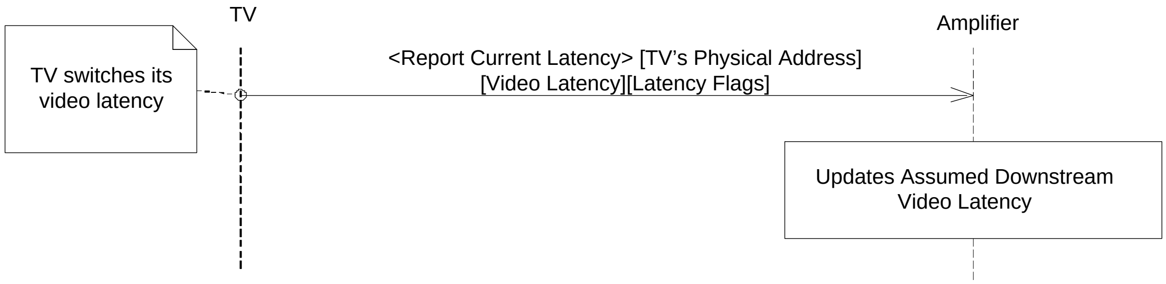


Figure 10-12: TV reports updated latency value(s) and flags when changing latency

When the Amplifier (or another device) starts rendering audio, it may broadcast the <Request Current Latency> message including the [Physical Address] of the targeted device to discover the actual latency value (and/or other information) of another device. The device at this targeted [Physical Address] shall respond with a (broadcast) <Report Current Latency> message with the current values, see example in Figure 10-13.

In other situations, there is no need to send <Request Current Latency> messages, and a device shall not send <Request Current Latency> messages more often than once every minute.

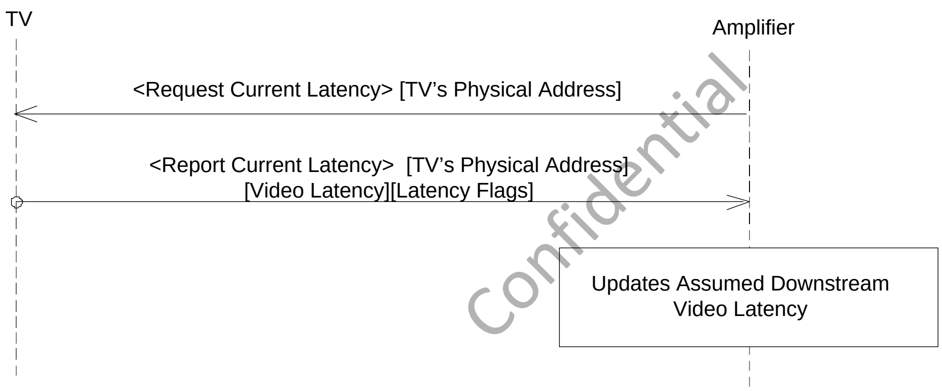


Figure 10-13: TV reports current latency value(s) and flags upon request

The latency values and flags shall be set as detailed in Section 10.7.1.

An Amplifier[1] supporting Dynamic Auto Lipsync which receives updated latency value(s) or flags from a downstream device shall use the [Video Latency] value as the Assumed Downstream Video Latency; this value is used to delay the audio in the Amplifier towards non-HDMI outputs (e.g. the Amplifier's speakers) to compensate for the video latency of the downstream device(s), (see Section 10.6.1.2). It shall not be used to delay the audio towards the HDMI output (see Section 10.5).

A Repeater[2] supporting Dynamic Auto Lipsync which receives updated latency value(s) or flags from a downstream device, or which changes its internal video/audio latency, shall calculate the video latency for its upstream devices based on the video latency reported by the downstream device plus its internal video latency, and shall broadcast this latency and the actual latency flags in the message <Report Current Latency> including its [Physical Address].

[1] The roles of Amplifier and Repeater are not mutually exclusive. See Section 10.7.
[2] The roles of Amplifier and Repeater are not mutually exclusive. See Section 10.7.

When an Amplifier or Repeater supporting Dynamic Auto Lipsync starts up, it shall acquire the latency values and flags of the downstream device(s) by broadcasting a <Request Current Latency> message, and shall use the latency value(s) and flags reported by the downstream device(s) to calculate the total latency value as described in the previous two paragraphs. If the device is a Repeater, it shall then broadcast the total video latency value along with the appropriate flags in a <Report Current Latency> message. This is to ensure that Amplifiers and Repeaters do not miss any TV latency changes that might have happened while they were in standby.

Figure 10-14 shows an example of the message flow with three devices (Amplifier and TV connected via an intermediate Repeater). When the Amplifier does not receive the last message depicted in this Figure (see Section 11.9.5 and H14b Section CEC 9.2 for response time and timeout), it shall assume the intermediate Repeater does not support the Dynamic Auto Lipsync feature, and can use the TV's reported latency values; this means that the (unknown) latency of the intermediate device is ignored, but this latency is expected to be small compared to the latency of the TV, so using the TV's reported values is likely to give a better result than ignoring the values received via <Report Current Latency>.

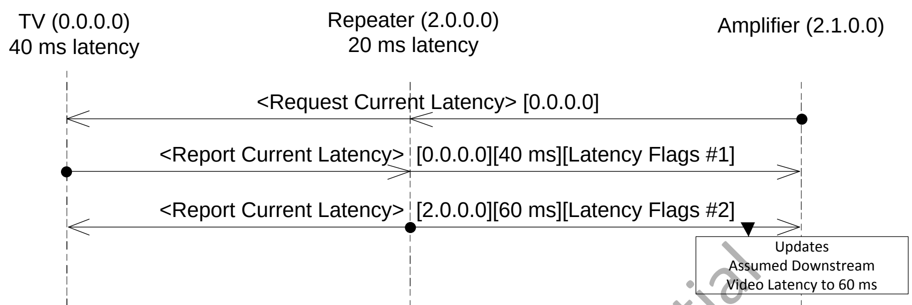


Figure 10-14: Three-device scenario, initiation by Amplifier's request

10.7.3 Latency of TV's Audio Outputs

TVs typically provide an audio signal to the Amplifier via an ARC connection, SPDIF or another output. TVs should apply delay compensation on this signal (i.e. audio outputs are in sync with video on the screen; the recommended case 2 described in Section 10.6.1.1), however, some don't, so the Amplifier does not know if it needs to apply further delay to such a signal. Consequently, the flag [Audio Output Compensated] is provided (see Section 10.7.1) to indicate whether the TV has applied delay to its audio outputs or not. It is assumed that the TV's delay compensation towards the various audio-only outputs (ARC, SPDIF, etc.) is identical.

In the case where the TV compensates the audio internally by the same amount as the internal video latency (recommended case 2 described in Section 10.6.1.1), it shall set the flag [Audio Output Compensated]=1, and the Amplifier shall not apply further delay to the audio signal from TV (irrespective of the reported latency) – see Figure 10-15 (left).

In the case where the TV does not compensate the audio internally, it shall set the flag [Audio Output Compensated]=2, and the Amplifier shall delay the audio signal from TV by the Assumed Downstream Video Latency – see Figure 10-15 (right).

In the case where the TV has an audio delay towards its audio outputs, which is neither 0 nor identical to the internal video latency, it shall set the flag [Audio Output Compensated]=3, and shall indicate the amount of delay that the TV has applied in the operand [Audio Output Delay]. In this case the Amplifier shall delay the audio signal from TV by the Assumed Downstream Video Latency minus the value indicated in [Audio Output Delay].

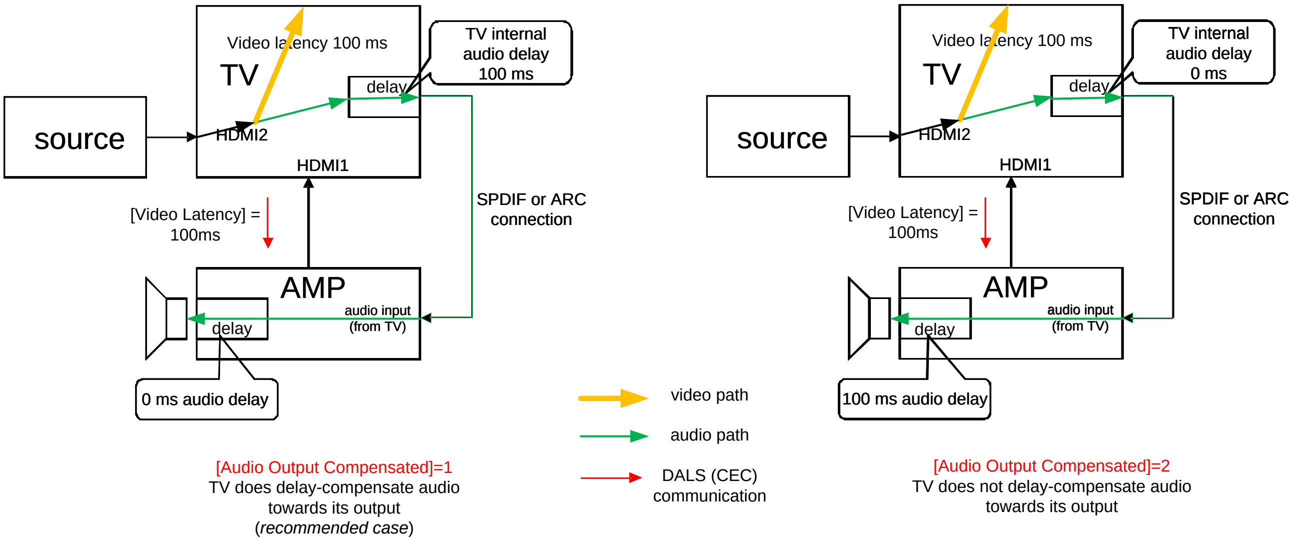


Figure 10-15: Operation of audio delay when [Audio Output Compensated]=1 versus 2

The flag [Audio Output Compensated] shall only be used by the Amplifier when it is rendering audio from the TV. For the case where the audio originates from the Amplifier internally, or from one of the Amplifier's (HDMI or other) inputs, the Amplifier shall delay the audio signal towards its speakers by the Assumed Downstream Video Latency, and ignore the value of [Audio Output Compensated] (and, if present, [Audio Output Delay]).

The operand [Audio Output Delay] shall only be sent by the TV and shall only be used by the Amplifier when [Audio Output Compensated]=3 and the Amplifier is rendering audio from the TV.

10.8 Handling of Hot Plug Detect (HPD) and +5V Power signals

The CEC's Logical Address Allocation (see Section 11.3.3 and H14b Section CEC 10) depends on a device having successfully discovered its Physical Address as read from the downstream EDID when HPD is high. Hence the Physical Address Discovery process (Section 10.9 and H14b Section 8.7.2) for a device with an HDMI output cannot complete until HPD is high on the inputs of the root device as well as any intermediate devices.

In order to allow optimal performance of a CEC system, each device needs to know its Physical Address to be able to allocate a Logical Address and announce its presence to other devices.

Therefore, CEC 2.0 Sink and Repeater devices shall assert the HPD signal (i.e. keep it at a high level), unless forbidden by other requirements, e.g.

- The need to de-assert the HPD line for at least 100 ms (H14b Section 8.5) to perform updates to the EDID.
- HPD low going pulse (at least 100 ms) for HDCP purposes.

The HPD signal depends on the incoming +5V Power signal (H14b Section 8.5). Namely, "The Hot Plug Detect pin may be asserted only when the +5V Power line from the Source is detected").

Therefore, CEC 2.0 Source and Repeater devices shall keep the 5V signal at high level.

The requirements in Section 10.8 of This Specification are not mandatory for devices which are unplugged from an AC power outlet and do not have battery power.

10.9 Discovery Algorithm

(‡) This section incorporates text from the HDMI Specification 1.4b Section 8.7.3. See Notice for copyright information.

The text in H14b Section 8.7.3 is extended as follows:

The following algorithm is used to allocate the physical address of each device whenever HPD from the Sink is de-asserted (i.e. at a low level) and upon power-up:

Sink de-asserts HPD (i.e. bring HPD to a low level) to all Source Devices.

```
If I am CEC root
    Set my_address to 0.0.0.0
Else
    Wait for a high level on HPD from Sink
    Query Sink for my_address of my connection (H14b Section 8.7.4)
    The device shall retain this physical address until HPD is removed (or the device is powered off).
End if
If device has connections for Source Devices then
    Label all possible connections to Source Devices uniquely starting from connection_label = 1 to
    the number of Source input connections
    If device has separate EDIDs for each Source connection then
        If my_address ends with 0 then
            Set each source_physical_address to my_address with the first 0 being replaced
            with connection_label.
        Else (i.e. beyond the fifth layer of the tree)
            Set each source_physical_address to F.F.F.F
        End if
    Else (note this case is deprecated, see H14b Section CEC 11 and CEC Appendix A)
        Set each source_physical_address to my_address
    End if
    Write source_physical_address to H14b VSDB in EDID for each Source connection
End if
```

Allow HPD on the Sink's HDMI inputs to be asserted (i.e. set to high level) when connected to HDMI Source Devices that have provided a high level on the +5V Power line (See H14b Section 8.5).

11 CEC 2.0, Consumer Electronic Control

(‡) This section and its subsections incorporate text from the HDMI Specification 1.4b Supplement 1. See Notice for copyright information.

11.1 Introduction

This section describes the CEC Version 2.0 specification. This CEC Version 2.0 specification incorporates the Version 1.4b specification by reference only; this CEC Version 2.0 specification only shows the extensions to the CEC Version 1.4b specification. Where sections exist in the CEC Version 1.4b specification but no corresponding sections exist in the CEC Version 2.0 specification, implementers are directed to the CEC Version 1.4b specification for details. In these cases, there is no extension to the text in the CEC Version 1.4b specification.

11.1.1 Relationship and compatibility with earlier version

This CEC 2.0 specification and subsequent future CEC specifications do not automatically supersede the CEC 1.4b specification. It is recommended that an Adopter implements the latest CEC specification to improve interoperability between new devices and older devices.

A CEC 2.0 device shall include a mechanism (e.g. through a user menu or other suitable user-controlled action) to switch the device between a state in which no modified messages are sent and a state in which CEC 2.0 messages are sent.

An Adopter can choose to implement CEC according to the CEC 1.4b specification, or to this CEC 2.0 specification (but not a mixture between the two specifications):

- If implemented according to the CEC 1.4b specification, the product shall pass compliance testing as defined in CTS 1.4b.
- If implemented according to the CEC 2.0 specification, the product shall pass compliance testing as defined in CTS 1.4b, as well as additional compliance testing defined in CTS 2.0.

CEC, as defined in This Specification, builds upon CEC as defined in the HDMI/CEC Specification, version 1.4b and extends it to include expanded sets of mandatory features to promote wider interoperability between all compliant devices. Clarifications and enhancements defined in CEC 2.0 are designed in such a way as to be backward compatible with devices implementing earlier versions of CEC.

Since CEC 2.0 was designed to be a backward compatible extension of CEC 1.4b, a device compliant with This Specification (CEC 2.0) will be backward compatible with existing compliant CEC 1.x devices, and the message behavior described in this CEC 2.0 specification will be compatible with the behavior of a device compliant with the CEC 1.4b specification.

Extending while maintaining backward compatibility is based on a number of design requirements for CEC devices:

1. The CEC 1.4b specification expects that when a device receives an unknown or known-but-not-implemented opcode, the CEC 1.4b device will ignore it if it was a broadcast message or respond with a <Feature Abort> if it was a directly addressed message. This allows future specifications (including this one) to allocate and use new opcodes for new features.

2. The CEC 1.4b specification expects a device to ignore received broadcast messages that CEC 1.4b specification only defines as directly addressed. This allows future specifications (including this one) to add semantics to "broadcast" usage of a previously "directly addressed" message.

3. The CEC 1.4b specification expects a device to <Feature Abort> (or ignore) a directly addressed message that the CEC 1.4b specification only defines as broadcast. This allows future specifications (including this one) to add semantics to "directly addressed" usage of a previously "broadcast" message.

4. The CEC 1.4b specification expects a device to ignore operand values that are reserved in the CEC 1.4b specification. This allows future specifications (including this one) to allocate semantics to previously reserved values.

5. This CEC 2.0 specification has similar or stricter expectations on a device than those listed in the above bullets 1 through 4 for the CEC 1.4b specification; they are detailed in the relevant sections of the CEC 2.0 Specification.

In order to maintain backward compatibility in CEC 2.0 and beyond,

- The HDMI Forum shall not expand the structure of the operands of CEC messages from CEC 1.x (they are all considered frozen) but may do so for CEC 2.0+ messages in later specifications. The values provided within existing operands may be extended (using previously reserved bits or values).
- The HDMI Forum shall not amend or modify the behavior of CEC 1.x messages, except for the purposes of improving interoperability and bandwidth with legacy devices. For this purpose, a clarification shall not be considered as an amendment or a modification to the behavior of the message. Such changes shall not have adverse effects on CEC 1.x devices.
- A CEC 2.0+ device may broadcast CEC 2.0+ messages at any time (because CEC 1.x defined that unrecognized messages are ignored by receiving CEC 1.x devices – see H14b Section CEC 12.3).
- A CEC 2.0+ device should not send directly addressed CEC 2.0+ messages to CEC 1.x devices at any time. (However, if it does, the CEC 1.x device has the option to <Feature Abort> them, and this is left to the 1.x adopter to determine the best course of action.) A CEC 2.0+ device shall handle any response from the CEC 1.x device gracefully.

11.1.2 Behavior with earlier versions

Table 11-1 summarizes extensions from CEC Version 1.4b to CEC Version 2.0.

Table 11-1: Behavior extensions

Feature	Section	Extension
Addressing	11.3.2	Version 2.0 provides extra Logical Addresses 12 and 13 to be used by certain devices in certain cases. Version 2.0 extends the “Video Processor” (Primary) Device Type to the more generic “Processor” to cover more use cases.
Routing Control	11.5 11.9.9	Support for <Set Stream Path> and sending <Inactive Source> was made mandatory in Version 2.0.
One Touch Record	11.2.2.2	Support for this Feature in Version 2.0 depends on the presence of the “TV supports <Record TV Screen>” bit in [Device Features], see Table 11-4.
System Information	11.2.3 11.2.4	<Report Features> and <Give Features> are new messages for Version 2.0. Support for <CEC Version> was made mandatory for Version 2.0.
Deck Control	11.2.2.2	Support for this Feature in Version 2.0 depends on the presence of the "supports being controlled by Deck Control" bit in [Device Features], see Table 11-4.
OSD Display	11.2.2.2	Support for this Feature in Version 2.0 depends on the presence of the "TV supports <Set OSD String>" bit in [Device Features], see Table 11-4 and Table 11-19
Device OSD Name Transfer	11.2.2.1	Support for <Set OSD Name> (for non-TV) and <Give OSD Name> (for TV) was made mandatory in Version 2.0.
Remote Control Pass Through	11.6	Support for this Feature was made mandatory in Version 2.0. Table 11-31 adds three new UI Command Codes (0x58, 0x59, and 0x5A). Table 11-31 also lists when certain UI Command Codes shall be supported.
Power	11.5	Use of <Report Power Status> with a broadcast address is new in Version 2.0.
System Audio Control	11.9.11	Support for this Feature was made mandatory in Version 2.0 for a TV and Amplifier as well as Generic Sources with volume/mute remote control functions, along with the <Give Audio Status> and < Report Audio Status> messages for an Amplifier. In addition, in Version 2.0 volume controls shall be sent to the TV when the System Audio Mode is [Off].
Audio Rate Control	11.2.2.2	Support for this Feature in Version 2.0 depends on the presence of the "Source supports <Set Audio Rate>" bit in [Device Features], see Table 11-4 and Table 11-25
Audio Return Channel Control	11.7	Support for this Feature in Version 2.0 depends on the presence of the "Source supports ARC Rx" and "Sink supports ARC Tx" bits in [Device Features], see Table 11-5 and Table 11-26. Version 2.0 requires that Audio Return Channel control negotiation and operation shall be done towards any adjacent HDMI device, irrespective of its Logical Address.
Dynamic Auto Lipsync	10.7	This feature was introduced in This Specification. Since all CEC messages for this feature are defined as broadcast messages, they will be ignored by Devices conforming to Version 1.4b or earlier.

11.2 Feature overview

11.2.1 Conformance Levels

The following paragraph is an addition to the text of H14b Section CEC 2.1:

Although CEC is optional when creating an HDMI product, if CEC 2.0, as defined in This Specification, is implemented, then all features and Opcodes defined as mandatory in CEC 2.0 shall be implemented (see Section 11.2.2). This shall include any feature advertised as being supported either implicitly via a device type (in [Primary Device Type] or [All Device Types]) or explicitly via the [Device Features] operand.

The implementation status of messages (mandatory or optional) is described in Section 11.2.2 and the relevant items in Table 11-13 to Table 11-29, and (when This Specification does not describe it) H14b CEC Table 8 through H14b CEC Table 28.

11.2.2 Classification and declaration of features

Devices implementing CEC 2.0 shall implement a mandatory set of features and commands. Most of these are mandatory for all devices; some are mandatory depending on the device's Device Type(s) – see Table 11-2 below. Some other features which are optional to be implemented are listed in Table 11-3 below.

Additionally, a mechanism using operand [Device Features] is defined which allows a device to indicate that certain features or functions are supported in combination with certain other conditions being met – see Table 11-4 and Table 11-5 below.

11.2.2.1 Mandatory features

Table 11-2 below lists the mandatory features which shall be implemented if any aspect of the CEC 2.0 specification is implemented; for some of these features, this requirement is conditional upon a device's declared Device Type(s), as specified in [Primary Device Type] and [All Device Types]. For each of the features, refer to the mentioned Sections and Tables for details of which messages and behavior are mandatory (this might depend on device type).

Table 11-2: Mandatory CEC features

Mandatory CEC feature	Mandatory for devices of certain Device Type (either specified in their [Primary Device Type] or in [All Device Types])	Section describing feature	Table describing messages related to feature
One Touch Play	All except Pure CEC Switches	H14b Section CEC 13.1	H14b CEC Table 8
Routing Control	All	Section 11.9.9 and H14b Section CEC 13.2	Table 11-13 and H14b CEC Table 9
Standby	All	Section 11.5.6 and H14b Section CEC 13.3	Table 11-14
Power state changes	All	Section 11.5	Table 11-13, Table 11-14, Table 11-22, Table 11-24, H14b CEC Table 8, H14b CEC Table 9
System Information	All	Section 11.2.3, 11.2.4, and H14b Section CEC 13.6	Table 11-16 and H14b CEC Table 13
Device OSD Name Transfer	All except Pure CEC Switches	Section 11.9.10 and H14b Section CEC 13.11	Table 11-20
Remote Control Pass Through	All except Pure CEC Switches	Section 11.6 and H14b Section CEC 13.13	Table 11-21
Power Status	All except Pure CEC Switches	Section 11.5 and H14b Section CEC 13.14	Table 11-22
General Protocol	All except Pure CEC Switches	Sections 11.9.7 and 11.9.8 and H14b Section CEC 12	Table 11-23
System Audio Control	TV Audio System Generic Sources with volume/mute remote control functions	Section 11.9.11 and H14b Section CEC 13.15	Table 11-24
One Touch Record	Recording Device	H14b Section CEC 13.4	Table 11-15

In addition to the above and below requirements, CEC devices shall adhere to the requirements listed throughout this section (Section 11), as well as the requirements listed in H14b Sections CEC 4 through CEC 10, CEC 12, CEC 14, and applicable parts of CEC 15 through CEC 17. For CEC Switches, additional requirements are listed in H14b Section CEC 11, which is extended as follows:

A CEC Switch shall interpret and send CEC messages which are mandatory for CEC Switches and can be switched by CEC messages.

All CEC-related features and functionality (for all mandatory features) in a device shall be enabled “out-of-the-box” so that a user can employ them without having to go through any setup steps when they first use the product.

A device with Device Type ”TV” that has dynamic menu capabilities should be able to present an overview of the devices (including their OSD names) that have been discovered so that the user can select a device from that list for viewing.

11.2.2.2 Optional features

Table 11-3 below lists a series of optional CEC features. For each of these features, if a device chooses to implement it, the device shall implement all messages and behavior marked as mandatory in the associated Tables and Sections for that feature.

Table 11-3: Optional CEC features

Optional CEC feature	Section describing feature	Table describing messages related to feature
One Touch Record	H14b Section CEC 13.4	Table 11-15
Deck Control	H14b Section CEC 13.7	Table 11-17
OSD Display	H14b Section CEC 13.10	Table 11-19
Audio Rate Control	H14b Section CEC 13.16	Table 11-25
Audio Return Channel Control	Section 11.7 and H14b Section CEC 13.17	Table 11-26

Table 11-4 and Table 11-5 below list combinations of certain device types, and certain supported messages (or other conditions). A device matching such combination shall set the associated bit(s) (see 3rd column in Table 11-4 and 2nd column in Table 11-5) in the operand [Device Features] of the <Report Features> message. In all other cases, a device shall set these bits to 0.

Table 11-4: When to set bits to 1 in [Device Features]

If a device has declared this Device Type…	…and if this/these message(s) are supported…	…then the device shall set this associated bit in [Device Features]
TV	<Record TV Screen> (as Follower)	"TV supports <Record TV Screen>"
TV	<Set OSD String> (as Follower)	"TV supports <Set OSD String>"
Playback Device or Recording Device	<Deck Control>, <Give Deck Status> and <Play> (all three as Follower) and <Deck Status> (as Initiator)	"supports being controlled by Deck Control"
Playback Device or Recording Device or Tuner	<Set Audio Rate> (as Follower)	"Source supports <Set Audio Rate>"

Table 11-5: When to set ARC bits to 1 in [Device Features]

If a device has this capability…	…then the device shall set this associated bit in [Device Features]…	…and the device shall support these messages…
Sink has ARC Tx capability on one or more HDMI inputs (note – ARC Tx might not be available on multiple inputs at the same time)	"Sink supports ARC Tx"	messages marked with "ARC Tx device" in Table 11-26
Source has ARC Rx capability on HDMI output	"Source supports ARC Rx"	messages marked with "ARC Rx device" in Table 11-26

11.2.2.3 Other features in CEC 1.4b

Some subsections in H14b Section CEC 13 and some Tables in Section 11.10 are not listed in Table 11-2 and Table 11-3. These sections and tables correspond to other features described in CEC 1.4b, which are summarized in Table 11-6 below and which are not extended relative to CEC 1.4b.

Table 11-6: Other CEC features

Other CEC feature	Section describing feature	Table describing messages related to feature
Timer Programming	H14b Section CEC 13.5	H14b CEC Table 12
Tuner Control	H14b Section CEC 13.8	H14b CEC Table 15
Vendor Specific Commands	Section 11.8, Section 11.9.4, and H14b Section CEC 13.9	Table 11-18 and H14b CEC Table 16
Device Menu Control	H14b Section CEC 13.12	H14b CEC Table 19
CDC	H14b Section CEC 13.18	H14b CEC Table 26

Vendor Specific Commands can be used by a CEC 2.0 device – but shall follow the requirements listed in Sections 11.8 and 11.9.4.

Device Menu Control is deprecated – see Section 11.6.1.

11.2.2.4 Other features using CEC messaging

The Dynamic Auto Lipsync feature, defined in Section 10.7 of This Specification, uses CEC messages (see Table 10-27); the operands for those messages are defined in Table 10-28 and Section 10.7.1 contains further information.

11.2.3 Feature Discovery

The <Report Features> message is used by a device to broadcast its features: a combination of its CEC version, the collection of Device Types in the device (operand [All Device Types]) and several other characteristics (operands [RC Profile] and [Device Features], see Section 11.6.4 and 11.2.2). It shall be sent by a device during/after the logical address allocation (see Section 11.3.3), and when requested by another device (this request can be done with a <Give Features> message to a particular device).

When a device makes such updates that one or more of the operands of the <Report Features> message change value, it shall broadcast a <Report Features> message with the up-to-date operand values.

The length of the operands [Device Features] and [RC Profile] is variable (1 ..N bytes) and is determined by bit 7 of each byte; if this bit is set (=1), then the following byte also belongs to the operand. This allows the creation of variable length operands for [Device Features] and [RC Profile]. CEC 2.0 needs only 1-byte versions of both operands, but devices implementing CEC 2.0 shall be able to handle this variable length mechanism. When encountering a longer operand than defined in CEC 2.0 for [Device Features] or [RC Profile], the Follower shall ACK all bytes, use the first byte(s) that are defined in the specification it was designed to, and ignore the additional bytes of the operand. When encountering a shorter operand than expected for [Device Features] or [RC Profile], the Follower shall ACK all bytes, use the received byte(s) as specified, and shall assume zero values for the bits in the bytes that were not received.

11.2.4 Version Discovery

The last paragraph of H14b Section CEC 13.6.2 is extended as follow:

A device may ask another device to indicate which Version of CEC the target device supports. It shall do this by sending a <Get CEC Version> message. The target device shall respond with a <CEC Version> message, which includes the relevant [CEC Version] operand.

A device should inquire which version of the CEC specification another device implements by sending the <Get CEC Version> message as described above. If <CEC Version> is not received (which is possible since this message was not mandatory before CEC Version 2.0), it shall assume the other device is designed to meet CEC Version 1.4b; if a response of <CEC Version> is received, the operand [CEC Version] shall be inspected to determine the CEC version of the other device. If a device already has received messages introduced in CEC Version 2.0 or later (e.g. <Report Features>) that contain the operand [CEC Version], it shall conclude that this device is designed according to CEC Version 2.0 or later.

If [CEC Version] contains a value that is not known to a device (i.e. [CEC Version] is newer than its own [CEC Version]), that device shall operate to the highest level of functionality that it has been designed for, and shall assume the newer specification version has retained backwards compatibility with the older versions. A 2.0 device that encounters a CEC version higher than "Version 2.0" shall operate as a 2.0 device.

See Table 11-30 for extension of the [CEC Version] operand.

11.3 Addressing

11.3.1 Physical Addresses

The text in H14b Section CEC 10.1 is extended as follows:

The algorithm defined in Section 10.9 is used to allocate the Physical Address of each device.

Whenever a new Physical Address (other than F.F.F.F) is discovered, a CEC device shall:

- allocate the Logical Address (see Section 11.3.3);
- report its supported features by broadcasting a <Report Features> message indicating the device's characteristics in operands [CEC Version], [All Device Types], [RC Profile], and [Device Features];
- report the association between its Logical and Physical Addresses by broadcasting a <Report Physical Address> message indicating the device's Primary Device Type.

This process allows any device to create a map of physical connections to Logical Addresses.

<Report Features> shall be sent before <Report Physical Address> so Followers can identify CEC 2.0 devices easily.

<Report Physical Address> should be sent no later than 1 second after <Report Features> (measured between the start bits of both messages).

In order to prevent duplicate messages unnecessarily consuming bandwidth, a Follower receiving <Report Features> or <Report Physical Address> from a device should not send inquiring messages that would result in a broadcast response (such as <Give Physical Address> and <Give Device Vendor ID>) back to that device for 2 seconds after receiving <Report Features> or <Report Physical Address> to give the Initiator the opportunity to broadcast its information.

Also, a device should not ask for static information (e.g. Vendor ID) that another device has already supplied.

11.3.2 Device Types and Logical Addresses

The text in H14b Section CEC 10.2 is extended as follows:

Each device appearing on the control signal line has a Logical Address which is allocated to only one device in the system (except for devices using Logical Address 15, where several devices may take this address with reduced functionality).

Each CEC Device has a (single) Primary Device Type from the following Table 11-7 depending on its characteristics, and advertises this in the cluster.

Table 11-7: Device Types

Device Type	Characteristics
TV	Render the video from HDMI input on a screen
Recording Device	Generic Source with recording functionality that is exposed via CEC Feature “One Touch Record”
Tuner	Generic Source with tuner functionality that is exposed via CEC feature “Tuner Control”
Playback Device	Generic Source which is not a Recording Device or Tuner
Audio System	Render the audio from HDMI input (or alternative audio input); implements System Audio Control feature
Pure CEC Switch	A device according to H14b Section CEC 11.1 which has no other functionality or Device Type
Processor	A device with all the following properties: • cannot itself become an Active Source; • has an HDMI output and at least one input (HDMI or non-HDMI); • passes video from input to output modified or unmodified; • has its own Physical Address; • requires direct addressing; • has no other device type

Note that [All Device Types] contains a bit for device type ’CEC Switch’. A device that includes a CEC Switch (see Section 11.2.2.1 and H14b Section CEC 11) shall set the corresponding bit in [All Device Types] – this includes all “Pure CEC Switches” and all “Processors”, as well as other devices that have a CEC Switch (e.g. Amplifier with HDMI input(s)).

A device shall try to allocate a Logical Address according to its [Primary Device Type] (see details below), and shall report this Primary Device Type as operand in <Report Physical Address> (see Section 11.3.1).

A Generic Source device that does not have a recorder or tuner – or only has them as a secondary feature – shall select ‘Playback Device’ as [Primary Device Type]. Some examples of such Source Devices that need to select ’Playback

Device' are media player, PC, game console, photo camera, a STB that does not implement the Tuner Control feature, a device that converts analog inputs to HDMI.

Where a physical device (i.e. an entity that the consumer considers a device) needs to expose multiple device types, it shall select a single primary device type from the above list, and shall try to allocate a Logical Address according to this primary device type, and report this primary device type in the operand [Primary Device Type] of the <Report Physical Address> message (see Section 11.3.1). The device shall indicate all the supported device types (including the primary device type) in operand [All Device Types] of <Report Features>.

Example: a recorder that also wants to expose its digital tuner functionality, selects 'Recording Device' as [Primary Device Type], and tries to allocate a Logical Address accordingly. In its [All Device Types] operand, it declares both 'Recording Device' and 'Tuner'.

For certain combinations of multiple device types in a single physical device, it is allowed to expose both device types with a separate Logical Address – so that the physical device will have two Logical Addresses. These combinations are listed in the below Table. Other combinations allocating multiple Logical Addresses are not allowed.

Table 11-8: Combinations of Device Types that may try to allocate multiple Logical Addresses

Device type #1	Device type #2	Use case / Comment
Audio System	Playback/Recording Device	Combination of playback/recording device inside Audio System enclosure ("home theater system")
TV	Playback/Recording Device	Combination of playback/recording device inside TV enclosure ("bolt-on" integration)

A device using this mechanism shall report from each Logical Address allocated:

- The appropriate Primary Device Type linked to the Logical Address,
- Both device types from the table above, as well as any additional device types that the device may have, in the [All Device Types] operand.

Example: a home theater system (combination of Audio System and Blu-ray player in single physical device enclosure) tries to allocate Logical Address 5 (for the Audio System) and one of the Logical Addresses associated with the 'Playback Device' (4, 8 or 11). On these allocated Logical Addresses, it reports the appropriate [Primary Device Type] ('Audio System' on Logical Address 5 and 'Playback Device' on the other Logical Address). On both Logical Addresses, it reports [All Device Types] with the bits for 'Audio System' and 'Playback Device' set (=1). If the device has additional device characteristics, it does not allocate additional Logical Addresses, but it sets additional bits in [All Device Types] operand instead.

For all combinations of multiple device types in a single device not listed in Table 11-8, the device shall select a (single) Primary Device Type and only allocate a single corresponding Logical Address; all of its other device types shall be reported using the [All Device Types] operand.

Some Logical Addresses are dedicated for certain Primary Device Types (see Table 11-9). The text below (from H14b Section CEC 10.2) describes the mandatory steps to select and try to allocate a Logical Address for a device, depending on its primary device type.

- A device with Primary Device Type 'TV' which has Physical Address 0.0.0.0, shall try to allocate the relevant 'TV' (0) Logical Address. If the 'TV'(0) Logical Address cannot be allocated it may try to allocate the 'Specific Use' (14) Logical Address (note that allocating the 'Specific Use' (14) Logical Address might result in reduced functionality being available);

- A device with Primary Device Type 'TV' at a Physical Address other than 0.0.0.0 shall try to allocate the 'Specific Use' (14) address. If address 14 is already allocated, it shall take the 'Unregistered' Logical Address (15);
- A device with Primary Device Type 'Audio System' shall try to allocate the relevant 'Audio System' (5) Logical Address;
- A device with Primary Device Type 'Playback Device' which can become an Active Source, shall try to allocate one of the 'Playback Device' Logical Addresses (4, 8, 11);
- A device with Primary Device Type 'Recording Device' which can become an Active Source, shall try to allocate one of the 'Recording Device' Logical Addresses (1, 2, 9);
- A device with Primary Device Type 'Tuner' which can become an Active Source, shall try to allocate one of the 'Tuner' Logical Addresses (3, 6, 7, 10);

For a Special Device (see Section 11.3.4) using a single CEC line (see H14b CEC Figure 9A and H14b CEC Figure 10A), or for the output (secondary CEC Line side) of a Special Device which has both primary and secondary CEC lines (see H14b CEC Figure 9B and H14b CEC Figure 10B):

- If it wants to advertise being a second TV, then it shall try to allocate 'Specific Use' (14) Logical Address. Such a device uses "TV" for [Primary Device Type] when sending a <Report Physical Address> message;
- If it wants to advertise being a Processor (see conditions in Table 11-7), then it shall try to allocate 'Specific Use' (14) Logical Address. Such a device uses "Processor" for [Primary Device Type] when sending a <Report Physical Address> message;
- Else if it wants to advertise any other functionality in the special device, such as a Tuner, it shall try to allocate a Logical Address for the Primary Device Type that it wishes to advertise.

For a Special Device which has both primary and secondary CEC lines, the input (primary CEC line) side shall try to allocate the relevant 'TV' (0) Logical Address.

If a device is a pure CEC Switch or CDC-only device according to H14b Supplement 2 or it does not want to advertise any functionality, it shall take the 'Unregistered' Logical Address (15).

'Specific Use' Logical Addresses (14) shall only be used for those cases described above.

For details on how to "try to allocate" a Logical Address, see Section 11.3.3.

When there are many devices of type 'Playback Device', 'Recording Device', 'Tuner' or 'Processor' in a system, a device might fail to allocate a Logical Address out of the pool of addresses for that particular Device Type, in which case it may try to allocate one of the addresses 12 or 13 (Backup 1 or 2, see Table 11-9). When using one of these 'backup' logical addresses, the [Primary Device Type] reported in <Report Physical Address> shall be the original Device Type. In all other cases, these "Backup" Logical Addresses shall not be used.

Note regarding interoperability: Devices using CEC version 1.4b or earlier might not expect the use of Logical Addresses 12 or 13, but it is likely to work better (at least not worse) than the alternative offered by CEC version 1.4b and earlier for this case (the device using the 'Unregistered' Logical Address, with much reduced functionality).

If a device tries to allocate a Logical Address, and it fails to allocate any of the possible Logical Addresses mentioned above, it can either take the 'Unregistered' Logical Address (15), or disable its CEC functionality.

Note that a device that has taken the 'Unregistered' Logical Address (15), will have reduced functionality since it cannot be directly addressed by other devices, and can only receive broadcast messages.

If a device has multiple instances of a particular functionality, it shall advertise only one instance. If a device has multiple tuners, it shall only expose one for control via CEC. In this case, it is up to the device itself to manage multiple tuners.

Table 11-9: Logical Addresses (extended from H14b CEC Table 5)

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	Backup 1 (for Device Types 'Playback Device', 'Recording Device', 'Tuner', 'Processor' if all dedicated Logical Addresses have been allocated)
13	Backup 2 (for Device Types 'Playback Device', 'Recording Device', 'Tuner', 'Processor' if all dedicated Logical Addresses have been allocated)
14	Specific Use
15	Unregistered (as Initiator address) Broadcast (as Destination address)

11.3.3 Logical Address allocation

The text in H14b Section CEC 10.2.1 is extended as follows:

Note that a Logical Address should only be allocated when a device has a valid Physical Address (i.e. not F.F.F.F), at all other times a device should take the 'Unregistered' Logical Address (15).

A device that wants to try to allocate a certain Logical Address shall send a <Polling Message> to the same address (e.g. 'TV' → 'TV' or 'Audio System' → 'Audio System'). If the <Polling Message> is not acknowledged, then the device knows the Logical Address is not being used, and shall take that Logical Address.

Where more than one possible Logical Address is available for the given device type (e.g. 'Tuner 1', 'Tuner 2', etc.), an address allocation procedure shall be carried out by a newly connected device. The device selects the first allocated address for that device type and sends a <Polling Message> to the same address (e.g. 'Tuner 1' → 'Tuner 1'). If the <Polling Message> is not acknowledged, then the device stops the procedure and keeps (allocates) that address.

If the first address is acknowledged, then the device selects the next address for that device type and repeats the process (e.g. 'Tuner 2' → 'Tuner 2'). Again, if the message is not acknowledged, the device keeps (allocates) that address.

This procedure continues until all possible 'type specific' Logical Addresses have been checked (possibly extended with the Backup addresses, see Section 11.3.2); if no 'type specific' Logical Addresses are available the device should take the unregistered address (15). Note that several physical devices might be sharing this address, and functionality will be reduced since a device with this address cannot be directly addressed by other devices, and can only receive broadcast messages.

A device may lose its Logical Address when it is disconnected or switched off. However, it may remember its previous Logical Address, so that the next time it is reconnected or switched on, it should begin the polling process at its previous Logical Address and try each other allowable Logical Address in sequence before taking the unregistered address. For example, if a STB that was previously allocated address 'Tuner 2' is reconnected, it would poll 'Tuner 2', 'Tuner 3', 'Tuner 4' and 'Tuner 1' (and then possibly the Backup addresses) before taking the unregistered address.

If a device loses its Physical Address at any time (e.g. it is unplugged) then its Logical Address should be set to 'Unregistered' (15).

A device that has allocated a Logical Address after the above process, shall report this to the system by sending messages as detailed in Section 11.3.1.

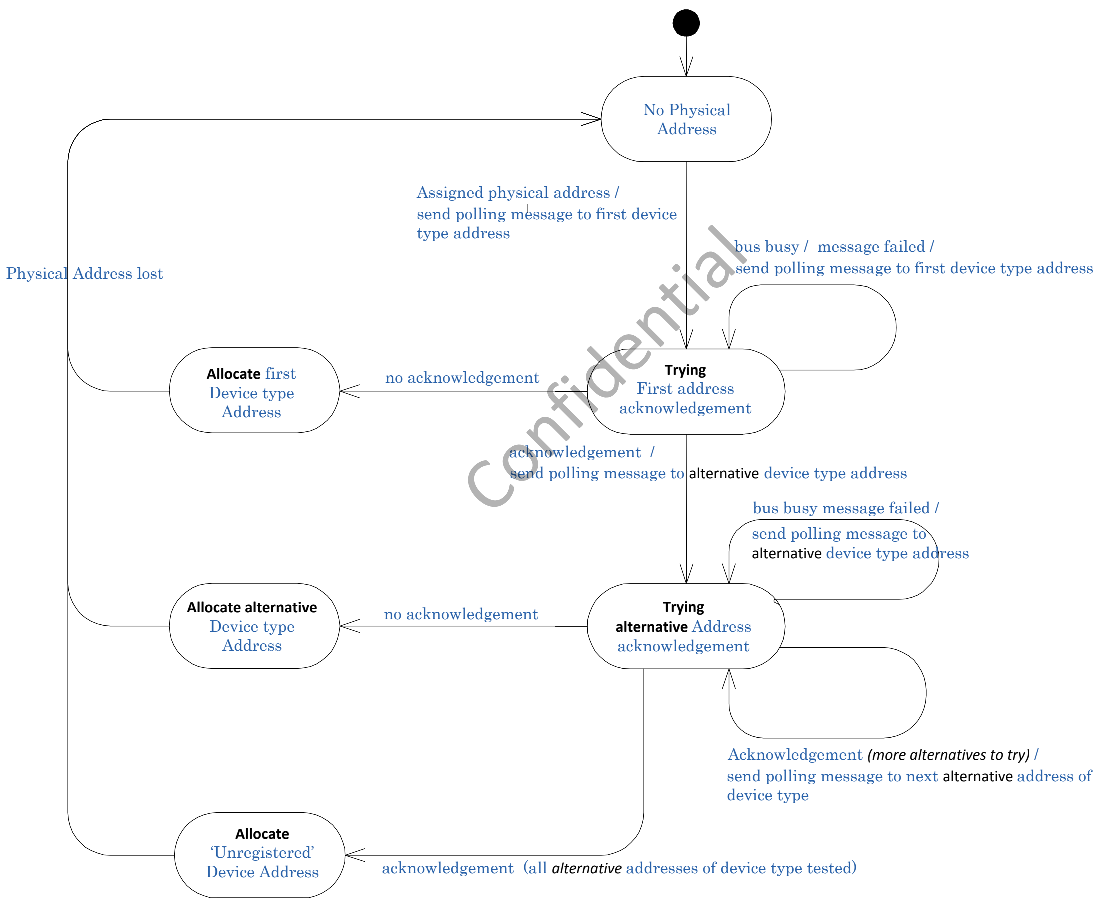


Figure 11-1: Logical Address Allocation *(Clarified from H14b CEC Figure 8)*

11.3.4 Logical Addressing for Special Devices

The last sentence of the first paragraph of H14b Section CEC 10.2.2 is extended as follows:

The display (panel) has Primary Device Type='TV' and tries to allocate Logical Address 'TV' (0).

The last paragraph on page CEC-22 of H14b Section CEC 10.2.2 is extended as follows:

H14b CEC Figure 10 shows how a second TV can be used. In both these examples, the second TV may try to allocate the Specific Use Logical Address (14) or the Logical Address of any other functionality in the TV, such as a Tuner – depending on its Primary Device Type, see Section 11.3.2 for full details. In the example with both primary and secondary CEC lines, the second TV tries to allocate one of these addresses on the secondary CEC Line.

11.3.5 Logical Addressing for Recording Devices

The last paragraph of H14b Section CEC 13.4.2 is extended as follows:

The TV should ignore a <Record TV Screen> message that comes from a non-Recording Device address, however it shall accept the message from a Logical Address used by a Recording Device.

11.4 Polling

The description of the EOM bit in the last paragraph of H14b Section CEC 6.1.3 is extended as follows:

A message with the EOM bit set to '1' in the Header Block (indicating no further blocks will follow; it is a one-block message) can be used to 'ping' or 'poll' other devices, to ascertain if they are connected. This is the <Polling Message> and the Initiator and Destination addresses will be different. It is also used in Section 11.3.3 for allocating Logical Addresses: in this case the Initiator and Destination addresses are the same.

The text concerning <Polling Message> in H14b Section CEC 12.2 is extended as follows:

If a <Polling Message> has not been ACK'd, the device is not present or is not in a state to respond, then repeated polling of these addresses should be limited.

Version 2.0 of the specification adds additional requirements on the frequency of sending the <Polling Message>, as excessive use of the <Polling Message> on the CEC line can significantly decrease bandwidth available for, or responsiveness of, time-sensitive or user-driven features. As more devices in a system are employing CEC, unrestricted polling becomes more of an issue.

A device shall not send any <Polling Message> which is not a re-transmission attempt (see Section 11.9.3) more frequently than once every 500 ms when the <Polling Message> is for a secondary or background task (which is not the direct result of a user-initiated action or Logical Address allocation).

A device shall not send a <Polling Message> which is not a re-transmission attempt (see Section 11.9.3) to the same address more frequently than once per Minimum Polling Period, which is defined as 14 seconds. Note that longer periods between <Polling Messages> are preferred.

Only in the following exceptional cases, is polling more frequently than the Minimum Polling Period allowed:

- When a TV comes out of standby or enables CEC, it may do an initial bus scan to determine which devices are present. To complete this scan as soon as possible, the TV may send <Polling Messages> more frequently than 500 ms during this initial bus scan.
- If System Audio Mode is ON, the TV may poll the Audio System no more than once every 5 seconds, in order to ensure that the Audio System is still connected.
- There is a special case where faster polling is allowed for "mobile" devices that get regularly plugged in/out, e.g. a digital camera, and that initiate a One-Touch Play feature once they are plugged in. (Background: Since some legacy CEC Switches do not assert their HPD line on inputs that are not selected, and to handle this case,

these devices may poll specifically for a TV on the CEC line to determine connectivity.) Therefore, the following behavior is allowed for these “mobile” devices:

- Prior to detecting connectivity, a device may poll the TV no more than once per second, and may use the One-Touch Play feature upon detecting a TV.
- Once the HPD line is detected to be high or polling the TV was successful, a device shall no longer poll once per second, and shall adhere to the default Minimum Polling Period - until the TV is no longer detected.

Polling can and should be skipped if other CEC traffic shows that a device is present. Hence, a device should not poll a certain Logical Address within at least one Minimum Polling Period after the following CEC events occur between the device that is polling and the device whose Logical Address is to be polled:

- A directly addressed message, sent to that Logical Address, was acknowledged.
- A directly addressed message has been sent from that Logical Address.
- A broadcast message has been sent from that Logical Address.

It is recommended that, if the device is capable of monitoring CEC traffic directed to other devices, then this capability should also be used to further reduce the need for polling. In this case, such a device should not poll a certain Logical Address for at least one Minimum Polling Period after it detects that that Logical Address acknowledged a directed message initiated from any Logical Address, or any message was sent from that Logical Address.

11.5 Power state changes

11.5.1 Normal Power State Changes – Sending

In systems, where only devices with CEC version 2.0 (or higher) are used, the messages described in this section shall be used to ensure reliable power state behavior.

If a device wants to become the Active Source (One Touch Play), it shall use either the <Image View On> or the <Text View On> message to wake up the TV as described in H14b Section CEC 13.1. It shall not use the message <User Control Pressed> with any of the power-related operands.

If a device wants to activate audio rendering through the System Audio Control, it shall use the message <System Audio Mode Request> (with [Physical Address] operand) to wake up the Audio System as described in Section 11.9.11 and H14b Section CEC 13.15.2; it shall not use the message <User Control Pressed> with any of the power-related operands.

If the TV wants a Source Device to become the Active Source, i.e. to send video and audio towards the TV (and possibly audio towards the Audio System), the TV shall use the <Set Stream Path> message to wake up the Source device as described in Section 11.9.9; it shall not use the message <User Control Pressed> with any of the power-related operands.

For CEC Switches, wakeup is implicit – if they are on the active path, they will be woken up by the messages described in Section 11.9.9, H14b Section CEC 11, and H14b Section CEC 13.2.2 – they do not need explicit wakeup messages.

For combined devices (e.g. Audio System + CEC Switch or Audio System + Playback Device), the device that wants to wakeup either part should just wakeup that part using the mechanisms described above.

If a device wants to send all devices into standby state, it shall use the broadcast <Standby> message; it shall not use the message <User Control Pressed> with any of the power-related operands.

11.5.2 Normal Power State Changes – Receiving

A TV shall wakeup (i.e. come out of Standby state into On state) upon receiving <Image View On> or <Text View On> as described in H14b Section CEC 13.1. It shall not trigger the wakeup upon receiving <Active Source> messages.

An Audio System shall wakeup upon receiving a <System Audio Mode Request> message with [Physical Address] operand as described in Section 11.9.11 and H14b Section CEC 13.15.2.

A Source Device shall wakeup upon receiving a <Set Stream Path> message with its Physical Address as operand, as described in Section 11.9.9.

All devices shall wakeup from Standby state upon receiving a <User Control Pressed> message with one of the operands [“Power On Function”] or [“Power Toggle Function”].

All devices shall transit from On to Standby state upon receiving a <User Control Pressed> message with one of the operands [“Power Off Function”] or [“Power Toggle Function”].

All devices shall go to the Standby state upon receiving a directly addressed or broadcast <Standby> message as described in Section 11.5.6 and H14b Section CEC 13.3.2.

Note: devices using (only) Logical Address 15 can only receive broadcast messages, and hence can only be woken up with a <Set Stream Path> message, and can only be sent to standby using a broadcast <Standby> message.

For combined devices (e.g. Audio System + CEC Switch or Audio System + Playback Device), the device that wants to wakeup either part should just wakeup that part. It is up to the Follower if the other part also wakes up or not. But note that if a TV (or other device) wakes up the Audio System to render the sound from the current Active Source, the Playback Device in the combined device should not become the Active Source (since that would switch away from watching the current Active Source).

For CEC Switches, wakeup is implicit – if they are on the active path, they shall wake up by the messages described in Section 11.9.9 and H14b Sections CEC 11 and CEC 13.2.2 – they do not need explicit wakeup messages.

11.5.3 Power State Changes when using mixed system (with legacy devices) - Sending

In systems where devices with CEC version 2.0 (or higher) are mixed with devices with lower CEC versions (i.e. “legacy” devices), the devices with CEC version 2.0 (or higher) shall use the messages from Section 11.5.1 towards the other devices with CEC version 2.0 (or higher).

When devices with CEC version 2.0 (or higher) operate with devices with lower CEC versions, for bringing another device out of the Standby State into the On state, the preferred mechanism described in Section 11.5.1 shall be attempted first. Only if the desired effect is not achieved with these messages (to be checked with <Give Device Power Status>), alternative attempts using a <User Control Pressed> message with the appropriate deterministic power-related operand (“Power On Function”, 0x6D, see Table 11-31) may be tried. Only as a last resort, a <User Control Pressed> message with operand “Power” (0x40, see Table 11-31) may be attempted. Success cannot be guaranteed when using operand "Power" (0x40) since some legacy devices use this as a toggle, some use it only to bring a device out of the Standby state, and some use it only to go to the Standby state.

If a device wants to send another device into the Standby state, it should[1] use the directly addressed <Standby> message rather than the <User Control Pressed> message with any of the power-related operands. Note that a device

[1] The directly addressed <Standby> message is mandatory for a legacy Follower, and the <User Control Pressed> message is not mandatory for a legacy Follower, so the method using <Standby> has a higher chance of success.

using only Logical Address 15 cannot be sent to Standby state using either the directly addressed <Standby> message or a <User Control Pressed> message; the only way to send such a device to Standby state is by broadcasting a <Standby> message – but this obviously will send all other devices to Standby state as well.

11.5.4 Power State Changes when using mixed system (with legacy devices) - Receiving

In systems where devices with CEC version 2.0 (or higher) are mixed with devices with lower CEC versions (i.e. "legacy" devices), the devices with CEC version 2.0 (or higher) might receive power-related messages from the legacy devices other than those mentioned in Sections 11.5.1 and 11.5.2.

In order to improve interoperability, this section describes further (in addition to those described in previous sections) mandatory behavior for devices with CEC version 2.0 (or higher).

- All devices shall wakeup from the Standby state upon receiving <User Control Pressed> with operand ["Power"].

11.5.5 Power States and Power State Transitions

CEC 2.0 (and higher) supports power state change notifications to decrease the usage of <Give Device Power Status> messages to appropriate cases only.

CEC defines the power states listed in Table 11-10 and the power state transitions listed in Table 11-11.

Table 11-10: Power States

No.	Power Status	Is stable power state?	Is intermediate power state?
1	On	Yes	No
2	Standby	Yes	No
3	In transition On to Standby	No	Yes
4	In transition Standby to On	No	Yes

Table 11-11: Power State Transitions

No.	Power State Transition from …	Power State Transition to …	Description	Broadcast message to send
1	“Standby”	“In transition Standby to On”	Starting transition from “Standby” to “On”	<Report Power Status> [“In transition Standby to On”]
2	“In transition Standby to On”	“On”	Finished transition from “Standby” to “On”	<Report Power Status> [“On”]
3	“In transition Standby to On”	“Standby”	Interruption of transition from “Standby” to “On” and finished transition back to “Standby”.	<Report Power Status> [“Standby”]
4	“Standby”	“On”	Fast transition from “Standby” to “On” within 2s.	<Report Power Status> [“On”]
5	“On”	“In transition On to Standby”	Starting transition from “On” to “Standby”	<Report Power Status> [“In transition On to Standby”]
6	“In transition On to Standby”	“Standby”	Finished transition from “On” to “Standby”	<Report Power Status> [“Standby”]
7	“In transition On to Standby”	“On”	Interruption of transition from “On” to “Standby” and finished transition back to “On”.	<Report Power Status> [“On”]
8	“On”	“Standby”	Fast transition from “On” to “Standby” within 2s.	<Report Power Status> [“Standby”]

All CEC devices shall notify all of their power state transitions to other CEC devices by broadcasting <Report Power Status> messages. Note – in this mechanism, the <Report Power Status> message is used as a broadcast message, so all devices in the system are aware.

At the start of each power state transition, i.e. from “Standby” to “On” or vice versa from “On” to “Standby”, all CEC devices shall broadcast either a <Report Power Status>[“In transition Standby to On”] or a <Report Power Status>[“In transition On to Standby”] message respectively. Immediately after that power state transition has finished with a CEC device being in the appropriate stable power state, that CEC device shall broadcast either a <Report Power Status>[“On”] or a <Report Power Status>[“Standby”] message respectively.

If an ongoing power state transition is interrupted and the CEC device has finished its transition back to the previous stable power state, e.g. when the user repeatedly presses the power button, that CEC device shall broadcast either a <Report Power Status>[“On”] or a <Report Power Status>[“Standby”] message corresponding to the previous stable power state (which is also the new power state).

When a CEC device knows that its power state transition, i.e. from “Standby” to “On” or vice versa from “On” to “Standby”, will be finished within 2 seconds, that CEC device should not send either a <Report Power Status>[“In transition Standby to On”] or a <Report Power Status>[“In transition On to Standby”] message respectively. In this case, immediately after that power state transition has finished with the CEC device being in the appropriate stable power state, the CEC device shall broadcast either a <Report Power Status>[“On”] or a <Report Power Status>[“Standby”] message respectively. Note – the recommendation to omit sending the ‘in transition’ message is in order to avoid many CEC messages being sent, e.g. when several CEC devices are powered-up simultaneously.

Whilst coming out of the “Standby” power state i.e. while in the “In transition Standby to On” power state, some devices may not be able to handle many CEC messages at the application layer. Therefore, CEC devices should refrain from sending CEC messages towards those CEC devices with the “In transition Standby to On” power state.

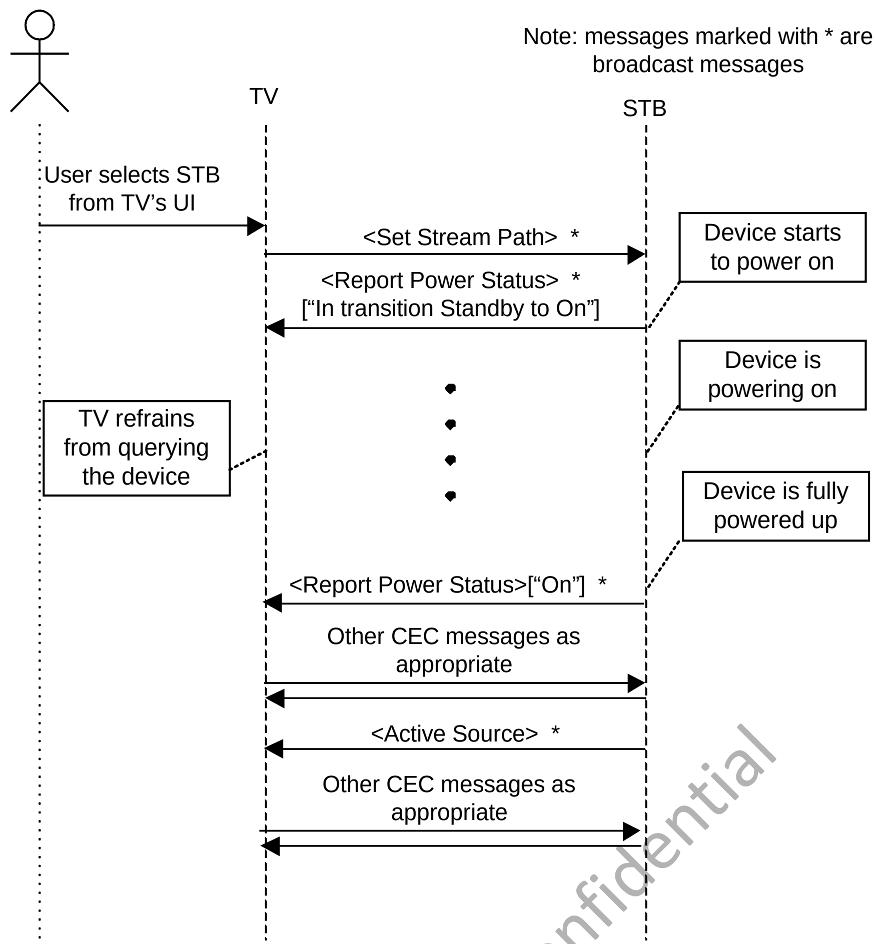


Figure 11-2: A typical scenario for a device waking up (transitions #1 and #2)

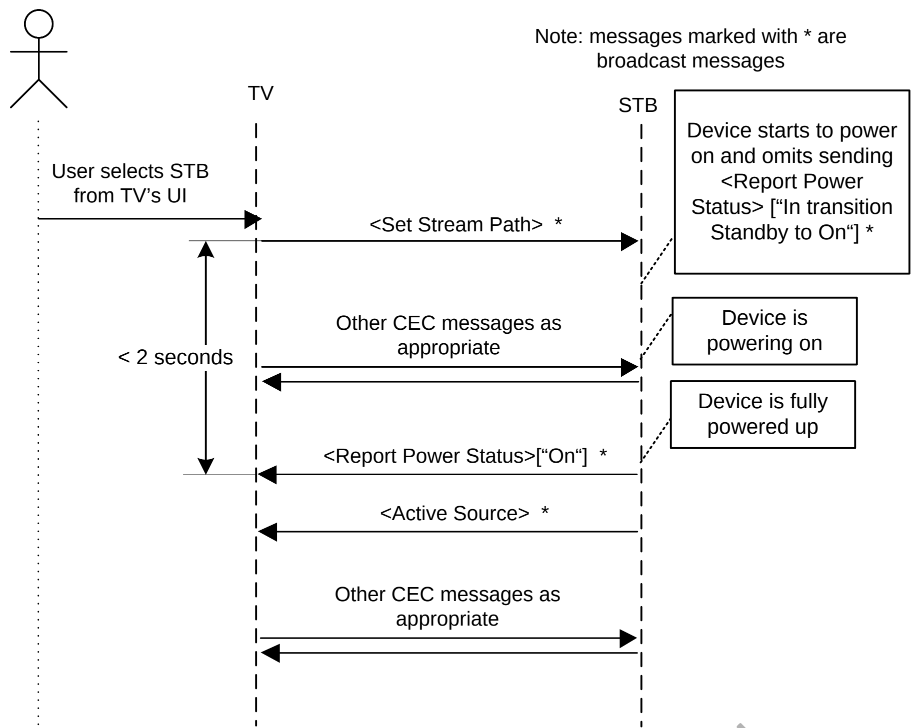


Figure 11-3: A typical scenario for a device waking up (transition #4)

For all power state transitions (#3, #6, and #8) in Table 11-11 that move a device into the "Standby" power state, the device shall broadcast a <Report Power Status> ["Standby"] message so that the other devices in the system are aware. Other devices can act appropriately in reaction to such message (e.g. TV can switch to another Source if it was watching this device, similar to <Inactive Source>, see Section 11.9.9).

11.5.6 System Standby

H14b Section CEC 13.3.2 is extended as follows:

The broadcast message <Standby> can be used to switch all CEC devices to the Standby state. A typical scenario where the user sets the whole system to the Standby state is shown below:

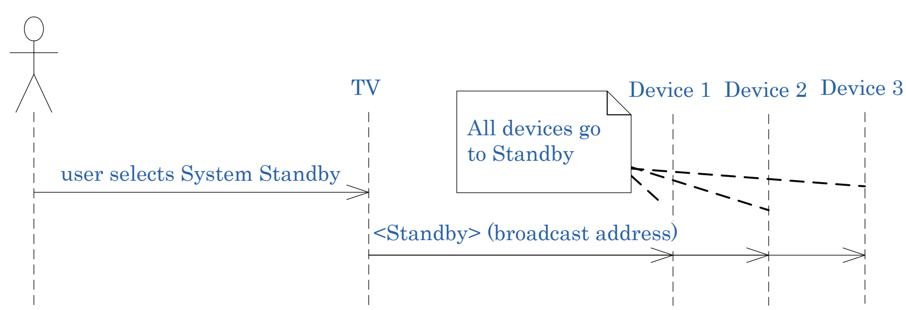


Figure 11-4: A typical scenario for the broadcast (system) Standby feature *(from H14b CEC Figure 13)*

The whole system can be set to the Standby state by a broadcast <Standby> message from any device in the system. It is manufacturer dependent when this message is sent.

Typically, a TV broadcasts a <Standby> message if the TV is switched off by the user, to bring the other devices into the Standby state when the TV is switched off ("single button switch-off").

A device can switch another device into the Standby state by sending the message <Standby> as a directly addressed message to it.

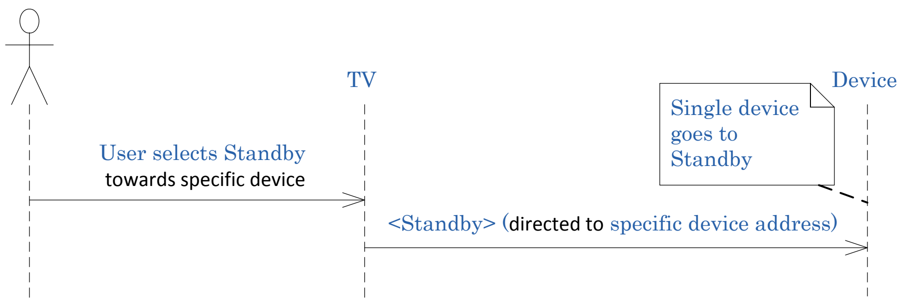


Figure 11-5: A typical scenario for the Standby feature to a specific device *(Clarified from H14b CEC Figure 14)*

When a source device is put to Standby by the user (e.g. by its own remote control or local key), it shall not broadcast a system <Standby> message unless explicitly requested by the user.

A <Standby> message is not a toggle and can only be used to send a device to the Standby state: other messages shall be used to activate a device, i.e. bring a device out of the Standby state (see Sections 11.5.1 and 11.5.3).

A device in the On state receiving a directly addressed or broadcast <Standby> message shall go into a Standby state. A device in a Standby state receiving a directly addressed or broadcast <Standby> message shall stay in a Standby state.

A <Standby> message should not interrupt any background tasks such as a recording - see Timed Recording, H14b section CEC 13.5.3.

Devices may ignore <Standby> messages if they are in an On state where going into the Standby state is not the appropriate action or due to device limitations it is not possible to go to the Standby state. For example:

- The device is recording (see Section 11.5.7);
- The device only has a mechanical power switch;
- It only provides limited facilities for external control of its power;
- The Standby function is disabled;
- It is a device, such as a PC, which is performing other functions that should be left running;
- High priority services, such as the reception of emergency announcements or similar, shall continue.

11.5.7 Recording

The penultimate paragraph of H14b Section CEC 13.4.2 and also the last paragraph of H14b Section CEC 13.5.2 are extended as follows (i.e. this applies for both directed and broadcast <Standby> messages):

When a recorder is making a recording, the <Standby> message should not interrupt a recording in progress. If the recorder receives a <Standby> message during the recording, it should react to the <Standby> message when the recording has finished unless it is the Active Source at the end of the recording.

11.6 Remote Control Pass Through

11.6.1 Relationship with other features

The CEC features Deck Control (see H14b Section CEC 13.7) and Tuner Control (see H14b Section CEC 13.8) have some functional overlap with functionality provided by Remote Control Pass Through. Since support for Remote Control Pass Through is mandatory (and Deck Control and Tuner Control are not mandatory), Remote Control Pass Through takes precedence over Deck Control and Tuner Control.

The Device Menu Control Feature (see H14b Section CEC 13.12) is superseded by Remote Control Pass Through, and is no longer needed for devices with CEC version 2.0 or higher.

To improve interoperability with TVs with lower CEC version, Sources conforming to CEC 2.0 or later versions thereof may need to send <Menu Status> [“Activated”] to such legacy TVs when they are the Active Source to make the TV forward remote control buttons – even if they have no menu on the screen (thus violating the description of this feature in CEC version 1.4b).

In operation with a TV with CEC version 2.0 or higher, the <Menu Status> message should not be sent.

11.6.2 Feature Description

H14b Section CEC 13.13.2 is extended as follows:

This feature is used to pass remote control commands received by one device (typically the TV) through to another device in the network. This feature will typically be used in situations where a device offers a remote control which is employed as a “single remote” for controlling other devices within the system. The device will receive the RC command and will typically pass the command through to the appropriate target device, see Section 11.6.4.

For an overview of the buttons and triggers that can be sent using this mechanism, see Table 11-31. Note that some of these triggers could be implemented through an ‘on-screen’ menu rather than as physical buttons on the remote (see Section 11.6.6, first paragraph). Other controls the forwarding device might have (example: gesture control for volume), that are mapped internally to a “User Operation” entry in Table 11-31 shall be forwarded in same way as the equivalent button.

When a remote control button which needs to be forwarded is pressed, (see Section 11.6.4, also for the destination of the forwarding), the Initiator shall send a <User Control Pressed> message with the [UI Command] corresponding to the button that is pressed, see Table 11-31. When that button is released, devices that do not implement Press and Hold operation (see Section 11.6.3 and H14b section CEC 13.13.3) shall send a <User Control Released> message immediately upon detection of the release. Another implementation is to send the <User Control Released> message shortly after the <User Control Pressed> message, even if the key has not yet been released. For devices that do implement Press and Hold operation, see Section 11.6.3 and H14b section CEC 13.13.3.

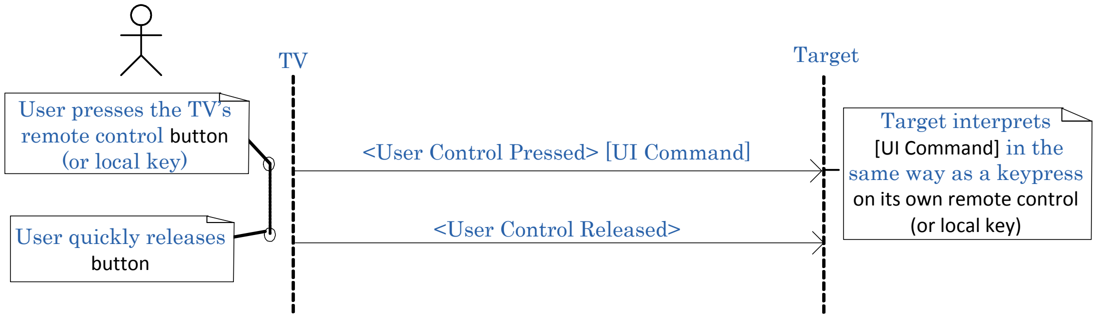


Figure 11-6: A typical scenario where the user presses and quickly releases the same button *(Clarified from H14b CEC Figure 22)*

The Initiator may send further <User Control Pressed> messages without interleaving <User Control Released> messages if a new button press occurs (with corresponding new value of [UI Command]) within the Initiator Repetition Time defined in H14b section CEC 13.13.3(1). This has the additional implicit effect of sending a <User Control Released> for the first button, see Figure 11-7 below:

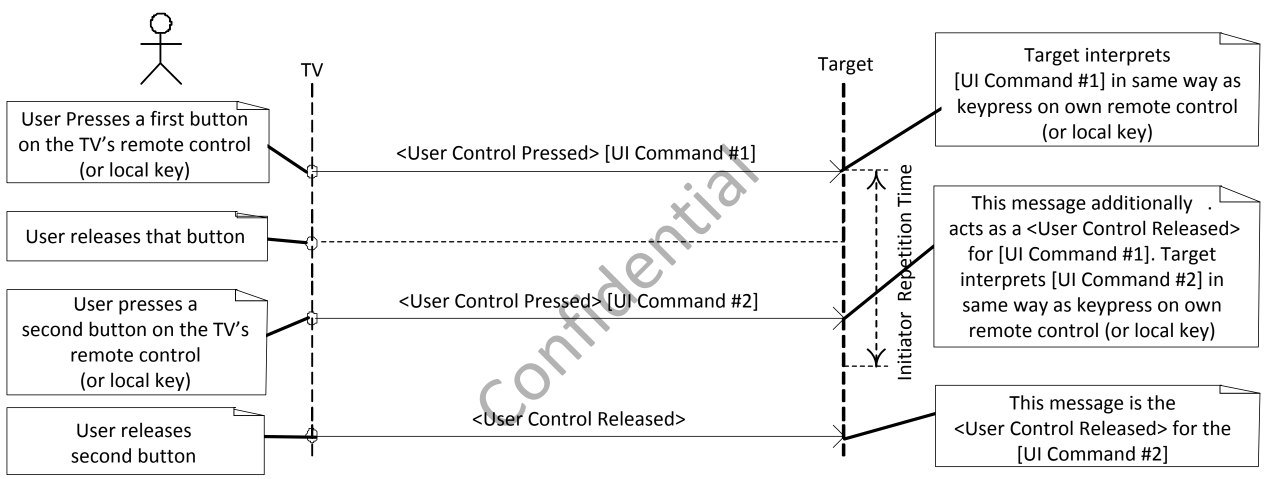


Figure 11-7: A typical scenario where the user quickly presses a second button

The <User Control Pressed> and <User Control Released> messages indicate that the user has pressed and released the relevant button on their remote control.

Table 11-31 indicates (in the last two columns) which [UI Commands] shall be supported by the Follower. For the non-Deterministic commands (i.e. those commands in Table 11-31 which are not mentioned in H14b CEC Table 6), the response may be device-dependent and the Follower shall interpret those <User Control Pressed> and <User Control Released> messages in the same way as when a user presses and releases the corresponding button on device's own remote controller (or local key). For commands where there is no such equivalent button on the device's own remote controller, the Follower should nevertheless attempt to mimic such behavior. For example: consider a device which has subtitling functionality, but has no direct button on its own remote control for this feature (i.e. the user can only select the feature from a menu or local key); when such a device receives <User Control Pressed>["Subpicture"], it should trigger the subtitling function/display.

For the Deterministic commands (i.e. those commands in H14b CEC Table 6), the Follower's response is detailed in H14b CEC Table 6, Table 11-31 and elsewhere in This Specification.

A device that has initiated a <User Control Pressed> message shall ensure that it sends a <User Control Released> message before going into the Standby state. In the event that the Initiator of the message is powered off or its HDMI cable is disconnected before sending a <User Control Released> message, the Follower will never receive the <User Control Released> message.

If a Follower does not receive a <User Control Released> message (or another <User Control Pressed> message) within an appropriate time period equal to the Follower Safety Timeout period, it shall assume that the button has been released and act accordingly. For details of the Follower Safety Timeout period see Section 11.6.3, which defines the Press and Hold Operation.

If a Follower receives a <User Control Pressed> message with an operand that it does not support, it may send a <Feature Abort> message with an [Abort Reason] of "Invalid operand". Note – Followers should be aware of and avoid situations where, for some [UI Command]s that may be repeatedly sent such as ["Volume Up"], this may cause many <Feature Abort> messages to be sent. Followers should avoid such situations by limiting the number of <Feature Abort> messages sent, e.g.

- For a [UI Command] such as ["Volume Up"], the Follower should not send <Feature Abort> at all.
- For a [UI Command] such as ["Video on Demand"], the Follower may send <Feature Abort>.

Note - Where the Follower does not support the operand of <User Control Pressed> in its current state, the [Abort Reason] of "Refused" is more appropriate.

11.6.3 Press and Hold Operation

(1) Initiator Behavior

The Initiator Behavior is described in H14b Section 13.13.3 (1) with following extensions.

The last sentence of the first paragraph following H14b CEC Figure 23 of H14b Section CEC 13.13.3 is extended as follows:

Implementers should note that using timings near the maximum value may result in incorrect Press and Hold behavior (as this is very close to the Follower Safety Timeout period of the Follower) and that using timings near the minimum value places an unnecessarily heavy load on the CEC line, so both of these cases should be avoided.

The text following H14b CEC Figure 24 in H14b Section CEC 13.13.3 is extended as follows:

Note that H14b CEC Figure 24 shows the case where only one button is pressed and released.

If the user presses and holds a button for a long time and then releases the first button and presses another button within the Initiator Repetition Time (see H14b section CEC 13.13.3 (1)), then it is not necessary to send a <User Control Released> message when the first button is released because the <User Control Pressed> with the second [UI Command] acts as the <User Control Released> message for the first button. If the second button is pressed after the Initiator Repetition Time, the Initiator shall send a <User Control Released> message.

An Initiator which implements Press and Hold behavior may send an additional <User Control Released> message just after the first <User Control Pressed> message, and start the actual Press and Hold behavior after some time (which can be larger than the Initiator Repetition Time and the Follower Safety Timeout). This has the effect of a single press followed by a Press and Hold sequence.

(2) Follower Behavior

The Follower Safety Timeout period of a Follower supporting Press and Hold operation shall not be less than 500ms and is recommended to be at least 550ms. The time between messages for the Follower Safety Timeout period is measured from the end of the message, i.e. when the Follower receives a Data Block where the EOM bit is set to '1'.

The Follower shall start the Press and Hold behavior (see H14b CEC figure 24) when another <User Control Pressed> message containing the same [UI Command] is received within the Follower Safety Timeout period.

- The Press and Hold behavior (e.g. increment step, speed, etc) is defined by the Follower and should have the same behavior when using CEC as when one of the buttons on its own remote control or one of the local keys is pressed;
- It is optional for the Follower to start the Press and Hold behavior earlier, i.e. before the second <User Control Pressed> message has been received. Note that if the Follower starts the Press and Hold behavior before the second <User Control Pressed> message, then the Follower might make several increments before the user can release the button – which might not be the user's intention, so this is discouraged.

The Follower shall stop its Press and Hold behavior for the previous [UI Command] when:

- A <User Control Released> message is received; or
- A <User Control Pressed> message containing a different (new) [UI Command] is received within the Follower Safety Timeout period; or
- The Follower Safety Timeout period has expired.

In the last two cases above, the Follower shall behave as if it has received a <User Control Released> message.

The Follower shall stop its Press and Hold behavior for the previous [UI Command] before it handles the <User Control Pressed> message for a new [UI command].

Whether a forwarded [UI Command] is executed as a single-shot event or part of a Press and Hold sequence is determined by the Follower (e.g. volume and cursor buttons could have press-and-hold behavior while color buttons would not have such behavior), and might also be state-dependent.

11.6.4 RC Button Forwarding Principles of Operation

H14b Section CEC 13.13.4 is extended as follows:

In order to allow for "single remote control" system operation, whereby, for example, the TV's remote controller is used to control other devices by sending User Control messages, the TV shall send to the appropriate device as many button presses as possible which the TV does not require itself or does not require in its current state, using <User Control Pressed> and associated <User Control Released> messages.

The "what you see is what you control" method is used to determine whether to forward a button or not: the buttons on the remote that are not needed for the TV's internal operation (in its current state) are forwarded to the device that is being watched (the Active Source). This forwarding behavior shall not depend on <Menu Request> or <Menu Status> messages being sent (see H14b Section CEC 13.12 for description of those messages); the TV shall forward button presses using the <User Control Pressed> and <User Control Released> messages to the device that is the current Active Source (see below for exception for buttons related to audio rendering).

During the time that the TV generates and displays an OSD overlay (e.g. menu or pop-up), the button presses related to this menu are handled in the TV and are not forwarded. Once the OSD overlay has been removed, normal forwarding is resumed.

The TV will need a remote control button (or another mechanism) to allow the user to switch to other Source Devices. Typical examples include: source button (gives list of Sources that can be selected from), home button (gives TV's home screen allowing user to select other functions or Sources).

The TV shall indicate to the other devices in the system which buttons and triggers can be generated by the TV. This information is contained in the operand [RC Profile ID], which is sent in the <Report Features> message. A Source Device may use this information to adapt its operation if necessary to the buttons and triggers that the TV can send (e.g. have different menu trees depending on TV's remote profile).

CEC 2.0 defines 4 remote control profiles:

- profile 1 = minimalistic zapper (low button count)
- profile 2 = intermediate between profile 1 and profile 3
- profile 3 = typical TV remote
- profile 4 = extended form of profile 3

The TV shall indicate the highest (largest) profile in operand [RC Profile ID] for which the user can initiate all the UI Commands marked in Table 11-31 for that Remote Control profile. If a TV does not support all UI Commands for any of these profiles, it shall set [RC Profile ID] to 0x00. An RC Profile ID value of 0x00 does not mean there are no buttons on the TV remote. The user may still be able to initiate any UI Commands this TV supports.

Both the buttons and triggers in the reported profile as well as additionally available buttons and triggers shall be forwarded as described above in the first paragraph of this Section 11.6.4, whenever the TV does not require these itself or does not require these in its current state, i.e. the list of buttons in the reported profile is not limiting the buttons/trigger that may and shall be sent.

The "single remote" principle can also be used with the remote control of another device, e.g. the remote control associated with a STB. In that case, the STB would forward buttons that it does not require, or does not require in its current state, to the TV or the Active Source as appropriate (see below for exception for audio rendering related buttons).

The handling of buttons related to audio rendering (Volume Up, Volume Down, Mute) and any other buttons (UI Commands in Table 11-31) related to audio rendering such as ["Select Sound Presentation"], ["Mute Function"] and ["Restore Volume Function"] is different from the above principles, and depends on the type of device (also see Section 11.9.11.4):

- For a TV, these are either handled inside the TV itself (if System Audio Mode is off) or forwarded to the Audio System (if System Audio Mode is on).
- For a Source Device, these are either sent to the TV (if System Audio Mode is off) or sent to the Audio System (if System Audio Mode is on).
- For an Audio System, these are either handled inside the Audio System itself (if System Audio Mode is on) or forwarded to the TV (if System Audio Mode is off).

If a device has not received a <Set System Audio Mode> ["On"] message, or it does not succeed in sending messages to Logical Address 5 (which will happen if no device with Logical Address 5 is present in the system), the device shall act according to the above rules for the case "System Audio Mode is off".

11.6.5 Reporting of capabilities related to Remote Control Pass Through

A TV shall indicate to the other devices in the system which buttons and triggers can be generated by the TV using the operand [RC Profile ID] in the <Report Features> message, as described in the previous section.

A device which is not a TV, and can be controlled via Remote Control Pass Through, shall indicate its support (as Follower) of the UI Commands related to menus (Device Root Menu, Device Setup Menu, Contents Menu, Media Top Menu, Media Context-Sensitive Menu) in the operand [RC Profile Source] that is sent in the <Report Features> message.

It is recommended that a TV adapt its operations according to the information from [RC Profile Source]. For example, a TV can limit the items on its on-screen menu to only those that are supported by the Source Device.

11.6.6 Other uses of <User Control Pressed>

H14b Section CEC 13.13.5 is extended as follows:

The <User Control Pressed> message may also be sent in other cases which are not the direct result of a user interaction, nor directly mapped to a Remote Control button. For example, a TV might offer the user a way to access the root menu of connected devices from a menu in the TV UI. If the user selects that item in the TV UI, the TV will send a <User Control Pressed> ["Root Menu"] to the corresponding device. The Initiator (the TV in this example) shall also send the corresponding <User Control Released> message.

If a Follower is not in a state where it can action those messages, e.g. it is in Standby, then it shall send a <Feature Abort> message with an [Abort Reason] of "Not in correct mode to respond".

In order to deterministically change the power status of the target device, it is recommended to use the relevant deterministic functions 0x6D, 0x6C or 0x6B instead of ["Power"] (0x40), because the UI Command Code ["Power"] (0x40) might not have predictable behavior (see Section 11.5.3). If it is necessary to deterministically change the power status of the target device by using 0x40, then the Initiator should first enquire the Power Status of the target device by sending a <Give Device Power Status> message. In this case, if the target device is already in the desired power state, then the Initiator shall not send a <User Control Pressed> ["Power"] message. Also refer to Section 11.5 for behavior with respect to power changes, in particular Section 11.5.3.

11.7 Audio Return Channel Control

H14b Section CEC 13.17.2 is extended as follows:

An HDMI Source with bit "Source supports ARC Rx" in [Device Features] set (=1) (see Table 11-5) shall allow ARC negotiation and operation with an adjacent HDMI Sink which has allocated any Logical Address in the range 0..14.

An HDMI Sink with bit "Sink supports ARC Tx" in [Device Features] set (=1) (see Table 11-5) shall allow ARC negotiation and operation with an adjacent HDMI Source which has allocated any Logical Address in the range 1..14.

11.8 Vendor Specific Messages

H14b Section CEC 13.9.2 is extended as follows:

This feature allows a set of vendor specific commands to be used to communicate between devices.

A device that supports vendor specific commands shall store a Vendor ID. A device shall broadcast a <Device Vendor ID> message after a successful initialization and address allocation to inform all other devices of its vendor ID. A device may request the Vendor ID of another device by sending a <Give Device Vendor ID> message to it. The Follower shall respond by broadcasting a <Device Vendor ID> message if it has a Vendor ID, or reply with a <Feature Abort> message with reason "[Unrecognized Opcode]" if it does not support Vendor Specific commands. In this way any device can determine the Vendor ID of another device.

A device shall attempt to transmit a directly addressed <Vendor Command> to another device only if it has obtained or received the Vendor ID of that device and it recognizes that Vendor ID. A device shall only send a <Vendor Command> if it has previously sent a <Device Vendor ID> message.

A Follower device may accept a <Vendor Command> from an Initiator of the same Vendor ID. With the agreement of the vendors involved, it is also possible for a device to accept a <Vendor Command> from devices made by other vendors. The Follower may accept a <Vendor Command> only if the Initiator's Vendor ID matches a Vendor ID on the Follower's internal list of acceptable Vendor IDs. It should ignore all messages coming from devices with Vendor IDs which it does not recognize and should send a <Feature Abort> message with reason "[Refused]". This behavior was not allowed in Versions before 1.3a and so a device that wishes to send <Vendor Command> messages between different vendors in this way shall first discover whether the target conforms to Version 1.3a or later, by sending a <Get CEC Version> message. A Follower conforming to Version 1.3a or later and supporting such <Vendor Command> messages between different vendors shall respond with a <CEC Version> message. If the Follower responds with a CEC Version of 1.3a or later, then the Initiator device can continue by sending the required <Vendor Command>. Note that sending a <Get CEC Version> message does not need to be done every time a device wishes to send a <Vendor Command> to another device having a different Vendor ID - if the Initiator already knows the CEC Version of the target then it is not necessary to send a <Get CEC Version> message.

If an Initiator device wants to send a <Vendor Command> and it does not know the Vendor ID of the Follower device, the Initiator device shall send a <Give Device Vendor ID> message to the Follower device before it sends the <Vendor Command>. The Follower device may respond to the received <Vendor Command>. It should only respond without previously sending a <Give Device Vendor ID> message if the Follower device already knows the Vendor ID of the initiating device.

The <Vendor Command With ID> message may be broadcast as well as directly addressed. This differs from the <Vendor Command> in that the first 3 bytes of the payload carry a Vendor ID which identifies the vendor or entity which defined the command. Devices which receive the <Vendor Command With ID> and which do not accept the Vendor ID contained in the command shall ignore this command and shall respond with a <Feature Abort> if the message was directly addressed to that receiving device using reason "[Refused]" (for an unrecognized Vendor ID) or "[Unrecognized Opcode]" (if it does not support <Vendor Command With ID>).

It is possible to send vendor specific remote control commands using the <Vendor Remote Button Down> and <Vendor Remote Button Up> messages. These messages use the mechanism and timing, as described in H14b section CEC 13.13.2 Remote Control Pass Through and Section 11.6 of This Specification, for <User Control Pressed> and <User Control Released> messages. Since the operand(s) (if any) of <Vendor Remote Button Down> and <Vendor Remote Button Up> commands are dependent on the Initiator's Vendor ID, a Follower shall ignore such messages originating from Initiator devices whose Vendor ID is not known to the Follower, or from devices where the Follower does not know the semantics of the operand(s) of these messages.

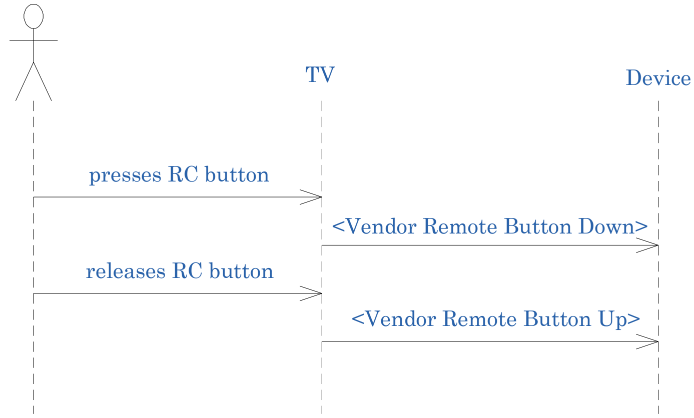


Figure 11-8: The messages sent in the Vendor Specific Commands feature *(from H14b CEC Figure 21)*

In addition it is possible to send other (non remote control key) vendor specific messages using the <Vendor Command> and <Vendor Command With ID> messages. The message parameter(s) can be used to communicate any additional (vendor defined) messages and data.

A device shall not restrict its transmission or reception of, or reaction to, CEC commands with other devices based on the Vendor ID of the other device, except for the <Vendor Command>, <Vendor Command with ID>, <Vendor Remote Button Down> and <Vendor Remote Button Up> messages.

An example of such disallowed behavior would be a TV that forwards remote control keypresses or sends opcodes such as <Play> only to devices with its own Vendor ID or other known Vendor IDs.

Additionally, a device shall not implement a <Vendor Command>, <Vendor Command with ID>, <Vendor Remote Button Down> or <Vendor Remote Button Up> message in order to replace a defined CEC command. After sending <User Control Pressed>, the <Vendor Remote Button Up> command shall not replace the corresponding <User Control Released> command.

A device may transmit a Vendor Specific command to another device in order to optimize CEC throughput (e.g. one Vendor Specific command replacing a series of standard CEC commands), or to simplify or extend a defined command, but the device shall use the defined CEC function if the Follower does not support the Vendor Specific implementation.

A device shall not withhold transmission of, or refuse to react to, a CEC command in favor of a Vendor Specific implementation.

11.9 Other topics and clarifications

11.9.1 Electrical parameters

H14b Section CEC 4 is extended as follows:

A device implementing CEC 2.0 shall:

- Conform to Table 11-12 when it is fully powered-Off (power removed). During the fully powered-Off state, the CEC line is not monitored, and the device cannot be addressed.

and

- Conform to H14b CEC Table 2 in all other power states. In these states (including the Standby state), the device shall keep monitoring the CEC line for any messages addressing that device, including any messages that bring the device out of Standby or require the device to provide a response (see Sections 11.2.2, 11.5, and H14b Section CEC 14.1.3) and act upon such messages. Also <Polling Messages> shall be ACKnowledged when the device is addressed to allow polling and Logical Address Allocation of other devices to function properly.

Table 11-12: CEC Electrical Specifications during the fully powered-Off state

Description	Value	Notes
Leakage current in fully powered-Off state	1.8μA max	1

Notes:

1 This effectively requires that the internal pull-up circuit shall be disconnected from the CEC line when the device is off. For example, this can be implemented by connecting an isolating diode between the CEC input pin and the internal pull-up circuit, such that diode is reverse-biased in the off state with an external device pulling-up the CEC line.

11.9.2 Measuring data bit timing

To determine the bit timing of the incoming signal, a Follower typically looks for the transients (edge detection) in the CEC signal. In order to allow for accurate determination of bit timings, as well as spurious pulses (see Section 11.9.3), it shall be able to determine rising and falling edges with an accuracy of ≤0.1 ms. Where a Follower uses sampling at constant interval to determine the value of the CEC signal, the sampling period shall be ≤0.1 ms.

11.9.3 Re-transmissions and errors

H14b Section CEC 7.1 is extended as follows:

A valid frame is considered lost and therefore shall be re-transmitted under the following conditions:

- If a frame is not acknowledged in a directly addressed message.
- If a frame is negatively acknowledged in a broadcast message.
- If the Initiator detects low impedance on the CEC line when it is transmitting high impedance and is not expecting a Follower asserted bit.
- If the Initiator detects the “error notification” described below.

Re-transmission can be attempted up to 3 times for a single message and shall be attempted at least once. The re-transmission shall be after the signal-free time described in H14b CEC Table 4. If the message to be re-transmitted is a <Polling Message> for a secondary task (see Section 11.4), then it is recommended to send only one re-transmission.

H14b Section CEC 7.4 is extended as follows:

If a Follower detects the existence of spurious pulses (a waveform not matching the timings from a falling to a rising edge or vice versa in H14b CEC Figure 4) on the CEC line, it shall notify all other devices (though primarily the Initiator) that a potential problem has occurred. The Follower may ignore spurious pulses with segments between the edges shorter than 0.1 ms (< 0.1 ms), consistent with the “≤ 0.1 ms” requirement for implementations using constant interval sampling (see Section 11.9.2).

An error is defined as a period between falling edges that is less than a minimum data bit period (i.e. too short to be a valid bit). Note that the start bit has different timing from normal data bits and is used to identify a valid CEC message (see H14b Section CEC 5.2.1). CEC Line Error checking shall start only after receiving a valid start bit.

Spurious pulses and errors are notified by the Follower generating a low bit period on the control signal line of 1.4-1.6 times the nominal data bit period. After such an error notification the original Initiator should stop sending its current frame and shall send a re-transmission, see above. Other devices receiving such an error notification shall not react with another error notification (in order to prevent endless loops of error notification signals).

11.9.4 Protocol Extensions

H14b Section CEC 8 is extended as follows:

In order to allow for extensions to the protocol in future releases of the specification, the current opcodes and parameters can be extended by adding further parameters (following the existing operands) onto them. If a CEC device receives a message with more operands than expected, it shall ACK the additional operands and shall ignore these additional operands, thus allowing future extensions to already existing commands. Followers shall interpret the expected (non-additional) operands normally.

For entirely new commands, new opcodes can be allocated in a future version of This Specification.

For the avoidance of doubt, the allocation of further operands, new opcodes and new addresses shall only be done by the HDMI Forum.

Note – for the commands <Vendor Command>, <Vendor Command with ID> and <Vendor Remote Button Down>, only the registered owner of the [Vendor ID] may define the operands – see Section 11.8.

11.9.5 Message response timing

H14b Section CEC 9.2 is extended as follows:

There are certain time constraints for messages to be obeyed at the application level. Devices should respond within 200ms and shall respond within the required maximum response time of 1 second. This response time is measured from the end of an incoming message, to the start of the transmission of the reply message.

This implies that the original Initiator might need to wait as long as 1.4 seconds for the response to be fully received, as it might take up to about 400 ms to send a message of maximum length – and possibly slightly more when there is other bus traffic.

In the following specific cases, the maximum response time of 1 second might be exceeded:

- When a new Source is selected using <Set Stream Path>, the <Active Source> response might be delayed until video is available (described in further detail in H14b Sections CEC 13.1.2 and CEC 13.2.2)
- When a Recording Device receives <Record On>, it might take several seconds to send an accurate <Record Status> (described in further detail in H14b Section CEC 13.4)

If a device cannot reply within the required timeout due to traffic on the CEC bus, it should send the response at the earliest opportunity. Unless it is explicitly described in the HDMI 1.4b Specification or in This Specification, it is up to the follower to accept or to ignore the response received after the response receiving timeout is expired.

11.9.6 Source Declaration

H14b Section CEC 12.1 is extended as follows:

For a device to act as a Source Device, it shall issue an <Active Source> message to declare its intention. Thus any presently active source shall act appropriately, see Section 11.9.9, and H14b Sections CEC 13.1.2 and CEC 13.2.2.

11.9.7 Protocol General Rules

The sixth paragraph (including the five point bullet list that follows) of H14b Section CEC 12.2 is extended as follows:

A Follower shall ignore a message coming from address 15 (unregistered) when it would require a directly addressed response, or cause the Follower to direct messages to a device at the unregistered address at a later time as a result.

11.9.8 Feature Abort

The first paragraph of H14b Section CEC 12.3 is extended as follows:

All devices shall support the message <Feature Abort>. If a device does not support the opcode of a directly addressed message that it has received or it is unable to deal with the message at present, or if there was something wrong with the transmitted frame at the high-level protocol layer, it shall respond with a <Feature Abort> message with the appropriate [Abort Reason].

The text in the last two paragraphs of H14b Section CEC 12.3 is extended as follows:

If the [Abort Reason] was anything other than "Unrecognized opcode", the Initiator may send the message again. It is recommended that it waits for at least 200ms in order to allow time for the Follower to recover from the state that caused the initial <Feature Abort> message.

A device shall not respond to a <Feature Abort> message with another <Feature Abort> message, in order to prevent endless sequences of such messages.

Note: <Feature Abort> is also used as a response to the <Abort> message during testing, see H14b Section CEC 12.4.

11.9.9 Routing Control

The 6th, 7th, and 8th paragraphs of H14b Section CEC 13.2.2 are extended as follows:

The user may select a device to view via the TV user interface. In contrast to the <Active Source> message (which is sent by the current active source to the TV), the <Set Stream Path> message is sent by the TV to the source device to request it to broadcast its path using an <Active Source> message. In this case, the TV should broadcast a <Set Stream Path> message with the Physical Address of the device it wants to display as a parameter. Any CEC Switches between the device and TV shall switch (if required) to ensure the device is on the active AV path. CEC Switches shall not send a <Routing Change> message in this case. This feature also ensures that non-CEC-compliant devices in the network can be switched to, if for instance they have been manually set up in the TV menu. A CEC device at the location specified by the <Set Stream Path> message shall come out of the Standby state (if necessary). If and when it has stable video to display, it shall broadcast an <Active Source> message and begin streaming its output.

Note: there is a special case when a TV switches to its internal tuner or to another non-HDMI source (e.g. Y/C, or a SCART socket on European market sets). In this case, it is the TV which broadcasts the <Active Source> message with address 0.0.0.0.

When the user has specifically sent the currently active device only to the Standby state (e.g. as the result of a user action using the device's local control, such as its own remote controller), it shall send an <Inactive Source> message with its own Physical Address as an operand. It is a manufacturer's decision to decide the TV's response: it may, for example, display its own internal tuner, or select another device for display. In these cases, the TV shall send a new <Active Source> message with its own Physical Address (0.0.0.0, when displaying its own internal tuner), or send a <Set Stream Path> message to a new device for display. An <Inactive Source> message shall also be sent when the Source Device has no correct HDMI video signal to be presented to the user, even if the device is not in the Standby state.

The 10th paragraph of H14b Section CEC 13.2.2 is extended as follows:

If a CEC Switch is at the new position indicated by the [New Address] operand of the <Routing Change> message then it shall broadcast a <Routing Information> message with the Physical Address of its current active path. If a CEC Switch is at the new position indicated by the operand of the <Routing Information> message then it shall broadcast a <Routing Information> message with the Physical Address of its current active path (input). In this way the all CEC Switches are aware of the route to the new source and the last <Routing Information> message contains the complete route (address) to the new selected source.

The 12th and following paragraphs of H14b Section CEC 13.2.2 are extended as follows:

A TV (when it is the CEC Root Device at Physical Address 0.0.0.0) shall not implement the <Routing Information> message as an Initiator.

Optionally, if the TV detects that the active source device has been de-selected by changing the Switch it may either switch to an internal service or may send a <Set Stream Path> message to the device at the new location to indicate that it should become the new active source. In this case, the TV shall wait for a minimum of 7 nominal data bit periods and a recommended maximum of 500ms before reacting to a <Routing Change> or <Routing Information> message to allow CEC Switches to relay any <Routing Information> messages that are required.

The following diagram (Figure 11-9) shows an example of the message flow when a user manually switches a CEC Switch. (CEC Switches are shown filled). In the example illustrated in Figure 11-9, the TV would receive the various <Routing Change> and <Routing Information> messages, and then (after the recommended waiting time of up to 500 ms, see above) send <Set Stream Path> [1.2.1.1]. The device at that address would then become active and send <Active Source> [1.2.1.1] after it has stable video.

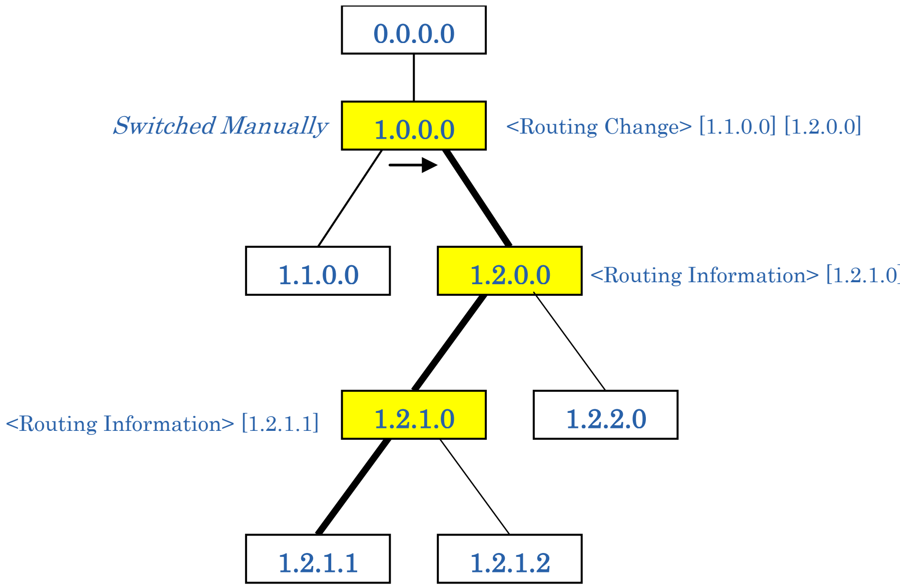


Figure 11-9: Example message flow, when a CEC Switch is manually switched *(from H14b CEC Figure 12)*

11.9.10 Device OSD Name Transfer

H14b Section CEC 13.11.2 is extended as follows:

This feature is used to request the preferred name of a device to be used in any on-screen display (e.g. menus), which reference that device. A device (e.g. the TV) may request another device's name by sending a directly addressed <Give OSD Name> message to it. The device shall respond with a <Set OSD Name> message. The device's name should then be used in on-screen references to it.

A device that has more than one Logical Address (see Section 11.3.2 and Table 11-8, e.g. Audio System with integrated Playback Device) shall respond with the same [OSD Name] for each Logical Address. It is recommended that the [OSD Name] refers to the complete physical product, rather than the individual CEC functionality, in order to avoid user confusion. It is manufacturer dependent how the individual CEC functionalities (e.g. the Audio System and the Playback Device in the above example) are presented to the user.

A TV with OSD/Menu generation capabilities shall send a <Give OSD Name> message whenever it discovers a new device that has been connected.

11.9.11 System Audio Control

The first paragraph of H14b Section CEC 13.15.2 is extended as follows:

This feature allows an Amplifier to render the audio for a source that is being displayed on a TV. When in this mode, the Amplifier uses the same source for audio, as the TV is using for video and provides the volume control and audio rendering functions, instead of the TV, which mutes its speakers. Also, the remote control commands related to audio rendering (e.g. volume +/- and mute buttons) from all devices shall be sent to the device that provides the audio rendering (see last part of Section 11.6.4).

The roles described as 'TV' and 'Amplifier' in this feature shall only be assumed by devices that have successfully been allocated Logical Addresses 0 and 5, respectively.

The paragraph above H14b CEC Figure 31 of H14b Section CEC 13.15.2 is extended as follows:

When the System Audio Mode is On, the Amplifier renders the audio and the Amplifier's volume can be set using the volume control of the Amplifier or other devices which have a volume control, such as the TV or a STB, using either the relevant user remote control or local controls on the device (e.g. physical Volume + / - keys or a rotary style control). Similarly, the mute status of the Amplifier can be controlled by the relevant "mute" remote control button (or other controls) of the various devices.

H14b CEC Figure 32 of H14b Section CEC 13.15.2 is clarified as follows:

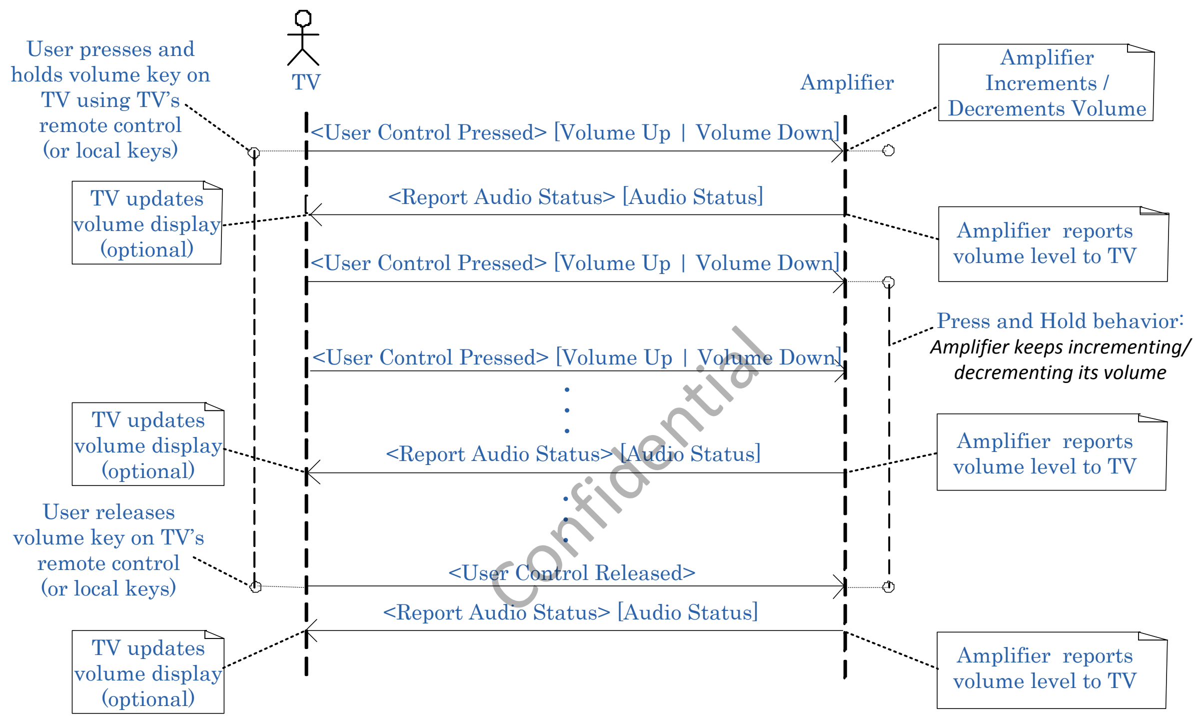


Figure 11-10: An example of TV and Amplifier implementing Press and Hold behavior *(Clarified from H14b CEC Figure 32)*

11.9.11.1 Reporting Audio Status

The 4th and following paragraphs after H14b CEC Figure 32 of H14b Section CEC 13.15.2 are extended as follows:

The <Give Audio Status> and <Report Audio Status> messages are mainly used so that the TV can display the audio status of the external Amplifier, for instance the current Mute status or a Volume level display. The <Give Audio Status > message is used to ask for the current audio status of a target Amplifier. The target device shall respond by sending a <Report Audio Status> message containing the Audio Status operand back to the device which sent the <Give Audio Status>.

After the relevant <User Control Pressed> message has been sent to adjust the volume, the Amplifier shall send <Report Audio Status> messages so that the TV may display updated volume indication as the volume changes. In this case, it is not recommended to send a <Report Audio Status> message more frequently than once every 500ms.

When the Amplifier is muted or unmuted, it shall send a <Report Audio Status> message so that the TV may display the updated mute status.

An Amplifier that does not have electronic control of volume or mute is excluded from the above requirements to send <Report Audio Status>.

11.9.11.2 System Audio Mode and Volume Control

The first paragraph on page CEC-60 of H14b Section CEC 13.15.2 is extended as follows:

While System Audio Mode is On:

- the TV or source shall not change their own internal volume levels;
- the Amplifier's local and remote controls shall also be active and able to control its volume.

When the System Audio Mode is Off, the TV renders the audio and the TV's volume can be set using the volume control of the TV or other devices which have volume control means, such as the STB or Amplifier, using either the relevant user remote control or local controls on the device (e.g. physical Volume + / - buttons or a rotary style control). Similarly, the mute status of the TV can be controlled by the relevant "mute" remote control button (or other controls) of the various devices.

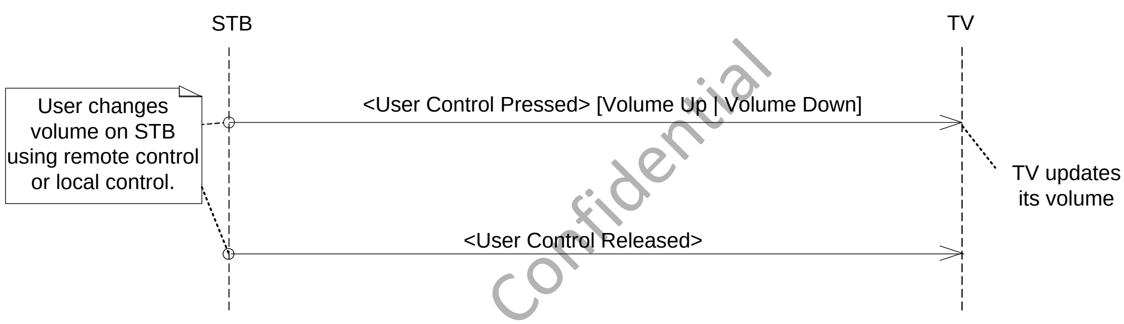


Figure 11-11: A Typical Operation of the volume control where the user presses and quickly releases a button

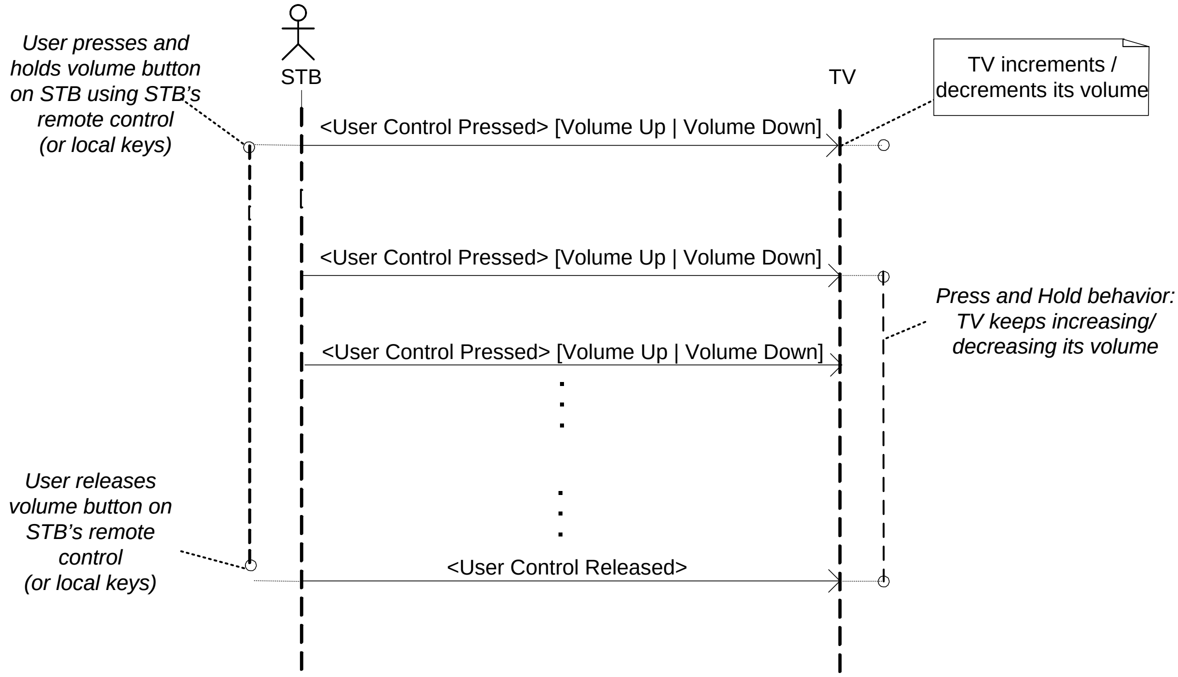


Figure 11-12: An example of TV and STB implementing Press and Hold behavior

Whenever the volume is changed by one of the above methods and the System Audio Mode is Off, the device that received the User's volume commands sends out a <User Control Pressed> with the relevant ["Volume Up"] or ["Volume Down"] operand to the TV. When the User releases the control, the device sends a <User Control Released> message to the TV. For further information on the User Control messages, press and hold, timing, etc., see Section 11.6.

Note that the behavior of the volume function will be determined by the implementation of the TV's volume control.

When the user wants to mute or unmute the TV's speakers while the System Audio Mode is Off and presses the "mute" button of a device's remote, the device (such as a STB or Amplifier) sends a <User Control Pressed> message with an operand of ["Mute"]. The behavior of this ["Mute"] message is determined by the TV. Alternatively, the device (such as a STB or Amplifier) may send a <User Control Pressed> message with an operand of ["Mute Function"] or ["Restore Volume Function"] (see Section 11.9.11.4 for further information).

Since the volume/mute controls of the device are forwarded to the TV, the device shall not change its own volume on the audio stream due to these volume/mute controls (and shall not mute the audio in the stream to the TV).

When System Audio Mode is off, the messages <Give Audio Status> and <Report Audio Status> are not used since the TV is aware of its own volume and can update the volume OSD autonomously without requiring informational messages from the Amplifier.

11.9.11.3 Discovering the Amplifier's Audio Format support

The second paragraph and H14b CEC Figure 34 of H14b Section CEC 13.15.3 are extended as follows:

The TV may also enquire if an Amplifier supports multiple audio formats by using one <Request Short Audio Descriptor> message, up to a maximum of 4 formats. In this case, the Amplifier responds with a <Report Short Audio Descriptor> message indicating which of the audio formats it supports, from the list in the corresponding <Request Short Audio Descriptor> message. If the Amplifier supports none of the requested formats, it shall respond with a <Feature Abort> ["Invalid Operand"] message.

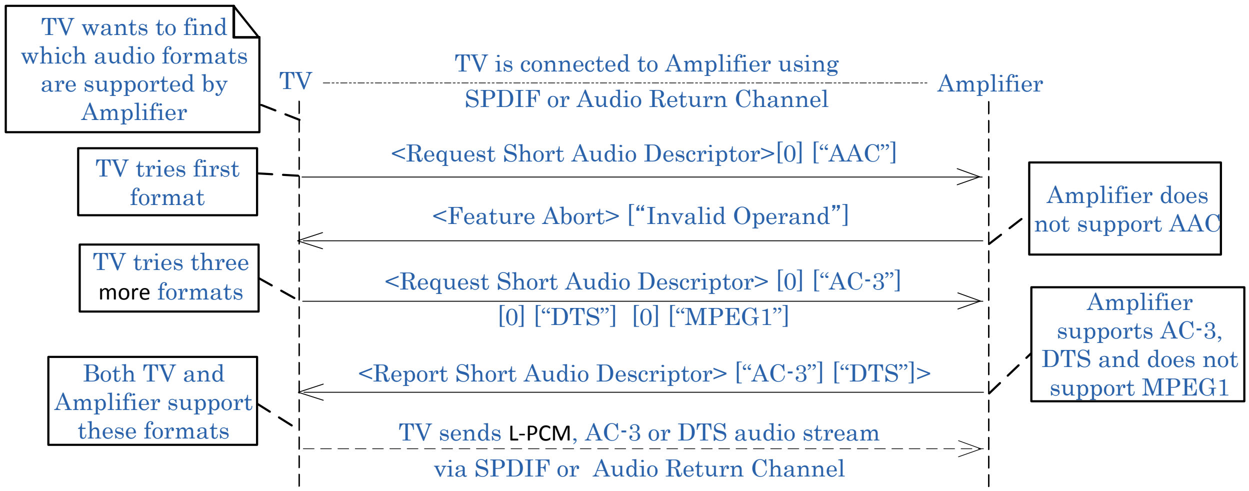


Figure 11-13: Typical Operation to discover the Audio Format capability of an Amplifier *(Clarified from H14b CEC Figure 34)*

11.9.11.4 Usage of remote control pass through

H14b Section CEC 13.15.4.5 is extended as follows:

When a device such as TV or STB offers a deterministic mute control mechanism and the user operates this mechanism in order to deterministically mute or unmute the Amplifier's speakers while the System Audio Mode is On, the device (such as a TV or STB) sends a <User Control Pressed> message with an operand of ["Mute Function"] or ["Restore Volume Function"]. Audio renderers such as the TV and the Amplifier shall support a <User Control Pressed> message with an operand of ["Mute"], ["Mute Function"] and ["Restore Volume Function"], see Table 11-31.

If the System Audio Mode is Off and the Amplifier receives a volume control (i.e. Volume Up, Volume Down or Mute) from its own remote control or local keypresses, it is up to Amplifier manufacturer's implementation to either forward the keypresses to the TV or to use this as a trigger to initiate System Audio Mode.

If a device such as a STB with volume control means receives its own remote control or local key keypresses for volume control, it shall forward such keypresses either to the Amplifier or to the TV, depending on whether System Audio Mode is On (send to Amplifier) or Off (send to TV).

Also see the last paragraphs of Section 11.6.4.

11.10 Message tables

H14b Section CEC 15 is extended as follows:

This section defines the individual messages used in CEC. It describes them and defines Messages and their parameters and expected responses. As CEC has no session layer, this section and the operands section (Section 11.12) effectively define the complete messaging system. Section 11.2.2.1 and Table 11-13 through Table 11-29 (see last two columns in those Tables) show which messages are mandatory. If a manufacturer implements any of the messages, then they shall be implemented as described in This Specification.

For devices that have taken Logical Address 15 (and have not allocated another Logical Address), the same exceptions as listed for Pure CEC Switches in Table 11-13 to Table 11-29 shall apply.

Some messages appear in multiple tables, as they are used by multiple features. If a message is mandatory according to one of the features or tables, and optional according to another feature or table, it is mandatory.

For requirements, messages, and operands not mentioned in This Specification see H14b Section CEC 15 through H14b Section CEC 17 and H14b Table CEC 8 through H14b CEC Table 28.

Table 11-13: Message Descriptions for the Routing Control Feature

H14b CEC Table 9 is extended as follows (note: messages for which the description is unchanged from HDMI 1.4b CEC Table 9 (i.e. <Active Source>, <Routing Change>, <Routing Information>) are not shown here – see HDMI 1.4b for those details):

Opcode	value	Description	Parameters	Parameter description	Response	Addressing		Mandatory for	
						Direct	Broadcast	Initiator	Follower
<Inactive Source>	0x9D	Used by the currently active source to inform the TV that it has no video to be presented to the user, or is going into the Standby state as the result of a local user command on the device.	[Physical Address]	The Physical Address of the device.	The TV may display its own internal tuner and shall send an <Active Source> with the address of the TV; or The TV may send <Set Stream Path> to another device for display.	•		All Sources	TV
<Request Active Source>	0x85	Used by a new device to discover the status of the system.	None		<Active Source> from the currently active source.		•		All, except for Pure CEC Switches and devices which cannot become a source.

Opcode	value	Description	Parameters	Parameter description	Response	Addressing		Mandatory for	
						Direct	Broadcast	Initiator	Follower
<Set Stream Path>	0x86	Used by the TV to request a streaming path from the specified Physical Address.	[Physical Address]	The Physical Address of the source device.	Any CEC Switches between the TV and the source device shall switch inputs according to the path defined in [Physical Address]. A CEC device at the new address shall come out of the Standby state, stream its output and broadcast an <Active Source> message.		•	TV with device selection menu	All Sources, CEC Switches

Table 11-14: Message Descriptions for the Standby Feature

H14b CEC Table 10 is extended as follows:

Opcode	value	Description	Parameters	Parameter description	Response	Addressing		Mandatory for	
						Direct	Broadcast	Initiator	Follower
<Standby>	0x36	Switches one or all devices into the Standby state. Can be used as a broadcast message or be addressed to a specific device. See Section 11.5.6 for important notes on the use of this message	None		Switch the device into the Standby state. [1] Ignore the message if already in the Standby state.	•	•	TV (sending as broadcast message)	All (receiving broadcast and directed message)

[1] Can be ignored if actively engaged in a recording or providing a source stream for a recording. See also Section 11.5.6 for other exceptions.

Table 11-15: Message Descriptions for the One Touch Record Feature

H14b CEC Table 11 is extended as follows:

Opcode	value	Description	Parameters	Parameter description	Response	Addressing		Mandatory for	
						Direct	Broadcast	Initiator	Follower
<Record Off>	0x0B	Requests a device to stop a recording.	None		Exit 'Recording' state.	•		Devices sending <Record On>	Recording Device
<Record On>	0x09	Attempt to record the specified source.	[Record Source]	Source to record, either analogue service, digital service, external source or own source (i.e. currently selected source).	Enter 'Recording' state and start recording if possible. Send the Initiator <Record Status>.	•		TV[1]	Recording Device
<Record Status>	0x0A	Used by a Recording Device to inform the Initiator of the message <Record On> about its status.	[Record Status Info]	The recording status of the device.		•		Recording Device	Devices sending <Record On>
<Record TV Screen>	0x0F	Request by the Recording Device to record the presently displayed source.	None		Initiate a recording using the <Record On> message, or send a <Feature Abort> ["Cannot provide source"] if the presently displayed source is not recordable.	•			TV

[1] If bit "TV supports <Record TV Screen>" is set (=1) in [Device Features]

Table 11-16: Message Descriptions for the System Information Feature

H14b CEC Table 13 is extended as follows (note: messages for which the description is unchanged from HDMI 1.4b CEC Table 13 (i.e. <Get Menu Language>) are not shown here – see HDMI 1.4b for those details):

Opcode	value	Description	Parameters	Parameter description	Response	Addressing		Mandatory for	
						Direct	Broadcast	Initiator	Follower
<CEC Version>[1]	0x9E	Used to indicate the version number of the CEC Specification which was used to design the device, in response to a <Get CEC Version>.	[CEC Version]	A value indicating the version number of the CEC Specification which was used to design the device.		•		All except Pure CEC Switches[2]	Devices that send <Get CEC Version>
<Get CEC Version>	0x9F	Used by a device to enquire which version number of the CEC Specification was used to design the Follower device.	None		The source responds with a <CEC Version> message indicating the version number of the CEC Specification which was used to design the Follower device.	•			All except Pure CEC Switches
<Give Physical Address>	0x83	A request to a device to return its Physical Address.	None		<Report Physical Address>	•			All, except for Pure CEC Switches
<Polling Message>	-	Used by any device for device discovery – similar to ping in other protocols.	None		Shall set a low level ACK.	•		All except for Pure CEC Switches	All except for Pure CEC Switches

[1] This message is also used in the Vendor Specific Command Feature

[2] Also except for devices that have taken Logical Address 15 (and have not allocated another Logical Address)

Opcode	value	Description	Parameters	Parameter description	Response	Addressing		Mandatory for	
						Direct	Broadcast	Initiator	Follower
<Report Physical Address>	0x84	Used to inform all other devices of the mapping between Physical and Logical Address of the Initiator.	[Physical Address] [Primary Device Type]	The device's Physical Address within the cluster, and its primary (main) device type.			•	All	TV
<Set Menu Language>	0x32	Used by a TV to indicate its currently selected menu language.	[Language]	The TV's current menu language.	Set the menu language as specified, if possible.		•	TV with OSD / Menu generation capabilities	All [1]
<Report Features>	0xA6	Used by a device to announce its version, features and device type(s)	[CEC Version] [All Device Types] [RC Profile] [Device Features]	The CEC version, flags on certain features and all the device types of a device. Note operands [RC Profile] and [Device Features] are variable length			•	All	
<Give Features>	0xA5	Used by a device to request another device's features	None		<Report Features>	•			All except for Pure CEC Switches[2]

[1] <Set Menu Language> is Mandatory as a Follower except for the following: TV, CEC Switches, Mobile Devices, other devices which are not able to change the language by CEC messages, e.g. a PC, and devices without OSD/ Menu generation capabilities.
[2] Also except for devices that have taken Logical Address 15 (and have not allocated another Logical Address).

Table 11-17: Message Descriptions for the Deck Control Feature

H14b CEC Table 14 is extended as follows:

Opcode	value	Description	Parameters	Parameter description	Response	Addressing		Mandatory for	
						Direct	Broadcast	Initiator	Follower
<Deck Control>	0x42	Used to control a device's media functions.	[Deck Control Mode]	The deck control requested.	Perform the specified actions, or return a <Feature Abort> message. It is device dependent whether or not a Skip Forward/Wind or Skip Backward /Rewind command is legal when in the 'Deck Inactive' state. If the device is in the Standby state and it receives an eject command, it should power on and eject its media.	•			Playback / Recording Device[1]
<Deck Status>	0x1B	Used to provide a deck's status to the Initiator of the <Give Deck Status> message.	[Deck Info]	Information on the device's current status.		•		Playback / Recording Device	
<Give Deck Status>	0x1A	Used to request the status of a device, regardless of whether or not it is the current active source.	[Status Request]	Allows the Initiator to request the status once or on all future state changes. Or to cancel a previous <Give Deck Status> ["On"] request.	<Deck Status>	•			Playback / Recording Device

[1] If bit "supports being controlled by Deck Control" is set (=1) in [Device Features]

Opcode	value	Description	Parameters	Parameter description	Response	Addressing		Mandatory for	
						Direct	Broadcast	Initiator	Follower
<Play>	0x41	Used to control the playback behavior of a source device.	[Play Mode]	Play mode required.	Perform the specified actions, or return a <Feature Abort> message. If media is available the device enters 'Deck Active' state. If the device is in the Standby state, has media available and the parameter is ["Play Forward"] it should power on.	•			Playback / Recording Device[1]

[1] If bit "supports being controlled by Deck Control" is set (=1) in [Device Features]

Table 11-18: Message Descriptions for the Vendor Specific Commands Feature

H14b CEC Table 16 is extended as follows (note: messages for which the description is unchanged from HDMI 1.4b CEC Table 16 (i.e. <Device Vendor ID>, <Give Device Vendor ID>, <Vendor Command>, <Vendor Command With ID>, <Vendor Remote Button Down>, <Vendor Remote Button Up>) are not shown here – see HDMI 1.4b for those details):

Opcode	value	Description	Parameters	Parameter description	Response	Addressing		Mandatory for	
						Direct	Broadcast	Initiator	Follower
<CEC Version>[1]	0x9E	Used to indicate the version number of the CEC Specification which was used to design the device, in response to a <Get CEC Version>	[CEC Version]	A value indicating the version number of the CEC Specification which was used to design the device.		•		All except for Pure CEC Switches[2]	Devices that send <Get CEC Version>
<Get CEC Version>	0x9F	Used by a device to enquire which version number of the CEC Specification was used to design the Follower device.	None		The source responds with a <CEC Version> message indicating the version number of the CEC Specification which was used to design the Follower device.	•		All devices that want to initiate a scenario with devices of specific other vendors using the <Vendor Command> message	All except for Pure CEC Switches

[1] This message is also used in the System Information Feature

[2] Also except for devices that have taken Logical Address 15 (and have not allocated another Logical Address)

Table 11-19: Message Descriptions for the OSD Display Feature

H14b CEC Table 17 is extended as follows:

Opcode	value	Description	Parameters	Parameter description	Response	Addressing		Mandatory for	
						Direct	Broadcast	Initiator	Follower
<Set OSD String>	0x64	Used to send a text message to output on a TV.	[Display Control] [OSD String]	Display timing. Text to be displayed.	TV displays the message.	•			TV[1]

Table 11-20: Message Descriptions for the Device OSD Name Transfer Feature

H14b CEC Table 18 is extended as follows:

Opcode	value	Description	Parameters	Parameter description	Response	Addressing		Mandatory for	
						Direct	Broadcast	Initiator	Follower
<Give OSD Name>	0x46	Used to request the preferred OSD name of a device for use in menus associated with that device.	None		<Set OSD Name>	•		TV with OSD / Menu generation capabilities	All except for TV and Pure CEC Switches[2]
<Set OSD Name>	0x47	Used to set the preferred OSD name of a device for use in menus associated with that device.	[OSD Name]	The preferred name of the device.	Store the name and use it in any menus associated with that device.	•		All except for TV and Pure CEC Switches	TV with OSD / Menu generation capabilities

[1] If bit "TV supports <Set OSD String>" is set (=1) in [Device Features]

[2] Also except for devices that have taken Logical Address 15 (and have not allocated another Logical Address)

Table 11-21: Message Descriptions for the Remote Control Pass Through Feature

H14b CEC Table 20 is extended as follows:

Opcode	value	Description	Parameters	Parameter description	Response	Addressing		Mandatory for	
						Direct	Broadcast	Initiator	Follower
<User Control Pressed> [1]	0x44	Used to indicate that the user pressed a remote control button or switched from one remote control button to another. Can also be used as a command that is not directly initiated by the user.	[UI Command], plus any necessary Additional Operands specified in H14b CEC Table 6 and H14b CEC Table 7.	Required UI command.	Update display or perform an action, as required.	•		All devices that have a remote control[2]	All except Pure CEC Switches[3] (see also Table 11-31)
<User Control Released> [1]	0x45	Indicates that user released a remote control button (the last one indicated by the <User Control Pressed> message). Also used after a command that is not directly initiated by the user.	None		Update display or perform an action, as required.	•		All devices that have a remote control	All except Pure CEC Switches (see also Table 11-31)

[1] This message is also used in the Device Menu Control and System Audio Features

[2] If at least one button on the remote control is not always needed for the own control of the device

[3] Also except for devices that have taken Logical Address 15 (and have not allocated another Logical Address)

Table 11-22: Message Descriptions for the Power Status Feature

H14b CEC Table 21 is extended as follows:

Opcode	value	Description	Parameters	Parameter description	Response	Addressing		Mandatory for	
						Direct	Broadcast	Initiator	Follower
<Give Device Power Status>	0x8F	Used to determine the current power status of a target device	None		<Report Power Status>	•			All except Pure CEC Switches[1]
<Report Power Status>	0x90	Used to inform a requesting device of the current power status	[Power Status]	The current power status		•	•	All except Pure CEC Switches	

[1] Also except for devices that have taken Logical Address 15 (and have not allocated another Logical Address).

Table 11-23: Message Descriptions for General Protocol messages

H14b CEC Table 22 is extended as follows:

Opcode	value	Description	Parameters	Parameter description	Response	Addressing		Mandatory for	
						Direct	Broadcast	Initiator	Follower
<Feature Abort>	0x00	Used as a response to indicate that the device does not support the requested message type, or that it cannot execute it at the present time.	[Feature Opcode] [Abort Reason]	The Opcode of the aborted message. The reason provides an indication as to whether the Follower does not support the message, or does support the message but cannot respond at the present time.	Assume that request is not supported or has not been actioned.	•		All except for Pure CEC Switches[1] (Generate if a message is not supported)	All, except for Pure CEC Switches
<Abort> Message	0xFF	This message is reserved for testing purposes.	None		A device shall always respond with a <Feature Abort> message containing any valid value for [Abort Reason]. Pure CEC Switches shall not respond to this message.	•			All, except for Pure CEC Switches

[1] Also except for devices that have taken Logical Address 15 (and have not allocated another Logical Address)

Table 11-24: Message Descriptions for the System Audio Control Feature

H14b CEC Table 23 is extended as follows:

Opcode	value	Description	Parameters	Parameter description	Response	Addressing		Mandatory for	
						Direct	Broadcast	Initiator	Follower
<Give Audio Status>	0x71	Requests an Amplifier to send its volume and mute status	None		<Report Audio Status>	•			Audio System[1]
<Give System Au dio Mode Status>	0x7D	Requests the status of the System Audio Mode	None		Amplifier sends a <System Audio Mode Status> message indicating status (On or Off)	•			Audio System
<Report Audio Status>	0x7A	Reports an Amplifier's volume and mute status	[Audio Status]	Volume and mute status		•		Audio System	
<Report Short Audio Descriptor>	0xA3	Report Audio Capability	[Short Audio Descriptor]	Up to 4 Short Audio Descriptor(s) identifying supported audio format(s)		•			
<Request Short Audio Descriptor>	0xA4	Request Audio Capability	[Audio Format ID and Code]	Up to 4 [Audio Format ID and Code] (s) (if needed)	<Report Short Audio Descriptor>	•			

[1] Except Audio System that has no electronic control of volume/mute

Opcode	value	Description	Parameters	Parameter description	Response	Addressing		Mandatory for	
						Direct	Broadcast	Initiator	Follower
<Set System Audio Mode>	0x72	Turns the System Audio Mode On or Off.	[System Audio Status]	Specifies if the System Audio Mode is On or Off.	If set to On, the TV mutes its speakers. The TV or STB sends relevant <User Control Pressed> or <User Control Released> as necessary. If set to Off, the TV unmutes its speakers. The TV or STB stop sending the volume-related <User Control Pressed> or <User Control Released> messages.	•	•	Audio System	TV, devices that can send <System Audio Mode Request>
<System Audio Mode Request>	0x70	A device implementing System Audio Control requests to use System Audio Mode to the Amplifier	[Physical Address]	Source to be used is the device specified at this address.	The Amplifier comes out of the Standby state (if necessary) and switches to the relevant connector for device specified by [Physical Address]. It then sends a <Set System Audio Mode> [On] message. <System Audio Mode Request> sent without a [Physical Address] parameter requests termination of the feature. In this case, the Amplifier sends a <Set System Audio Mode> [Off] message.	•		TV	Audio System

Opcode	value	Description	Parameters	Parameter description	Response	Addressing		Mandatory for	
						Direct	Broadcast	Initiator	Follower
<System Audio Mode Status>	0x7E	Reports the current status of the System Audio Mode	[System Audio Status]	Current system Audio Mode	If [On], the device requesting this information can send the volume-related <User Control Pressed> or <User Control Released> messages.	•		Audio System	TV, devices that can send <System Audio Mode Request>
<User Control Pressed>[1]	0x44	Used to indicate that the user pressed a remote control button or switched from one remote control button to another. Can also be used as a command that is not directly initiated by the user.	[UI Command] of "Volume Up", "Volume Down" or "Mute", "Mute Function", "Restore Volume Function".	Relevant UI command issued by user.	Increase or Decrease the volume of the Amplifier, or mute/unmute the Amplifier.	•		TV, Audio System, Generic Sources with volume / mute remote control functions	Audio System, TV
<User Control Released>	0x45	Indicates that user released a remote control button (the last one indicated by the <User Control Pressed> message). Also used after a command that is not directly initiated by the user.	None		Stop increasing or decreasing the volume	•		TV, Audio System, Generic Sources with volume / mute remote control functions	Audio System, TV

[1] This message is also used in the Device Menu Control and RC Passthrough Features

Table 11-25: Message Descriptions for the Audio Rate Control Feature

H14b CEC Table 24 is extended as follows:

Opcode	value	Description	Parameters	Parameter description	Response	Addressing		Mandatory for	
						Direct	Broadcast	Initiator	Follower
<Set Audio Rate>	0x9A	Used to control audio rate of Source Device.	[Audio Rate]	The audio rate requested.	Source adapts audio rate.	•			Generic Source [1]

Table 11-26: Message Descriptions for the Audio Return Channel Control Feature

H14b CEC Table 25 is extended as follows:

Opcode	value	Description	Parameters	Parameter description	Response	Addressing		Mandatory for	
						Direct	Broadcast	Initiator	Follower
<Initiate ARC>	0xC0	Used by an ARC Rx device to activate the ARC functionality in an ARC Tx device.	None		The ARC functionality in the ARC Tx device is activated	•		ARC Rx device[2]	ARC Tx device[3]
<Report ARC Initiated>	0xC1	Used by an ARC Tx device to indicate that its ARC functionality has been activated.	None			•		ARC Tx device	ARC Rx device
<Report ARC Terminated>	0xC2	Used by an ARC Tx device to indicate that its ARC functionality has been deactivated.	None			•		ARC Tx device	ARC Rx device

[1] If bit "Source supports <Set Audio Rate>" is set (=1) in [Device Features]
[2] The device shall also set (=1) the "Source supports ARC Rx" bit in [Device Features]
[3] The device shall also set (=1) the "Sink supports ARC Tx" bit in [Device Features]

<Request ARC Initiation>	0xC3	Used by an ARC Tx device to request an ARC Rx device to activate the ARC functionality in the ARC Tx device.	None		ARC Rx device sends an <Initiate ARC> message	•			ARC Rx device[1]
<Request ARC Termination>	0xC4	Used by an ARC Tx device to request an ARC Rx device to deactivate the ARC functionality in the ARC Tx device.	None		ARC Rx device sends a <Terminate ARC>	•			ARC Rx device
<Terminate ARC>	0xC5	Used by an ARC Rx device to deactivate the ARC functionality in an ARC Tx device.	None		The ARC functionality in the ARC Tx device is deactivated	•		ARC Rx device	ARC Tx device[2]

[1] The device shall also set (=1) the "Source supports ARC Rx" bit in [Device Features]
[2] The device shall also set (=1) the "Sink supports ARC Tx" bit in [Device Features]

Table 11-27: Message Descriptions for the Dynamic Auto Lipsync feature

This feature is described in Section 10.7

Opcode	value	Description	Parameters	Parameter description	Response	Addressing		Mandatory for	
						Direct	Broadcast	Initiator	Follower
<Request Current Latency>	0xA7	Used by Amplifier (or other device) to request current latency values	[Physical Address]		The device at target Physical Address sends <Report Current Latency> with current values		•	Amplifier Repeater	TV Repeater
<Report Current Latency>	0xA8	Used by TV (or other Initiator) to report updates of latency	[Physical Address] [Video Latency] [Latency Flags] [Audio Output Delay][1]	Current video latency and related flags	The Amplifier (or other device) uses the reported values		•	TV Repeater	Amplifier Repeater

11.11 Message Dependencies

The following additions are made to H14b CEC Table 27:

Table 11-28: Message dependencies when receiving a message

If device does not <Feature Abort> the following message with "Unrecognized opcode" :	It shall not <Feature Abort> the following message(s) with "Unrecognized opcode":	It shall be able to send the message(s):
<Give Features>	-	<Report Features>
<Request Current Latency>	-	<Report Current Latency>

The following additions are made to H14b CEC Table 28:

[1] Operand [Audio Output Delay] is only present when [Audio Output Compensated] (part of [Latency Flags])=3

Table 11-29: Message dependencies when sending a message

If device ever sends the following message:	It shall be able to send the message(s):	It shall not <Feature Abort> the following message(s) with "Unrecognized opcode":
<Report Current Latency>	-	<Request Current Latency>
<Report Features>	-	<Give Features>

11.12 Operand Descriptions

The following extensions are made to H14b CEC Table 29.

Notes:

- The HDMI 1.4b operand [Device Type] has been extended as [Primary Device Type].
- Operands that are unchanged from HDMI 1.4b are not shown here – see HDMI 1.4b for those Operands.
- The Dynamic Auto Lipsync feature defined in Section 10.7 also uses CEC messages (see Table 10-27); the operands for those messages are defined in Table 10-28.
- The operand [Audio Format ID and Code] is extended as defined in Table 9-1 to accommodate additional audio formats (Section 9.1).

Table 11-30: Operand Descriptions

Name	Range Description		Length	Purpose
[All Device Types]	TV	Bit 7	1 byte	Default=0; set bit to 1 for each Device Type that the device supports, including the device's Primary Device Type (which is also indicated in [Primary Device Type] operand of <Report Physical Address>)
	Recording Device	Bit 6		
	Tuner	Bit 5		
	Playback Device	Bit 4		
	Audio System	Bit 3		
	CEC Switch	Bit 2		
	Reserved	Bit 1-0		
[CEC Version]	Deprecated (will not be used for future specifications)	0x00 - 0x03	1 byte	Indicates the version number of the CEC Specification which was used to design the device. Values 0x05 and below shall not be used for devices implementing CEC 2.0. A higher value indicates a more recent version of the Specification.
	"Version 1.3a"	0x04		
	"Version 1.4 or Version 1.4a or Version 1.4b"	0x05		
	"Version 2.0"	0x06		
	Reserved (for future specifications that are backward compatible with versions 2.0, 1.4/1.4a/1.4b, and 1.3a)	0x07 – 0x3F		
	Reserved	0x40 – 0xFF		

Name		Range Description		Length	Purpose
[Device Features]		[Device Features 1]; Or: [Device Features 1] to ….[Device Features N] (where N≥2)		1 .. N bytes	Bitwise field if certain feature (under certain conditions) is supported; Default=0, set bit to 1 in certain cases specified in Table 11-4 and Table 11-5. Length is variable, determined with [Device Features Extension] flag in each byte.
	[Device Features 1]	[Device Features Extension]	Bit 7	1 byte	First byte of [Device Features] Bitwise field if certain feature (under certain conditions) is supported; Default=0, set bit to 1 in certain cases specified in Table 11-4 and Table 11-5. [Device Features Extension] indicates whether the immediately following byte is a continuation of the [Device Features] operand or not. If bit 7 = 1, [Device Features] extends into the next byte. The last byte of [Device Features] shall have its [Device Features Extension] = 0 to indicate that is the end of the [Device Features] operand. CEC 2.0 defines only 1 byte for [Device Features]. For compatibility with future versions of the standard, Followers shall be able to handle variable length versions of this operand using the [Device Features Extension] mechanism.
		"TV supports <Record TV Screen>"	Bit 6		
		"TV supports <Set OSD String>"	Bit 5		
		"supports being controlled by Deck Control"	Bit 4		
		"Source supports <Set Audio Rate>"	Bit 3		
		"Sink supports ARC Tx"	Bit 2		
		"Source supports ARC Rx"	Bit 1		
		reserved	Bit 0		
	[Device Features 2] to [Device Features N]	[Device Features Extension]	Bit 7	1 byte	Continuation of [Device Features] operand. [Device Features Extension] indicates if this is the last byte of the operand (0) or not (1).
		reserved	Bit 6 - 0		
	[Device Features Extension]	this is the last byte of [Device Features]	0	1 bit	Indicates if this is the last byte of the operand (0) or not (1).
		[Device Features] continues in next byte	1		
[Primary Device Type]		"TV"	0	1 byte	Used by a device to indicate its primary device type. See Table 11-7 when certain Primary Device Type can be used
		"Recording Device"	1		
		Reserved	2		
		"Tuner"	3		
		"Playback Device"	4		
		"Audio System"	5		
		"Pure CEC Switch"	6		
		"Processor"	7		

Name		Range Description		Length	Purpose
[RC Profile]		[RC Profile 1]; Or: [RC Profile 1] .. [RC Profile N] (where N≥2)		1 .. N bytes	Summarizes characteristics of a TV or device as relevant for Remote Control Pass Through Length is variable, determined with [RC Profile Extension] flag in each byte.
	[RC Profile 1]	[RC Profile Extension]	Bit 7	1 byte	First byte of [RC Profile] [RC Profile Extension] indicates whether the immediately following byte is a continuation of the [RC Profile] operand or not. If bit 7 = 1, [RC Profile] extends into the next byte. The last byte of [RC Profile] shall have its [RC Profile Extension] = 0 to indicate that is the end of the [RC Profile] field. CEC 2.0 defines only 1 byte for [RC Profile]. For compatibility with future versions of the standard, Followers shall be able to handle variable length versions of this operand using the [RC Profile Extension] mechanism Initiator with [Primary Device Type]=”TV” shall set bit 6 to 0. Initiator with other [Primary Device Type] shall set bit 6 to 1.
		If bit=0, [RC Profile TV] in bits 5-0 If bit=1, [RC Profile Source] in bits 5-0	Bit 6		
		[RC Profile TV] or [RC Profile Source], as determined by bit 6	Bit 5-0		

Name	Range Description		Length	Purpose
[RC Profile 2] to [RC Profile N]	[RC Profile Extension]	Bit 7	1 byte	Continuation of [RC Profile] operand. [RC Profile Extension] indicates if this is the last byte of the operand (0) or not (1).
	reserved	Bit 6-0		
[RC Profile Source]	reserved	Bit 5	6 bits	Default = 0. Initiator sets each bit to 1 corresponding to UI Commands that will yield a useful result when receiving those as operand in a <User Control Pressed> message
	“Source can handle UI Command 0x09, “Device Root Menu” ”	Bit 4		
	“Source can handle UI Command 0x0A, “Device Setup Menu” ”	Bit 3		
	“Source can handle UI Command 0x0B, “Contents Menu” ”	Bit 2		
	“Source can handle UI Command 0x10, “Media Top Menu” ”	Bit 1		
	“Source can handle UI Command 0x11, “Media Context-Sensitive Menu” ”	Bit 0		
[RC Profile TV]	reserved	Bit 5-4	6 bits	TV's characteristics for Remote Control Pass Through
	[RC Profile ID]	Bit 3-0		
[RC Profile ID]	None of these profiles	0x0	4 bits	Remote Control profile (see Section 11.6.4 and appropriate columns in Table 11-31)
	“RC Profile 1”	0x2		
	“RC Profile 2”	0x6		
	“RC Profile 3”	0xA		
	“RC Profile 4”	0xE		
	reserved	Other values		
[Vendor ID]	0x000000≤N≤0xFFFFFF (n is the 24-bit unique IEEE Organizationally Unique Identifier (OUI) obtained from the IEEE Registration Authority Committee (RAC)).		3 bytes	Identifier for a specific vendor or entity defining Vendor Specific commands.

H14b CEC Table 30 is extended as follows:

Table 11-31: UI Command Codes

Operation ID	User Operation	Notes for the deterministic IDs (0x60-0x6F): this is mandatory follower behavior; for the other IDs, typical behavior for a CEC 2.0 device is described (see Section 11.6.2); behavior for device with earlier CEC version may be different.	Remote Control profile				Mandatory for non-TV as Follower	Mandatory for TV as Follower
			1	2	3	4		
0x00	Select / OK (d)	Select the highlighted item		x	x	x	x	
(d) the name of this UI Command was updated in CEC 2.0								
0x01	Up	4-way cursor control		x	x	x	x	
0x02	Down			x	x	x	x	
0x03	Left			x	x	x	x	
0x04	Right			x	x	x	x	
0x05	Right-Up							
0x06	Right-Down							
0x07	Left-Up							
0x08	Left-Down							
0x09	Device Root Menu (d)	Trigger the top-level menu of a device. This menu might depend on the device's current state.			x	x	x	
0x0A	Device Setup Menu (d)	Trigger the setup function of a device			x	x	x	
0x0B	Contents Menu	Trigger an overview of content available on the device. Special case: For a disc-based player without other functions, this corresponds to the disc's main menu.			x	x	x	
0x0C	Favorite Menu							
0x0D	Back (d)	Exit current menu. For multi-level menus, this moves to previous menu/screen.		x	x	x	x	
0x0E - 0x0F	Reserved	*(shall not be used)*						
0x10	Media Top Menu	Trigger the display of the main menu available for the currently playing media, e.g. disc root menu (DVD) / Top Menu (Blu-ray)			x	x	x	

Operation ID	User Operation	Notes for the deterministic IDs (0x60-0x6F): this is mandatory follower behavior; for the other IDs, typical behavior for a CEC 2.0 device is described (see Section 11.6.2); behavior for device with earlier CEC version may be different.	Remote Control profile 1	 2	 3	 4	Mandatory for non-TV as Follower	Mandatory for TV as Follower
0x11	Media Context-sensitive Menu	Trigger the display of a context-sensitive media-related menu (e.g. DVD Menu or Blu-ray Popup Menu), typically containing functions to adapt the playback of the currently playing content.			x	x	x	
0x12 - 0x1C	Reserved	*(shall not be used)*						
0x1D	Number Entry Mode	Select/toggle an available Number Entry Mode that may be implemented on a device, such as: 1..12-button entry mode, 0..9-button entry mode, single vs. multiple digit entry.						
0x1E	Number 11	Used (along with 0x20..0x29) for systems with 12 numeric buttons						
0x1F	Number 12							
0x20	Number 0 or Number 10	"Number 0" for systems with 10 numeric buttons; "Number 10" for systems with 12 numeric buttons			x	x	x	
0x21 - 0x29	Numbers 1-9	Numeric buttons			x	x	x	
0x2A	Dot	To allow entry of decimal numbers, e.g. "channel 6.2"						
0x2B	Enter	(NOTE - this is not the common OK/select/enter button; see 0x00 for that)						
0x2C	Clear							
0x2D - 0x2E	Reserved	*(shall not be used)*						
0x2F	Next Favorite					x	x	
0x30	Channel Up	Change to next channel in numeric/preferred order	x	x	x	x	x	
0x31	Channel Down	Change to previous channel in numeric/preferred order	x	x	x	x	x	
0x32	Previous Channel	Change to channel previously watched				x	x	
0x33	Sound Select	Select audio stream (e.g. when multiple languages or audio streams) are available						
0x34	Input Select							

Operation ID	User Operation	Notes for the deterministic IDs (0x60-0x6F): this is mandatory follower behavior; for the other IDs, typical behavior for a CEC 2.0 device is described (see Section 11.6.2); behavior for device with earlier CEC version may be different.	Remote Control profile 1	 2	 3	 4	Mandatory for non-TV as Follower	Mandatory for TV as Follower
0x35	Display Information	Show/hide additional information related to currently watched content				x	x	
0x36	Help							
0x37	Page Up	Scroll one page upwards				x		
0x38	Page Down	Scroll one page downwards				x		
0x39 - 0x3F	Reserved	*(shall not be used)*						
0x40	Power	*deprecated*; Only to be used towards legacy devices as described in Section 11.5.3					M (c)	M (c)
M (c) = this shall be supported as Follower by all devices since legacy devices might send this, see Section 11.5.4								
0x41	Volume Up	Increase the volume	x (a)	x (a)	x (a)	x (a)	M (b)	M (b)
0x42	Volume Down	Decrease the volume	x (a)	x (a)	x (a)	x (a)	M (b)	M (b)
0x43	Mute	Toggle the mute status	x (a)	x (a)	x (a)	x (a)	M (b)	M (b)
(a) = when the device has volume controls M (b) = this shall be supported as Follower when the device can render audio and/or control volume of audio								
0x44	Play	Start playback; if already playing, might go to "pause" stop		x	x	x	x	
0x45	Stop	Stop playback		x	x	x	x	
0x46	Pause	Pause playback; if already paused, might toggle back to normal playback		x	x	x	x	
0x47	Record	Start/Toggle/Stop(*) recording				x	x (e)	
(e) if the device has recording functionality (*) depending upon Follower semantics/implementation								
0x48	Rewind	Reverse playback through content; repeated presses might cycle through various speed options (including "x1")			x	x	x	
0x49	Fast forward	Continue playback in forward direction at higher speed; repeated presses might cycle through various speed options (including "x1")			x	x	x	
0x4A	Eject	Eject; open/close tray					x (f)	
(f) if the device can eject some media								

Operation ID	User Operation	Notes for the deterministic IDs (0x60-0x6F): this is mandatory follower behavior; for the other IDs, typical behavior for a CEC 2.0 device is described (see Section 11.6.2); behavior for device with earlier CEC version may be different.	Remote Control profile 1	2	3	4	Mandatory for non-TV as Follower	Mandatory for TV as Follower
0x4B	Skip Forward (d)	Skip forward – could be a fixed timestep forward within same content item ("skip 1 minute"), skip forward towards a marker ("goto next chapter"), or goto next content item ("next item in playlist" or next channel from tuner).			x	x	x	
0x4C	Skip Backward (d)	Skip backward – could be a fixed timestep backward within same content item ("back 1 minute"), backwards towards a marker ("goto previous chapter"), or goto previous content item ("previous item in playlist" or previous channel from tuner)			x	x	x	
0x4D	Stop-Record	Stop the recording					x (e)	
0x4E	Pause-Record	Pause the recording; if already paused, could restart the recording					x (e)	
(e) if the device has recording functionality								
0x4F	Reserved	*(shall not be used)*						
0x50	Angle	For source material with multiple viewpoints, select one of them to be viewed						
0x51	Sub picture	Start/stop/control the subtitle or closed caption functionality				x	x	
0x52	Video on Demand					x	x	
0x53	Electronic Program Guide					x	x	
0x54	Timer Programming							
0x55	Initial Configuration							
0x56	Select Broadcast Type	Select broadcast mode specified in operand [UI Broadcast Type], see Table 11-30, H14b Table CEC 29, and H14b Section CEC 13.13.7.						

Operation ID	User Operation	Notes for the deterministic IDs (0x60-0x6F): this is mandatory follower behavior; for the other IDs, typical behavior for a CEC 2.0 device is described (see Section 11.6.2); behavior for device with earlier CEC version may be different.	Remote Control profile				Mandatory for non-TV as Follower	Mandatory for TV as Follower
			1	2	3	4		
0x57	Select Sound Presentation	Select sound presentation mode specified in operand [UI Sound Presentation Control], see Table 11-30, H14b Table CEC 29, and H14b Section CEC 13.13.7.						
0x58 (g)	Audio Description	Start/stop/control the Audio Description functionality. Audio Description refers to an audio service that helps blind and visually impaired consumers understand the action in a program. Note – in some countries, this is referred to as "Video Description"				x	x	
0x59 (g)	Internet	Start/stop/control the "Internet" functionality of a device. This could be a set of applications using internet connectivity in some way					x	
0x5A (g)	3D mode	Control the display/processing mode related to 3D. Can be sent from a Source (where such a button is available on the remote) to the TV. The "3D" button on the TV remote will typically be handled in the TV itself, and not be forwarded.						
(g) these UI Commands are introduced in CEC 2.0; devices with earlier CEC version do not support these UI Commands								
0x5B - 0x5F	Reserved	*(shall not be used)*						
0x60	Play Function	Select the playback mode specified in additional operand [Play Mode], see H14b CEC Table 6.						
0x61	Pause-Play Function	Pause playback. If repeated, the device shall remain in the paused state.						
0x62	Record Function	Start recording. If repeated, the device shall remain in the record state without interrupting the recording.						
0x63	Pause-Record Function	Pause the recording. If repeated, the device shall remain paused.						
0x64	Stop Function	Stop the media. If repeated, the device shall remain stopped.						

Operation ID	User Operation	Notes for the deterministic IDs (0x60-0x6F): this is mandatory follower behavior; for the other IDs, typical behavior for a CEC 2.0 device is described (see Section 11.6.2); behavior for device with earlier CEC version may be different.	Remote Control profile 1	2	3	4	Mandatory for non-TV as Follower	Mandatory for TV as Follower
0x65	Mute Function	Mute audio output. If repeated, the audio shall stay muted.					M (b)	M (b)
0x66	Restore Volume Function	Restores audio output to the value before it was muted.					M (b)	M (b)
M (b) = this shall be supported as Follower when the device can render audio and/or control volume of audio								
0x67	Tune Function	Select a channel specified in the additional operand [Channel Identifier], see H14b CEC Table 6.						
0x68	Select Media Function	Select Media within a device specified in additional operand [UI Function Media], see H14b CEC Table 6.						
0x69	Select A/V Input Function	Select an A/V input specified in additional operand [UI Function Select A/V input], see H14b CEC Table 6.						
0x6A	Select Audio Input Function	Select an Audio input specified in additional operand [UI Function Select Audio input], see H14b CEC Table 6.						
0x6B	Power Toggle Function	Toggle the device's power state (On / Standby)					M	M
0x6C	Power Off Function	Put the device into the Standby state. If repeated, the device stays in the Standby state.					M	M
0x6D	Power On Function	Put the device into the On (non-Standby) state. If repeated, the device stays in the active state.					M	M
M = this shall be supported as Follower by all devices, see Section 11.5								
0x6E - 0x70	Reserved	*(shall not be used)*						
0x71	F1 (Blue)	Blue button/function		x	x	x	x	
0x72	F2 (Red)	Red button/function		x	x	x	x	
0x73	F3 (Green)	Green button/function		x	x	x	x	
0x74	F4 (Yellow)	Yellow button/function		x	x	x	x	
0x75	F5							
0x76	Data	Control a data application associated with the currently viewed channel, e.g. teletext or data broadcast application (MHEG, MHP, HbbTV, etc.)					x	

Operation ID	User Operation	Notes for the deterministic IDs (0x60-0x6F): this is mandatory follower behavior; for the other IDs, typical behavior for a CEC 2.0 device is described (see Section 11.6.2); behavior for device with earlier CEC version may be different.	Remote Control profile				Mandatory for non-TV as Follower	Mandatory for TV as Follower
			1	2	3	4		
0x77 - 0xFF	Reserved	*(shall not be used)*						

Appendix A 3D Audio and Multi-Stream Audio Applications (Informative)

This section provides application scenarios for 3D Audio and Multi-Stream Audio. These examples show the capability of Source and Sink Devices compliant with This Specification to transport 3D Audio and Multi-Stream Audio.

A.1 Example Scenario for 3D Audio

Figure A-1 shows how 3D Audio samples can be transported from a BDP to a TV. This example assumes the following:

- The Source (e.g. a BDP) and the Sink (e.g. a TV) are all devices compliant with This Specification.
- The Source transmits L-PCM 48kHz 22.2 channel audio streams to the Sink.
- The Sink is capable of receiving L-PCM 48kHz 22.2 channel audio samples and transmitting each individual audio stream to its associated speaker.
- The transmitted Video Format is 1080p60.

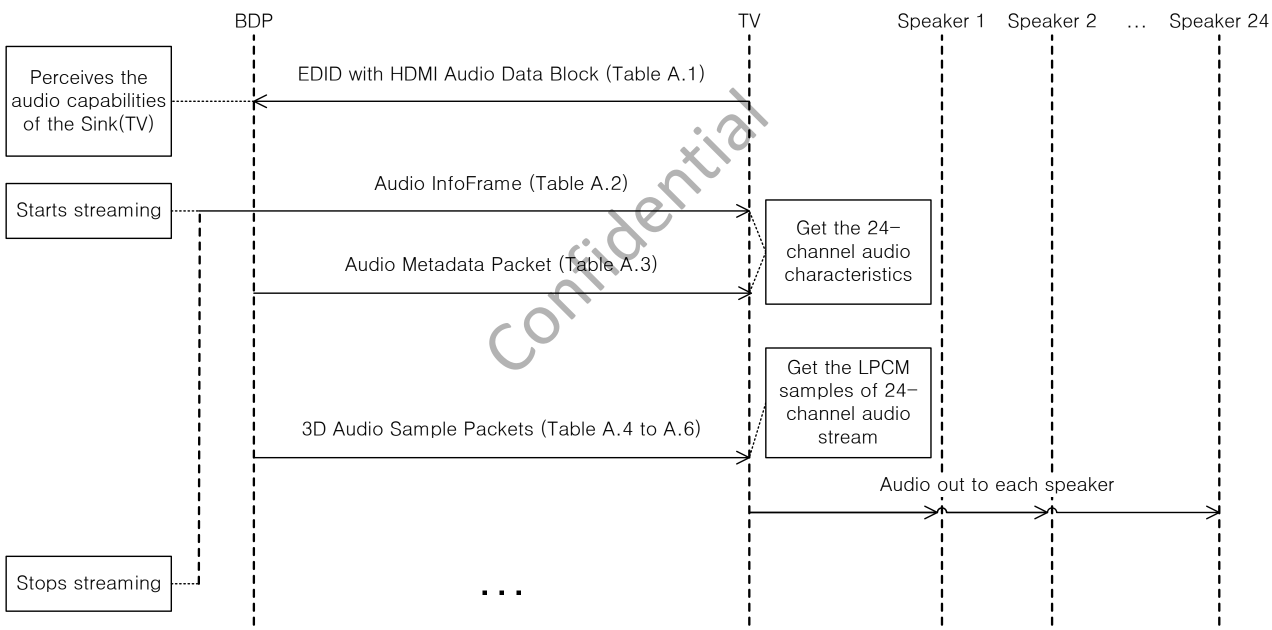


Figure A-1: Example Scenario for 3D Audio

The TV contains a CEA-861-F compliant E-EDID data structure accessible through the DDC. In order to support 3D Audio transmission, the E-EDID includes the HDMI Audio Data Block as well as other necessary data blocks. The BDP receives the HDMI Audio Data Block and recognizes the 3D Audio capabilities of the TV as described in Table A-1.

Table A-1: Example of the HDMI Audio Data Block for 22.2 channels

Byte \ Bit #	7	6	5	4	3	2	1	0
1	Tag code=7 (Use Extended Tag)			L = 11 (0b1011)				
2	Extended Tag Code = 18 (0x12)							
3	Rsvd (0)					Supports_ MS_ NonMixed = 0	Max_Stream_Count = 0b00	
4	Rsvd (0)					NUM_HDMI_3D_AD = 0b01		
5	0	0	0	0	Audio Format Code = 1			
6	0	0	0	Max Number of channels – 1 = 23 (0b10111)				
7	0	192 kHz (0)	176.4 kHz (0)	96 kHz (1)	88.2 kHz (1)	48 kHz (1)	44.1 kHz (1)	32 kHz (1)
8	0	0	0	0	0	24 bit (0)	20 bit (0)	16 bit (1)
9	FLW/FRW (0)	Rsvd (0)	FLC/FRC (1)	BC (1)	BL/BR (1)	FC (1)	LFE1 (1)	FL/FR (1)
10	TpSiL/TpSiR (1)	SiL/SiR (1)	TpBC (1)	LFE2 (1)	LS/RS (0)	TpFC (1)	TpC (1)	TpFL/TpFR (1)
11	0	0	0	LSd/LRd (0)	TpLS/TpRS (0)	BtFL/BtFR (1)	BtFC (1)	TpBL/TpBR (1)
12	ACAT = 2 (0b0010)				0	0	0	0

Byte 1, 2, 3, and 4 indicate the header of the HDMI Audio Data Block. NUM_HDMI_3D_AD is set to 0b01 to indicate support for 3D Audio transmission. Supports_MS_NonMixed and Max_Stream_Count are set to 0 because the TV in this example does not support Multi-Stream Audio transmission.

Byte 5, 6, 7, and 8 constitute the HDMI 3D Audio Descriptor which describes the 3D Audio characteristics of the TV. Audio format code, Max Number of channels-1, sampling frequency, and sample size are also shown here.

Byte 9, 10, 11, and 12 constitute the HDMI 3D Speaker Allocation Descriptor which describes the active speakers for 22.2 channels (SMPTE 2036-2).

The BDP will send an Audio InfoFrame and Audio Metadata Packet to the TV after reading the EDID from the TV. In this case, Channel Count and Channel/Speaker Allocation information are carried using the Audio Metadata Packet instead of Audio InfoFrame. 3D_CC and 3D_CA in the Audio Metadata Packet describe Channel Count and Channel/Speaker Allocation information for 22.2 channels audio streams, respectively. Table A-2 shows an example of the Audio InfoFrame payload for 22.2-channel audio transmission. Table A-3 shows an example of the Audio Metadata Packet payload for 22.2-channel audio transmission.

Table A-2: Example of the Audio InfoFrame payload for 22.2 channels

Byte \ Bit #	7	6	5	4	3	2	1	0
PB0	Checksum							
PB1	CT3 (0)	CT2 (0)	CT1 (0)	CT0 (0)	Reserved (0)	CC2 (0)	CC1 (0)	CC0 (0)
PB2	Reserved (0)			SF2 (0)	SF1 (0)	SF0 (0)	SS1 (0)	SS0 (0)
PB3	Format depends on coding type (i.e. CT0...CT3)							
PB4	CA7 (0)	CA6 (0)	CA5 (0)	CA4 (0)	CA3 (0)	CA2 (0)	CA1 (0)	CA0 (0)
PB5	DM_INH (0)	LSV3 (0)	LSV2 (0)	LSV1 (0)	LSV0 (0)	Rsvd (0)	LFEP BL1 (0)	LFEP BL0 (0)
PB6	Reserved (0)							
PB7	Reserved (0)							
PB8	Reserved (0)							
PB9	Reserved (0)							
PB10	Reserved (0)							
PB11-PB27	Reserved (0)							

Table A-3: Example of the Audio Metadata Packet for 22.2-channel Audio

Byte \ Bit #	7	6	5	4	3	2	1	0
HB0	0	0	0	0	1	1	0	1
HB1	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	3D_AUDIO = 1
HB2	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	NUM_AUDIO_STR = 0b00		NUM_VIEWS = 0b00	
PB0	Rsvd (0)	Rsvd (0)	Rsvd (0)	3D_CC4 (1)	3D_CC3 (0)	3D_CC2 (1)	3D_CC1 (1)	3D_CC0 (1)
PB1	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	ACAT = 0x02			
PB2	3D_CA7 (0)	3D_CA6 (0)	3D_CA5 (0)	3D_CA4 (0)	3D_CA3 (0)	3D_CA2 (1)	3D_CA1 (0)	3D_CA0 (0)
PB3-PB27	Reserved (0)							

The BDP transmits 22.2-channel audio samples through the 3D Audio Sample Packets. Each 3D Audio Sample Packet supports up to 8 audio channels and thus, three consecutive 3D Audio Sample Packets are needed to send one 22.2-channel audio sample. As described in Section 9.3.3, sample_start is used to designate the first 3D Audio Sample Packet. In this example, three 3D Audio Sample Packets may be populated as shown in Table A-4, Table A-5, and Table A-6.

Note that PCUV refers to the parity bit (P), channel status (C), user data (U), and validity bit (V) as defined in IEC 60958-1. Refer to H14b Section 5.3.4 for the exact layout of PCUV within each Subpacket.

Table A-4: Example of the first 3D Audio Sample Packet for 22.2 channels

Byte \ Bit #	7	6	5	4	3	2	1	0
HB0	0	0	0	0	1	0	1	1
HB1	0	0	0	sample_ start (1)	sample_ present. sp3 (1)	sample_ present. sp2 (1)	sample_ present .sp1 (1)	sample_ present. sp0 (1)
HB2	B.3	B.2	B.1	B.0	sample_flat. sp3 (0)	sample_flat. sp2 (0)	sample_flat. sp1 (0)	sample_flat. sp0 (0)
SB0 - SB2	Channel 1 / Sample N							
SB3 - SB5	Channel 2 / Sample N							
SB6	PCUV of Ch 2				PCUV of Ch 1			
SB7 - SB9	Channel 3 / Sample N							
SB10 - SB12	Channel 4 / Sample N							
SB13	PCUV of Ch 4				PCUV of Ch 3			
SB14 - SB16	Channel 5 / Sample N							
SB17 - SB19	Channel 6 / Sample N							
SB20	PCUV of Ch 6				PCUV of Ch 5			
SB21 - SB23	Channel 7 / Sample N							
SB24 - SB26	Channel 8 / Sample N							
SB27	PCUV of Ch 8				PCUV of Ch 7			

Table A-5: Example of the second 3D Audio Sample Packet for 22.2 channels

Byte \ Bit #	7	6	5	4	3	2	1	0
HB0	0	0	0	0	1	0	1	1
HB1	0	0	0	sample_ start (0)	sample_ present. sp3 (1)	sample_ present. sp2 (1)	sample_ present .sp1 (1)	sample_ present. sp0 (1)
HB2	B.3	B.2	B.1	B.0	sample_flat. sp3 (0)	sample_flat. sp2 (0)	sample_flat. sp1 (0)	sample_flat. sp0 (0)
SB0 - SB2	Channel 9 / Sample N							
SB3 - SB5	Channel 10 / Sample N							
SB6	PCUV of Ch 10				PCUV of Ch 9			
SB7 - SB9	Channel 11 / Sample N							
SB10 - SB12	Channel 12 / Sample N							
SB13	PCUV of Ch 12				PCUV of Ch 11			
SB14 - SB16	Channel 13 / Sample N							
SB17 - SB19	Channel 14 / Sample N							
SB20	PCUV of Ch 14				PCUV of Ch 13			
SB21 - SB23	Channel 15 / Sample N							
SB24 - SB26	Channel 16 / Sample N							
SB27	PCUV of Ch 16				PCUV of Ch 15			

Table A-6: Example of the third 3D Audio Sample Packet for 22.2 channels

Byte \ Bit #	7	6	5	4	3	2	1	0
HB0	0	0	0	0	1	0	1	1
HB1	0	0	0	sample_ start (0)	sample_ present. sp3 (1)	sample_ present. sp2 (1)	sample_ present .sp1 (1)	sample_ present. sp0 (1)
HB2	B.3	B.2	B.1	B.0	sample_flat. sp3 (0)	sample_flat. sp2 (0)	sample_flat. sp1 (0)	sample_flat. sp0 (0)
SB0 - SB2	Channel 17 / Sample N							
SB3 - SB5	Channel 18 / Sample N							
SB6	PCUV of Ch 18				PCUV of Ch 17			
SB7 - SB9	Channel 19 / Sample N							
SB10 - SB12	Channel 20 / Sample N							
SB13	PCUV of Ch 20				PCUV of Ch 19			
SB14 - SB16	Channel 21 / Sample N							
SB17 - SB19	Channel 22 / Sample N							
SB20	PCUV of Ch 22				PCUV of Ch 21			
SB21 - SB23	Channel 23 / Sample N							
SB24 - SB26	Channel 24 / Sample N							
SB27	PCUV of Ch 24				PCUV of Ch 23			

A.2 Example scenario for Multi-Stream Audio

Figure A-2 below shows how Multi-Stream Audio can be transmitted from a BDP to a TV. This example assumes the following:

- The Source (e.g. a BDP) and Sink (e.g. a TV) are all devices compliant with This Specification.
- The Source and Sink have entered dual-view gaming mode.
- The Source transmits two audio streams, one for each view.
- The Sink is capable of sending two audio streams to two separate headphones.
- The transmitted Video Format is HDMI 3D 1080p60.

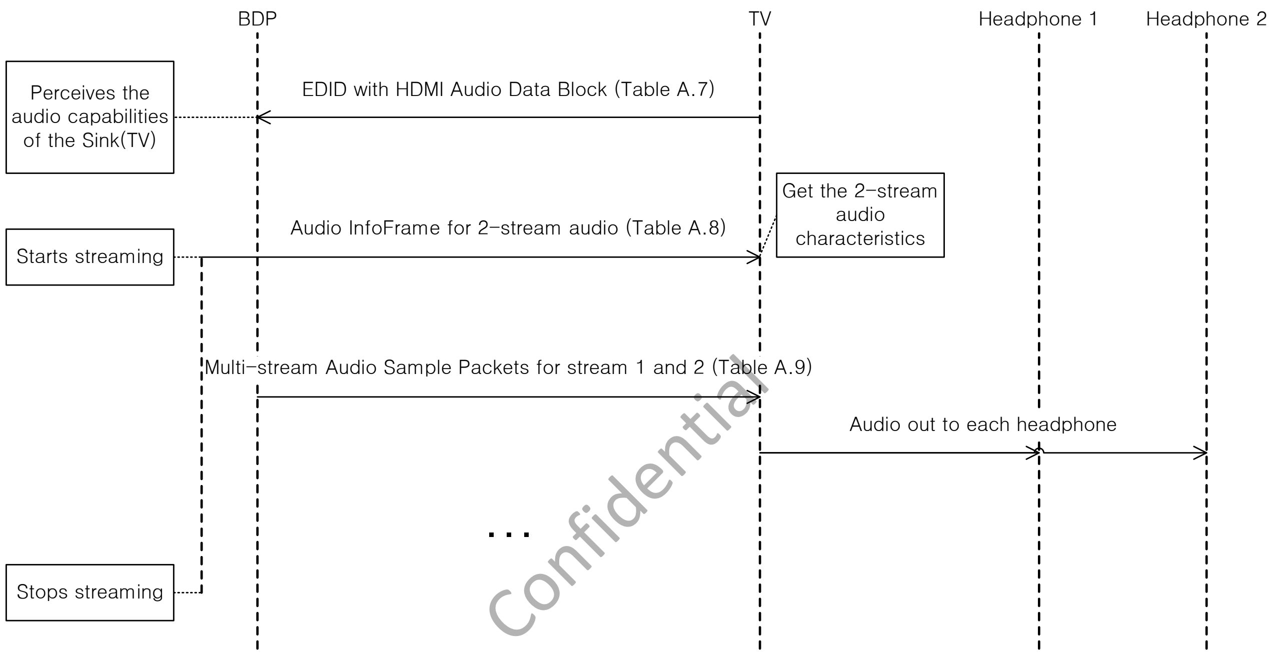


Figure A-2: Example Scenario for Multi-Stream Audio

The TV contains a CEA-861-F compliant E-EDID data structure accessible through the DDC. In order to support Multi-Stream Audio, the E-EDID includes the HDMI Audio Data Block as well as other necessary data blocks. The BDP receives the HDMI Audio Data Block and recognizes the Multi-Stream Audio capabilities of the TV as described in Table A-7.

Table A-7: Example of the HDMI Audio Data Block for two audio streams

Byte \ Bit #	7	6	5	4	3	2	1	0
1	Tag code=7 (Use Extended Tag)			L = 3 (0b011)				
2	Extended Tag Code = 18 (0x12)							
3	Rsvd (0)					Supports_ MS_Non Mixed =0	Max_Stream_Count = 0b01	
4	Rsvd (0)					NUM_HDMI_3D_AD = 0b000		

Byte 1, 2, 3, and 4 indicate the header of the HDMI Audio Data Block. Max_Stream_Count is set to 0b01 because the Sink can handle two independent audio streams as described above. NUM_HDMI_3D_AD is set to 0b000 because the TV in this example does not support 3D Audio transmission.

The BDP will send an Audio InfoFrame to the TV after reading the EDID from the TV. Contrary to the 3D Audio transmission scenario, CC and CA in the Audio InfoFrame are used to transmit Channel Count and Channel/Speaker Allocation information, respectively. Table A-8 shows an example of the Audio InfoFrame payload for transmitting two audio streams.

Table A-8: Example of the Audio InfoFrame payload for two audio streams

Byte \ Bit #	7	6	5	4	3	2	1	0
PB0	Checksum							
PB1	CT3 (0)	CT2 (0)	CT1 (0)	CT0 (0)	Reserved (0)	CC2 (0)	CC1 (0)	CC0 (1)
PB2	Reserved (0)			SF2 (0)	SF1 (0)	SF0 (0)	SS1 (0)	SS0 (0)
PB3	Format depends on coding type (i.e. CT0...CT3)							
PB4	CA7 (0)	CA6 (0)	CA5 (0)	CA4 (0)	CA3 (0)	CA2 (0)	CA1 (0)	CA0 (0)
PB5	DM_INH (0)	LSV3 (0)	LSV2 (0)	LSV1 (0)	LSV0 (0)	Rsvd (0)	LFEP BL1 (0)	LFEP BL0 (0)
PB6	Reserved (0)							
PB7	Reserved (0)							
PB8	Reserved (0)							
PB9	Reserved (0)							
PB10	Reserved (0)							
PB11-PB27	Reserved (0)							

The BDP sends the Multi-Stream Audio Sample Packets which include stereo audio samples for two independent audio streams. That is, the first Subpacket includes a stereo audio sample from the first audio stream (Stream 0) and the second Subpacket includes a stereo audio sample from the second audio stream (Stream 1). In this example, the Multi-Stream Audio Sample Packets may be populated as shown in Table A-9.

Table A-9: Example of the Multi-Stream Audio Sample Packet for two audio streams

Byte \ Bit #	7	6	5	4	3	2	1	0
HB0	0	0	0	0	1	1	1	0
HB1	0	0	0	0	stream_ present. sp3 (0)	stream_ present. sp2 (0)	stream_ present. sp1 (1)	stream_ present. sp0 (1)
HB2	B.3	B.2	B.1	B.0	stream_ flat. sp3 (0)	stream_ flat. sp2 (0)	stream_ flat. sp1 (0)	stream_ flat. sp0 (0)
SB0 - SB2	Channel 1 / Sample N (Stream 0)							
SB3 - SB5	Channel 2 / Sample N (Stream 0)							
SB6	PCUV of Ch 2 (Stream 0)				PCUV of Ch 1 (Stream 0)			
SB7 - SB9	Channel 1 / Sample N (Stream 1)							
SB10 - SB12	Channel 2 / Sample N (Stream 1)							
SB13	PCUV of Ch 2 (Stream 1)				PCUV of Ch 1 (Stream 1)			
SB14 - SB16	Empty (0)							
SB17 - SB19								
SB20								
SB21 - SB23	Empty (0)							
SB24 - SB26								
SB27								

The BDP sends the Audio Metadata Packet to indicate which audio stream is associated with either the Left or Right stereoscopic picture in the 3D Video Format. In this example, the Audio Metadata Packet may be populated as shown in Table A-10. The Audio Metadata Descriptor for Stream 0 is placed in PB0 through PB4. In PB0, Multiview_Left is set (=1) to indicate that Stream 0 is associated with the Left stereoscopic picture in the 3D Video Format. Similarly, the second Audio Metadata Descriptor (PB5 through PB9) indicates that Stream 1 is associated with the Right stereoscopic picture in the 3D Video Format. In this example, language information or supplementary audio (i.e., audio for visually/hearing impaired) is not transmitted and thus the corresponding fields are set to 0.

Table A-10: Example of the Audio Metadata Packet for two audio streams

Byte \ Bit #	7	6	5	4	3	2	1	0
HB0	0	0	0	0	1	1	0	1
HB1	Rsvd (0)							3D_AUDIO (0)
HB2	Rsvd (0)				NUM_AUDIO_STR=0b01		NUM_VIEWS = 0b01	
PB0	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Multiview_ Right (0)	MultiView_ Left (1)
PB1	LC_Valid (0)	Rsvd (0)	Rsvd (0)	Suppl_A_ Valid (0)	Suppl_A_ Mixed (0)	Suppl_A_Typ = 0b000		
PB2	Language_Code = 0x000000							
PB3								
PB4								
PB5	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	Rsvd (0)	MultiView_ Right (1)	MultiView_ Left (0)
PB6	LC_Valid (0)	Rsvd (0)	Rsvd (0)	Suppl_A_ Valid (0)	Suppl_A_ Mixed (0)	Suppl_A_Typ = 0b000		
PB7	Language_Code = 0x000000							
PB8								
PB9								
PB10-PB27	Reserved (0)							

A.3 Example use-cases for Multi-Stream Audio

Figure A-3 and Table A-11 summarize examples of use cases that utilize the Multi-Stream Audio feature as described in This Specification.

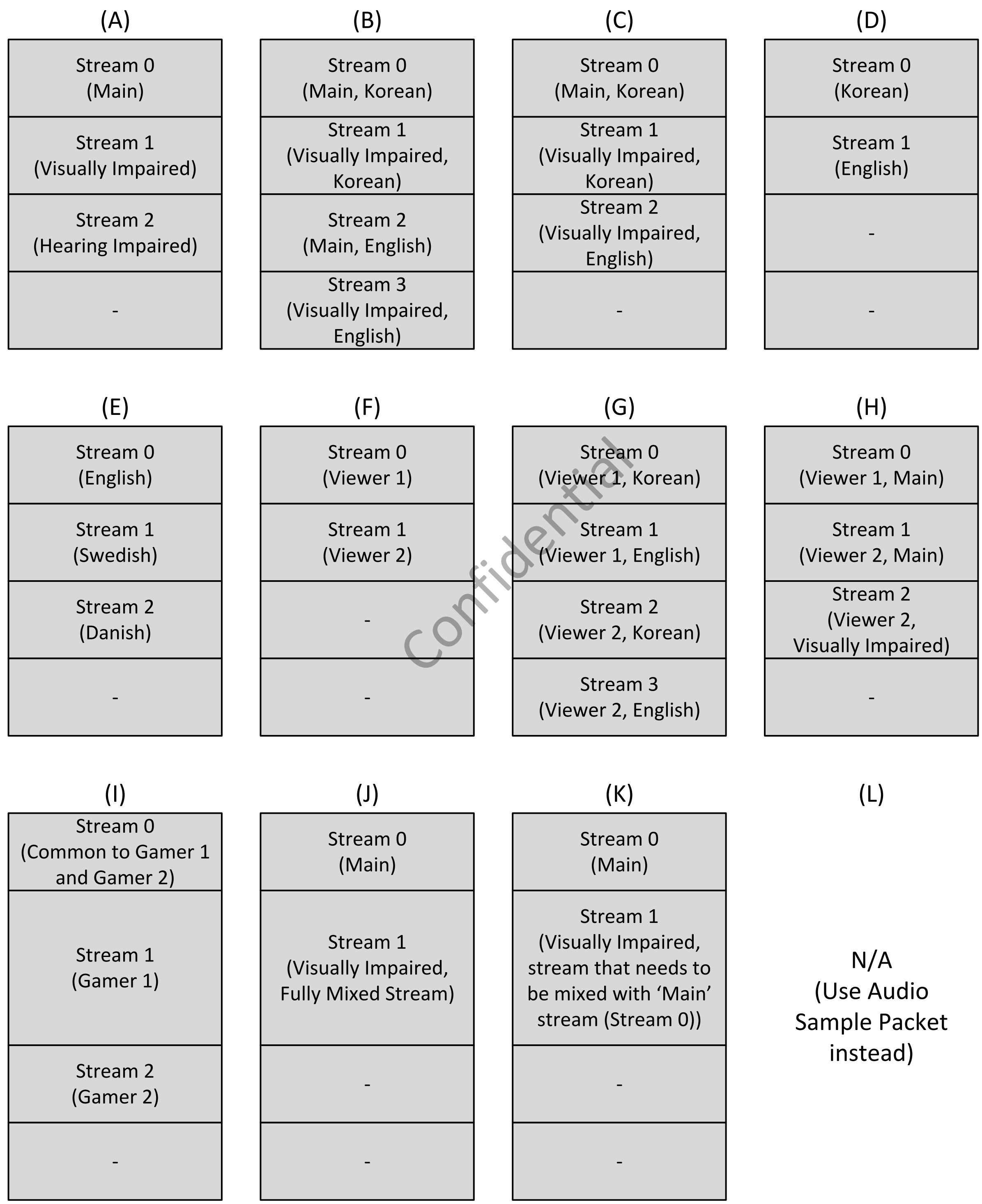


Figure A-3: Example use-cases for Multi-Stream Audio

Table A-11: Example use-cases for Multi-Stream Audio and associated signaling

<table>
<tr><th>Figure A-3</th><th>Use case</th><th>Content in stream 0..3</th><th>Signaling used by Source in Audio Metadata Packet</th><th>Signaling used by Sink in Audio Data Block</th></tr>
<tr><td>(A)</td><td>Single video, with audio description (single view)</td><td>Stream 0 = main

Stream 1 = visually impaired

Stream 2 = hearing impaired</td><td>
<table>
<tr><th></th><th colspan="4">Audio Metadata Descriptor</th></tr>
<tr><th></th><th>0</th><th>1</th><th>2</th><th>3</th></tr>
<tr><td>NUM_VIEWS</td><td colspan="4">0b00</td></tr>
<tr><td>NUM_AUDIO_STR</td><td colspan="4">0b01</td></tr>
<tr><td>MultiView_Left</td><td>0</td><td>0</td><td>0</td><td>-</td></tr>
<tr><td>MultiView_Right</td><td>0</td><td>0</td><td>0</td><td>-</td></tr>
<tr><td>LC_Valid</td><td>0</td><td>0</td><td>0</td><td>-</td></tr>
<tr><td>Suppl_A_Valid</td><td>0</td><td>1</td><td>1</td><td>-</td></tr>
<tr><td>Suppl_A_Mixed</td><td>-</td><td>1</td><td>1</td><td>-</td></tr>
<tr><td>Suppl_A_Type</td><td>-</td><td>1 or 2</td><td>3</td><td>-</td></tr>
<tr><td>Language_Code</td><td>-</td><td>-</td><td>-</td><td>-</td></tr>
</table>
</td><td>Max_Stream_Count=0b10 or more</td></tr>
<tr><td>(B)</td><td>Single video, with bi-lingual audio description (single view)</td><td>Stream 0 = main, Korean

Stream 1 = visually impaired, Korean

Stream 2 = main, English

Stream 3 = visually impaired, English</td><td>
<table>
<tr><th></th><th colspan="4">Audio Metadata Descriptor</th></tr>
<tr><th></th><th>0</th><th>1</th><th>2</th><th>3</th></tr>
<tr><td>NUM_VIEWS</td><td colspan="4">0b00</td></tr>
<tr><td>NUM_AUDIO_STR</td><td colspan="4">0b11</td></tr>
<tr><td>MultiView_Left</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>MultiView_Right</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>LC_Valid</td><td>1</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>Suppl_A_Valid</td><td>0</td><td>1</td><td>0</td><td>1</td></tr>
<tr><td>Suppl_A_Mixed</td><td>-</td><td>1</td><td>-</td><td>1</td></tr>
<tr><td>Suppl_A_Type</td><td>-</td><td>1 or 2</td><td>-</td><td>1 or 2</td></tr>
<tr><td>Language_Code</td><td>“kor”</td><td>“kor”</td><td>“eng”</td><td>“eng”</td></tr>
</table>
</td><td>Max_Stream_Count = 0b11</td></tr>
</table>

<table>
<tr><th>Figure A-3</th><th>Use case</th><th>Content in stream 0..3</th><th>Signaling used by Source in Audio Metadata Packet</th><th>Signaling used by Sink in Audio Data Block</th></tr>
<tr><td>(C)</td><td>Single video, with bi-lingual audio description (single view)</td><td>Stream 0 = main, Korean

Stream 1 = visually impaired, Korean

Stream 2 = visually impaired, English</td><td>
<table>
<tr><th></th><th colspan="4">Audio Metadata Descriptor</th></tr>
<tr><th></th><th>0</th><th>1</th><th>2</th><th>3</th></tr>
<tr><td>NUM_VIEWS</td><td colspan="4">0b00</td></tr>
<tr><td>NUM_AUDIO_STR</td><td colspan="4">0b10</td></tr>
<tr><td>MultiView_Left</td><td>0</td><td>0</td><td>0</td><td>-</td></tr>
<tr><td>MultiView_Right</td><td>0</td><td>0</td><td>0</td><td>-</td></tr>
<tr><td>LC_Valid</td><td>1</td><td>1</td><td>1</td><td>-</td></tr>
<tr><td>Suppl_A_Valid</td><td>0</td><td>1</td><td>1</td><td>-</td></tr>
<tr><td>Suppl_A_Mixed</td><td>-</td><td>1</td><td>1</td><td>-</td></tr>
<tr><td>Suppl_A_Type</td><td>-</td><td>1 or 2</td><td>1 or 2</td><td>-</td></tr>
<tr><td>Language_Code</td><td>“kor”</td><td>“kor”</td><td>“eng”</td><td>-</td></tr>
</table>
</td><td>Max_Stream_Count = 0b10 or more</td></tr>
<tr><td>(D)</td><td>Single video, with 2 audio tracks (single view)</td><td>Stream 0 = Korean

Stream 1 = English</td><td>
<table>
<tr><th></th><th colspan="4">Audio Metadata Descriptor</th></tr>
<tr><th></th><th>0</th><th>1</th><th>2</th><th>3</th></tr>
<tr><td>NUM_VIEWS</td><td colspan="4">0b00</td></tr>
<tr><td>NUM_AUDIO_STR</td><td colspan="4">0b01</td></tr>
<tr><td>MultiView_Left</td><td>0</td><td>0</td><td>-</td><td>-</td></tr>
<tr><td>MultiView_Right</td><td>0</td><td>0</td><td>-</td><td>-</td></tr>
<tr><td>LC_Valid</td><td>1</td><td>1</td><td>-</td><td>-</td></tr>
<tr><td>Suppl_A_Valid</td><td>0</td><td>0</td><td>-</td><td>-</td></tr>
<tr><td>Suppl_A_Mixed</td><td>-</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td>Suppl_A_Type</td><td>-</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td>Language_Code</td><td>“kor”</td><td>“eng”</td><td>-</td><td>-</td></tr>
</table>
</td><td>Max_Stream_Count = 0b01 or more</td></tr>
<tr><td>(E)</td><td>Single video, with 3 audio tracks (single view)</td><td>Stream 0 = English

Stream 1 = Swedish

Stream 2 = Danish</td><td>
<table>
<tr><th></th><th colspan="4">Audio Metadata Descriptor</th></tr>
<tr><th></th><th>0</th><th>1</th><th>2</th><th>3</th></tr>
<tr><td>NUM_VIEWS</td><td colspan="4">0b00</td></tr>
<tr><td>NUM_AUDIO_STR</td><td colspan="4">0b10</td></tr>
<tr><td>MultiView_Left</td><td>0</td><td>0</td><td>0</td><td>-</td></tr>
<tr><td>MultiView_Right</td><td>0</td><td>0</td><td>0</td><td>-</td></tr>
<tr><td>LC_Valid</td><td>1</td><td>1</td><td>1</td><td>-</td></tr>
<tr><td>Suppl_A_Valid</td><td>0</td><td>0</td><td>0</td><td>-</td></tr>
<tr><td>Suppl_A_Mixed</td><td>-</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td>Suppl_A_Type</td><td>-</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td>Language_Code</td><td>“eng”</td><td>“swe”</td><td>“dan”</td><td>-</td></tr>
</table>
</td><td>Max_Stream_Count = 0b10 or more</td></tr>
</table>

<table>
<tr><th>Figure A-3</th><th>Use case</th><th>Content in stream 0..3</th><th>Signaling used by Source in Audio Metadata Packet</th><th>Signaling used by Sink in Audio Data Block</th></tr>
<tr><td>(F)</td><td>Two independent video, each with own audio (dual view)</td><td>Stream 0 = viewer 1

Stream 1 = viewer 2</td><td>
<table>
<tr><th></th><th colspan="4">Audio Metadata Descriptor</th></tr>
<tr><th></th><th>0</th><th>1</th><th>2</th><th>3</th></tr>
<tr><td>NUM_VIEWS</td><td colspan="4">0b01</td></tr>
<tr><td>NUM_AUDIO_STR</td><td colspan="4">0b01</td></tr>
<tr><td>MultiView_Left</td><td>1</td><td>0</td><td>-</td><td>-</td></tr>
<tr><td>MultiView_Right</td><td>0</td><td>1</td><td>-</td><td>-</td></tr>
<tr><td>LC_Valid</td><td>0</td><td>0</td><td>-</td><td>-</td></tr>
<tr><td>Suppl_A_Valid</td><td>0</td><td>0</td><td>-</td><td>-</td></tr>
<tr><td>Suppl_A_Mixed</td><td>-</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td>Suppl_A_Type</td><td>-</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td>Language_Code</td><td>-</td><td>-</td><td>-</td><td>-</td></tr>
</table>
</td><td>Max_Stream_Count = 0b01 or more</td></tr>
<tr><td>(G)</td><td>Two independent video, each with own bi-lingual audio (dual view)</td><td>Stream 0 = viewer 1, Korean

Stream 1 = viewer 1, English

Stream 2 = viewer 2, Korean

Stream 3 = viewer 2, English</td><td>
<table>
<tr><th></th><th colspan="4">Audio Metadata Descriptor</th></tr>
<tr><th></th><th>0</th><th>1</th><th>2</th><th>3</th></tr>
<tr><td>NUM_VIEWS</td><td colspan="4">0b01</td></tr>
<tr><td>NUM_AUDIO_STR</td><td colspan="4">0b11</td></tr>
<tr><td>MultiView_Left</td><td>1</td><td>1</td><td>0</td><td>0</td></tr>
<tr><td>MultiView_Right</td><td>0</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td>LC_Valid</td><td>1</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>Suppl_A_Valid</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>Suppl_A_Mixed</td><td>-</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td>Suppl_A_Type</td><td>-</td><td>-</td><td>-</td><td>-</td></tr>
<tr><td>Language_Code</td><td>“kor”</td><td>“eng”</td><td>“kor”</td><td>“eng”</td></tr>
</table>
</td><td>Max_Stream_Count = 0b11</td></tr>
<tr><td>(H)</td><td>Two independent video, one with audio description (dual view)</td><td>Stream 0 = viewer 1, main

Stream 1 = viewer 2, main

Stream 2 = viewer 2, visually impaired</td><td>
<table>
<tr><th></th><th colspan="4">Audio Metadata Descriptor</th></tr>
<tr><th></th><th>0</th><th>1</th><th>2</th><th>3</th></tr>
<tr><td>NUM_VIEWS</td><td colspan="4">0b01</td></tr>
<tr><td>NUM_AUDIO_STR</td><td colspan="4">0b10</td></tr>
<tr><td>MultiView_Left</td><td>1</td><td>0</td><td>0</td><td>-</td></tr>
<tr><td>MultiView_Right</td><td>0</td><td>1</td><td>1</td><td>-</td></tr>
<tr><td>LC_Valid</td><td>0</td><td>0</td><td>0</td><td>-</td></tr>
<tr><td>Suppl_A_Valid</td><td>0</td><td>0</td><td>1</td><td>-</td></tr>
<tr><td>Suppl_A_Mixed</td><td>-</td><td>-</td><td>1</td><td>-</td></tr>
<tr><td>Suppl_A_Type</td><td>-</td><td>-</td><td>1 or 2</td><td>-</td></tr>
<tr><td>Language_Code</td><td>-</td><td>-</td><td>-</td><td>-</td></tr>
</table>
</td><td>Max_Stream_Count =0b10 or more</td></tr>
</table>

<table>
<tr><th>Figure A-3</th><th>Use case</th><th>Content in stream 0..3</th><th>Signaling used by Source in Audio Metadata Packet</th><th>Signaling used by Sink in Audio Data Block</th></tr>
<tr><td>(I)</td><td>Two-player gaming with common audio, with private audio for team communication (dual view)</td><td>Stream 0 = common to gamer 1 and gamer 2

Stream 1 = gamer 1

Stream 2 = gamer 2</td><td><table>
<tr><th></th><th colspan="4">Audio Metadata Descriptor</th></tr>
<tr><th></th><th>0</th><th>1</th><th>2</th><th>3</th></tr>
<tr><td>NUM_VIEWS</td><td colspan="4">0b01</td></tr>
<tr><td>NUM_AUDIO_STR</td><td colspan="4">0b10</td></tr>
<tr><td>MultiView_Left</td><td>1</td><td>1</td><td>0</td><td>-</td></tr>
<tr><td>MultiView_Right</td><td>1</td><td>0</td><td>1</td><td>-</td></tr>
<tr><td>LC_Valid</td><td>0</td><td>0</td><td>0</td><td>-</td></tr>
<tr><td>Suppl_A_Valid</td><td>0</td><td>1</td><td>1</td><td>-</td></tr>
<tr><td>Suppl_A_Mixed</td><td>-</td><td>0</td><td>0</td><td>-</td></tr>
<tr><td>Suppl_A_Type</td><td>-</td><td>4</td><td>4</td><td>-</td></tr>
<tr><td>Language_Code</td><td>-</td><td>-</td><td>-</td><td>-</td></tr>
</table></td><td>Max_Stream_Count = 0b10 or more
Supports_MS_NonMixed=1</td></tr>
<tr><td>(J)</td><td>Single video, with audio description (single view)</td><td>Stream 0 = main

Stream 1 = visually impaired, fully mixed stream</td><td><table>
<tr><th></th><th colspan="4">Audio Metadata Descriptor</th></tr>
<tr><th></th><th>0</th><th>1</th><th>2</th><th>3</th></tr>
<tr><td>NUM_VIEWS</td><td colspan="4">0b00</td></tr>
<tr><td>NUM_AUDIO_STR</td><td colspan="4">0b01</td></tr>
<tr><td>MultiView_Left</td><td>0</td><td>0</td><td>-</td><td>-</td></tr>
<tr><td>MultiView_Right</td><td>0</td><td>0</td><td>-</td><td>-</td></tr>
<tr><td>LC_Valid</td><td>0</td><td>0</td><td>-</td><td>-</td></tr>
<tr><td>Suppl_A_Valid</td><td>0</td><td>1</td><td>-</td><td>-</td></tr>
<tr><td>Suppl_A_Mixed</td><td>-</td><td>1</td><td>-</td><td>-</td></tr>
<tr><td>Suppl_A_Type</td><td>-</td><td>1 or 2</td><td>-</td><td>-</td></tr>
<tr><td>Language_Code</td><td>-</td><td>-</td><td>-</td><td>-</td></tr>
</table></td><td>Max_Stream_Count = 0b01 or more</td></tr>
<tr><td>(K)</td><td>Single video, with audio description (single view)</td><td>Stream 0 = main

Stream 1 = visually impaired, stream that needs to be mixed with ‘main’ stream (Stream 0)</td><td><table>
<tr><th></th><th colspan="4">Audio Metadata Descriptor</th></tr>
<tr><th></th><th>0</th><th>1</th><th>2</th><th>3</th></tr>
<tr><td>NUM_VIEWS</td><td colspan="4">0b00</td></tr>
<tr><td>NUM_AUDIO_STR</td><td colspan="4">0b01</td></tr>
<tr><td>MultiView_Left</td><td>0</td><td>0</td><td>-</td><td>-</td></tr>
<tr><td>MultiView_Right</td><td>0</td><td>0</td><td>-</td><td>-</td></tr>
<tr><td>LC_Valid</td><td>0</td><td>0</td><td>-</td><td>-</td></tr>
<tr><td>Suppl_A_Valid</td><td>0</td><td>1</td><td>-</td><td>-</td></tr>
<tr><td>Suppl_A_Mixed</td><td>-</td><td>0</td><td>-</td><td>-</td></tr>
<tr><td>Suppl_A_Type</td><td>-</td><td>1 or 2</td><td>-</td><td>-</td></tr>
<tr><td>Language_Code</td><td>-</td><td>-</td><td>-</td><td>-</td></tr>
</table></td><td>Max_Stream_Count = 0b01 or more
Supports_MS_NonMixed=1</td></tr>
</table>

Figure A-3	Use case	Content in stream 0..3	Signaling used by Source in Audio Metadata Packet	Signaling used by Sink in Audio Data Block
(L)	Two independent video, with same audio (dual view)	n/a (not using Multi-Stream Audio Sample Packet, but Audio Sample Packet instead)	n/a	Max_Stream_Count = 0b00 or more (or no ADB in EDID)

Appendix B 3D Audio Speaker Placement & Channel Allocation (informative)

This section provides Speaker Placement and Channel Allocation information for 3D Audio transmission.

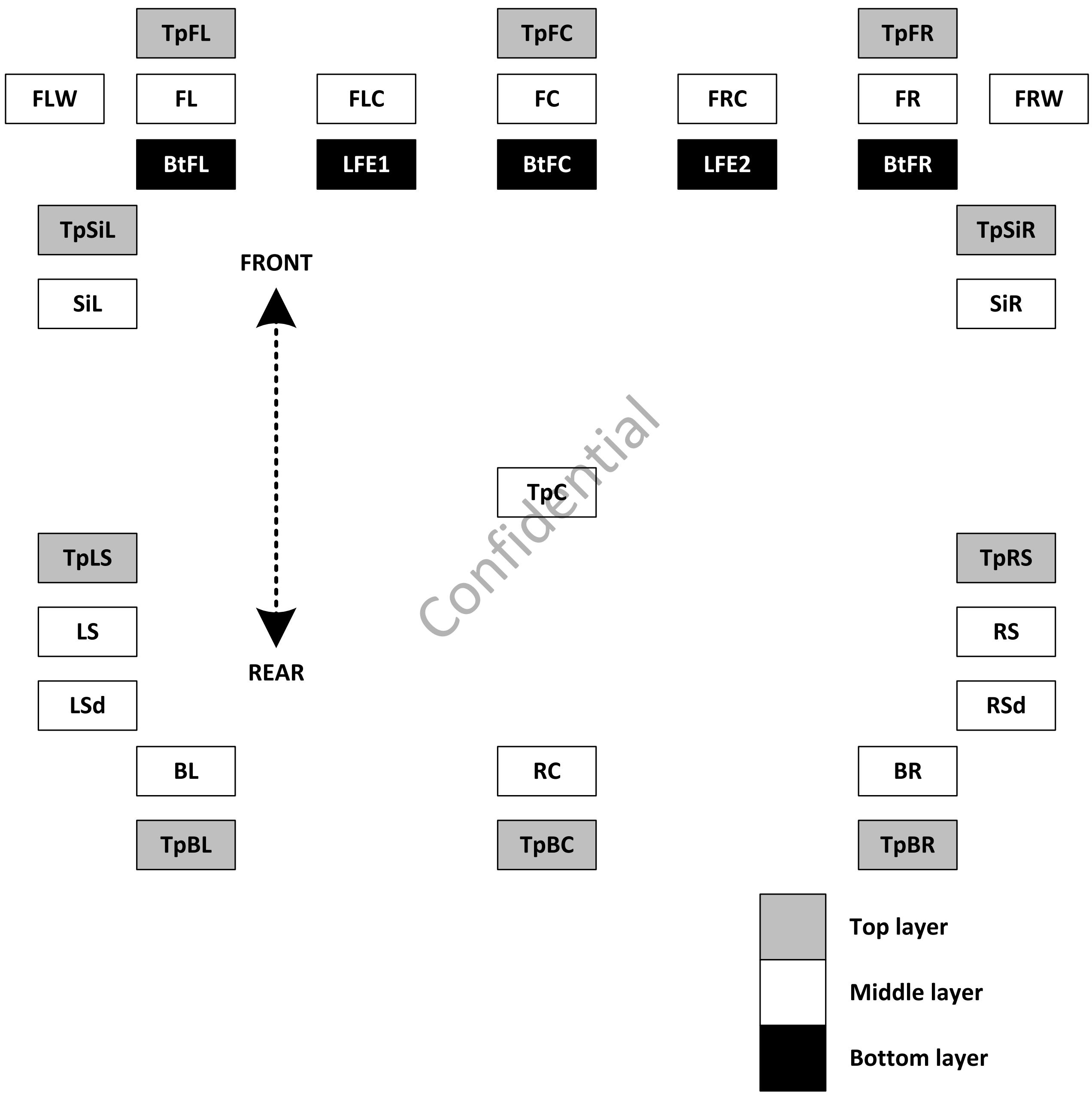


Figure B-1: 3D Speaker Placement

Table B-1: Audio Channel Description & Abbreviation Comparison (CEA/ITU/SMPTE/IEC)

Abbreviation				Description
CEA-861	ITU (Type B 10.2ch)	SMPTE (22.2ch)	IEC (30.2ch)	
FL	L	FL	FL	Front Left
FR	R	FR	FR	Front Right
LFE	LFE1	LFE1	LFE1	Low Frequency Effect 1
FC	C	FC	FC	Front Center
RL	LB	BL	BL	Back Left
RR	RB	BR	BR	Back Right
FLW	•	•	FLW	Front Left Wide
FRW	•	•	FRW	Front Right Wide
FLH	LH	TpFL	TpFL	Top Front Left
FRH	RH	TpFR	TpFR	Top Front Right
RC	•	BC	BC	Back Center
•	LS	•	LS	Left Surround
•	RS	•	RS	Right Surround
•	LFE2	LFE2	LFE2	Low Frequency Effect 2
FLC	•	FLC	FLC	Front Left Center
FRC	•	FRC	FRC	Front Right Center
RLC	•	•	•	Rear Left Center
RRC	•	•	•	Rear Right Center
FCH	•	TpFC	TpFC	Top Front Center
TC	•	TpC	TpC	Top Center
•	•	SiL	SiL	Side Left
•	•	SiR	SiR	Side Right
•	•	TpBL	TpBL	Top Back Left
•	•	TpBR	TpBR	Top Back Right
•	•	TpSiL	TpSiL	Top Side Left
•	•	TpSiR	TpSiR	Top Side Right
•	•	BtFC	BtFC	Bottom Front Center
•	•	BtFL	BtFL	Bottom Front Left
•	•	BtFR	BtFR	Bottom Front Right
•	CH	TpBC	TpBC	Top Back Center
•	•	•	TpLS	Top Left Surround
•	•	•	TpRS	Top Right Surround
•	•	•	LSd	Left Surround direct
•	•	•	RSd	Right Surround direct

Appendix C Recommended N and Expected CTS Values

(‡) This section incorporates text from the HDMI Specification 1.4b Appendix D.2. See Notice for copyright information.

The recommended value of N for several standard pixel clock rates at several Deep Color modes are shown below. It is recommended that Sources with non-coherent clocks use the values listed for a TMDS clock of “Other”.

Table C-1: 30 bits/Pixel: Recommended N and expected CTS for Audio Sample Frequency or Frame Rate of 32 kHz and multiples thereof

TMDS Character Rate	32 kHz		64 kHz		128 kHz		256 kHz	
(Mcsc)	N	CTS	N	CTS	N	CTS	N	CTS
31.5/1.001	9152	70312 – 70313 (70312.5)$^{\Delta}$	18304	70312 – 70313 (70312.5) $^{\Delta}$	36608	70312 – 70313 (70312.5)$^{\Delta}$	73216	70312 – 70313 (70312.5)$^{\Delta}$
31.5	4096	31500	8192	31500	16384	31500	32768	31500
33.75	4096	33750	8192	33750	16384	33750	32768	33750
33.75*1.001	8192	67567 – 67568 (67567.5)$^{\Delta}$	16384	67567 – 67568 (67567.5)$^{\Delta}$	32768	67567 – 67568 (67567.5)$^{\Delta}$	65536	67567 – 67568 (67567.5)$^{\Delta}$
67.5	4096	67500	8192	67500	16384	67500	32768	67500
67.5*1.001	8192	135135	16384	135135	32768	135135	65536	135135
92.8125/1.001	11648	263671 – 263672 (263671.875)$^{\Diamond}$	23296	263671 – 263672 (263671.875)$^{\Theta}$	46592	263671 – 263672 (263671.875)$^{\Theta}$	93184	263671 – 263672 (263671.875)$^{\Theta}$
92.8125	8192	185625	16384	185625	32768	185625	65536	9185625
185.625/1.001	11648	527343 – 527344 (527343.75)$^{\Diamond}$	23296	527343 – 527344 (527343.75)$^{\Diamond}$	46592	527343 – 527344 (527343.75)$^{\Diamond}$	93184	527343 – 527344 (527343.75)$^{\Diamond}$
185.625	4096	185625	8192	185625	16384	185625	32768	185625
371.25/1.001	11648	527343 – 527344 (527343.75)$^{\Diamond}$	23296	527343 – 527344 (527343.75)$^{\Diamond}$	46592	527343 – 527344 (527343.75)$^{\Diamond}$	93184	527343 – 527344 (527343.75)$^{\Diamond}$
371.25	6144	556875	12288	556875	24576	556875	49152	556875
Other	4096	measured	8192	measured	16384	measured	32768	measured

$^{\Delta}$ Fractional portion is 0.5: Dither between the values with a (1/2 duty Cycle).
$^{\Diamond}$ Fractional portion is 0.75: Dither between the values with a 3/4 duty cycle (3 with the larger value and 1 with the smaller value)
$^{\Theta}$ Fractional portion is 0.875: Dither between the values with a 7/8 duty cycle (7 with the larger value and 1 with the smaller value)

Table C-2: 30 bits/Pixel: Recommended N and expected CTS for Audio Sample Frequency or Frame Rate of 44.1 kHz and multiples thereof

TMDS Character Rate	44.1 kHz		88.2 kHz		176.4 kHz		352.8 kHz	
(Mcsc)	**N**	**CTS**	**N**	**CTS**	**N**	**CTS**	**N**	**CTS**
31.5/1.001	14014	78125	28028	78125	56056	78125	112112	78125
31.5	6272	35000	12544	35000	25088	35000	50176	35000
33.75	6272	37500	12544	37500	25088	37500	50176	37500
33.75*1.001	12544	75075	25088	75075	50176	75075	100352	75075
67.5	6272	75000	12544	75000	25088	75000	50176	75000
67.5*1.001	6272	75075	12544	75075	25088	75075	50176	75075
92.8125/1.001	17836	292968 – 292969 (292968.75)[◊]	35672	292968 – 292969 (292968.75)[◊]	71344	292968 – 292969 (292968.75)[◊]	142688	292968 – 292969 (292968.75)[◊]
92.8125	6272	103125	12544	103125	25088	103125	50176	103125
185.625/1.001	17836	585937 – 585938 (585937.5)[Δ]	35672	585937 – 585938 (585937.5)[Δ]	71344	585937 – 585938 (585937.5)[Δ]	142688	585937 – 585938 (585937.5)[Δ]
185.625	6272	206250	12544	206250	25088	206250	50176	206250
371.25/1.001	17836	585937 – 585938 (585937.5)[Δ]	35672	585937 – 585938 (585937.5)[Δ]	71344	585937 – 585938 (585937.5)[Δ]	142688	585937 – 585938 (585937.5)[Δ]
371.25	4704	309375	9408	309375	18816	309375	37632	309375
Other	6272	measured	12544	measured	25088	measured	50176	measured

[Δ] Fractional portion is 0.5: Dither between the values with a 1/2 duty Cycle).

[◊] Fractional portion is 0.75: Dither between the values with a 3/4 duty cycle (3 with the larger value and 1 with the smaller value)

Table C-3: 30 bits/Pixel: Recommended N and expected CTS for Audio Sample Frequency or Frame Rate of 48 kHz and multiples thereof

TMDS Character Rate	48 kHz		96 kHz		192 kHz		384 kHz	
(Mcsc)	**N**	**CTS**	**N**	**CTS**	**N**	**CTS**	**N**	**CTS**
31.5/1.001	9152	46875	18304	46875	36608	46875	73216	46875
31.5	6144	31500	12288	31500	24576	31500	49152	31500
33.75	6144	33750	12288	33750	24576	33750	49152	33750
33.75*1.001	8192	45045	16384	45045	32768	45045	65536	45045
67.5	6144	67500	12288	67500	24576	67500	49152	67500
67.5*1.001	8192	90090	16384	90090	32768	90090	65536	90090
92.8125/1.001	11648	175781 – 175782 (175781.25)[◘]	23296	175781 – 175782 (175781.25)[◘]	46592	175781 – 175782 (175781.25)[◘]	93184	175781 – 175782 (175781.25)[◘]
92.8125	12288	185625	24576	185625	49152	185625	98304	185625
185.625/1.001	11648	351562 – 351563 (351562.5)[Δ]	23296	351562 – 351563 (351562.5)[Δ]	46592	351562 – 351563 (351562.5)[Δ]	93184	351562 – 351563 (351562.5)[Δ]
185.625	6144	185625	12288	185625	24576	185625	49152	185625
371.25/1.001	11648	703125	23296	703125	46592	703125	93184	703125
371.25	5120	309375	10240	309375	20480	309375	40960	309375
Other	6144	measured	12288	measured	24576	measured	49152	measured

[◘] Fractional portion is 0.25: Dither between the values with a 1/4 duty Cycle (3 with the smaller value and 1 with the larger value)

[Δ] Fractional portion is 0.5: Dither between the values with a (1/2 duty Cycle).

Table C-4: 36bits/Pixel: Recommended N and expected CTS for Audio Sample Frequency or Frame Rate of 32 kHz and multiples thereof

TMDS Character Rate	32 kHz		64 kHz		128 kHz		256 kHz	
(Mcsc)	**N**	**CTS**	**N**	**CTS**	**N**	**CTS**	**N**	**CTS**
37.8/1.001	9152	84375	18304	84375	36608	84375	73216	84375
37.8	4096	37800	8192	37800	16384	37800	32768	37800
40.5	4096	40500	8192	40500	16384	40500	32768	40500
40.5*1.001	8192	81081	16384	81081	32768	81081	65536	81081
81	4096	81000	8192	81000	16384	81000	32768	81000
81*1.001	4096	81081	8192	81081	16384	81081	32768	81081
111.375/1.001	11648	316406 – 316407 (316406.25)▫	23296	316406 – 316407 (316406.25)▫	46592	316406 – 316407 (316406.25)▫	93184	316406 – 316407 (316406.25)▫
111.375	4096	111375	8192	111375	16384	111375	32768	111375
222.75/1.001	11648	632812 – 632813 (632812.5)$^{\Delta}$	23296	632812 – 632813 (632812.5)$^{\Delta}$	46592	632812 – 632813 (632812.5)$^{\Delta}$	93184	632812 – 632813 (632812.5)$^{\Delta}$
222.75	4096	222750	8192	222750	16384	222750	32768	222750
445.5/1.001	5824	632812 – 632813 (632812.5)$^{\Delta}$	11648	632812 – 632813 (632812.5)$^{\Delta}$	23296	632812 – 632813 (632812.5)$^{\Delta}$	46592	632812 – 632813 (632812.5)$^{\Delta}$
445.5	4096	445500	8192	445500	16384	445500	32768	445500
Other	4096	measured	8192	measured	16384	measured	32768	measured

▫ Fractional portion is 0.25: Dither between the values with a 1/4 duty Cycle (3 with the smaller value and 1 with the larger value)

$^{\Delta}$ Fractional portion is 0.5: Dither between the values with a (1/2 duty Cycle).

Table C-5: 36 bits/Pixel: Recommended N and expected CTS for Audio Sample Frequency or Frame Rate of 44.1 kHz and multiples thereof

TMDS Character Rate	44.1 kHz		88.2 kHz		176.4 kHz		352.8 kHz	
(Mcsc)	N	CTS	N	CTS	N	CTS	N	CTS
37.8/1.001	7007	46875	14014	46875	28028	46875	56056	46875
37.8	6272	42000	12544	42000	25088	42000	50176	42000
40.5	6272	45000	12544	45000	25088	45000	50176	45000
40.5*1.001	6272	45045	12544	45045	25088	45045	50176	45045
81	6272	90000	12544	90000	25088	90000	50176	90000
81*1.001	6272	90090	12544	90090	25088	90090	50176	90090
111.375/1.001	17836	351562 – 351563 (351526.5)Δ	35672	351562 – 351563 (351526.5)Δ	71344	351562 – 351563 (351526.5)Δ	142688	351562 – 351563 (351526.5)Δ
111.375	6272	123750	12544	123750	25088	123750	50176	123750
222.75/1.001	17836	703125	35672	703125	71344	703125	142688	703125
222.75	6272	247500	12544	247500	25088	247500	50176	247500
445.5/1.001	8918	703125	17836	703125	35672	703125	71344	703125
445.5	4704	371250	9408	371250	18816	371250	37632	371250
Other	6272	measured	12544	measured	25088	measured	50176	measured

Δ Fractional portion is 0.5: Dither between the values with a (1/2 duty Cycle).

Table C-6: 36 bits/Pixel: Recommended N and expected CTS for Audio Sample Frequency or Frame Rate of 48 kHz and multiples thereof

TMDS Character Rate	48 kHz		96 kHz		192 kHz		384 kHz	
(Mcsc)	N	CTS	N	CTS	N	CTS	N	CTS
37.8/1.001	9152	56250	18304	56250	36608	56250	73216	56250
37.8	6144	37800	12288	37800	24576	37800	49152	37800
40.5	6144	40500	12288	40500	24576	40500	49152	40500
40.5*1.001	8192	54054	16384	54054	32768	54054	65536	54054
81	6144	81000	12288	81000	24576	81000	49152	81000
81*1.001	6144	81081	12288	81081	24576	81081	49152	81081
111.375/1.001	11648	210937 – 210938 (210937.5)$^{\Delta}$	23296	210937 – 210938 (210937.5)$^{\Delta}$	46592	210937 – 210938 (210937.5)$^{\Delta}$	93184	210937 – 210938 (210937.5)$^{\Delta}$
111.375	6144	111375	12288	111375	24576	111375	49152	111375
222.75/1.001	11648	421875	23296	421875	46592	421875	93184	421875
222.75	6144	222750	12288	222750	24576	222750	49152	222750
445.5/1.001	5824	421875	11648	421875	23296	421875	46592	421875
445.5	5120	371250	10240	371250	20480	371250	40960	371250
Other	6144	measured	12288	measured	24576	measured	49152	measured

$^{\Delta}$ Fractional portion is 0.5: Dither between the values with a (1/2 duty Cycle).

Table C-7: 48 bits/Pixel: Recommended N and expected CTS for Audio Sample Frequency or Frame Rate of 32 kHz and multiples thereof

TMDS Character Rate	32 kHz		64 kHz		128 kHz		256 kHz	
(Mcsc)	N	CTS	N	CTS	N	CTS	N	CTS
50.4/1.001	4576	56250	9152	56250	18304	56250	36608	56250
50.4	4096	50400	8192	50400	16384	50400	32768	50400
54	4096	54000	8192	54000	16384	54000	32768	54000
54*1.001	4096	54054	8192	54054	16384	54054	32768	54054
108	4096	108000	8192	108000	16384	108000	32768	108000
108*1.001	4096	108108	8192	108108	16384	108108	32768	108108
148.5/1.001	11648	421875	23296	421875	46592	421875	93184	421875
148.5	4096	148500	8192	148500	16384	148500	32768	148500
297/1.001	11648	843750	23296	843750	46592	843750	93184	843750
297	4096	297000	8192	297000	16384	297000	32768	297000
594/1.001	5824	843750	11648	843750	23296	843750	46592	843750
594	3072	445500	6144	445500	12288	445500	24576	445500
Other	4096	measured	8192	measured	16384	measured	32768	measured

Table C-8: 48 bits/Pixel: Recommended N and expected CTS for Audio Sample Frequency or Frame Rate of 44.1 kHz and multiples thereof

TMDS Character Rate (Mcsc)	44.1 kHz		88.2 kHz		176.4 kHz		352.8 kHz	
	N	CTS	N	CTS	N	CTS	N	CTS
50.4/1.001	7007	62500	14014	62500	28028	62500	56056	62500
50.4	6272	56000	12544	56000	25088	56000	50176	56000
54	6272	60000	12544	60000	25088	60000	50176	60000
54*1.001	6272	60060	12544	60060	25088	60060	50176	60060
108	6272	120000	12544	120000	25088	120000	50176	120000
108*1.001	6272	120120	12544	120120	25088	120120	50176	120120
148.5/1.001	17836	468750	35672	468750	71344	468750	142688	468750
148.5	6272	165000	12544	165000	25088	165000	50176	165000
297/1.001	8918	468750	17836	468750	35672	468750	71344	468750
297	6272	330000	12544	330000	25088	330000	50176	330000
594/1.001	4459	468750	8918	468750	17836	468750	35672	468750
594	4704	495000	9408	495000	18816	495000	37632	495000
Other	6272	measured	12544	measured	25088	measured	50176	measured

Table C-9: 48 bits/Pixel: Recommended N and expected CTS for Audio Sample Frequency or Frame Rate of 48 kHz and multiples thereof

TMDS Character Rate	48 kHz		96 kHz		192 kHz		384 kHz	
(Mcsc)	N	CTS	N	CTS	N	CTS	N	CTS
50.4/1.001	6864	56250	13728	56250	27456	56250	54912	56250
50.4	6144	50400	12288	50400	24576	50400	49152	50400
54	6144	54000	12288	54000	24576	54000	49152	54000
54*1.001	6144	54054	12288	54054	24576	54054	49152	54054
108	6144	108000	12288	108000	24576	108000	49152	108000
108*1.001	6144	108108	12288	108108	24576	108108	49152	108108
148.5/1.001	11648	281250	23296	281250	46592	281250	93184	281250
148.5	6144	148500	12288	148500	24576	148500	49152	148500
297/1.001	5824	281250	11648	281250	23296	281250	46592	281250
297	6144	297000	12288	297000	24576	297000	49152	297000
594/1.001	5824	562500	11648	562500	23296	562500	46592	562500
594	5120	495000	10240	495000	20480	495000	40960	495000
Other	6144	measured	12288	measured	24576	measured	49152	measured

Appendix D Dynamic Auto Lipsync and Source Devices (Informative)

For a Source Device which does not have a separate (non-HDMI) audio output, the features Auto Lipsync (ALS, see Section 10.6) and Dynamic Auto Lipsync (DALS, see Section 10.7) are not relevant; it will always send synchronized video and audio on the HDMI output.

When the Source Device does have a separate audio output (e.g. analog stereo or SPDIF) to a legacy Amplifier, which does not have delay facilities (see Figure D-1), the device may (in certain situations) need to delay the audio on the audio output connected to the Amplifier. This delay should be set to match the video latency in the TV:

- If the TV does not support ALS or DALS, the user will have to manually set the delay in the Source Device.
- If both the TV and Source Device support ALS, the Source Device uses the video latency value read from the TV's EDID to set its audio delay to match the reported TV (average) latency.
- If both the TV and Source Device support DALS, the Source Device uses the video latency value carried in DALS messages to set its audio delay to match the reported TV (actual) latency.

Note – if the Amplifier has its own delay capability, it may be more appropriate to use that delay capability and to not apply delay in the Source Device.

Note – on its HDMI output, this Source Device will always send synchronized video and audio, irrespective of the delay values used on the separate audio output(s).

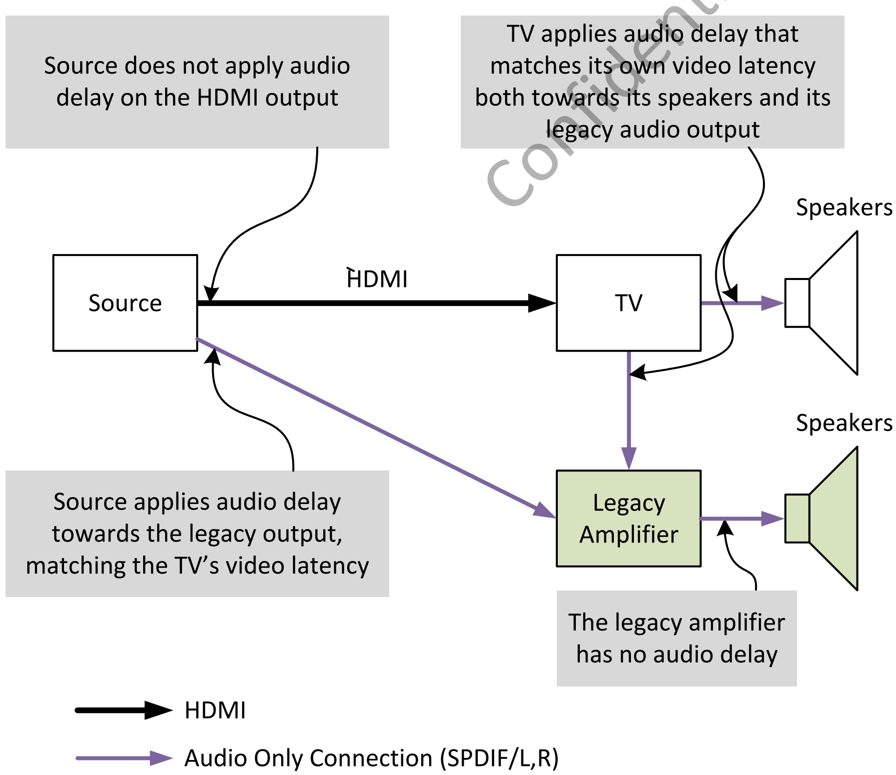


Figure D-1: Latency Compensation with Legacy Amplifier

Appendix E Signaling in AVI InfoFrame and VSIF for various Video Formats

This Appendix illustrates how the structures in the AVI InfoFrame, H14b VSIF and HF-VSIF are used when transmitting various Video Formats.

Table E-1 applies for 2D Video Formats, Table E-2 applies for 3D Video Formats (when none of features listed in Table 10-1 is active) and Table E-3 applies for 3D Video Formats when one (or more) of the features in Table 10-1 is active.

Table E-1: Signaling for 2D Video Formats

Video Format	AVI InfoFrame		VSIF
	VIC	Y2,Y1,Y0	
Traditional formats(1), (3)	1-64	000,001,010	none(2)
"21:9" (64:27)(3)	65-92 103-107	000,001,010	none(2)
2160p24/25/30Hz (1), (3), (6)	0	000,001,010	**H14b VSIF** with HDMI_Video_Format=001 HDMI_VIC=01..04
2160p25/30Hz (VIC 99,100)(3)	99, 100	000,001,010	none(2)
2160p50/60Hz (4:2:0)(3)	96, 97 101, 102 106, 107	011	none(2)
2160p50/60Hz (4:4:4)	96, 97 101, 102 106, 107	000,001,010	none(2)
Non-VIC format(1), (3)	0	000,001,010	none(2)
Not allowed(5)	*93-95,98*		

Notes
(1) Signaling for these formats is defined in H14b.
(2) Even though an H14b VSIF is not necessary for these use cases, when switching back from 3D to 2D, it might be useful to send an H14b VSIF with HDMI_Video_Format=0b000 (see Appendix F).
(3) When Deep Color is applied, the TMDS Character Rate will/might exceed 340 Mcsc.
(5) These VICs (as listed in Table 10-2) are not permitted to be used in 2D mode; instead the H14b VSIF and HDMI_VIC=01..04 signaling is used.
(6) These formats are defined as “4K” Video Formats in H14b Section 8.2.3.1.

Table E-2: Signaling for 3D Video Formats (when none of the features in Table 10-1 is active)

Video Format	AVI InfoFrame		VSIF
	VIC	**Y2,Y1,Y0**	
Traditional formats[(1), (3)]	1-64	000,001,010	**H14b VSIF** with HDMI_Video_Format=010 3D_Structure=FP[(4)]/TaB/SbS *if SbS*: 3D_Ext_Data *optional*: 3D_Meta (see H14b Appendix H)
"21:9" (64:27)[(3)]	65-92 103-107	000,001,010	
2160p24/25/30Hz [(3), (7)]	93-95 98	000,001,010	
2160p25/30Hz (VIC 99, 100)[(3)]	99, 100	000,001,010	
2160p50/60Hz (4:2:0)[(3)]	96, 97 101, 102 106, 107	011	
2160p50/60Hz (4:4:4)	96, 97 101, 102 106, 107	000,001,010	
Non-VIC format[(1), (3)]	0	000,001,010	

Notes
(1) Signaling for these formats is defined in H14b.
(3) When Deep Color is applied, the TMDS Character Rate might exceed 340 Mcsc.
(4) When 3D-Frame-Packing is applied, the TMDS Character Rate will/might exceed 340 Mcsc, and possibly might exceed 600 Mcsc.
(7) The 2D versions of these formats are defined as “4K” Video Formats in H14b Section 8.2.3.1.

Table E-3: Signaling for 3D Video Formats (when one or more features in Table 10-1 is active)

Video Format	AVI InfoFrame		VSIF
	VIC	**Y2,Y1,Y0**	
Traditional formats[(3)]	1-64	000,001,010	**HF-VSIF** with 3D_Valid=1 3D_F_Structure=FP[(4)]/TaB/SbS if SbS: 3D_F_Ext_Data *if "Dual View" and/or "Independent View" are active*: 3D_Additional_Info_Present=1 *if "3D OSD Disparity" is active*: 3D_Disparity_Data_present=1 *optional*: 3D_Meta (see H14b Appendix H)
"21:9" (64:27)[(3)]	65-92 103-107	000,001,010	
2160p24/25/30Hz [(3), (7)]	93-95 98	000,001,010	
2160p25/30Hz (VIC 99, 100) [(3)]	99, 100		
2160p50/60Hz (4:2:0)[(3)]	96, 97 101, 102 106, 107	011	
2160p50/60Hz (4:4:4)	96, 97 101, 102 106, 107	000,001,010	
Non-VIC format[(3)]	0	000,001,010	

Notes
(3) When Deep Color is applied, the TMDS Character Rate might exceed 340 Mcsc.
(4) When 3D-Frame-Packing is applied, the TMDS Character Rate might exceed 340 Mcsc, and possibly might exceed 600 Mcsc.
(7) The 2D versions of these formats are defined as “4K” Video Formats in H14b Section 8.2.3.1.

Example 1

A Source sends 2D video at 1920x1080p, 59.94/60 Hz, 16:9 aspect ratio, with VIC=16 in the AVI InfoFrame. No H14b VSIF or HF-VSIF is sent.

Then the Source switches to 3D side-by-side of the same format; the Source keeps sending the same AVI InfoFrame (VIC-field=16), and starts to send the H14b VSIF with HDMI_Video_Format=0b010 and 3D_Structure=0b1000 (side-by-side, half).

Finally, the Source additionally starts the 3D OSD Disparity feature; the Source keeps sending the same AVI InfoFrame (VIC-field=16), stops sending the H14b VSIF and instead sends an HF-VSIF, with 3D_Valid=1 and 3D_F_Structure=0b1000 (side-by-side, half), 3D_DisparityData_present=1 and appropriate DisparityData.

Example 2

A Source sends 2D video at 3840x2160p, 23.98/24 Hz, 16:9 aspect ratio. Since this is a format defined in HDMI 1.4b, the H14b VSIF is used with HDMI_VIC=3, and the VIC field in the AVI InfoFrame is set to 0.

Then the Source switches to 3D side-by-side of the same format; the Source updates the VIC-field in the AVI InfoFrame to 93, it continues to send the H14b VSIF (without HDMI_VIC but now with 3D signaling: HDMI_Video_Format=0b010 and 3D_Structure=0b1000 (side-by-side, half)).

Finally, the Source additionally starts the 3D OSD Disparity feature; the Source keeps sending the same AVI InfoFrame (VIC-field=93), stops sending the H14b VSIF, and starts to send the HF-VSIF, with 3D_Valid=1 and 3D_F_Structure=0b1000 (side-by-side, half), 3D_DisparityData_present=1 and appropriate DisparityData.

Appendix F Use of H14b VSIF for 3D-2D Transitions (Informative)

Context: HDMI 1.4b Section 8.2.3 defines the H14b VSIF for use with the transmission of 3D video. This section clarifies device behavior, specifically, to improve the user experience when switching from 3D to 2D with some HDMI 1.4b Sink devices, which have been found to remain in 3D mode even when the Source has stopped the transmission of the H14b VSIF.

When there is any change in the H14b VSIF, the Sink should begin to adapt its display processing accordingly within 1 second. This includes changes from “no H14b VSIF is being transmitted” to “H14b VSIF is being transmitted” and vice versa.

When a Source changes its transmission from 3D to 2D, the Source should signal the end of 3D transmission by sending an H14b VSIF with an HDMI_Video_Format that does not indicate 3D (0b000 or 0b001, depending on the new Video Format now being transmitted), after the change from 3D to 2D, for at least 2 seconds or until re-start of HDMI video is necessary. The background of this recommendation is the observation that the Sinks mentioned in the first paragraph of this section will respond properly (i.e. go to 2D mode) when receiving the H14b VSIF indicating non-3D.

When the Source stops transmitting the H14b VSIF, the Sink should interpret this as indicating that transmission of features described in this InfoFrame has stopped (e.g. transition from 3D, as previously signaled in the InfoFrame, to (default) 2D).