Video Electronics Standards Association

VESA DisplayID Standard

Version 2.0

11 September, 2017

www.vesa.org

Purpose

This Standard defines flexible data formats that organize interface and display configuration information in a structure stored within a display product. A host has access to the structure over the video interface connection. The host uses this data to automatically setup and optimize the video interface and image signals while facilitating plug and play operation with minimal or no user intervention.

Summary

DisplayID v2.0 describes the second-generation version of VESA *DisplayID Standard*. Use of *DisplayID v2.0* structures is intended to replace use of EDID structures to describe capabilities of new display devices. *DisplayID v2.0* contains several new features that better permit its use in a wide range of applications, including PC monitors, consumer television products, embedded displays (e.g., laptop LCD panels), and other display products. The DisplayID structure definition is intended to be a non-ambiguous, simple, and compact data declaration that provides information about modern displays that are used by the system. This information includes model and specific unit identification information, colorimetry, color depth, color space, native gamma EOTF, audio feature support, supported timings, and supported formats that are to be supplied to the host video source over an appropriate communications channel (e.g., VESA E-DDC). The Source device may then use this information to automatically configure itself to optimally support the display in use.

Notes: *DisplayID v2.0 retains the same structure as DisplayID v1.3, but has been updated with new data blocks that are designed to better accommodate modern display requirements and capabilities.*

*DisplayID structures are **not** directly backward-compatible with earlier EDID/E-EDID definitions, but do include many of the same and/or similar data field definitions from those earlier standards. In many cases, similar data fields have been extended to provide greater magnitude and/or precision than was possible with EDID/E-EDID.*

Contents

Purpose 1

Summary 1

Preface 8

Intellectual Property 8
Trademarks 8
Support for this Standard 8
Patents 9
Acknowledgments 9
Revision History 10

Section 1 Overview 11

1.1 Summary 11
1.2 Background and Changes from Previous EDID Definitions 11
1.3 Standard Objectives 12
1.4 Reference Documents 12
1.5 Terminology 14
1.6 Conventions 15
1.6.1 Precedence 15
1.6.2 Keywords 15
1.6.3 Numbering 15
1.6.4 Data Format Conventions 16

Section 2 DisplayID Structure and Format Definition 17

Section 3 Data Blocks – Description and Usage 22

3.1 Data Block Format and Definitions 24
3.2 DisplayID Section – A Visual Representation 26

Section 4 Data Block Definitions 28

4.1 Product Identification Data Block 29
4.1.1 Manufacturer/Vendor ID Field 30
4.1.2 Product ID Code Field 30
4.1.3 Serial Number Field 31
4.1.4 Week of Manufacture/Model Tag and Year of Manufacture/ Model Year Fields 32
4.1.5 Size of Product Name String and Product Name String Fields 33

4.2 Display Parameters Data Block 34
4.2.1 Horizontal Image Size and Vertical Image Size Fields 36
4.2.2 Horizontal Pixel Count and Vertical Pixel Count Fields (Native Format) 37
4.2.3 Feature Support Flags Field 37
4.2.4 Native Color Chromaticity Fields 39
4.2.5 Native Luminance-related Fields 42
4.2.6 Native Color Depth and Display Device Technology Fields 43
4.2.7 Native Gamma EOTF Field 44
4.3 Video Timing Mode-related Data Blocks 44
4.3.1 Type VII Timing – Detailed Timing Data Block 45
4.3.2 Type VIII Timing – Enumerated Timing Code Data Block 50
4.3.3 Type IX Timing – Formula-based Data Block 52
4.4 Dynamic Video Timing Range Limits Data Block 54
4.5 Display Interface Features Data Block 55
4.5.1 Supported Interface Color Depth-related Fields 57
4.5.2 Minimum Pixel Rate at Which YCbCr 4:2:0 Encoding Is Supported Field 60
4.5.3 Supported Interface Audio Capability and Feature Flags Field 60
4.5.4 Supported Interface Color Space and EOTF-related Fields 61
4.6 Stereo Display Interface Data Block 65
4.6.1 Stereo Interface Method Code Field 67
4.7 Tiled Display Topology Data Block 76
4.7.1 Tiled Display and Tile Capabilities Field 77
4.7.2 Tiled Display Topology and Tile Location Fields 78
4.7.3 Tile Size Field 80
4.7.4 Tile Pixel Multiplier and Tile Bezel-related Information Fields 81
4.7.5 Tiled Display Topology ID-related Fields 82
4.8 ContainerID Data Block 84
4.8.1 ContainerID Field 85
4.9 Vendor-specific Data Block 86
4.9.1 Manufacturer/Vendor ID Field 87
4.10 CTA DisplayID Data Block 88

Appendix A Fixed-length DisplayID Section Example 90

Appendix B VESA Organization Vendor-specific Data Block 96

Appendix C Scan Orientation Clarification 98

Tables

Table 1: Main Contributors to *DisplayID v2.0* 9
Table 2: Revision History 10

Table 1-1: Reference Documents (Normative) 12
Table 1-2: Reference Documents (Informative) 13
Table 1-3: Terminology 14
Table 1-4: Keywords 15
Table 1-5: Numbering 15
Table 1-6: Data Format Conventions 16

Table 2-1: Variable-length DisplayID Section Structure 18
Table 2-2: Fixed-length DisplayID Section Structure 19
Table 2-3: DisplayID Base Section Review 20
Table 2-4: DisplayID Extension Section Review 21

Table 3-1: Data Block Tag Allocation 23
Table 3-2: DisplayID Data Block Format 24
Table 3-3: DisplayID Section Visual Representation 26

Table 4-1: Product Identification Data Block Fields 29
Table 4-2: Manufacturer/Vendor ID Field Format 30
Table 4-3: Product ID Code Field Format 30
Table 4-4: Serial Number Field Format 31
Table 4-5: Week of Manufacture/Model Tag and Year of Manufacture/Model Year Field Formats 32
Table 4-6: Size of Product Name String and Product Name String Field Formats 33
Table 4-7: Display Parameters Data Block Fields 34
Table 4-8: Horizontal Image Size and Vertical Image Size Field Formats 36
Table 4-9: Horizontal Pixel Count and Vertical Pixel Count Field Formats (Native Format) 37

Table 4-10: Feature Support Flags Field Format 38
Table 4-11: Native Color Chromaticity Field Formats 39
Table 4-12: 12-bit Binary-format Fraction Representation (Examples) 41
Table 4-13: Native Luminance-related Field Formats 42
Table 4-14: Native Color Depth and Display Device Technology Field Format 43
Table 4-15: Native Gamma EOTF Field Format 44
Table 4-16: Video Timing Mode-related Data Block Fields 44
Table 4-17: Type VII Timing – Detailed Timing Data Block Format 46
Table 4-18: Type VII Detailed Timing Descriptor Format 46
Table 4-19: Type VIII Timing – Enumerated Timing Code Data Block Format 50
Table 4-20: Type IX Timing – Formula-based Data Block Format 52
Table 4-21: Type IX Formula-based Timing Descriptor Format 53
Table 4-22: Dynamic Video Timing Range Limits Data Block Fields 54
Table 4-23: Display Interface Features Data Block Fields 55
Table 4-24: Supported Interface Color Depth-related Field Formats 57
Table 4-25: Minimum Pixel Rate at Which YCbCr 4:2:0 Encoding Is Supported Field Format 60
Table 4-26: Supported Interface Audio Capability and Feature Flags Field Format 60
Table 4-27: Supported Interface Color Space and EOTF-related Field Formats 61
Table 4-28: Stereo Display Interface Data Block Fields 65
Table 4-29: Stereo Interface Method Codes 67
Table 4-30: Frame/Field Sequential Stereo Interface Method Stereo Display Interface Data Block Fields 68
Table 4-31: Side-by-side Stereo Interface Method Stereo Display Interface Data Block Fields 69
Table 4-32: Pixel-interleaved Stereo Interface Method Stereo Display Interface Data Block Fields 70
Table 4-33: Dual Interface, Left and Right Separate Stereo Interface Method Stereo Display Interface Data Block Fields 71
Table 4-34: Multi-view Stereo Interface Method Stereo Display Interface Data Block Fields 73
Table 4-35: Stacked Frame Stereo Interface Method Stereo Display Interface Data Block Fields 74

Table 4-36: Proprietary Stereo Interface Methods Stereo Display Interface Data Block Fields 75
Table 4-37: Tiled Display Topology Data Block Fields 76
Table 4-38: Tiled Display and Tile Capabilities Field Format 77
Table 4-39: Tiled Display Topology and Tile Location Field Formats. 78
Table 4-40: Tile Size Field Formats 80
Table 4-41: Tile Pixel Multiplier and Tile Bezel-related Information Field Formats 81
Table 4-42: Tiled Display Topology ID-related Field Formats. 82
Table 4-43: Tiled Display Manufacturer/Vendor ID Field Format 82
Table 4-44: Tiled Display Product ID Code Field Format 83
Table 4-45: Tiled Display Serial Number Field Format 83
Table 4-46: ContainerID Data Block Fields 84
Table 4-47: ContainerID Field Format 85
Table 4-48: Vendor-specific Data Block Fields 86
Table 4-49: Manufacturer/Vendor ID Field Format 87
Table 4-50: CTA Data Block Encapsulation 88

Table A-1: Fixed-length DisplayID Section Example 90

Table B-1: VESA Organization Vendor-specific Data Block 96

Figures

Figure 4-1: Video Timing Parameter Definitions 49

Figure C-1: Scan Orientation Settings for Example 1920x1080 Landscape Panel at Various Viewing Configurations 98

Figure C-2: Scan Orientation Settings for Example 1080x1920 Portrait Panel at Various Viewing Configurations 98

Figure C-3: Scan Orientation Settings Used with an Example Landscape 1920x1080 Timing to Implement Landscape Display Devices 99

Figure C-4: Scan Orientation Settings Used with an Example Portrait 1080x1920 Timing to Implement Landscape Display Devices 99

Preface

Intellectual Property

While every precaution has been taken in the preparation of this Standard, the Video Electronics Standards Association and its contributors assume no responsibility for errors or omissions, and make no warranties, expressed or implied, of functionality or suitability for any purpose.

Trademarks

All trademarks used within this Standard are the property of their respective owners.

DisplayPort is a trademark and VESA is a registered trademark of the Video Electronics Standards Association.

CTA is a trademark of the Consumer Technology Association.

CIE is a trademark of the International Commission on Illumination.

DVI is a trademark of the Digital Display Working Group.

HDMI and High-Definition Multimedia Interface are trademarks or registered trademarks of HDMI Licensing Administrator, Inc. in the United States and other countries.

I^2C is a trademark of NXP Semiconductors (formerly Philips Semiconductors).

Support for this Standard

Clarifications and application notes might exist that further support this Standard. To obtain the latest Standard and any support documentation, contact VESA.

If you have a product that incorporates DisplayID, ask the company that manufactured your product for assistance. If you are a manufacturer, VESA can assist you with any clarification that you might require.

Submit all comments or reported errors to support@vesa.org.

Patents

VESA® draws attention to the fact that compliance with this Standard might involve the use of a patent or other intellectual property right (collectively, "IPR") concerning DisplayID. VESA takes no position concerning the evidence, validity, and/or scope of this IPR. At the time of publication, there are currently no IPRs specific to DisplayID to list in this Standard.

Attention is drawn to the possibility that some of the elements of this VESA Standard might be the subject of IPRs external to this Standard other than those identified above. VESA shall not be held responsible for identifying any or all such IPRs, and has made no inquiry into the possible existence of any such IPRs.

THIS STANDARD IS BEING OFFERED WITHOUT ANY WARRANTY WHATSOEVER, AND IN PARTICULAR, ANY WARRANTY OF NON-INFRINGEMENT IS EXPRESSLY DISCLAIMED. ANY IMPLEMENTATION OF THIS STANDARD SHALL BE MADE ENTIRELY AT THE IMPLEMENTER'S OWN RISK, AND NEITHER VESA, NOR ANY OF ITS MEMBERS OR SUBMITTERS, SHALL HAVE ANY LIABILITY WHATSOEVER TO ANY IMPLEMENTER OR THIRD PARTY FOR ANY DAMAGES OF ANY NATURE WHATSOEVER DIRECTLY OR INDIRECTLY ARISING FROM THE IMPLEMENTATION OF THIS STANDARD.

Acknowledgments

This Standard would not have been possible without the efforts of VESA's DisplayPort Task Group. In particular, Table 1 lists the individuals and their companies that contributed significant time and knowledge to this version of the Standard.

Table 1: Main Contributors to *DisplayID v2.0*

Company	Name	Contribution
Advanced Micro Devices, Inc.	Aric Cyr	Technical Contributor
	Syed A. Hussain	Task Group Vice Chair and Document Editor
Analogix Semiconductor, Inc.	Mehran Badii	Technical Contributor
Apple, Inc.	Bob Ridenour	Task Group Chair
Avatar Tech Pubs	Trish McDermott	Technical Writer
Broadcom Limited	Chris Pasqualino	Technical Contributor
EIZO, Inc.	Bamba Yoshinobu	Technical Contributor
JVC Kenwood Corporation	Takayuki Kashahara	Technical Contributor
NVIDIA Corporation	Kai Chen	Technical Contributor
	Robert Schutten	Technical Contributor
Samsung Display America Lab	Dale Stolitzka	Technical Contributor
Sony Corporation	Robert Blanchard	Technical Contributor

Revision History

Table 2: Revision History

Date	Version	Description
September 11, 2017	2.0	Initial release of the Standard.

1 Overview

1.1 Summary

This Standard, henceforth referred to as *DisplayID v2.0*, defines a framework and data structures for describing physical and performance attributes and supported timings. *DisplayID v2.0* also provides other relevant information regarding a display product to enable the video source, and driving that display to automatically configure itself for optimum display usage. At least one variable-length data structure (i.e., the "DisplayID Base" or "Base section") shall be provided under this system to provide basic identification, features, supported timings, and other information regarding a given display product to its host and/or video source device.

A method shall also be described for extending this basic information set through additional variable-length structures, referred to as "DisplayID Extensions," that comprise various forms of data blocks. The DisplayID data structure shall be independent of the communication protocol (i.e., video interface) that is used between the host and display product.

1.2 Background and Changes from Previous EDID Definitions

Note: *The following is provided as an informative summary only; this section does not establish requirements and/or definitions under DisplayID v2.0.*

The VESA Extended Display Identification Data (EDID) Standard, first introduced in 1994, established the most widely used data format for conveying display identification and description information to video sources (e.g., PCs and set-top boxes), which permitted such devices to configure themselves for optimum display use. EDID continues to be widely used within the display industry, particularly for computer displays and digital television, despite an implicit assumption in the EDID Base definition that the display is a direct-view CRT type.

DisplayID v2.0, intended as a true second-generation EDID, is a more-flexible and extensible display identification data format that is designed to meet the needs of a wide range of display types, technologies, and applications. The most obvious difference between *DisplayID v2.0* and its EDID predecessors is the use of a modular structure that is based on the concept of "data blocks," individually defined and self-contained data formats that each provide a specific set of related display information.

Data block definitions may be modified or added to the overall DisplayID system, as needed. *DisplayID v2.0* also allows for data blocks to be defined by other organizations, in particular the CTA. In addition, these data blocks may, within certain restrictions, be "mixed and matched," as needed, throughout the overall DisplayID structure. Use of this system also means that there are no longer separately defined and unique extensions, as was the case with EDID. Like the DisplayID Base section itself, DisplayID Extensions are created, as needed, from the defined data blocks.

A number of data block definitions have been developed to support *DisplayID v2.0*. In addition, the DisplayID system expects to leverage previously developed CTA data blocks for use in the CTA EDID Extension, as defined by *CTA-861-G* and higher. With the range of available data blocks, it is expected that the basic DisplayID system shall be applicable to desktop monitors, television monitors, receivers, and display devices (e.g., providing ID information on the LCD panel used in a laptop), as well as non-display devices that might be connected to video sources (e.g., repeaters or interface-translator products).

1.3 Standard Objectives

VESA developed *DisplayID v2.0* to meet, exceed, and/or complement certain criteria to provide information in a compact format that can allow a graphics subsystem to be configured based on the attached display's capabilities.

1.4 Reference Documents

Table 1-1 and Table 1-2 list the various reference documents that are used within this Standard. Users of this Standard are advised to ensure that they have the latest versions/revisions of reference standards/specifications and documents.

Table 1-1: Reference Documents (Normative)

Document	Version/ Revision[a]	Publication Date	Referenced As
Adobe® RGB (1998) Color Image Encoding	Version 2005-05	May 2005	*Adobe RGB*
ANSI/CTA-861, A DTV Profile for Uncompressed High Speed Digital Interface – see global.ihs.com (formerly known as *CEA-861*)	Revision G	November 1, 2016	*CTA-861-G*
BT.601, Studio encoding parameters of digital television for standard 4:3 and wide screen 16:9 aspect ratios[b]	Version 7	March 2011	*ITU-R BT.601*
BT.709, Parameter values for the HDTV standards for production and international programme exchange[b]	Version 6	June 2015	*ITU-R BT.709*
BT.1886, Reference electro-optical transfer function for flat panel displays used in HDTV studio projection[b]	Version 0	March 2011	*ITU-R BT.1886*
BT.2020, Parameter values for ultra-high definition television systems for production and international programme exchange[b]	Version 2	October 2015	*ITU-R BT.2020*
CIE 15.2 Colorimetry	Revision 86	1986	*CIE 15.2 r86*
CIE 1931 xy Chromaticity Diagram	Revision 1931	1931	*CIE 1931*
*CIE ISO 11664-5:2016, Colorimetry – Part 5: CIE 1976 L*u*v* Colour Space and u', v' Uniform Chromaticity Scale Diagram*	Revision 2016	September 2016	*CIE 1976*
IEC 61966-2-1:1999, Multimedia systems and equipment – Colour measurement and management – Part 2-1: Colour management – Default RGB colour space - sRGB		October 18, 1999	*IEC 61966-2-1*

Table 1-1: Reference Documents (Normative) (Continued)

Document	Version/ Revision[a]	Publication Date	Referenced As
IEEE 754-2008, IEEE Standard for Floating-Point Arithmetic		August 29, 2008	*IEEE 754*
ISO/IEC 8859-1:1998, Information Technology – 8-bit single-byte coded graphic character sets – Part 1: Latin alphabet No. 1 – ASCII codes		April 16, 1998	*ISO/IEC 8859-1*
SMPTE RP 431-2, D-Cinema Quality – Reference Projector and Environment[c]	2011	April 6, 2011	*SMPTE RP 431-2*
SMPTE ST 2084, Dynamic Range Electro-Optical Transfer Function of Mastering Reference Displays[c]	2014	August 16, 2014	*SMPTE ST 2084*
VESA Display Monitor Timing (DMT) Standard[d]	Version 1.13	February 8, 2013	*DMT Standard*
VESA Coordinated Video Timing (CVT) Standard[d]	Version 1.2	February 8, 2013	*CVT Standard*

a. All references include subsequently published errata, specification change notices or engineering change notices, etc.
b. Published by International Telecommunication Union (ITU). See www.itu.int.
c. Published by Society of Motion Picture and Television Engineers.
d. See www.vesa.org/vesa-member/downloads/.

Table 1-2: Reference Documents (Informative)

Document	Version/ Revision[a]	Publication Date	Referenced As
Digital Display Working Group, *Digital Visual Interface (DVI) Specification*	Revision 1.0	April 2, 1999	*DVI r1.0*
High-Definition Multimedia Interface (HDMI) Specification – see www.hdmi.org	Version 1.4b Version 2.0b	October 11, 2011 March 3, 2016	*HDMI v1.4b* *HDMI v2.0b* *HDMI Specification*
IETF RFC 4122, A Universally Unique IDentifier (UUID) URN Namespace		July 2005	*IETF RFC 4122*
LVDS (defined by ANSI/TIA/EIA-644-A)		2002	
VESA DisplayPort (DP) Standard[b]	Version 1.4[c]	February 23, 2016	*DP Standard*
VESA Enhanced Display Data Channel (E-DDC) Standard[b]	Version 1.3	September 11, 2017	*E-DDC Standard*
VESA Enhanced Extended Display Identification Data (E-EDID) Standard[b]	Release A.2	September 25, 2006	*E-EDID Standard*
VESA Glossary of Terms[b]	Current	Current	
VESA Intellectual Property Rights (IPR) Policy[b]	200D	March 27, 2017	

a. All references include subsequently published errata, specification change notices or engineering change notices, etc.
b. See www.vesa.org/vesa-member/downloads/.
c. This version of the referenced Standard is correct at the time of publication of this Standard. In the event that a later version of the referenced Standard is published, reference should be made to the latest published version.

1.5 Terminology

DisplayID v2.0 uses slightly different terminology to refer to the overall DisplayID structure's components than earlier EDID standards. Table 1-3 lists several key definitions.

Table 1-3: Terminology

Term	Definition
cd/m^2	Candela per square meter (formerly referred to as "nits").
Data Block or Block	Defined set of related information that is used to construct the DisplayID sections. Except for a few bytes at the beginning and end of each section (e.g., the Checksum bytes or the Structure Version/Revision byte), all *DisplayID v2.0* information shall be supplied as part of a data block. ***Note:*** *Although DisplayID v2.0 defines many of the data blocks that can be used in DisplayID, other data blocks, including those defined by other standards organizations (such as the CTA), may be encountered within a given DisplayID structure.*
Descriptor(s)	Collection of fields, strings, elements, flags, and/or tags.
Element(s)	Key piece(s) of information regarding the display. May refer to a single bit flag or an entire section. See Section 4.1.1 for an example.
Field	One or more contiguous bits or bytes within a data block that are used to convey a particular piece of information. For example, the Horizontal Pixel Count field within the Display Parameters data block comprises two bytes that specify the total number of physical pixels that the display device provides (see Section 4.2.2). May refer to a group of related flags within a given byte, –or– group of bits within a given byte, that provide a numeric value.
Flag(s)	Single bits that, when set or cleared, provide information regarding a given parameter or feature in "yes/no," "present/not present," terms, etc. Indicate a Boolean choice of support or non-support. Typical elements refer to a single flag or collection of flags, any or all of which may be supported in DisplayID. See Section 4.1.4 for an example.
Section	Collection of variable-length data blocks of up to 256-byte portions of the DisplayID structure. Comprises the Base section, as well as extensions if extensions are provided. All DisplayID structures shall have a Base section, and may also include one or more extension sections.
String	Contiguous bytes that are to be interpreted as text information (i.e., a series of characters that are read in the order in which they appear within the DisplayID block). Unless otherwise specified, all strings provided under DisplayID are provided in standard ASCII format (*ISO/IEC 8859-1*; see Section 1.3).
Structure	Body of information provided per *DisplayID v2.0* (i.e., the DisplayID Base of up to 256 bytes of information (the "Base section"), plus any and all extensions, each of which may be up to 256 bytes in length). Provided as one contiguous space that is calculated from the sum of all individual section sizes.
Tag(s)	Used when only a single choice is supported or referenced within a block (e.g., block identifiers). See Section 4.1.1 for an example.

1.6 Conventions

1.6.1 Precedence

If there is a conflict between text, figures, and tables, the precedence shall be tables, figures, and then text.

1.6.2 Keywords

Table 1-4 lists keywords that differentiate between the levels of requirements and options within this Standard.

Table 1-4: Keywords

Keyword	Definition
May	Indicates a choice with no implied preference.
Optional	Describes features not mandated by this Standard. However, if an optional feature is implemented, the feature shall be implemented as defined by this Standard (optional normative).
RESERVED	Indicates reserved bits, bytes, words, fields, and code values that are set aside for future standardization. Their use and interpretation may be specified by future extensions to this Standard and, unless otherwise stated, shall not be used or adapted by vendor implementation. A reserved bit, byte, word, or field shall be cleared to 0 by the sender and shall be ignored by the receiver. RESERVED field values shall not be sent by the sender and, if received, shall be ignored by the receiver.
Shall	Indicates a mandatory requirement. Designers are mandated to implement all such requirements to ensure interoperability with other compliant devices.
Should	Indicates flexibility of choice with a preferred alternative. Equivalent to the phrases “is recommended” and “it is recommended that.”

1.6.3 Numbering

Table 1-5 lists the different types of numbering used within this Standard.

Table 1-5: Numbering

Numbering Format	Definition
Binary	Numbers that are immediately followed by a lowercase “b” (e.g., 01b).
Decimal	Numbers that are not immediately followed by either a “b” or “h” (e.g., 5).
Hexadecimal	Numbers that are immediately followed by a lowercase “h” (e.g., A0h).

1.6.4 Data Format Conventions

The DisplayID structure is designed to be compact in its representation of data to fit the most information into a limited space. To accomplish this, variable data lengths are used according to the particular element's needs. These include fields from a single bit up to 256 bytes in length. Except where explicitly stated, the conventions listed in Table 1-6 shall be used in all cases.

Table 1-6: Data Format Conventions

Data Length	Convention Used	Example
1 to 7 bits	Stored in the order stated.	
8 bits (1 byte)	Stored at the location stated.	
16 bits (2 bytes)	Bytes are in a binary format (not binary-coded decimal format) that is stored at locations that are specified with the least significant byte (LSB) in the first location.	1280 decimal = 0500h. Stored 00h at the first location and 05h at the next location.
Character string	Bytes are ASCII, stored in the order in which they appear within the string.	"ACED" stored 41h at the first location, 43h at the second location, 45h at the third location, and 44h at the fourth location.

2 DisplayID Structure and Format Definition

DisplayID, like earlier EDID standards, is based on the concept that essential display identification and configuration data is declared in one or more contiguous sections. DisplayID is divided into 256 variable-length sections of up to 256 bytes each. The first section is referred to as the Base EDID in the original EDID nomenclature, –or– DisplayID Base as of *DisplayID v2.0*. Subsequent sections are referred to as "extensions."

DisplayID sections are collections of related elements, listed individually or grouped into data blocks and further into sub-blocks, referred to as "descriptors."

Addresses for each descriptor within a data block are provided based on the beginning of that block, in hexadecimal format. Addresses for individual field offsets within a descriptor are provided in decimal-format bytes or bits.

Under *DisplayID v2.0*, however, the entire content, including the Base section and any extensions, may be constructed in a flexible manner from any number of elements and/or predefined data blocks or descriptors. This method is derived from, and intended to be compatible with, a method first developed by the CTA in its CTA EDID Extension Block, Version 3 (see *CTA-861-G*). In defining that extension, the CTA developed a powerful model in which related display data elements may be collected into variable-length data blocks, each with a unique identifying tag, which may then be mixed and matched, as needed, in the creation of a CTA extension. VESA has since adopted this model for its DisplayID structure. The entire DisplayID structure provided here, including the Base section and any and all extensions that may appear following the Base section, can be allocated, as needed, using defined data block structures (both VESA and CTA definitions) that have only a few restrictions.

A display product's DisplayID structure shall identify at least one Video Timing mode, provide details on that timing, and declare that Video Timing mode as the preferred mode for that display product. The identification of a timing mode as preferred, the definition of preferred, and the prioritizing of Video Timing modes within the DisplayID structure, are described in Section 4.3.

Certain elements are needed in the Base section, and certain elements may appear only once within the entire DisplayID structure. These restrictions are detailed within the individual block definitions. Table 2-1 describes the format of a variable-length DisplayID structure in which the Checksum byte immediately follows the data block's end.

Table 2-1: Variable-length DisplayID Section Structure

Offset	Value	Description/Format	Mandatory
00h	20h	**DisplayID Structure** Version 2, Revision 0.	✔
01h	00h – FBh	**Bytes in Section** Section length, excluding the five mandatory bytes. Value is *N* - 5.[a]	✔
03h – 02h		**Display Product Primary Use Case and Extension Count**	
02h	0h – Fh	**Display Product Primary Use Case**	✔
03h	00h – FFh	**Extension Count** RESERVED. Cleared to all 0s for the Extension section and total Extension Count within the Base section.	✔
04h	Block	**Data Block** See Section 4.	
…		**…**	
(*N* - 2)h[a]	Block	**Last Byte of Last Valid Data Block**	
(*N* - 1)h[a]	00h – FFh	**Checksum**	✔

a. N represents the total number of bytes within the DisplayID section.

The primary difference between Base and Extension sections is that in an extension section, the Extension Count bytes are cleared to all 0s. The Display Product Primary Use Case field shall be declared only in the first DisplayID section.

The DisplayID framework allows a section to be defined as a fixed size with unused bytes that are filled with dummy values (0s) between the section's last valid data block and its last byte, the Checksum byte. Table 2-2 describes the format of a fixed-length DisplayID structure.

Note: *See Appendix A for a fixed-length DisplayID section example.*

Table 2-2: Fixed-length DisplayID Section Structure

Offset	Value	Description/Format	Mandatory
00h	20h	**Display Structure** Version 2, Revision 0.	✔
01h	00h – FBh	**Bytes in Section** Section length, excluding the five mandatory bytes. Value is *N* - 5.[a]	✔
03h – 02h		**Display Product Primary Use Case and Extension Count**	
02h	0h – Fh	**Display Product Primary Use Case**	✔
03h	00h – FFh	**Extension Count** RESERVED. Cleared to all 0s for the Extension section and total Extension Count within the Base section.	✔
04h	Block	**Data Block** See Section 4.	
…		**…**	
…h	Block	**Last Byte of Last Valid Data Block**	
(… + 1)h	00h	**Fill Data**	
…		**Fill Data**	
(*N* - 2)h[a]	00h	**Fill Data**	
(*N* - 1)h[a]	00h – FFh	**Checksum**	✔

a. N represents the total number of bytes within the DisplayID section.

Table 2-3 and Table 2-4 describe the DisplayID Base and Extension section elements, respectively.

Table 2-3: DisplayID Base Section Review

Offset	Bit #	Register
00h	**DisplayID Structure Version 2, Revision 0**	
	3:0	**Revision** 0h = Revision 0.
	7:4	**Version** 0h = RESERVED. 2h = Version 2.
01h	7:0	**Bytes in Section** Section length, excluding the five mandatory bytes. Value is *N* - 5.[a]
02h	**Display Product Primary Use Case**	
	3:0	**Display Product Primary Use Case** Use case ranges from 0 through 15. 0h = Used for Extension sections that have the same primary use case as the Base section. 1h = Test Structure; test equipment only. 2h = None of the listed primary use cases; generic display. 3h = Television (TV) display. 4h = Desktop productivity display. 5h = Desktop gaming display. 6h = Presentation display. 7h = Head-mounted Virtual Reality (VR) display. 8h = Head-mounted Augmented Reality (AR) display. 9h through Fh = RESERVED.
	7:4	RESERVED, cleared to all 0s.
03h	7:0	**Extension Count** RESERVED. Cleared to all 0s for the Extension section and total Extension Count within the Base section. Value ranges from 00h through FFh.
04h	7:0	**Data Block** See Section 3.
…		**…**
(*N* - 1)h[a]		**Checksum** Checksum ranges from 0 through 255. Value ranges from 00h through FFh.

a. N represents the total number of bytes within the DisplayID section.

Table 2-4: DisplayID Extension Section Review

Offset	Bit #	Register
00h	**DisplayID Structure Version 2, Revision 0**	
	3:0	**Revision** 0h = Revision 0.
	7:4	**Version** 0h = RESERVED. 2h = Version 2.
01h	7:0	**Bytes in Section** Section length, excluding the five mandatory bytes. Value is *N* - 5.[a]
02h	7:0	**Extension Section** Value is 00h.
03h	7:0	**Extension Count** Value ranges from 01h through FFh.
04h	7:0	**Data Block** See Section 3.
…		**…**
(*N* - 1)h[a]		**Checksum** Value ranges from 00h through FFh.

a. N represents the total number of bytes within the DisplayID section.

3 Data Blocks – Description and Usage

As noted in Section 2, the DisplayID structure differs from the earlier EDID structure. Virtually all information provided under DisplayID is provided in the form of data blocks. Data blocks are predefined modules containing a given set of related information. With certain restrictions, data blocks may be used as needed and at the manufacturer's discretion throughout the DisplayID structure.

Typically, data blocks do not have a fixed length, although some specific block definitions may establish a fixed length for that type of block. Data blocks shall be distinguished through a block tag, which is always the first byte. The second byte contains block revision- and data block-related information. Following that byte shall be a single byte that defines the block length, as the remaining number of bytes to follow in that block up to a maximum of 248 bytes, while excluding the mandatory tag, block revision, and length bytes. This shall permit a single data block to fill the remainder of any 256-byte section of the DisplayID structure – a single 251-byte data block, plus the mandatory five bytes, headers, and checksum, for that DisplayID section.

VESA and the CTA both have defined data blocks. To simplify distinguishing between them, the range of possible block tags has been evenly divided so that the most significant bit (bit 7) serves as a VESA/CTA flag. VESA controls tags within the range of 0 through 127 decimal. The CTA or other organizations control tags within the range of 128 through 255 decimal. Although VESA and the CTA may both provide information describing the other's data blocks, as is done within this section and Section 4, the information is provided only for reference. To ensure compliance, all DisplayID structure implementers shall consult the relevant standards of the appropriate organization for the data block(s) in question.

Notes: *VESA-defined data block specifications are provided in Section 4. For editing and revision purposes, however, each shall be considered as a separate specification and assigned a revision number that applies only to that data block. The overall DisplayID version/revision number, as indicated on the cover page of this Standard, applies to all sections except for the data block specifications. Readers are cautioned to note the document revision date on the cover page, and to check with the VESA office to ensure that they are using the current revision/version of all related standards/specifications.*

Section 4.10 describes how a CTA data block can be mapped within a DisplayID data block.

Table 3-1 lists the data blocks that are defined as of *DisplayID v2.0*.

Table 3-1: Data Block Tag Allocation

Block Tag	Data Block Name	Mandatory Block[a]	Defined in
1Fh – 00h	**RESERVED** RESERVED for legacy data blocks for DisplayID Structure *v1.x*. and shall **not** be used with new DisplayID Structure *v2.x*.		
20h	**Product Identification Data Block**	Yes	Section 4.1
21h	**Display Parameters Data Block**	Yes, for Display Use	Section 4.2
22h	**Type VII Timing – Detailed Timing Data Block**	Yes, for Display Use	Section 4.3.1
23h	**Type VIII Timing – Enumerated Timing Code Data Block**	No	Section 4.3.2
24h	**Type IX Timing – Formula-based Timing Data Block**	No	Section 4.3.3
25h	**Dynamic Video Timing Range Limits Data Block**	No	Section 4.4
26h	**Display Interface Features Data Block**	Yes, for Display Use	Section 4.5
27h	**Stereo Display Interface Data Block**	No	Section 4.6
28h	**Tiled Display Topology Data Block**	No	Section 4.7
29h	**ContainerID Data Block**	Yes, for Multi-function Devices	Section 4.8
7Dh – 2Ah	**RESERVED for Additional VESA-defined Data Blocks**		
7Eh	**Vendor-specific Data Block**	No	Section 4.9
80h – 7Fh	**RESERVED**		
81h	**CTA DisplayID Data Block**	No	Section 4.10
FFh – 82h	**RESERVED** RESERVED for additional data blocks related to external standards organization(s).		

a. "Yes, for Display Use" indicates that the Display Product Primary Use Case field of the DisplayID structure for the relevant product shall be populated with a value within the range of 2h through 8h.

3.1 Data Block Format and Definitions

All *DisplayID v2.0* data blocks follow the basic format listed in Table 3-2. The block's first byte contains an 8-bit tag (see Table 3-1), the second byte contains the block revision and data block, and the third byte contains the data block's length. Three bits of the data block revision number are provided as the lower three bits of the second byte; the byte's remaining five bits may be used as specified in the particular data block definition in question. Following these three mandatory bytes shall be the actual payload of information conveyed by that block, per its specification as provided here or elsewhere. Eight payload length bits are provided in the third byte of each data block. However, the maximum length of any data block is 251 bytes, –or– a full 256-byte section. (For example, in this case, there would be 251 bytes for the data block itself, comprising a 3-byte data block header plus 248-byte payload, plus the four mandatory section header bytes, plus the Checksum byte that is needed for all sections.)

Note: *Under this system, 00h shall be a valid payload length It is possible that a data block may be defined such that its mere presence conveys significant information to the Source device, –or– that all necessary information is conveyed in the second byte's five block-specific bits.*

Table 3-2: DisplayID Data Block Format

Offset	Bit #	Description/Format	Flag	Tag
00h	7:0	**Data Block Identification** Value ranges from 00h through FFh.		✔
01h	**Block Revision and Other Data**			
	2:0	**Block Revision** Revision ranges from 0 through 7. 000b = Revision 0 (default).		
	7:3	**RESERVED** Block-specific. Cleared to all 0s.	✔	✔
02h	7:0	**Number of Payload Bytes** Number of payload bytes within the data block ranges from 0 through 248. Value ranges from 00h through F8h.		
03h	7:0	**First Data Payload Byte** Descriptor.		
04h	7:0	**Second Data Payload Byte** Descriptor, if present.		
…		**…**		

The Block Revision field shall be incremented whenever a new data block field is defined that does not break compatibility with legacy revisions. In case backward-compatibility cannot be achieved, a new block tag shall be used to define the new data block. The following are cases in which the Block Revision field shall be incremented:

- Adding a new field at the end of the data block (the Number of Payload Bytes field shall also be updated for this case)
- Redefining previously RESERVED bit(s)
- Extending the range of a previously RESERVED field

Any variable-length field within a data block shall be set up such that the field's length can be determined, without ambiguity. This allows for future expansion of data blocks without breaking compatibility.

3.2 DisplayID Section – A Visual Representation

Table 3-3 visually describes DisplayID section format and packing with common data blocks.

Table 3-3: DisplayID Section Visual Representation

Header Name			Field Name
DisplayID Section	Section Header		Structure Version/Revision
			Section Size
			Display Product Primary Use Case
			Extension Count
	Product Identification Data Block	Block Header	Block Tag
			Block Revision and Other Data
			Number of Payload Bytes
		Block Payloads	Block Payload Byte 0
			Block Payload Byte 1
			Block Payload Byte 2
			Block Payload Byte 3
			…
			Block Payload Last Byte *N*
	Display Parameters Data Block	Block Header	Block Tag
			Block Revision and Other Data
			Number of Payload Bytes
		Block Payloads	Block Payload Byte 0
			Block Payload Byte 1
			Block Payload Byte 2
			Block Payload Byte 3
			…
			Block Payload Last Byte *N*

Table 3-3: DisplayID Section Visual Representation (Continued)

Header Name			Field Name
DisplayID Section	Display Interface Features Data Block	Block Header	Block Tag
			Block Revision and Other Data
			Number of Payload Bytes
		Block Payloads	Block Payload Byte 0
			Block Payload Byte 1
			Block Payload Byte 2
			Block Payload Byte 3
			…
			Block Payload Last Byte *N*
	Video Timing Mode Data Block(s)	Block Header	Block Tag
			Block Revision and Other Data
			Number of Payload Bytes
		Block Payloads	Block Payload Byte 0
			Block Payload Byte 1
			Block Payload Byte 2
			Block Payload Byte 3
			…
			Block Payload Last Byte *N*
	Dynamic Video Timing Range Limits Data Block	Block Header	Block Tag
			Block Revision and Other Data
			Number of Payload Bytes
		Payloads	Block Payload Byte 0
			Block Payload Byte 1
			…
			Block Payload Last Byte *N*
	Checksum		Checksum

4 Data Block Definitions

This section provides specifications for all VESA-defined data blocks that are defined for use within the DisplayID structure. As previously noted, valid DisplayID data blocks may be defined by other organizations, most notably the CTA, and individual VESA data block definitions shall be considered as separate standards by VESA and carry their own revision number and revision date.

The overall DisplayID version and revision number shall not be incremented for revisions to the individual data block definitions; these shall increment only for revisions applied to previous sections of this document, including changes to the overall DisplayID structure definition or the basic data block structure (e.g., location and/or size of the tag code or length fields, etc.). The overall DisplayID version/revision numbers shall also not be incremented in the case of adding new tag code assignments. The document revision date, however, shall be updated for any changes to one or more data block definitions, –or– the addition or deletion of data block definitions and/or tag code numbers.

The requirement definition for each data block (i.e., whether a field is required) is relevant for DisplayID Base sections.

Tables summarizing the requirements for each display product primary use case defined by Table 3-1, along with the tag code, latest revision, and any restrictions, are provided at the beginning of each data block's definition section.

4.1 Product Identification Data Block

The DisplayID Base section shall include a mandatory Product Identification data block as its first data block. The Product Identification data block is composed of several fields that shall be used to uniquely identify the monitor.

Table 4-1 lists the size and order of each field, with all addresses relative to the beginning of the data block.

Table 4-1: Product Identification Data Block Fields

Offset	Bit #	Description/Format	Defined in	Tag
00h	7:0	**Product Identification Data Block** Value is 20h.		✔
01h	**Block Revision and Other Data**			
	2:0	**Block Revision** Revision ranges from 0 through 7. 000b = Revision 0 (default).		
	7:3	**RESERVED** Cleared to all 0s.		
02h	7:0	**Number of Payload Bytes in Block** Number of payload bytes within the data block ranges from 12 through 248. Value ranges from 0Ch through F8h.		
05h – 03h	23:0	**Manufacturer/Vendor ID** Descriptor.	Section 4.1.1	
07h – 06h	15:0	**Product ID Code** Descriptor.	Section 4.1.2	
0Bh – 08h	31:0	**Serial Number** Optional Descriptor.	Section 4.1.3	
0Ch	7:0	**Week of Manufacture/Model Tag** Descriptor.	Section 4.1.4	
0Dh	7:0	**Year of Manufacture/Model Year** Descriptor.	Section 4.1.4	
0Eh	7:0	**Size of Product Name String** Descriptor.	Section 4.1.5	
FBh – 0Fh		**Product Name String** Optional Descriptor.	Section 4.1.5	

4.1.1 Manufacturer/Vendor ID Field

The Manufacturer/Vendor ID field listed in Table 4-1 is a **required** element of the Product Identification data block. The Manufacturer/Vendor ID field contains the display manufacturer's 3-byte IEEE Organizationally Unique Identifier (OUI) code, which identifies the display's manufacturer or vendor.

The Institute of Electrical and Electronics Engineers (IEEE) issues IEEE OUIs. Contact the IEEE at http://standards.ieee.org/develop/regauth/oui/ for registration and/or further details.

Table 4-2 lists the field format.

Table 4-2: Manufacturer/Vendor ID Field Format

Offset	Byte #	Bit #	Description/Format
03h	0	7:0	**IEEE OUI First Byte** Byte code.
04h	1	7:0	**IEEE OUI Second Byte** Byte code.
05h	2	7:0	**IEEE OUI Third Byte** Byte code.

4.1.2 Product ID Code Field

The Product ID Code field listed in Table 4-1 is a **required** element of the Product Identification data block. The Product ID Code field contains a 2-byte vendor-assigned Product ID code. The field's length helps to differentiate between multiple models from the same manufacturer. If this field is used to represent a Product ID code (e.g., a model number), the number shall be stored in hexadecimal format with the least significant byte (LSB) listed first. Table 4-3 lists the field format.

Table 4-3: Product ID Code Field Format

Offset	Byte #	Bit #	Description/Format
06h	0	7:0	**Product ID Code LSB** Value ranges from 00h through FFh.
07h	1	7:0	**Product ID Code MSB** Value ranges from 00h through FFh.

4.1.3 Serial Number Field

The Serial Number field listed in Table 4-1 is an **optional** element of the Product Identification data block. The Serial Number field is a 32-bit serial number that shall be used to differentiate between individual instances of the same model of display product. When used, this field's bit order follows the order listed in Table 4-4. The four bytes of the serial number are listed with the LSB first. The serial number ranges from 0 through 4,294,967,295. This serial number shall be a number only – it shall **not** represent an ASCII code. If this field is not used, however, a value of "00h, 00h, 00h, 00h" shall be entered.

Table 4-4: Serial Number Field Format

Offset	Byte #	Bit #								Description
		7	6	5	4	3	2	1	0	
08h	0	7	6	5	4	3	2	1	0	**ID Serial Number**
09h	1	15	14	13	12	11	10	9	8	
0Ah	2	23	22	21	20	19	18	17	16	
0Bh	3	31	30	29	28	27	26	25	24	

4.1.4 Week of Manufacture/Model Tag and Year of Manufacture/Model Year Fields

The Week of Manufacture/Model Tag and Year of Manufacture/Model Year fields listed in Table 4-1 are **required** Product Identification data block elements. The Week of Manufacture/Model Tag field, when used to indicate the week, shall be programmed to a value within the range of 1 through 52. If a week is not declared, the value shall be cleared to all 0s. If a model year is to be declared, the value shall be programmed to FFh and offset 0Dh shall indicate the model year.

Otherwise, the Year of Manufacture/Model Year field shall be used to represent the Gregorian calendar year in which the display was manufactured. In either case, the value used to indicate the year shall be stored as an offset from the year 2000, as derived from the following equation:

Stored Value = (Year of Manufacture - 2000)

For example, a display manufactured in 2016 would contain value 10h (00010000b) in offset 0Dh. If the Year of Manufacture/Model Year field is not used, the byte would be cleared to all 0s.

Table 4-5 lists the field formats.

Table 4-5: Week of Manufacture/Model Tag and Year of Manufacture/Model Year Field Formats

Offset	Byte #	Bit #	Description/Format	Flag
0Ch	0	7:0	**Week of Manufacture/Model Tag** 00h = No week specified. 01h through 34h = Week number within the year of manufacture. Week number ranges from 1 through 52. 35h through FEh = RESERVED. FFh = Model year is specified in Byte 0Dh.	 ✔ ✔
0Dh	1	7:0	**Year of Manufacture/Model Year** 00h through 0Eh = RESERVED. 0Fh through FFh = Year of manufacture: • Model year when Byte 0Ch = FFh • Gregorian year when Byte 0Ch ≠ FFh	

4.1.5 Size of Product Name String and Product Name String Fields

The Size of Product Name String field listed in Table 4-1 is a **required** element of the Product Identification data block. The Size of Product Name String field provides the Product Name String field length, in bytes. If the length is cleared to all 0s, the Product Name String field does **not** exist.

The Product Name String field listed in Table 4-1 is an **optional** element of the Product Identification data block. The Product Name String field may be used to provide additional model and/or product identification information in the form of an ASCII character string of variable length, up to a maximum size that fills up a DisplayID section. The string's first character is stored at offset 0Fh, the second character is stored at offset 10h, etc.

Table 4-6 lists the field formats.

Table 4-6: Size of Product Name String and Product Name String Field Formats

Offset	Byte #	Bit #	Description/Format	Flag
0Eh	0	7:0	**Size of Product Name String** 00h = No string specified (not typical use). 01h through ECh = Number of bytes within the Product Name String field ranges from 1 through 236.	✔
FBh – 0Fh	236:1		**Product Name String** Value ranges from 00h through FFh.	

4.2 Display Parameters Data Block

The Display Parameters data block is composed of several fields that shall define the monitor's global parameters. No more than one Display Parameters data block shall be exposed within the DisplayID section. Display Parameters is a **mandatory** data block for product primary use case as a display and is **optional** otherwise.

Table 4-7 lists the size and order of each field, with all addresses relative to the beginning of the data block.

Table 4-7: Display Parameters Data Block Fields

Offset	Bit #	Description/Format	Defined in	Tag
00h	7:0	**Display Parameters Data Block** Value is 21h.		✔
01h	**Block Revision and Other Data**			
	2:0	**Block Revision** Revision ranges from 0 through 7. 000b = Revision 0 (default).		
	6:3	**RESERVED** Cleared to all 0s.		
	7	**Image Size Multiplier** 0 = Horizontal Image Size and Vertical Image Size fields (offsets 03h through 06h) have 0.1-mm precision (default). 1 = Horizontal Image Size and Vertical Image Size fields have 1.0-mm precision.	Section 4.2.1	
02h	7:0	**Number of Payload Bytes in Block** 1Dh = Data block is composed of 29 payload bytes.		
04h– 03h	15:0	**Horizontal Image Size** Descriptor.	Section 4.2.1	
06h– 05h	15:0	**Vertical Image Size** Descriptor.	Section 4.2.1	
08h– 07h	15:0	**Horizontal Pixel Count** Descriptor.	Section 4.2.2	
0Ah– 09h	15:0	**Vertical Pixel Count** Descriptor.	Section 4.2.2	
0Bh	7:0	**Feature Support Flags** Descriptor.	Section 4.2.3	
0Eh – 0Ch	23:0	**Native Color Chromaticity (Primary Color 1 Chromaticity)** Descriptor.	Section 4.2.4	

Table 4-7: Display Parameters Data Block Fields (Continued)

Offset	Bit #	Description/Format	Defined in	Tag
11h – 0Fh	23:0	**Native Color Chromaticity (Primary Color 2 Chromaticity)** Descriptor.	Section 4.2.4	
14h – 12h	23:0	**Native Color Chromaticity (Primary Color 3 Chromaticity)** Descriptor.	Section 4.2.4	
17h – 15h	23:0	**Native Color Chromaticity (White Point Chromaticity)** Descriptor.	Section 4.2.4	
19h – 18h	15:0	**Native Maximum Luminance (Full Coverage)** Descriptor.	Section 4.2.5	
1Bh – 1Ah	15:0	**Native Maximum Luminance (10% Rectangular Coverage)** Descriptor.	Section 4.2.5	
1Dh – 1Ch	15:0	**Native Minimum Luminance** Descriptor.	Section 4.2.5	
1Eh	7:0	**Native Color Depth and Display Device Technology** Descriptor.	Section 4.2.6	
1Fh	7:0	**Native Gamma EOTF** Descriptor.	Section 4.2.7	

4.2.1 Horizontal Image Size and Vertical Image Size Fields

The Horizontal Image Size and Vertical Image Size fields of the Display Parameters data block, along with the Image Size Multiplier bit (offset 01h, bit 7), shall define the active image area's display size. Two bytes are provided for the horizontal dimensions, two bytes are provided for the vertical dimensions, and one bit is provided for the multiplier, in that order. The values stored here shall be interpreted as 16-bit numbers that provide the size along the horizontal or vertical axis, respectively, in units of 0.1 or 1.0mm, based on the Image Size Multiplier bit value:

- **Image Size Multiplier bit is cleared to 0** = Range of sizes that may be stored for either field is 0.1 through 6,553.5mm, in each direction
- **Image Size Multiplier bit is set to 1** = Range of sizes that may be stored for either field is 1.0 through 65,535.0mm, in each direction

The multiplier's minimum value should be used, when possible, to achieve maximum precision.

For display technologies that do **not** have a precisely fixed image size or native pixel format (e.g., CRT displays), the values stored here shall comprise the maximum image size for any supported format or timing. This is not necessarily the physical screen's limitation – the maximum image size should be the maximum portion of the screen that the manufacturer has designated as usable for image display. If the Horizontal Image Size and/or Vertical Image Size fields are cleared to all 0s, the system shall **not** make any assumptions regarding display size.

Note: *If a projection display provides an image of indeterminate size, it shall be appropriate to clear both fields to all 0s.*

Table 4-8 lists the field formats.

Table 4-8: Horizontal Image Size and Vertical Image Size Field Formats

Offset	Byte #	Bit #	Description/Format[a]
04h – 03h	1:0	15:0	**Horizontal Image Size** • Range shall be 0.1 through 6,553.5mm when Image Size Multiplier = 0 • Range shall be 1.0 through 65,535.0mm when Image Size Multiplier = 1
06h – 05h	1:0	15:0	**Vertical Image Size** • Range shall be 0.1 through 6,553.5mm when Image Size Multiplier = 0 • Range shall be 1.0 through 65,535.0mm when Image Size Multiplier = 1

a. *If a projection display provides an image of indeterminate size, it shall be appropriate to clear both fields to all 0s.*

4.2.2 Horizontal Pixel Count and Vertical Pixel Count Fields (Native Format)

The 2-byte Horizontal Pixel Count and Vertical Pixel Count fields of the Display Parameters data block shall define the display device's native format, in pixels. For display technologies that do **not** provide a fixed native format (e.g., CRT displays), all four bytes shall contain all 0s. The image or device aspect ratio may **not** be inferred from this information because there is no requirement that the device's pixels shall be square or that the horizontal and vertical directions shall have an equal number of pixels per unit distance. Table 4-9 lists the field formats.

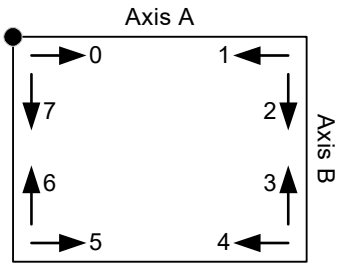
Table 4-9: Horizontal Pixel Count and Vertical Pixel Count Field Formats (Native Format)

Offset	Byte #	Bit #	Description/Format
08h – 07h	1:0	15:0	**Horizontal Pixel Count** Native format, horizontal pixels. Number of pixels ranges from 0 through 65,535. Value ranges from 0000h through FFFFh.
0Ah – 09h	1:0	15:0	**Vertical Pixel Count** Native format, vertical pixels. Number of pixels ranges from 0 through 65,535. Value ranges from 0000h through FFFFh.

4.2.3 Feature Support Flags Field

The Feature Support Flags field of the Display Parameters data block is a single byte that provides flags that shall be used to indicate the display's support for various features or functions. Table 4-10 lists the field format.

Table 4-10: Feature Support Flags Field Format

Offset	Bit #	Description/Format
0Bh	2:0	**Scan Orientation** Scan Orientation with respect to normal viewing position. Axis A and B map to the horizontal and vertical pixels as seen from the normal viewing position, and may not map to the horizontal and vertical dimensions in the specified timing declarations. (See Appendix C for further clarification.) Unless otherwise governed by display interface-specific rules (e.g., DisplayPort or HDMI), the Source device shall support Scan Orientation 0, and may also support other scan orientations. 000b = Left to right, top to bottom. 001b = Right to left, top to bottom. 010b = Top to bottom, right to left. 011b = Bottom to top, right to left. 100b = Right to left, bottom to top. 101b = Left to right, bottom to top. 110b = Bottom to top, left to right. 111b = Top to bottom, left to right. 
	4:3	**Luminance Information** When Native Luminance-related fields in the Display Parameters data block contain positive values, this field indicates how to interpret those fields. If the Native Luminance-related fields contain a negative 0 value, -0, this field shall be cleared to all 0s. 00b = Non-zero maximum luminance information contained within this block is exposed as a minimum guaranteed value. 01b = Non-zero maximum luminance information contained within this block is provided as a guidance for the Source device. ***Note:*** *The Sink device has an option to use -0 values for luminance fields that indicate to the Source device to not use Native Luminance-related fields.* All other values are RESERVED.
	5	**RESERVED** Cleared to 0.
	6	**Color Information** 0 = All color information contained within this block shall be provided in terms of *CIE 1931* (*x*, *y*) coordinates (default). 1 = All color information contained within this block shall be provided in terms of *CIE 1976* (*u'*, *v'*) coordinates.
	7	**Audio Speaker Information** 0 = Audio speakers shall be integrated into the display (default). 1 = Audio speakers shall not be integrated into the display. Separate speaker connection shall be supplied by an external jack.

4.2.4 Native Color Chromaticity Fields

The Native Color Chromaticity fields shall provide information regarding the display's native color characteristics, including the primary colors' chromaticity aspects and display's white points, as defined by the 3-byte fields that are listed for each in Table 4-11. Values may be provided as 12-bit (*x*, *y*) color coordinates per *CIE 1931*, –or– as 12-bit (*u'*, *v'*) color coordinates per *CIE 1976*. The Color Information bit of the Display Parameters data block's Feature Support Flags field (offset 0Bh, bit 6) shall determine whether the colors are indicated in (x, y) or (*u'*, *v'*) coordinate format.

Table 4-11: Native Color Chromaticity Field Formats

Offset	Byte #	Bit #	Description/Format
0Eh – 0Ch	**Native Color Chromaticity (Primary Color 1 Chromaticity)**		
0Ch	0	7:0	**Primary Color 1 *x* or *u'* Value Low Bits or Standard Color Space Identifier** Color *x* or *u'* value bits 7 through 0, respectively. Value ranges from 00h through FFh.
0Dh	1	11:8	**Primary Color 1 *x* or *u'* Value High Bits** Color *x* or *u'* value bits 11 through 8, respectively. Value ranges from 0h through Fh.
		15:12	**Primary Color 1 *y* or *v'* Value Low Bits** Color *y* or *v'* value bits 3 through 0, respectively. Value ranges from 0h through Fh.
0Eh	2	23:16	**Primary Color 1 *y* or *v'* Value High Bits** Color *y* or *v'* value bits 11 through 4, respectively. Value ranges from 00h through FFh.
11h – 0Fh	**Native Color Chromaticity (Primary Color 2 Chromaticity)**		
0Fh	0	7:0	**Primary Color 2 *x* or *u'* Value Low Bits or Standard Color Space Identifier** Color *x* or *u'* value bits 7 through 0, respectively. Value ranges from 00h through FFh.
10h	1	11:8	**Primary Color 2 *x* or *u'* Value High Bits** Color *x* or *u'* value bits 11 through 8, respectively. Value ranges from 0h through Fh.
		15:12	**Primary Color 2 *y* or *v'* Value Low Bits** Color *y* or *v'* value bits 3 through 0, respectively. Value ranges from 0h through Fh.
11h	2	23:16	**Primary Color 2 *y* or *v'* Value High Bits** Color *y* or *v'* value bits 11 through 4, respectively. Value ranges from 00h through FFh.

Table 4-11: Native Color Chromaticity Field Formats (Continued)

Offset	Byte #	Bit #	Description/Format
14h – 12h	**Native Color Chromaticity (Primary Color 3 Chromaticity)**		
12h	0	7:0	**Primary Color 3 *x* or *u′* Value Low Bits or Standard Color Space Identifier** Color *x* or *u′* value bits 7 through 0, respectively. Value ranges from 00h through FFh.
13h	1	11:8	**Primary Color 3 *x* or *u′* Value High Bits** Color *x* or *u′* value bits 11 through 8, respectively. Value ranges from 0h through Fh.
		15:12	**Primary Color 3 *y* or *v′* Value Low Bits** Color *y* or *v′* value bits 3 through 0, respectively. Value ranges from 0h through Fh.
14h	2	23:16	**Primary Color 3 *y* or *v′* Value High Bits** Color *y* or *v′* value bits 11 through 4, respectively. Value ranges from 00h through FFh.
17h – 15h	**Native Color Chromaticity (White Point Chromaticity)**		
15h	0	7:0	**White Point *x* or *u′* Value Low Bits or Standard Color Space Identifier** Color *x* or *u′* value bits 7 through 0, respectively. Value ranges from 00h through FFh.
16h	1	11:8	**White Point *x* or *u′* Value High Bits** Color *x* or *u′* value bits 11 through 8, respectively. Value ranges from 0h through Fh.
		15:12	**White Point *y* or *v′* Value Low Bits** Color *y* or *v′* value bits 3 through 0, respectively. Value ranges from 0h through Fh.
17h	2	23:16	**White Point *y* or *v′* Value High Bits** Color *y* or *v′* value bits 11 through 4, respectively. Value ranges from 00h through FFh.

4.2.4.1 Primary Color Definition and Ordering

"Primary colors" are defined as those that correspond to physical sub-pixels or other sources of pure color, controlled by a given physical or logical channel of the device interface or video signal. The ordering of the primary colors, as provided in this data block, shall correspond to standard ordering of the primary colors as they correspond to the video signal channels as given in the applicable interface/video signal standard. For example, in a typical RGB color representation, "**R**" (**red**), "**G**" (**green**), and "**B**" (**blue**) are the primary colors and are described in that order.

4.2.4.2 White Point Definition and Ordering

"**White** point" describes the color that the display produces when all primary channels are driven to their maximum saturation value (e.g., in an 8-bit color RGB display, **white** is the color that results from all three R, G, and B channels each being driven to a value of 255).

4.2.4.3 Data Format

The chromaticity and white point values to be stored in the Native Color Chromaticity fields shall be initially expressed as decimal-format fractional numbers, accurate to the thousandths place.

Each value shall be stored in this data block as a 12-bit binary-format fraction. In such a fractional representation, a value of 1 for the bit immediately to the right of the decimal point (the most significant bit of the value provided, in this case bit 11) represents a value of 2^{-1}. A value of 1 in the rightmost bit (i.e., the least significant bit, or bit 0) represents a value of 2^{-12}.

Using this representation, all color coordinates provided in these bytes should be accurate within ±0.0001 of the actual value. Table 4-12 lists three examples.

Table 4-12: 12-bit Binary-format Fraction Representation (Examples)

Actual Value	Binary-format Value	Converted Back to Decimal Format
0.610	100111000011	0.610107
0.307	010011101001	0.306884
0.150	001001100110	0.149902

4.2.5 Native Luminance-related Fields

The Native Luminance-related fields provide information related to the display's native luminance in various data transmission conditions with differing maximum and 0 code coverage. The information shall be consistent with the physical display such that the display's actual measurement in the described condition shall result in a similar value to that exposed in DisplayID, within a reasonable tolerance. If luminance information is provided, all Native Luminance-related fields shall contain positive values. If luminance information is **not** provided, all Native Luminance-related fields shall contain a negative 0 value, -0 (0x8000). Table 4-13 lists the field formats.

Table 4-13: Native Luminance-related Field Formats

Offset	Byte #	Bit #	Description/Format
19h – 18h	1:0	15:0	**Native Maximum Luminance (Full Coverage)** Native maximum luminance, in candela per square meter (cd/m^2), that shall be physically possible to attain on the display with all pixels programmed to maximum code. Value shall be encoded in *IEEE 754* half-precision binary floating-point format. A positive value indicates that the field contains valid luminance data. A negative 0 value, -0, indicates that the field does not contain valid luminance data and shall not be used. All other negative values are RESERVED.
1Bh – 1Ah	1:0	15:0	**Native Maximum Luminance (10% Rectangular Coverage)** Native maximum luminance, in candela per square meter (cd/m^2), that shall be physically possible to attain on the display with a 10% rectangular patch programmed to maximum code in the middle of the screen, while the remainder of the screen is programmed to 0 code. Value shall be encoded in *IEEE 754* half-precision binary floating-point format. A positive value indicates that the field contains valid luminance data. A negative 0 value, -0, indicates that the field does not contain valid luminance data and shall not be used. All other negative values are RESERVED.
1Dh – 1Ch	1:0	15:0	**Native Minimum Luminance** Native minimum luminance, in candela per square meter (cd/m^2), that shall be physically possible to attain on the display, using a corner box pattern in which a Source device sends a code of 0 everywhere except at the 2.5% rectangular patches at the corners of the screen, which are programmed to maximum code. Value shall be encoded in *IEEE 754* half-precision binary floating-point format. A positive value indicates that the field contains valid luminance data. A negative 0 value, -0, indicates that the field does not contain valid luminance data and shall not be used. All other negative values are RESERVED.

4.2.6 Native Color Depth and Display Device Technology Fields

The Native Color Depth and Display Device Technology fields provide information related to the display's native color depth and the technology used by the display device's "glass." Table 4-14 lists the field formats.

Table 4-14: Native Color Depth and Display Device Technology Field Format

Offset	Bit #	Description/Format
1Eh	2:0	**Native Color Depth** Value shall be selected based on the Sink device's ability to use all available bits per component (bpc) for display output rendering. 000b = Not defined. Source device shall apply display interface-specific rules. 001b = 6bpc. 010b = 8bpc. 011b = 10bpc. 100b = 12bpc. 101b = 16bpc. 110b and 111b = RESERVED.
	3	**RESERVED** Cleared to 0.
	6:4	**Display Device Technology** Describes the technology used by the display device's "glass." The Source device may optionally use the information provided by this field as a hint to optimize the Source device's content for the relevant technology. The Sink device shall **not** expect a guaranteed behavior of the Source device based on the information provided in this field. 000b = Technology type is not specified. 001b = Active Matrix LCD technology. 010b = Organic LED technology. All other values are RESERVED.
	7	**RESERVED** Cleared to 0.

4.2.7 Native Gamma EOTF Field

The Native Gamma EOTF field provides information related to the native gamma Electro-Optical Transfer Function (EOTF) that the display supports. Exposing a specific gamma value indicates that the Sink device shall be able to decode content generated by the Source device at that gamma value. Table 4-15 lists the field formats.

Table 4-15: Native Gamma EOTF Field Format

Offset	Bit #	Description/Format
1Fh	7:0	**Native Gamma EOTF** Defines the gamma range, from 1.00 through 3.54, as follows: *Field Value = (Gamma × 100) - 100* Value ranges from 00h through FFh. FFh = No gamma information shall be provided.

4.3 Video Timing Mode-related Data Blocks

The Video Timing Mode-related data blocks can declare support for multiple timings, using three different descriptor formats. Table 4-16 lists the size and order of each field, with all addresses relative to the beginning of the data block.

Table 4-16: Video Timing Mode-related Data Block Fields

Offset	Bit #	Description/Format/Priority	Defined in	Tag
00h	7:0	**Video Timing Mode-related Data Blocks** 22h = Type VII Timing – Detailed. 23h = Type VIII Timing – Enumerated Timing Code. 24h = Type IX Timing – Formula-based.	 Section 4.3.1 Section 4.3.2 Section 4.3.3	✔
01h	**Block Revision and Other Data**			
	2:0	**Block Revision** Revision ranges from 0 through 7. 000b = Revision 0 (default).		
	7:3	**RESERVED** Timing-specific. Cleared to all 0s.		
02h	7:0	**Number of Payload Bytes** Number of payload bytes ranges from 0 through 248. Value ranges from 00h through F8h.		
03h	7:0	**First Video Timing Mode Byte** Descriptor. Priority 1.		
03h + N^a	7:0	**Second Video Timing Mode Byte** Descriptor, if present. Priority 2.		
…		…		

a. N represents the total number of bytes within the DisplayID section.

Notes: Preferred Timings and Timing Prioritization

As noted in Section 2, most DisplayID structures shall identify at least one timing as the preferred timing for that product; in addition, other timings may also be identified as preferred or supported elsewhere within the DisplayID structure.

A preferred timing is defined as one that shall, in the manufacturer's opinion, result in the product's optimum performance within its intended use or application. If two or more timings are identified as preferred or supported within the DisplayID structure, the Source device shall use the following priorities when selecting which timing to use:

1 *First timing identified in the Base section, either as the first timing block (of any type), within a Video data block, –or– as a timing code within the appropriate data block.*

2 *Other timing (if any) provided in the DisplayID structure that is identified as preferred.*

3 *Other timing (if any) indicated as supported by way of one of the methods listed in item 1.*

4 *Standard 640x480, 60-Hz progressive-scan timing that is to be used as a base video mode for any display that does not otherwise identify a supported timing, –or– whose DisplayID information cannot be read.*

4.3.1 Type VII Timing – Detailed Timing Data Block

The Type VII Detailed Timing data block shall be composed of single or multiple 20-byte Detailed Timing descriptors. Type VII Detailed Timing is a **mandatory** data block for product primary use case as a display and is **optional** otherwise.

The *DisplayID v2.0* definition of the Type VII Detailed Timing data block is similar to that of the *DisplayID v1.3* Type I Detailed Timing data block, except for Bytes 0, 1, 2 of the descriptor fields, which carry the pixel clock information. With Type VII, higher-precision pixel clock support can be exposed. The field's range allows the pixel clock rate to be defined as 0.001 through 16,777.216MP/s.

Table 4-17 and Table 4-18 list the data block's structure and descriptor formats, respectively.

Table 4-17: Type VII Timing – Detailed Timing Data Block Format

Offset	Bit #	Description/Format/Priority	Tag
00h	7:0	**Type VII Timing – Detailed Timing Data Block** Value is 22h.	✔
01h	**Block Revision and Other Data**		
	2:0	**Block Revision** Revision ranges from 0 through 7. 000b = Revision 0 (default).	
	7:3	**RESERVED** Cleared to all 0s.	
02h	7:0	**Number of Payload Bytes in Block** Number of payload bytes within the data block ranges from 20 through 240 ($1 \leq N \leq 12$).[a] Value is based on $N \times 20 - 14$h, 28h, 3Ch, etc., through F0h. All other values are RESERVED.	
16h – 03h	159:0	**First Type VII Detailed Timing Descriptor** 20-byte descriptor. Priority 1.	
2Ah – 17h	159:0	**Second Type VII Detailed Timing Descriptor** 20-byte descriptor, if present. Priority 2.	
…		**…**	

a. N represents the number of Detailed Timing Descriptors in the data block.

Table 4-18: Type VII Detailed Timing Descriptor Format

Byte #	Bit #	Description/Format	Flag
2:0	**Pixel Clock (in kHz)** Rate range is defined as 0.001 through 16,777.216MP/s.		
0	7:0	**Pixel Clock (in kHz) Low Bits 7:0** Value ranges from 00h through FFh.	
1	15:8	**Pixel Clock (in kHz) Middle Bits 15:8** Value ranges from 00h through FFh.	
2	23:16	**Pixel Clock (in kHz) High Bits 23:16** Value ranges from 00h through FFh.	

Table 4-18: Type VII Detailed Timing Descriptor Format (Continued)

Byte #	Bit #	Description/Format	Flag
3	**Timing Options**		✔
	3:0	**Aspect Ratio** 0h = 1:1 4h = 16:9 1h = 5:4 5h = 16:10 2h = 4:3 6h = 64:27 3h = 15:9 7h = 256:135 8h = Aspect ratio shall be calculated by using the Horizontal Active Image Pixels and Vertical Active Image Lines fields (Bytes 5:4 and 13:12, respectively). 9h through Fh = RESERVED.	
	4	**Interface Frame Scanning Type** 0 = Progressive scan frame. 1 = Interlaced scan frame.	✔
	6:5	**3D Stereo Support** 00b = This timing shall always be displayed in mono (no stereo). 01b = This timing shall always be displayed in stereo. 10b = This timing shall be displayed in mono or stereo, depending on a user action (such as wearing stereo glasses). 11b = RESERVED	✔
	7	**Preferred Detailed Timing** 1 = Preferred Detailed timing.	
5:4	**Horizontal Active Image Pixels** Number of pixels ranges from 1 through 65,536.		
4	7:0	**Horizontal Active Image Pixels Low Bits 7:0** Value ranges from 00h through FFh.	
5	15:8	**Horizontal Active Image Pixels High Bits 15:8** Value ranges from 00h through FFh.	
7:6	**Horizontal Blank Pixels** Number of pixels ranges from 1 through 65,536.		
6	7:0	**Horizontal Blank Pixels Low Bits 7:0** Value ranges from 00h through FFh.	
7	15:8	**Horizontal Blank Pixels High Bits 15:8** Value ranges from 00h through FFh.	

Table 4-18: Type VII Detailed Timing Descriptor Format (Continued)

Byte #	Bit #	Description/Format	Flag
9:8	**Horizontal Offset (Front Porch)** Number of pixels ranges from 1 through 32,768.		
8	7:0	**Horizontal Offset (Front Porch) Low Bits 7:0** Value ranges from 00h through FFh.	
9	14:8	**Horizontal Offset (Front Porch) High Bits 14:8** Value ranges from 00h through 7Fh.	
	15	**Horizontal Sync Polarity** 0 = Negative. 1 = Positive.	
11:10	**Horizontal Sync Width** Number of pixels ranges from 1 through 65,536.		
6	7:0	**Horizontal Sync Width Low Bits 7:0** Value ranges from 00h through FFh.	
7	15:8	**Horizontal Sync Width High Bits 15:8** Value ranges from 00h through FFh.	
13:12	**Vertical Active Image Lines** Number of lines ranges from 1 through 65,536.		
4	7:0	**Vertical Active Image Lines Low Bits 7:0** Value ranges from 00h through FFh.	
5	15:8	**Vertical Active Image Lines High Bits 15:8** Value ranges from 00h through FFh.	
15:14	**Vertical Blank Lines** Number of lines ranges from 1 through 65,536.		
14	7:0	**Vertical Blank Lines Low Bits 7:0** Value ranges from 00h through FFh.	
15	15:8	**Vertical Blank Lines High Bits 15:8** Value ranges from 00h through FFh.	
17:16	**Vertical Sync Offset (Front Porch)** Number of lines ranges from 1 through 32,768.		
16	7:0	**Vertical Sync Offset (Front Porch) Low Bits 7:0** Value ranges from 00h through FFh.	
17	14:8	**Vertical Sync Offset (Front Porch) High Bits 14:8** Value ranges from 00h through 7Fh.	
	15	**Vertical Sync Polarity** 0 = Negative. 1 = Positive.	

Table 4-18: Type VII Detailed Timing Descriptor Format (Continued)

Byte #	Bit #	Description/Format	Flag
19:18	**Vertical Sync Width** Number of lines ranges from 1 through 65,536.		
18	7:0	**Vertical Sync Width Low Bits 7:0** Value ranges from 00h through FFh.	
19	15:8	**Vertical Sync Width High Bits 15:8** Value ranges from 00h through FFh.	

4.3.1.1 Support for Interlaced Video Timing Modes

Past EDID Standards did not provide for specifying vertical blanking and sync offsets between two interlaced timing-related fields. The CTA addressed this by using frame-based timing in its use of the EDID Detailed Timing data blocks, as defined in *CTA-861-B* and higher. In *DisplayID v2.0*, VESA harmonizes its own definition for use of these detailed timings, in the case of interlaced formats, with the CTA norms. For details regarding interlaced frame transport over a specific display interface, see the relevant interface standard or specification (e.g., DisplayPort or HDMI) to address timing parameters' division over the interlace fields and interlace-format signaling.

4.3.1.2 Additional Requirements and Information Regarding Borders

This section provides additional information and requirements regarding the use of borders in the Type VII Timing – Detailed Timing data block.

Note: *The concept of border areas dates back to the early days of CRT-based computer monitors, and refers to a portion of the displayed image that is outside the addressable area (i.e., the area that the host system's graphics controller uses for displaying information). The video source may set these border areas to* ***white*** *or some other solid color to provide a visible frame around the active video area. Border use is rare in current PC industry practice, and is of questionable value for non-CRT display types.*

For these reasons, borders have been removed from the detailed timing structure and shall be implicitly considered to be cleared to all 0s.

Figure 4-1 provides additional information on the relationship between border times and other defined horizontal and vertical timing parameters.

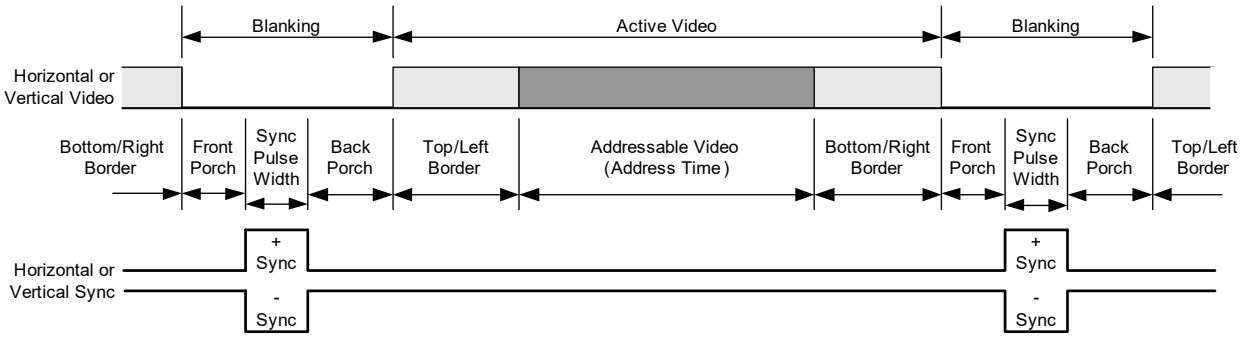


Figure 4-1: Video Timing Parameter Definitions

4.3.2 Type VIII Timing – Enumerated Timing Code Data Block

The Type VIII Enumerated Timing Code data block provides a means for exposing support for timings based on enumerating timing codes within the data block. Timing codes shall be defined by a relevant standards organization such as VESA, the CTA, or HDMI, in DMT Timings (*DMT Standard*), CTA Video Information Code (VIC) Timings, and/or HDMI VIC Timings, respectively. Timing priority shall be in the order listed. A DisplayID section can have multiple Type VIII Timing data blocks. Each Type VIII Timing data block shall include only timings based on a single Timing Code type. Type VIII Enumerated Timing Code is an **optional** data block.

Note: *Type VIII Timing data blocks contain CTA VIC timing codes, rather than Short Video Descriptors (SVDs). Native formats shall be inferred from information in the Display Parameters data block, primarily the native format Horizontal Pixel Count and Vertical Pixel Count fields (offsets 08h – 07h and 0Ah – 09h, respectively) of the Display Parameters data block.*

Table 4-19 lists the data block's structure.

Table 4-19: Type VIII Timing – Enumerated Timing Code Data Block Format

Offset	Bit #	Description/Format/Priority
00h	7:0	**Type VIII Timing – Enumerated Timing Code Data Block** Value is 23h.
01h	**Block Revision and Other Data**	
	2:0	**Block Revision** Revision ranges from 0 through 7. 000b = Revision 0 (default).
	3	**Timing Code Size** 0 = 1-byte descriptor timing code. 1 = 2-byte descriptor timing code.
	5:4	**RESERVED** Cleared to all 0s.
	7:6	**Timing Code Type** 00b = DMT timing code. 01b = CTA VIC timing code. 10b = HDMI VIC timing code. 11b = RESERVED timing code type.
02h	7:0	**Number of Payload Bytes in Block** Number of payload bytes within the data block ranges from 1 through 248. Value ranges from 01h through F8h. All other values are RESERVED.

Table 4-19: Type VIII Timing – Enumerated Timing Code Data Block Format (Continued)

Offset	Bit #	Description/Format/Priority
03h	7:0	**First Timing Code** Descriptor. Priority 1.
03h + *N*[a]	7:0	**Second *N*-byte Timing Code** *N*-byte descriptor, if present. Priority 2.
03h + 2*N*[a]	7:0	**Third *N*-byte Timing Code** *N*-byte descriptor, if present. Priority 3.
…		**…**

a. N represents the timing code length that is specified in the Timing Code Size field (i.e., one or two bytes).

4.3.3 Type IX Timing – Formula-based Data Block

The Type IX Formula-based data block shall provide a means for a Sink device to expose timing support that is derived from an industry-standard formula. Block Revision 0 supports *CVT Standard v1.2* and higher's timing with standard blanking, and *CVT Standard v1.1* and higher's reduced blanking. Timing priority shall be in the order listed. Type IX Formula-based is an **optional** data block.

Table 4-20 and Table 4-21 list the data block's structure and descriptor formats, respectively.

Table 4-20: Type IX Timing – Formula-based Data Block Format

Offset	Bit #	Description/Format/Priority	Tag
00h	7:0	**Type IX Timing – Formula-based Timing Data Block** Value is 24h.	✔
01h	**Block Revision and Other Data**		
	2:0	**Block Revision** Revision ranges from 0 through 7. 000b = Revision 0 (default).	
	7:3	**RESERVED** Cleared to all 0s.	
02h	7:0	**Number of Payload Bytes in Block** Number of payload bytes within the data block ranges from 6 through 246 ($1 \le N \le 41$).[a] Value is based on $N \times 6 - 6$h, Ch, 12h, etc., through F6h. All other values are RESERVED.	
08h – 03h	47:0	**First Type IX Formula-based Timing Descriptor** 6-byte descriptor. Priority 1.	
0Eh – 09h	47:0	**Second Type IX Formula-based Timing Descriptor** 6-byte descriptor, if present. Priority 2.	
…		**…**	

a. N represents the number of formula-based timing descriptors in the data block.

Table 4-21: Type IX Formula-based Timing Descriptor Format

Byte #	Bit #	Description/Format	Flag	Tag
0	**Timing Options**			
	2:0	**Timing Formula/Algorithm** 000b = *CVT Standard v1.2* and higher's standard blanking. 001b = *CVT Standard v1.1* and higher's reduced blanking. 010b = *CVT Standard v1.2* and higher's reduced blanking. All other values are RESERVED.		✔
	3	**RESERVED** Cleared to 0.		
	4	**Video-optimized Refresh Rate Support** 0 = Refresh rate × (1000 / 1001) shall not be supported. 1 = Refresh rate × (1000 / 1001) shall be supported.	✔	
	6:5	**3D Stereo Support** 00b = This timing shall always be displayed in mono (no stereo). 01b = This timing shall always be displayed in stereo. 10b = This timing shall be displayed in mono or stereo, depending on a user action (such as wearing stereo glasses). 11b = RESERVED.	✔	
	7	**RESERVED** Cleared to 0.		
2:1	**Horizontal Active Image Pixels** Number of pixels ranges from 1 through 65,536.			
1	7:0	**Horizontal Active Image Pixels Low Bits 7:0** Value ranges from 00h through FFh.		
2	15:8	**Horizontal Active Image Pixels High Bits 15:8** Value ranges from 00h through FFh.		
4:3	**Vertical Active Image Lines** Number of lines ranges from 1 through 65,536.			
3	7:0	**Vertical Active Image Lines Low Bits 7:0** Value ranges from 00h through FFh.		
4	15:8	**Vertical Active Image Lines High Bits 15:8** Value ranges from 00h through FFh.		
5	7:0	**Refresh Rate** Refresh rate ranges from 1 through 256Hz. Value ranges from 00h through FFh.		

4.4 Dynamic Video Timing Range Limits Data Block

The Dynamic Video Timing Range Limits data block shall be used to convey the supported timing range of vertical frequencies. Any Sink device that exposes this data block shall indicate support for a seamless, continuous vertical frequency device that can support the seamless timing change for any timing that lies within the specified minimum and maximum pixel rates. Dynamic Video Timing Range Limits is an **optional** data block in Base sections.

Table 4-22 lists the size and order of each field, with all addresses relative to the beginning of the data block.

Table 4-22: Dynamic Video Timing Range Limits Data Block Fields

Offset	Bit #	Description/Format	Flag	Tag
00h	7:0	**Dynamic Video Timing Range Limits Data Block** Value is 25h.		✔
01h	**Block Revision and Other Data**			
	2:0	**Block Revision** Revision ranges from 0 through 7. 000b = Revision 0 (default).		
	7:3	**RESERVED** Cleared to all 0s.		
02h	7:0	**Number of Payload Bytes in Block** 09h = Data block is composed of nine payload bytes. All other values are RESERVED.		
05h– 03h	23:0	**Minimum Pixel Clock (in kHz)** Required. Specifies the minimum pixel rate above which dynamic video timing changes shall be supported, as per the reported scheme. Range shall be defined as 0.001 through 16,777.216MP/s. Value ranges from 000000h through FFFFFFh.		
08h – 06h	23:0	**Maximum Pixel Clock (in kHz)** Required. Specifies the maximum pixel rate below which dynamic video timing changes shall be supported, as per the reported scheme. Range shall be defined as 0.001 through 16,777.216MP/s. Value ranges from 000000h through FFFFFFh.		
09h	7:0	**Minimum Vertical Refresh Rate** Required. Specifies the display device's minimum dynamic vertical frequency. Vertical frequency shall be defined as follows: *(pixel_clock / horizontal_total) / vertical_total* Range shall be defined as 0 through 255Hz. Value ranges from 00h through FFh.		

Table 4-22: Dynamic Video Timing Range Limits Data Block Fields (Continued)

Offset	Bit #	Description/Format	Flag	Tag
0Ah	7:0	**Maximum Vertical Refresh Rate** Required. Specifies the display device's maximum dynamic vertical frequency. Range shall be defined as 0 through 255Hz. Value ranges from 00h through FFh.		
0Bh	**Dynamic Video Timing Range Support Flags**		✔	
	6:0	**RESERVED** Cleared to all 0s.		
	7	**Seamless Dynamic Video Timing Support** 0 = Seamless Dynamic Video Timing change shall not be supported with a fixed horizontal pixel rate and dynamic vertical blanking. 1= Seamless Dynamic Video Timing change shall be supported with a fixed horizontal pixel rate and dynamic vertical blanking.		

4.5 Display Interface Features Data Block

The Display Interface Features data block shall provide information about the supported pixel encoding format along with the supported color depths. The data block also provides information about the supported basic audio rates. Display Interface Features is a **mandatory** data block for product primary use case as a display and is **optional** otherwise.

Table 4-23 lists the size and order of each field, with all addresses relative to the beginning of the data block.

Table 4-23: Display Interface Features Data Block Fields

Offset	Bit #	Description/Format	Defined in	Tag
00h	7:0	**Display Interface Features Data Block** Value is 26h.		✔
01h	**Block Revision and Other Data**			
	2:0	**Block Revision** Revision ranges from 0 through 7. 000b = Revision 0 (default).		
	7:3	**RESERVED** Cleared to all 0s.		
02h	7:0	**Number of Payload Bytes in Block** 09h + *N* = Data block is composed of nine + *N* payload bytes.[a]		
03h	7:0	**Supported Interface Color Depth for RGB Encoding** Descriptor.	Section 4.5.1	
04h	7:0	**Supported Interface Color Depth for YCbCr 4:4:4 Encoding** Descriptor.	Section 4.5.1	

Table 4-23: Display Interface Features Data Block Fields (Continued)

Offset	Bit #	Description/Format	Defined in	Tag
05h	7:0	**Supported Interface Color Depth for YCbCr 4:2:2 Encoding** Descriptor.	Section 4.5.1	
06h	7:0	**Supported Interface Color Depth for YCbCr 4:2:0 Encoding** Descriptor.	Section 4.5.1	
07h	7:0	**Minimum Pixel Rate at Which YCbCr 4:2:0 Encoding Is Supported** Descriptor.	Section 4.5.2	
08h	7:0	**Supported Interface Audio Capability and Feature Flags** Descriptor.	Section 4.5.3	
09h	7:0	**Supported Interface Color Space and EOTF Standard Combination 1** Descriptor.	Section 4.5.4	
0Ah	7:0	**Supported Interface Color Space and EOTF Standard Combination 2** Descriptor. RESERVED.		
0Bh	7:0	**Number of Additional Supported Interface Color Space and EOTF (*N*)** Descriptor.[a]	Section 4.5.4	
0Bh + 1	7:0	**Additional Supported Interface Color Space and EOTF #1** Descriptor.	Section 4.5.4	
…		**…**		
0Bh + *N*[a]	7:0	**Additional Supported Interface Color Space and EOTF #*N*** Descriptor.	Section 4.5.4	

a. N represents the number of additional supported interface color space and EOTF combinations.

4.5.1 Supported Interface Color Depth-related Fields

The four Supported Interface Color Depth-related fields – one byte each for RGB, YCbCr 4:4:4, YCbCr 4:2:2, and YCbCr 4:2:0 pixel encodings, as listed in Table 4-24 – define the supported color depths for that particular pixel encoding:

- **Supported** – Color depth is supported for all DisplayID-exposed timings.
- **No support indicated** – Sink device has not explicitly reported a color-depth capability in the Display Interface Features data block; thus, the Source device shall rely on other DisplayID data blocks, –or– an interface-specific rule, to determine color-depth support. Examples of other data blocks that may report color-depth capability include the CTA YCbCr 4:2:0, HDMI vendor-specific, and HDMI Forum Vendor-specific data blocks.

Table 4-24: Supported Interface Color Depth-related Field Formats

Offset	Bit #	Description/Format
03h	**Supported Interface Color Depth for RGB Encoding**	
	0	**6 bits per Primary Color** 0 = No support indicated. 1 = Supported.
	1	**8 bits per Primary Color** 0 = No support indicated. 1 = Supported.
	2	**10 bits per Primary Color** 0 = No support indicated. 1 = Supported.
	3	**12 bits per Primary Color** 0 = No support indicated. 1 = Supported.
	4	**14 bits per Primary Color** 0 = No support indicated. 1 = Supported.
	5	**16 bits per Primary Color** 0 = No support indicated. 1 = Supported.
	7:6	**RESERVED** Cleared to all 0s.

Table 4-24: Supported Interface Color Depth-related Field Formats (Continued)

Offset	Bit #	Description/Format
04h	**Supported Interface Color Depth for YCbCr 4:4:4 Encoding**	
	0	**6 bits per Primary Color** 0 = No support indicated. 1 = Supported.
	1	**8 bits per Primary Color** 0 = No support indicated. 1 = Supported.
	2	**10 bits per Primary Color** 0 = No support indicated. 1 = Supported.
	3	**12 bits per Primary Color** 0 = No support indicated. 1 = Supported.
	4	**14 bits per Primary Color** 0 = No support indicated. 1 = Supported.
	5	**16 bits per Primary Color** 0 = No support indicated. 1 = Supported.
	7:6	**RESERVED** Cleared to all 0s.

Table 4-24: Supported Interface Color Depth-related Field Formats (Continued)

Offset	Bit #	Description/Format
05h	**Supported Interface Color Depth for YCbCr 4:2:2 Encoding**	
	0	**8 bits per Primary Color** 0 = No support indicated. 1 = Supported.
	1	**10 bits per Primary Color** 0 = No support indicated. 1 = Supported.
	2	**12 bits per Primary Color** 0 = No support indicated. 1 = Supported.
	3	**14 bits per Primary Color** 0 = No support indicated. 1 = Supported.
	4	**16 bits per Primary Color** 0 = No support indicated. 1 = Supported.
	7:5	**RESERVED** Cleared to all 0s.
06h	**Supported Interface Color Depth for YCbCr 4:2:0 Encoding**	
	0	**8 bits per Primary Color** 0 = No support indicated. 1 = Supported.
	1	**10 bits per Primary Color** 0 = No support indicated. 1 = Supported.
	2	**12 bits per Primary Color** 0 = No support indicated. 1 = Supported.
	3	**14 bits per Primary Color** 0 = No support indicated. 1 = Supported.
	4	**16 bits per Primary Color** 0 = No support indicated. 1 = Supported.
	7:5	**RESERVED** Cleared to all 0s.

4.5.2 Minimum Pixel Rate at Which YCbCr 4:2:0 Encoding Is Supported Field

The Minimum Pixel Rate at Which YCbCr 4:2:0 Encoding Is Supported field indicates the minimum pixel rate at which the Sink device shall support YCbCr 4:2:0 encoding, as indicated in the Supported Interface Color Depth for YCbCr 4:2:0 Encoding field (offset 06h). The Source device shall use YCbCr 4:2:0 encoding only for DisplayID-exposed modes that meet or exceed the listed pixel rate. A value of 00h in this field shall indicate that the Sink device can support YCbCr 4:2:0 encoding at all DisplayID-exposed modes. Table 4-25 defines the byte format.

Table 4-25: Minimum Pixel Rate at Which YCbCr 4:2:0 Encoding Is Supported Field Format

Offset	Bit #	Description
07h	7:0	**Minimum Pixel Rate at Which YCbCr 4:2:0 Encoding Is Supported** YCbCr 4:2:0 Minimum Pixel Rate Requirement = 74.25MP/s × field value 00h = Sink device can support YCbCr 4:2:0 encoding at all DisplayID-exposed modes.

4.5.3 Supported Interface Audio Capability and Feature Flags Field

The Supported Interface Audio Capability and Feature Flags field is a single byte that provides single-bit flags, as defined in Table 4-26, that shall be used to indicate the display's support for various features or functions. Bits 7, 6, and 5 define the display's basic audio capabilities.

Table 4-26: Supported Interface Audio Capability and Feature Flags Field Format

Offset	Bit #	Description/Format
08h	4:0	**RESERVED** Cleared to all 0s.
	5	**48-kHz Sample Rate Supported** 0 = Audio shall not be supported at the 48-kHz sample rate. 1 = Audio shall be supported at the 48-kHz sample rate.
	6	**44.1-kHz Sample Rate Supported** 0 = Audio shall not be supported at the 44.1-kHz sample rate. 1 = Audio shall be supported at the 44.1-kHz sample rate.
	7	**32-kHz Sample Rate Supported** 0 = Audio shall not be supported at the 32-kHz sample rate. 1 = Audio shall be supported at the 32-kHz sample rate.

4.5.4 Supported Interface Color Space and EOTF-related Fields

The Supported Interface Color Space and EOTF-related fields list the supported standard Color Space and EOTF that the display can decode. Table 4-27 lists the field formats.

Table 4-27: Supported Interface Color Space and EOTF-related Field Formats

Offset	Bit #	Description/Format
09h	**Supported Interface Color Space and EOTF Standard Combination 1**	
	0	**Color Space and EOTF as Defined in sRGB Specification** Support as defined in *IEC 61966-2-1*. 0 = No support indicated. 1 = Supported.
	1	**Color Space and EOTF as Defined in ITU-R BT.601 Specification** Support as defined in *ITU-R BT.601*. 0 = No support indicated. 1 = Supported.
	2	**Color Space as Defined in ITU-R BT.709 Specification and EOTF as Defined in ITU-R BT.1886 Specification** Support as defined in *ITU-R BT.709* and *ITU-R BT.1886*. 0 = No support indicated. 1 = Supported.
	3	**Color Space and EOTF as Defined in Adobe RGB Specification** Support as defined in *Adobe RGB*. 0 = No support indicated. 1 = Supported.
	4	**Color Space and EOTF as Defined in DCI-P3 Specification** Support as defined in *SMPTE RP 431-2*. 0 = No support indicated. 1 = Supported.
	5	**Color Space and EOTF as Defined in ITU-R BT.2020 Specification** Support as defined in *ITU-R BT.2020*. 0 = No support indicated. 1 = Supported.
	6	**Color Space as Defined in ITU-R BT.2020 Specification and EOTF as Defined in SMPTE ST 2084 Specification** Support as defined in *ITU-R BT.2020* and *SMPTE ST 2084*. 0 = No support indicated. 1 = Supported.
	7	**RESERVED** Cleared to 0.

Table 4-27: Supported Interface Color Space and EOTF-related Field Formats (Continued)

Offset	Bit #	Description/Format
0Ah	**Supported Interface Color Space and EOTF Standard Combination 2**	
	7:0	**RESERVED** Cleared to all 0s.
0Bh	**Number of Additional Supported Interface Color Space and EOTF (*N*)**	
	2:0	**Number of Additional Supported Interface Color Space and EOTF (*N*)** Indicates how many additional interface color space and EOTF combinations (*N*) are supported. A corresponding number, *N*, of Additional Supported Interface Color Space and EOTF bytes shall be exposed. This field shall be cleared to all 0s if additional supported interface color space and EOTF bytes do **not** need to be exposed. Value ranges from 000b through 111b.
	7:3	**RESERVED** Cleared to all 0s.

Table 4-27: Supported Interface Color Space and EOTF-related Field Formats (Continued)

Offset	Bit #	Description/Format
0Bh + 1	**Additional Supported Interface Color Space and EOTF #1**	
	3:0	**Supported Interface EOTF** 0h = Supported EOTF is not defined. Source device shall apply display interface-specific rules. 1h = EOTF shall be as defined in sRGB Color Space (*IEC 61966-2-1*). 2h = EOTF shall be as defined in *ITU-R BT.601* Color Space. 3h = EOTF shall be as defined in *ITU-R BT.1886* for *ITU-R BT.709* Color Space. 4h = EOTF shall be as defined in *Adobe RGB* Color Space. 5h = EOTF shall be as defined in DCI-P3 Color Space (*SMPTE RP 431-2*). 6h = EOTF shall be as defined in *ITU-R BT.2020* Color Space. 7h = Gamma function EOTF within the range of 1.00 through 3.54 (gamma value as per the Native Gamma EOTF field (offset 1Fh) in the Display Parameters data block). 8h = *SMPTE ST 2084* EOTF. 9h = Hybrid Log EOTF. Ah = Custom EOTF (details shall be defined in another data block). All other values are RESERVED.
	7:4	**Supported Interface Color Space** 0h = Supported Color Space is not defined in this field. Source device shall apply display interface-specific rules. 1h = sRGB Color Space (*IEC 61966-2-1*). 2h = *ITU-R BT.601* Color Space. 3h = *ITU-R BT.709* Color Space. 4h = *Adobe RGB* Color Space. 5h = DCI-P3 Color Space (*SMPTE RP 431-2*). 6h = *ITU-R BT.2020* Color Space. 7h = Custom Color Space (details shall be defined in another data block). All other values are RESERVED.
…		**…**

Table 4-27: Supported Interface Color Space and EOTF-related Field Formats (Continued)

Offset	Bit #	Description/Format
0Bh + *N*	**Additional Supported Interface Color Space and EOTF #*N***	
	3:0	**Supported Interface EOTF** 0h = Supported EOTF is not defined. Source device shall apply display interface-specific rules. 1h = EOTF shall be as defined in sRGB Color Space (*IEC 61966-2-1*). 2h = EOTF shall be as defined in *ITU-R BT.601* Color Space. 3h = EOTF shall be as defined in *ITU-R BT.1886* for *ITU-R BT.709* Color Space. 4h = EOTF shall be as defined in *Adobe RGB* Color Space. 5h = EOTF shall be as defined in DCI-P3 Color Space (*SMPTE RP 431-2*). 6h = EOTF shall be as defined in *ITU-R BT.2020* Color Space. 7h = Gamma function EOTF within the range of 1.00 through 3.54 (gamma value as per the Native Gamma EOTF field (offset 1Fh) in the Display Parameters data block). 8h = *SMPTE ST 2084* EOTF. 9h = Hybrid Log EOTF. Ah = Custom EOTF (details shall be defined in another data block). All other values are RESERVED.
	7:4	**Supported Interface Color Space** 0h = Supported Color Space is not defined in this field. Source device shall apply display interface-specific rules. 1h = sRGB Color Space (*IEC 61966-2-1*). 2h = *ITU-R BT.601* Color Space. 3h = *ITU-R BT.709* Color Space. 4h = *Adobe RGB* Color Space. 5h = DCI-P3 Color Space (*SMPTE RP 431-2*). 6h = *ITU-R BT.2020* Color Space. 7h = Custom Color Space (details shall be defined in another data block). All other values are RESERVED.

4.6 Stereo Display Interface Data Block

Although the Stereo Display Interface data block is **optional**, this data block shall be present whenever the 3D Stereo Support flags in any Detailed Timing descriptor indicate that the display is capable of displaying stereoscopic images. The data block provides information regarding the data format that is needed for sending stereo image pairs across the interface to the display.

A Block Revision 0 Stereo Display Interface data block contains one sub-block that describes the interface method to the stereoscopic display. Byte 03h specifies the sub-block's length so that other sub-blocks can be added in the future. The sub-block contains a 1-byte Stereo Interface Method Code field (offset 04h) and a set of Interface Method-specific parameters.

A Block Revision 1 Stereo Display Interface data block adds a provision for optionally exposing a supported timing descriptor for a given 3D stereo interface descriptor. The 3D Stereo Timing Support field (offset 01h, bits 7:6) shall be used to indicate which stereo method to apply to the specified timing.

Table 4-28 lists the size and order of each field, including the corresponding timing support, with all addresses relative to the beginning of the data block. Whenever the 3D Stereo Timing Support field indicates that Timing Code shall be listed as part of the Stereo Display Interface data block (bit 6 of the field (offset 01h) is set to 1), the data block shall have additional timing descriptor(s) following the Interface Method-specific Parameters field (offset 05h).

Table 4-28: Stereo Display Interface Data Block Fields

Offset	Bit #	Description/Format	Tag
00h	7:0	**Stereo Display Interface Data Block** Value is 27h.	✔
01h	**Block Revision and Other Data**		
	2:0	**Block Revision** Revision ranges from 0 through 7. 000b = Revision 0 (default).	
	5:3	**RESERVED** Cleared to all 0s.	
	7:6	**3D Stereo Timing Support** 00b = Data block shall apply only to timing descriptors that explicitly report 3D capability. 01b = Data block shall apply only to timing descriptors that explicitly report 3D capability, and the Timing Code shall be listed as part of the Stereo Display Interface data block. 10b = Data block shall apply to all listed timing descriptors within any of the timing blocks. 11b = Only the Timing Code shall be listed as part of the Stereo Display Interface data block.	
02h	7:0	**Number of Payload Bytes in Block** Value is based on $N + 2$ (where N represents the number of payload bytes in the stereo interface method-specific parameters), and ranges from 02h through FFh.	

Table 4-28: Stereo Display Interface Data Block Fields (Continued)

Offset	Bit #	Description/Format	Tag
03h	7:0	**Number of Bytes in Stereo Interface Method** Number of bytes value is based on *N* + 1 (where *N* represents the number of bytes in the stereo interface method-specific parameters).	
04h	7:0	**Stereo Interface Method Code** Descriptor. See Section 4.6.1.	
05h	7:0	**Interface Method-specific Parameters** Descriptor for interface method-specific parameters (e.g., *N* bytes).	
…		**…** Descriptor.	
N + 04h	7:0	Descriptor.	
N + 05h	**3D Timing Descriptor** Exists only when bit 6 of the 3D Stereo Timing Support field (offset 01h) is set to 1.		
	4:0	**Number of Supported Timing Code** Value ranges from 00h through 1Fh (e.g., *M*).	
	5	**RESERVED** Cleared to 0.	
	7:6	**Timing Code Type** 00b = DMT timing code. 01b = CTA VIC timing code. 10b = HDMI VIC timing code. 11b = RESERVED timing code type.	
N + 06h	7:0	**1-byte Timing Code – First Code** 1-byte descriptor.	
…		**…** 1-byte descriptor.	
N + *M* + 06h	7:0	**1-byte Timing Code – *M*th Code** 1-byte descriptor.	

Table 4-28: Stereo Display Interface Data Block Fields (Continued)

Offset	Bit #	Description/Format	Tag
N + *M* + 07h	**3D Timing Descriptor** Exists only when bit 6 of the 3D Stereo Timing Support field (offset 01h) is set to 1.		
	4:0	**Number of Supported Timing Code** Value ranges from 00h through 1Fh (e.g., *P*).	
	5	**RESERVED** Cleared to 0.	
	7:6	**Timing Code Type** 00b = DMT timing code. 01b = CTA VIC timing code. 10b = HDMI VIC timing code. 11b = RESERVED timing code type.	
N + *M* + 08h	7:0	**1-byte Timing Code** 1-byte descriptor.	
…		**…** 1-byte descriptor.	
N + *M* + *P* + 08h	7:0	**1-byte Timing Code** 1-byte descriptor.	

4.6.1 Stereo Interface Method Code Field

The 1-byte Stereo Interface Method Code field shall encode the interface method, as defined in Table 4-29. Depending on the Code value, there are one, two, or eight bytes of additional parameters that further describe the interface data format.

Table 4-29: Stereo Interface Method Codes

Code (Offset 04h Field Value)	Interface Method	# of Parameter Bytes	Defined in
00h	Frame/Field Sequential Stereo.	1	Section 4.6.1.1
01h	Side-by-side Stereo.	1	Section 4.6.1.2
02h	Pixel-interleaved Stereo.	8	Section 4.6.1.3
03h	Dual Interface, Left and Right Separate.	1	Section 4.6.1.4
04h	Multi-view.	2	Section 4.6.1.5
05h	Stacked Frame Stereo.	1	Section 4.6.1.6
FEh – 06h	RESERVED.	–	
FFh	Proprietary Stereo Interface Methods.	0	Section 4.6.1.7

4.6.1.1 Stereo Interface Method – Frame/Field Sequential Stereo

The Frame/Field Sequential stereo interface method indicates that the display shall expect an image sequence that consists of left, right, left, etc. The display includes a selection device (e.g., shutter glasses or Z-screen) that derives left/right information through a VESA Standard Connector for Stereoscopic Display Hardware or through another standard method. (Other proprietary interfaces shall be supported only for legacy devices.) Table 4-30 defines the format for this interface method.

Table 4-30: Frame/Field Sequential Stereo Interface Method Stereo Display Interface Data Block Fields

Offset	Bit #	Description/Format	Tag
00h	7:0	**Stereo Display Interface Data Block** Value is 27h.	✔
01h	**Block Revision and Other Data**		
	2:0	**Block Revision** Revision ranges from 0 through 7. 000b = Revision 0 (default).	
	6:3	**RESERVED** Cleared to all 0s.	
	7	**3D Stereo Timing Support** 0 = Data block shall apply only to timing descriptors that explicitly report 3D capability. 1 = Data block shall apply to all listed timing descriptors within any of the timing blocks.	
02h	7:0	**Number of Payload Bytes in Block** 03h = Data block is composed of three payload bytes.	
03h	7:0	**Number of Bytes in Stereo Interface Method** 02h = Two bytes.	
04h	7:0	**Stereo Interface Method Code** 00h = Frame/Field Sequential Stereo.	
05h	**Stereo Polarity Descriptor**		
	0	**Stereo Polarity Descriptor** ***Note:*** *The polarity shall be 0 for devices that use a VESA Standard Connector for Stereoscopic Display Hardware. For other methods, the polarity should also be 0 (i.e., the Stereo Sync signal is 1 when transmitting a left-eye image).* 0 = Stereo Sync shall be a logical 1 when transmitting a left-eye image, and a logical 0 when transmitting a right-eye image. 1 = Stereo Sync shall be a logical 1 when transmitting a right-eye image, and a logical 0 when transmitting a left-eye image.	
	7:1	**RESERVED** Cleared to all 0s.	

4.6.1.2 Stereo Interface Method – Side-by-side Stereo

The Side-by-side stereo interface method indicates that the display shall expect a single image, the left half of which corresponds to the first of the stereo pair and the right half of which corresponds to the second of the stereo pair.

No scaling is assumed. For example, an 800x600 display that takes side-by-side data would declare a detailed timing descriptor of resolution 1600x600 marked as always being displayed in stereo.

Table 4-31 defines the format for this interface method.

Table 4-31: Side-by-side Stereo Interface Method Stereo Display Interface Data Block Fields

Offset	Bit #	Description/Format	Tag
00h	7:0	**Stereo Display Interface Data Block** Value is 27h.	✔
01h	**Block Revision and Other Data**		
	2:0	**Block Revision** Revision ranges from 0 through 7. 000b = Revision 0 (default).	
	6:3	**RESERVED** Cleared to all 0s.	
	7	**3D Stereo Timing Support** 0 = Data block shall apply only to timing descriptors that explicitly report 3D capability. 1 = Data block shall apply to all listed timing descriptors within any of the timing blocks.	
02h	7:0	**Number of Payload Bytes in Block** 03h = Data block is composed of three payload bytes.	
03h	7:0	**Number of Bytes in Stereo Interface Method** 02h = Two bytes.	
04h	7:0	**Stereo Interface Method Code** 01h = Side-by-side Stereo.	
05h	**View Identity Descriptor**		
	0	**View Identity Descriptor** 0 = Left half of the image represents the left-eye view. Right half of the image represents the right-eye view. 1 = Left half of the image represents the right-eye view. Right half of the image represents the left-eye view.	
	7:1	**RESERVED** Cleared to all 0s.	

4.6.1.3 Stereo Interface Method – Pixel-interleaved Stereo

The Pixel-interleaved stereo interface method indicates that the display shall expect a single image in which pixels from the left- and right-eye views are interleaved in a repeating pattern that is defined by the eight parameter bytes. Table 4-32 defines the format for this interface method.

Table 4-32: Pixel-interleaved Stereo Interface Method
Stereo Display Interface Data Block Fields

Offset	Bit #	Description/Format	Tag
00h	7:0	**Stereo Display Interface Data Block** Value is 27h.	✔
01h	**Block Revision and Other Data**		
	2:0	**Block Revision** Revision ranges from 0 through 7. 000b = Revision 0 (default).	
	6:3	**RESERVED** Cleared to all 0s.	
	7	**3D Stereo Timing Support** 0 = Data block shall apply only to timing descriptors that explicitly report 3D capability. 1 = Data block shall apply to all listed timing descriptors within any of the timing blocks.	
02h	7:0	**Number of Payload Bytes in Block** 0Ah = Data block is composed of 10 payload bytes.	
03h	7:0	**Number of Bytes in Stereo Interface Method** 09h = Nine bytes.	
04h	7:0	**Stereo Interface Method Code** 02h = Pixel-interleaved Stereo.	
0Ch – 05h	63:0	**8x8 Interleave Pattern Descriptor** Defines an 8x8 pixel pattern that starts at the display's top-left pixel, which is then horizontally and vertically repeated until the display's bottom-right pixel is reached. The interleave pattern's first byte describes the topmost line, the next byte the second line, etc. (e.g., bit 7 of each byte describes the leftmost image pixel, bit 6 describes the first pixel from the left, etc.). 0 = Pixel position for the descriptor bit shall be a right-eye image pixel. 1 = Pixel position for the descriptor bit shall be a left-eye image pixel. ***Note:*** *2-way interleaved stereo can easily be represented in the Pixel-interleaved stereo interface method. For example, the 2-way interleaved stereo, right-eye image on even lines can be represented as FF00FF00FF00FF00h, with the first byte (FFh) placed at offset 05h, the second byte (00h) placed at offset 06h, etc., with the last byte (00h) placed at offset 0Ch.*	

4.6.1.4 Stereo Interface Method – Dual Interface, Left and Right Separate

The Dual Interface, Left and Right Separate stereo interface method applies to display devices that have dual interfaces. One interface shall carry the image sequence for the left-eye view, and the other interface shall carry the image sequence for the right-eye view. To support a wide variety of stereo display techniques, this interface method shall support optional mirroring. Table 4-33 defines the format for this interface method.

Table 4-33: Dual Interface, Left and Right Separate Stereo Interface Method Stereo Display Interface Data Block Fields

Offset	Bit #	Description/Format	Tag
00h	7:0	**Stereo Display Interface Data Block** Value is 27h.	✔
01h	**Block Revision and Other Data**		
	2:0	**Block Revision** Revision ranges from 0 through 7. 000b = Revision 0 (default).	
	6:3	**RESERVED** Cleared to all 0s.	
	7	**3D Stereo Timing Support** 0 = Data block shall apply only to timing descriptors that explicitly report 3D capability. 1 = Data block shall apply to all listed timing descriptors within any of the timing blocks.	
02h	7:0	**Number of Payload Bytes in Block** 03h = Data block is composed of three payload bytes.	
03h	7:0	**Number of Bytes in Stereo Interface Method** 02h = Two bytes.	
04h	7:0	**Stereo Interface Method Code** 03h = Dual Interface, Left and Right Separate.	

Table 4-33: Dual Interface, Left and Right Separate Stereo Interface Method Stereo Display Interface Data Block Fields (Continued)

Offset	Bit #	Description/Format	Tag
05h	**Left and Right Polarity and Mirroring Descriptors**		
	0	**Left and Right Polarity Descriptor** ***Note:*** *If the Source device encounters two left- or two right-eye view devices, the Source device shall arbitrarily assign left-eye view to one device and right-eye view to the other device.* 0 = This interface carries the right-eye view. 1 = This interface carries the left-eye view.	
	2:1	**Mirroring Descriptor** 00b = No mirroring. 01b = Left/right are mirrored. 10b = Top/bottom are mirrored. 11b = RESERVED.	
	7:3	**RESERVED** Cleared to all 0s.	

4.6.1.5 Stereo Interface Method – Multi-view

The Multi-view stereo interface method indicates a display with more than two views. For example, in addition to the standard display, a lenticular or barrier-type auto-stereoscopic display is also included. Table 4-34 defines the format for this interface method.

Table 4-34: Multi-view Stereo Interface Method
Stereo Display Interface Data Block Fields

Offset	Bit #	Description/Format	Tag
00h	7:0	**Stereo Display Interface Data Block** Value is 27h.	✔
01h	**Block Revision and Other Data**		
	2:0	**Block Revision** Revision ranges from 0 through 7. 000b = Revision 0 (default).	
	6:3	**RESERVED** Cleared to all 0s.	
	7	**3D Stereo Timing Support** 0 = Data block shall apply only to timing descriptors that explicitly report 3D capability. 1 = Data block shall apply to all listed timing descriptors within any of the timing blocks.	
02h	7:0	**Number of Payload Bytes in Block** 04h = Data block is composed of four payload bytes.	
03h	7:0	**Number of Bytes in Stereo Interface Method** 03h = Three bytes.	
04h	7:0	**Stereo Interface Method Code** 04h = Multi-view.	
05h	7:0	**Number of Views Descriptor** Indicates the number of discrete full color views that shall be needed to compose the image to drive this display. Number of Views descriptors shall be greater than 2 (value of 03h or higher).	
06h	7:0	**View Interleaving Method Code Descriptor** VESA shall assign the view-interleaving method code at the request of a display manufacturer. The manufacturer shall provide VESA with a description of the view-interleaving method at the time of the request. Number of View-interleaving Method Code descriptors ranges from 0 through 255.	

4.6.1.6 Stereo Interface Method – Stacked Frame Stereo

The Stacked Frame stereo interface method indicates that the display shall expect a single image, the top and bottom portions of which correspond to the first and second of the stereo pair, respectively. Between the two image portions are additional unused image lines whose count is equal to the timing's vertical blank.

No scaling is assumed. For example, an 800x600 display with a vertical blank of 36 lines that takes stacked frame stereo would declare a detailed timing descriptor of resolution 800x600. This detailed timing shall be used as-is for monoscopic timings. For stereo timings, resolution output to the display should be increased to 800x1236 (i.e., 800x600 for the top and bottom portions, plus an additional 36 unused lines between them).

Table 4-35 defines the format for this interface method.

Table 4-35: Stacked Frame Stereo Interface Method
Stereo Display Interface Data Block Fields

Offset	Bit #	Description/Format	Tag
00h	7:0	**Stereo Display Interface Data Block** Value is 27h.	✔
01h	**Block Revision and Other Data**		
	2:0	**Block Revision** Revision ranges from 0 through 7. 000b = Revision 0 (default).	
	6:3	**RESERVED** Cleared to all 0s.	
	7	**3D Stereo Timing Support** 0 = Data block shall apply only to timing descriptors that explicitly report 3D capability. 1 = Data block shall apply to all listed timing descriptors within any of the timing blocks.	
02h	7:0	**Number of Payload Bytes in Block** 03h = Data block is composed of three payload bytes.	
03h	7:0	**Number of Bytes in Stereo Interface Method** 02h = Two bytes.	
04h	7:0	**Stereo Interface Method Code** 05h = Stacked Frame Stereo.	
05h	**View Identity Descriptor**		
	0	**View Identity Descriptor** 0 = Top and bottom portions of the image represent the left- and right-eye views, respectively. 1 = RESERVED. **Shall not be used.**	
	7:1	**RESERVED** Cleared to all 0s. **Shall not be used.**	

4.6.1.7 Stereo Interface Method – Proprietary Stereo Interface Methods

For stereo displays that use a proprietary interface method, a Stereo Display Interface data block is needed and the Stereo Interface Method Code should be programmed to FFh. Elsewhere in the DisplayID structure, vendors should include a Vendor-specific data block that defines the stereo parameters for the display's proprietary stereo interface method. Table 4-36 defines the format for this interface method.

Table 4-36: Proprietary Stereo Interface Methods
Stereo Display Interface Data Block Fields

Offset	Bit #	Description/Format	Tag
00h	7:0	**Stereo Display Interface Data Block** Value is 27h.	✔
01h	**Block Revision and Other Data**		
	2:0	**Block Revision** Revision ranges from 0 through 7. 000b = Revision 0 (default).	
	6:3	**RESERVED** Cleared to all 0s.	
	7	**3D Stereo Timing Support** 0 = Data block shall apply only to timing descriptors that explicitly report 3D capability. 1 = Data block shall apply to all listed timing descriptors within any of the timing blocks.	
02h	7:0	**Number of Payload Bytes in Block** 02h = Data block is composed of two payload bytes.	
03h	7:0	**Number of Bytes in Stereo Interface Method** 01h = One byte.	
04h	7:0	**Stereo Interface Method Code** FFh = Proprietary Stereo Interface Methods.	

4.7 Tiled Display Topology Data Block

The Tiled Display Topology data block shall be used to describe whether tiled displays are implemented in a single enclosure, –or– as a collection of separate physical displays configured within a tiled display topology. The Source device may use information provided by the data block to determine the best way to automatically configure the display and/or to provide useful information to the user with regard to display configuration. The Tiled Display Topology data block is **optional** in a Base section.

Table 4-37 lists the size and order of each field, with all addresses relative to the beginning of the data block.

Table 4-37: Tiled Display Topology Data Block Fields

Offset	Bit #	Description/Format	Defined in	Tag
00h	7:0	**Tiled Display Topology Data Block** Value is 28h.		✔
01h	**Block Revision and Other Data**			
	2:0	**Block Revision** Revision ranges from 0 through 7. 000b = Revision 0 (default).		
	7:3	**RESERVED** Cleared to all 0s.		
02h	7:0	**Number of Payload Bytes in Block** 16h = Data block is composed of 22 payload bytes.		
03h	7:0	**Tiled Display and Tile Capabilities**	Section 4.7.1	
06h, 04h	7:4, 7:0	**Tiled Display Topology**	Section 4.7.2	
06h – 05h	3:0, 7:0	**Tile Location**	Section 4.7.2	
0Ah – 07h	31:0	**Tile Size**	Section 4.7.3	
0Fh – 0Bh	39:0	**Tile Pixel Multiplier and Tile Bezel-related Information**	Section 4.7.4	
18h – 10h	71:0	**Tiled Display Topology ID**	Section 4.7.5	

4.7.1 Tiled Display and Tile Capabilities Field

The Tiled Display and Tile Capabilities field details the tiled display behavior of various tile configurations (driven by the Source device) that shall expose the tiled display physical enclosure information. Table 4-14 lists the field formats.

Table 4-38: Tiled Display and Tile Capabilities Field Format

Offset	Bit #	Description/Format
03h	2:0	**Tile Behavior when It Is the Only Tile Receiving an Image from the Source** 000b = Behavior cannot be described by other values defined in this revision of the data block. 001b = Image is displayed at the location specified by the Tile Location field (offsets 06h – 05h). 010b = Image is scaled to fit the entire tiled display. 011b = Image is cloned to all other tiles within the entire tiled display. All other values are RESERVED.
	4:3	**Tile Behavior when More than One Tile and Fewer than Total Number of Tiles are Driven by the Source** ***Note:*** *When the total number of tiles is two, this field (bits 4:3) becomes* ***don't care*** *because the number of tiles driven by the Source device cannot be more than one and fewer than the total number of tiles.* 00b = Behavior cannot be described by other values defined in this revision of the data block. 01b = When this tile is receiving an image, the image is displayed at the location specified by the Tile Location field (offsets 06h – 05h). All other values are RESERVED.
	4:0	**Tile Behavior when Subsets of the Tiles of the Entire Tiled Display Are Receiving Images from the Source** ***Note:*** *If all tiles are receiving images from the Source device, each image is displayed at the location specified by the Tile Location field (offsets 06h – 05h).*
	5	**RESERVED** Cleared to 0.
	6	**Tile Bezel Descriptor** 0 = Tile Bezel Information descriptor is not available. Offsets 0Ch through 0Fh shall be cleared to all 0s. ***Note:*** *The Tile Pixel Multiplier shall be cleared to all 0s for this case.* 1 = Offsets 0Ch through 0Fh contain the Tile Bezel Information descriptor. The Tile Pixel Multiplier in the Tile Pixel Multiplier and Tile Bezel-related Information field at offset 0Bh shall be a non-zero value. When bit 7 is set to 1, this bit shall also be set to 1.
	7	**Physical Display Enclosure** 0 = Tiled display consists of multiple physical display enclosures. 1 = Tiled display is within a single physical display enclosure.

Note: *If a Sink device exposes a non-native mode in DisplayID, the Source device may send that mode as-is and the Sink device shall appropriately handle stitching of the tiles across the seam for a single enclosure display case. The stitching shall work across all supported scaling options that the Sink device exposes for that resolution in the on-screen display. If the Sink device cannot handle stitching across the seam with a scaled image for a given resolution, the Sink device should do one of the following:*

- ***Not** report the mode in DisplayID, –or–*
- *If the mode is reported in DisplayID, the Sink device should center the image and **not** expose the scaling option in the on-screen display for that resolution*

4.7.2 Tiled Display Topology and Tile Location Fields

These fields describe the tiled display topology and tile location. The lower four bits and upper two bits of the Tiled Display Topology fields are located at offsets 04h and 06h, respectively. The lower four bits and upper two bits of the Tile Location fields are located at offsets 05h and 06h, respectively. Table 4-39 lists the field formats.

Table 4-39: Tiled Display Topology and Tile Location Field Formats

Offset	Bit #	Description/Format
04h	**Total Number of Tiles (Low Bits)**	
	3:0	**Total Number of Vertical Tiles 3:0** Value ranges from 0h through Fh. Used in combination with bits 5:4 in offset 06h (high bits) to define the total number of vertical tiles, which ranges from 1 through 64.
	7:4	**Total Number of Horizontal Tiles 3:0** Value ranges from 0h through Fh. Used in combination with bits 7:6 in offset 06h (high bits) to define the total number of horizontal tiles, which ranges from 1 through 64.
05h	**Tile Location (Low Bits)**	
	3:0	**Vertical Tile Location 3:0** Value ranges from 0h through Fh. Used in combination with bits 1:0 in offset 06h (high bits) to define the vertical tile location, which ranges from 1 through 64.
	7:4	**Horizontal Tile Location 3:0** Value ranges from 0h through Fh. Used in combination with bits 3:2 in offset 06h (high bits) to define the horizontal tile location, which ranges from 1 through 64.

Table 4-39: Tiled Display Topology and Tile Location Field Formats (Continued)

Offset	Bit #	Description/Format
06h	**Tile Location and Total Number of Tiles (High Bits)**	
	1:0	**Vertical Tile Location 5:4** Value ranges from 0h through 3h. Used in combination with bits 3:0 in offset 05h (low bits) to define the vertical tile location, which ranges from 1 through 64.
	3:2	**Horizontal Tile Location 5:4** Value ranges from 0h through 3h. Used in combination with bits 7:4 in offset 05h (low bits) to define the horizontal tile location, which ranges from 1 through 64.
	5:4	**Total Number of Vertical Tiles 5:4** Value ranges from 0h through 3h. Used in combination with bits 3:0 in offset 04h (low bits) to define the total number of vertical tiles, which ranges from 1 through 64.
	7:6	**Total Number of Horizontal Tiles 5:4** Value ranges from 0h through 3h. Used in combination with bits 7:4 in offset 04h (low bits) to define the total number of horizontal tiles, which ranges from 1 through 64.

4.7.3 Tile Size Field

The Tile Size field describes the horizontal and vertical size of individual tiles (and **not** the size of the entire tiled display) at native resolution. Table 4-40 lists the field formats.

Table 4-40: Tile Size Field Formats

Offset	Byte #	Bit #	Description/Format
07h	0	7:0	**Horizontal Size 7:0** Value ranges from 00h through FFh. Used in combination with bits 7:0 in offset 08h (high bits) to define the horizontal tile size, which ranges from 1 through 65,536 pixels.
08h	1	7:0	**Horizontal Size 15:8** Value ranges from 00h through FFh. Used in combination with bits 7:0 in offset 07h (low bits) to define the horizontal tile size, which ranges from 1 through 65,536 pixels.
09h	2	7:0	**Vertical Size 7:0** Value ranges from 00h through FFh. Used in combination with bits 7:0 in offset 0Ah (high bits) to define the vertical tile size, which ranges from 1 through 65,536 lines.
0Ah	3	7:0	**Vertical Size 15:8** Value ranges from 00h through FFh. Used in combination with bits 7:0 in offset 09h (low bits) to define the vertical tile size, which ranges from 1 through 65,536 lines.

4.7.4 Tile Pixel Multiplier and Tile Bezel-related Information Fields

These fields describe the Tile Pixel Multiplier and Tile Bezel-related information. If bezel information is indicated to be present (i.e., bit 6 of the Tiled Display and Tile Capabilities field is set to 1), the Tile Pixel Multiplier value shall be non-zero. Table 4-41 lists the field formats.

Note: *The bezel size in pixels calculation was originally based on the Tile Pixel Multiplier representing a unit of pixels per centimeter and bezel size in millimeters; however, if required, any generic multiplier can be used to achieve higher pixel precision.*

Table 4-41: Tile Pixel Multiplier and Tile Bezel-related Information Field Formats

Offset	Byte #	Bit #	Description/Format
0Bh	0	7:0	**Tile Pixel Multiplier** Value ranges from 00h through FFh for pixel multipliers of 0 through 255.
0Ch	1	7:0	**Tile Top Bezel Size** *Top Bezel in pixels = (Tile Pixel Multiplier × Tile Top Bezel Size × 0.1)* Value ranges from 00h through FFh for bezel sizes of 0 through 255.
0Dh	2	7:0	**Tile Bottom Bezel Size** *Bottom Bezel in pixels = (Tile Pixel Multiplier × Tile Bottom Bezel Size × 0.1)* Value ranges from 00h through FFh for bezel sizes of 0 through 255.
0Eh	3	7:0	**Tile Right Bezel Size** *Right Bezel in pixels = (Tile Pixel Multiplier × Tile Right Bezel Size × 0.1)* Value ranges from 00h through FFh for bezel sizes of 0 through 255.
0Fh	4	7:0	**Tile Left Bezel Size** *Left Bezel in pixels = (Tile Pixel Multiplier × Tile Left Bezel Size × 0.1)* Value ranges from 00h through FFh for bezel sizes of 0 through 255.

4.7.5 Tiled Display Topology ID-related Fields

The Tiled Display Manufacturer/Vendor ID, Tiled Display Product ID Code, and Tiled Display Serial Number fields comprise the Tiled Display Topology ID field descriptors. The Source device uses these descriptors to uniquely identify the topology of single enclosure and multiple enclosure (i.e., a group of discrete displays) topologies. These fields shall be populated with valid, unique values. For single enclosure topologies, these fields should match the Product Identification data block fields (although even if the Serial Number field in the Product Identification data block is unused, the Tiled Display Serial Number shall be populated with a unique value). For multiple enclosure topologies, these fields may differ from the Product Identification data block fields because the fields are expected to be unique to each discrete display. Table 4-42 lists the field formats.

Note: *Associating tiles with a specific tiled display is necessary to be able to produce expected behavior when more than one tiled display is connected to a Source device. Without this correct association, tiles from different tiled displays may be grouped together by the Source device, thereby resulting in tiled images potentially being sent to, and displayed on, the wrong physical display device.*

Table 4-42: Tiled Display Topology ID-related Field Formats

Offset	Byte #	Bit #	Description/Format
12h – 10h	2:0	23:0	**Tiled Display Manufacturer/Vendor ID** Descriptor.
14h – 13h	4:3	15:0	**Tiled Display Product ID Code** Descriptor.
18h – 15h	8:5	31:0	**Tiled Display Serial Number** Descriptor.

4.7.5.1 Tiled Display Manufacturer/Vendor ID Field

The Tiled Display Manufacturer/Vendor ID field listed in Table 4-43 is a **required** element of the Tiled Display Topology data block. The Tiled Display Manufacturer/Vendor ID field contains the display manufacturer's 3-byte IEEE OUI code, which identifies the display's manufacturer or vendor. Table 4-43 lists the field format.

The IEEE issues IEEE OUIs. Contact the IEEE at http://standards.ieee.org/develop/regauth/oui/ for registration and/or further details.

Table 4-43: Tiled Display Manufacturer/Vendor ID Field Format

Offset	Byte #	Bit #	Description/Format
10h	0	7:0	**IEEE OUI First Byte** Byte code.
11h	1	7:0	**IEEE OUI Second Byte** Byte code.
12h	2	7:0	**IEEE OUI Third Byte** Byte code.

4.7.5.2 Tiled Display Product ID Code Field

The Tiled Display Product ID Code field listed in Table 4-42 is a **required** element of the Tiled Display Topology data block. The Tiled Display Product ID Code field contains a 2-byte vendor-assigned product code. The field's length helps to differentiate between multiple models from the same manufacturer. If this field is used to represent a Product ID code (e.g., a model number), the number shall be stored in hexadecimal format with the LSB listed first. Table 4-44 lists the field format.

Table 4-44: Tiled Display Product ID Code Field Format

Offset	Byte #	Bit #	Description/Format
13h	0	7:0	**Product ID Code LSB** Value ranges from 00h through FFh.
14h	1	7:0	**Product ID Code MSB** Value ranges from 00h through FFh.

4.7.5.3 Tiled Display Serial Number Field

The Tiled Display Serial Number field listed in Table 4-42 is a **required** element of the Tiled Display Topology data block. The Tiled Display Serial Number field is a 32-bit serial number that shall be used to differentiate between individual instances of the same display product model. This field's bit order follows the order listed in Table 4-45. The four bytes of the serial number are listed with the LSB first. The serial number ranges from 1 through 4,294,967,295. This serial number shall be a number only – it shall **not** represent an ASCII code. Value "00h, 00h, 00h, 00h" is RESERVED and shall **not** be used.

Note: *Unlike the Serial Number field in the Product Identification data block, a value of 0 does **not** indicate that this field is unused.*

Table 4-45: Tiled Display Serial Number Field Format

Offset	Byte #	Bit #								Description
		7	6	5	4	3	2	1	0	
15h	0	7	6	5	4	3	2	1	0	**ID Serial Number**
16h	1	15	14	13	12	11	10	9	8	
17h	2	23	22	21	20	19	18	17	16	
18h	3	31	30	29	28	27	26	25	24	

4.8 ContainerID Data Block

The ContainerID data block provides a unique identifier that shall be used to associate the display device, and other features (e.g., audio devices) described in the DisplayID structure, with instances of other related devices (notably USB devices). Related devices shall typically reside within the same physical enclosure (e.g., a display device and camera device that are part of the same monitor enclosure). Container ID is a **mandatory** data block for product primary use case as a multi-function device and is **optional** otherwise.

Table 4-46 lists the size and order of each field, with all addresses relative to the beginning of the data block.

Table 4-46: ContainerID Data Block Fields

Offset	Bit #	Description/Format	Defined in	Tag
00h	7:0	**ContainerID Data Block** Value is 29h.		✔
01h	**Block Revision and Other Data**			
	2:0	**Block Revision** Revision ranges from 0 through 7. 000b = Revision 0 (default).		
	7:3	**RESERVED** Cleared to all 0s.		
02h	7:0	**Number of Payload Bytes in Block** 10h = Data block is composed of 16 payload bytes.		
12h – 03h	127:0	**ContainerID** 16-byte Universally Unique Identifier (UUID).	Section 4.8.1	

4.8.1 ContainerID Field

The ContainerID field is a 128-bit UUID that is unique to an individual device instance. The same ContainerID value provided in the ContainerID data block should be provided by other instances of devices that:

- Reside within the same enclosure, –or–
- Are otherwise to be associated with the specific device instance that is described by the ContainerID data block

The 16-byte UUID is typically displayed in human-readable form as a series of hexadecimal characters and hyphens that separate the bytes into fields, such as:

11223344-5566-7788-99AA-BBCCDDEEFF00

The individual fields, bytes, or bits of a UUID may have different meanings, depending on how the UUID is constructed; however, the 16-byte UUID itself shall be interpreted as a 16-byte binary-format value with no magnitude. The UUID's first byte (11h in the example) is placed at offset 03h, the second byte (22h in the example) is placed at offset 04h, etc., with the last byte (00h in the example) placed at offset 12h.

Note: *For details regarding how to generate a UUID, see IETF RFC 4122.*

Table 4-47 lists the ContainerID field format.

Table 4-47: ContainerID Field Format

Offset	Byte #	Bit #	Description/Format
03h	0	7:0	**UUID First Byte** Byte code.
04h	1	7:0	**UUID Second Byte** Byte code.
…			**…**
12h	15	7:0	**UUID Sixteenth Byte** Byte code.

4.9 Vendor-specific Data Block

The Vendor-specific data block shall be used for proprietary implementations that are not supported by *DisplayID v2.0*. The data block can also be used for a standards organization-specific data block. This block should be used only when the required data cannot be conveyed by a standard data block that has already been defined by VESA or the CTA. The vendor is then encouraged to propose a new standard data block to VESA. Vendor-specific is an **optional** data block in Base sections.

The data block shall conform to the format listed in Table 4-48 to ensure that the DisplayID structure's generic parser works correctly. The Number of Payload bytes for this block shall include the three bytes needed for a unique Manufacturer/Vendor ID in addition to the size of the vendor-specific data.

Appendix B shows the VESA Vendor-specific data block that exposes the embedded DisplayPort-specific data structure.

Table 4-48: Vendor-specific Data Block Fields

Offset	Bit #	Description/Format	Defined in	Tag
00h	7:0	**Vendor-specific Data Block** Value is 7Eh.		✔
01h	**Block Revision and Other Data**			
	2:0	**Block Revision** Revision ranges from 0 through 7. 000b = Revision 0 (default).		
	7:3	**RESERVED** Cleared to all 0s.		
02h	7:0	**Number of Payload Bytes in Block** Number of payload bytes within the data block ranges from 3 through 248. Value ranges from 03h through F8h.		
05h – 03h	23:0	**Manufacturer/Vendor ID** Descriptor.	Section 4.9.1	
06h – …		**Vendor-specific Data**		

4.9.1 Manufacturer/Vendor ID Field

The Manufacturer/Vendor ID field listed in Table 4-48 is a **required** element of the Vendor-specific data block. The Manufacturer/Vendor ID field contains the display manufacturer's 3-byte IEEE OUI code, which identifies the display's manufacturer or vendor. Table 4-49 lists the field format.

The IEEE issues IEEE OUIs. Contact the IEEE at http://standards.ieee.org/develop/regauth/oui/ for registration and/or further details.

Table 4-49: Manufacturer/Vendor ID Field Format

Offset	Byte #	Bit #	Description/Format
03h	0	7:0	**IEEE OUI First Byte** Byte code. Byte length ranges from 0 through 255.
04h	1	7:0	**IEEE OUI Second Byte** Byte code. Byte length ranges from 0 through 255.
05h	2	7:0	**IEEE OUI Third Byte** Byte code. Byte length ranges from 0 through 255.

4.10 CTA DisplayID Data Block

Identification code 81h shall be used for the **optional** DisplayID block that describes how to encapsulate CTA data blocks onto a DisplayID data block framework.

Table 4-50 describes how to map CTA blocks with an appropriate block tag code, associated block length, and the block payload on the CTA DisplayID data block. This shall allow leverage of *CTA-861*-defined speaker allocation, audio descriptor, colorimetry data block, and others, as-is, without redefining them in DisplayID.

Table 4-50: CTA Data Block Encapsulation

Offset	Bit #	Description/Format	Flag	Tag
00h	7:0	**Data Block Identification** Value is 81h.		✔
01h	**Block Revision and Other Data**			
	2:0	**Block Revision** Revision ranges from 0 through 7. 000b = Revision 0 (default).		
	7:3	**RESERVED** Block-specific. Cleared to all 0s.	✔	✔
02h	7:0	**Number of Payload Bytes** Number of payload bytes ranges from 0 through 248. Value ranges from 00h through F8h.		
03h	**CTA Block 1 Tag Code and Block 1 Length**			
	4:0	**Block Length** Associated data block length (e.g., L1). Value ranges from 00h through 1Fh.		
	7:5	**Tag Code** Based on the latest *CTA-861*. Value ranges from 000b through 111b.		
04h	7:0	**CTA Block 1 Descriptor 1** Descriptor.		
05h	7:0	**CTA Block 1 Descriptor 2** Descriptor.		
…		**…**		
1 + L1	7:0	**CTA Block 1 Descriptor L1** Descriptor.		

Table 4-50: CTA Data Block Encapsulation (Continued)

Offset	Bit #	Description/Format	Flag	Tag
2 + L1	**CTA Block 2 Tag Code and Block 2 Length**			
	4:0	**Block Length** Associated block length (e.g., L2). Value ranges from 00h through 1Fh.		
	7:5	**Tag Code** Based on the latest *CTA-861*. Value ranges from 000b through 111b.		
3 + L1	7:0	**CTA Block 2 Descriptor 1** Descriptor.		
4 + L1	7:0	**CTA Block 2 Descriptor 2** Descriptor.		
…		**…**		
2 + L1 + L2	7:0	**CTA Block 2 Descriptor L2** Descriptor.		

A Fixed-length DisplayID Section Example

Table A-1: Fixed-length DisplayID Section Example

Offset	Raw Value	Field Name	Interpretation
0h	0x20	**DisplayID Structure Version**	Version 2, Revision 0.
1h	0x86	**Bytes in Section**	Section length = 134 bytes.
2h	0x04	**Display Product Primary Use Case**	Desktop Productivity Display.
3h	0x00	**Extension Count**	Extension count = 0.
4h	0x20	**Data Block Name**	Product Identification data block.
5h	0x00	**Block Revision and Other Data**	Bits 7:3 = RESERVED = 00000b. Bits 2:0 = Block Revision 0 = 000b.
6h	0x18	**Number of Payload Bytes in Block**	Payload length = 24 bytes.
7h	0x12	**Manufacturer/Vendor ID**	IEEE-assigned ID = 12-34-56.
8h	0x34		
9h	0x56		
Ah	0x34	**Product ID Code**	Vendor-assigned Product ID code = 0x1234.
Bh	0x12		
Ch	0x78	**Serial Number**	Vendor-assigned Serial Number = 0x12345678.
Dh	0x56		
Eh	0x34		
Fh	0x12		
10h	0x01	**Week of Manufacture/Model Tag**	Week of manufacture = 1.
11h	0x11	**Year of Manufacture/Model Year**	Year of manufacture = 2017.
12h	0x0C	**Size of Product Name String**	String length = 12 bytes.
13h	0x44	**Product Name String**	Vendor-assigned Product Name String = “Display Name”.
14h	0x69		
15h	0x73		
16h	0x70		
17h	0x6C		
18h	0x61		
19h	0x79		
1Ah	0x20		
1Bh	0x4E		
1Ch	0x61		
1Dh	0x6D		
1Eh	0x65		

Table A-1: Fixed-length DisplayID Section Example (Continued)

Offset	Raw Value	Field Name	Interpretation
1Fh	0x21	**Data Block Name**	Display Parameters data block.
20h	0x00	**Block Revision and Other Data**	Bit 7 = Image Size Multiplier = 0 (0.1-mm precision). Bits 6:3 = RESERVED = 0000b. Bits 2:0 = Block Revision 0 = 000b.
21h	0x1D	**Number of Payload Bytes in Block**	Data block is composed of 29 payload bytes.
22h 23h	0x00 0x10	**Horizontal Image Size**	409.6mm.
24h 25h	0x70 0x08	**Vertical Image Size**	216.0mm.
26h 27h	0x00 0x10	**Horizontal Pixel Count**	4096 horizontal pixels.
28h 29h	0x70 0x08	**Vertical Pixel Count**	2160 vertical pixels.
2Ah	0x00	**Feature Support Flags**	Bit 7 = Audio Speaker Information = 0 (Audio speakers are integrated into the display). Bit 6 = Color Information = 0 (specified in terms of *CIE 1931* (*x*, *y*) coordinates). Bit 5 = RESERVED = 0. Bits 4:3 = Luminance Information = 00b (minimum guaranteed value). Bits 2:0 = Scan Orientation = 000b (left to right, top to bottom).
2Bh 2Ch 2Dh	0xCC 0xEA 0x51	**Native Color Chromaticity (Red)**	Red X = 0.675, Y = 0.320.
2Eh 2Fh 30h	0x45 0x54 0xA6	**Native Color Chromaticity (Green)**	Green X = 0.267, Y = 0.650.
31h 32h 33h	0x66 0x42 0x0D	**Native Color Chromaticity (Blue)**	Blue X = 0.150, Y = 0.052.
34h 35h 36h	0xFD 0x34 0x54	**Native Color Chromaticity (White)**	White X = 0.312, Y = 0.329.
37h 38h	0x40 0x5E	**Native Maximum Luminance (Full Coverage)**	400cd/m^2.

Table A-1: Fixed-length DisplayID Section Example (Continued)

Offset	Raw Value	Field Name	Interpretation
39h 3Ah	0x40 0x5E	**Native Maximum Luminance (10% Rectangular Coverage)**	400cd/m^2.
3Bh 3Ch	0x00 0x38	**Native Minimum Luminance**	0.5cd/m^2.
3Dh	0x13	**Native Color Depth and Display Device Technology**	Bit 7 = RESERVED = 0. Bits 6:4 = Display Device Technology = 001b (Active Matrix LCD). Bit 3 = RESERVED = 0. Bits 2:0 = Native Color Depth = 011b (10bpc).
3Eh	0x78	**Native Gamma EOTF**	Gamma = 2.2.
3Fh	0x26	**Data Block Name**	Display Interface Features data block.
40h	0x00	**Block Revision and Other Data**	Bits 7:3 = RESERVED = 00000b. Bits 2:0 = Block Revision 0 = 000b.
41h	0x09	**Number of Payload Bytes in Block**	Data block is composed of nine + *N* payload bytes.
42h	0x06	**Supported Interface Color Depth for RGB Encoding**	Supports 10 and 8 bits per primary color.
43h	0x00	**Supported Interface Color Depth for YCbCr 4:4:4 Encoding**	Not supported.
44h	0x00	**Supported Interface Color Depth for YCbCr 4:2:2 Encoding**	Not supported.
45h	0x00	**Supported Interface Color Depth for YCbCr 4:2:0 Encoding**	Not supported.
46h	0x00	**Minimum Pixel Rate at Which YCbCr 4:2:0 Encoding Is Supported**	Minimum pixel rate is 0MP/s.
47h	0x60	**Supported Interface Audio Capability and Feature Flags**	Bit 7 = 32-kHz Sample Rate Supported = 0 (not supported). Bit 6 = 44.1-kHz Sample Rate Supported = 1 (supported). Bit 5 = 48-kHz Sample Rate Supported = 1 (supported). Bits 4:0 = RESERVED = 00000b.
48h	0x01	**Supported Interface Color Space and EOTF Standard Combination 1**	Bit 7 = RESERVED = 0. Bit 6 = *ITU-R BT.2020* / *SMPTE ST 2084* = 0 (not supported). Bit 5 = *ITU-R BT.2020* = 0 (not supported). Bit 4 = DCI-P3 (*SMPTE RP 431-2*) = 0 (not supported). Bit 3 = *Adobe RGB* = 0 (not supported). Bit 2 = *ITU-R BT.709* / *ITU-R BT.1886* = 0 (not supported). Bit 1 = *ITU-R BT.601* = 0 (not supported). Bit 0 = sRGB (*IEC 61966-2-1*) = 1 (supported).

Table A-1: Fixed-length DisplayID Section Example (Continued)

Offset	Raw Value	Field Name	Interpretation
49h	0x00	**Supported Interface Color Space and EOTF Standard Combination 2**	Bits 7:0 = RESERVED = 00000000b.
4Ah	0x00	**Number of Additional Supported Interface Color Space and EOTF (*N*)**	Bits 7:3 = RESERVED = 00000b. Bits 2:0 = 000b = Zero additional supported combinations.
4Bh	0x22	**Data Block Name**	Type VII Detailed Timing data block.
4Ch	0x00	**Block Revision and Other Data**	Bits 7:3 = RESERVED = 00000b. Bits 2:0 = Block Revision 0 = 000b.
4Dh	0x3C	**Number of Payload Bytes in Block**	Data block is composed of 60 payload bytes.
4Eh	0xC8	**Timing 1**	Pixel clock rate = 556.744MP/s.
4Fh	0x7E		
50h	0x08		Bit 7 = 1 (Preferred Detailed timing).
51h	0x88		Bits 6:5 = 00b (Timing always mono – no stereo).
52h	0xFF		Bit 4 = 0 (Progressive scan frame).
53h	0x0F		Bits 3:0 = 1000b (Aspect Ratio calculated from pixel dimensions).
54h	0x4F		
55h	0x00		
56h	0x07		Horizontal Active Image Pixels = 4096.
57h	0x80		Horizontal Blank Pixels = 80.
58h	0x1F		Horizontal Sync Offset (front porch) = 8.
59h	0x00		Horizontal Sync Polarity = 1 (positive).
5Ah	0x6F		Horizontal Sync Width = 32.
5Bh	0x08		
5Ch	0x3D		Vertical Active Image Lines = 2160.
5Dh	0x00		Vertical Blank Lines = 62.
5Eh	0x00		Vertical Sync Offset (front porch) = 1.
5Fh	0x00		Vertical Sync Polarity = 0 (negative).
60h	0x07		Vertical Sync Width = 8.
61h	0x00		

Table A-1: Fixed-length DisplayID Section Example (Continued)

Offset	Raw Value	Field Name	Interpretation
62h	0x5D	**Timing 2**	Pixel clock rate = 234.590MP/s.
63h	0x94		
64h	0x03		Bit 7 = 0 (Not a preferred detailed timing).
65h	0x08		Bits 6:5 = 00b (Timing always mono – no stereo).
66h	0xFF		Bit 4 = 0 (Progressive scan frame).
67h	0x09		Bits 3:0 = 1000b (Aspect Ratio calculated from pixel dimensions).
68h	0x4F		
69h	0x00		
6Ah	0x07		Horizontal Active Image Pixels = 2560.
6Bh	0x80		Horizontal Blank Pixels = 80.
6Ch	0x1F		Horizontal Sync Offset (front porch) = 8.
6Dh	0x00		Horizontal Sync Polarity = 1 (positive).
6Eh	0x9F		Horizontal Sync Width = 32.
6Fh	0x05		
70h	0x28		Vertical Active Image Lines = 1440.
71h	0x00		Vertical Blank Lines = 41.
72h	0x00		Vertical Sync Offset (front porch) = 1.
73h	0x00		Vertical Sync Polarity = 0 (negative).
74h	0x07		Vertical Sync Width = 8.
75h	0x00		

Table A-1: Fixed-length DisplayID Section Example (Continued)

Offset	Raw Value	Field Name	Interpretation
76h	0xC7	**Timing 3**	Pixel clock rate = 133.320MP/s.
77h	0x08		
78h	0x02		Bit 7 = 0 (Not a preferred detailed timing).
79h	0x08		Bits 6:5 = 00b (Timing always mono – no stereo).
7Ah	0x7F		Bit 4 = 0 (Progressive scan frame).
7Bh	0x07		Bits 3:0 = 1000b (Aspect Ratio calculated from pixel dimensions).
7Ch	0x4F		
7Dh	0x00		
7Eh	0x07		Horizontal Active Image Pixels = 1920.
7Fh	0x80		Horizontal Blank Pixels = 80.
80h	0x1F		Horizontal Sync Offset (front porch) = 8.
81h	0x00		Horizontal Sync Polarity = 1 (positive).
82h	0x37		Horizontal Sync Width = 32.
83h	0x04		
84h	0x1E		Vertical Active Image Lines = 1080.
85h	0x00		Vertical Blank Lines = 31.
86h	0x00		Vertical Sync Offset (front porch) = 1.
87h	0x00		Vertical Sync Polarity = 0 (negative).
88h	0x07		Vertical Sync Width = 8.
89h	0x00		
8Ah		**Checksum**	

B VESA Organization Vendor-specific Data Block

Table B-1: VESA Organization Vendor-specific Data Block

Offset	Bit #	Description/Format	Tag
00h	7:0	**Vendor-specific Data Block** Value is 7Eh.	✔
01h	**Block Revision and Other Data**		
	2:0	**Block Revision** Revision ranges from 0 through 7. 000b = Revision 0 (default).	
	7:3	**RESERVED** Cleared to all 0s.	
02h	7:0	**IEEE OUI First Byte** Byte length ranges from 0 through 255. 3Ah = First VESA IEEE OUI byte.	
03h	7:0	**IEEE OUI Second Byte** Byte length ranges from 0 through 255. 02h = First VESA IEEE OUI byte.	
04h	7:0	**IEEE OUI Third Byte** Byte length ranges from 0 through 255. 92h = First VESA IEEE OUI byte.	
05h	**VESA Data Structure Type**		
	2:0	**Data Structure Type** 000b = Embedded DisplayPort (eDP). 001b = External DisplayPort. All other values are RESERVED for future use.	
	6:3	**RESERVED** Cleared to all 0s.	
	7	**Default Color Space and EOTF Handling** 0 = MISC signaling of “RGB unspecified color space” by a Source device is interpreted by a Sink device as sRGB color space and EOTF. 1 = MISC signaling of “RGB unspecified color space” by a Source device is interpreted by a Sink device as native color space and EOTF as specified in the Display Parameters data block.	

Table B-1: VESA Organization Vendor-specific Data Block (Continued)

Offset	Bit #	Description/Format	Tag
06h	3:0	**Number of Pixels in Horizontal Pixel Count Overlapping an Adjacent Panel Segment** Number of pixels ranges from 0 through 8. Value range of 9 through 15 is RESERVED.	
	4	**RESERVED** Cleared to 0.	
	6:5	**Multi-SST Operation** 00b = Not supported (Conventional Single-Stream Transport). 01b = Two streams (number of links shall be two or four). 10b = Four streams (number of links shall be four). 11b = RESERVED for future use.	
	7	**RESERVED** Cleared to 0.	

C Scan Orientation Clarification

This appendix clarifies use of the Scan Orientation bits of the Feature Support Flags field (offset 0Bh, bits 2:0) in the Display Parameters data block.

Figure C-1 illustrates the various scan orientation settings for an example 1920x1080 landscape panel at various viewing configurations. In this example, the native timing exposed in the DisplayID structure is a normal 1920x1080 timing that represents a left-to-right scan panel.

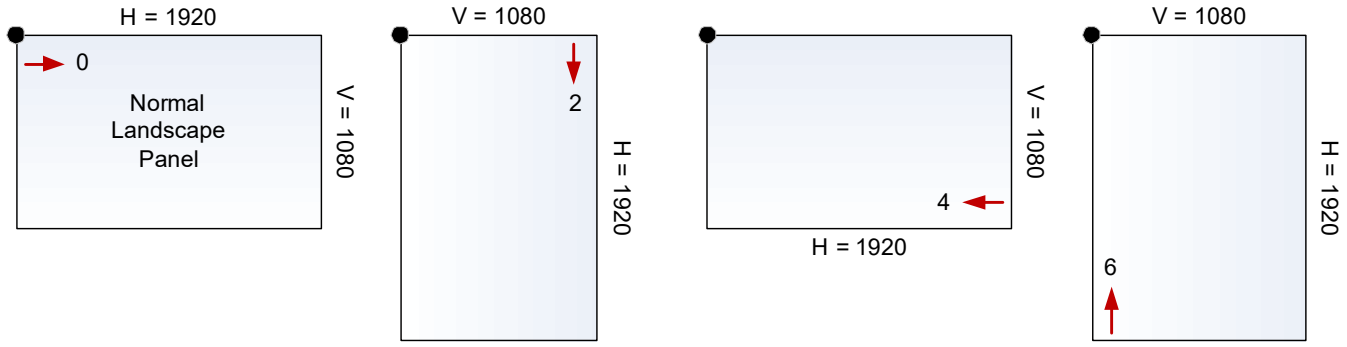


Figure C-1: Scan Orientation Settings for Example 1920x1080 Landscape Panel at Various Viewing Configurations

Figure C-2 illustrates the various scan orientation settings for an example 1080x1920 portrait panel at various viewing configurations. In this example, the native timing exposed in the DisplayID structure is a normal 1080x1920 timing that represents a left-to-right scan panel.

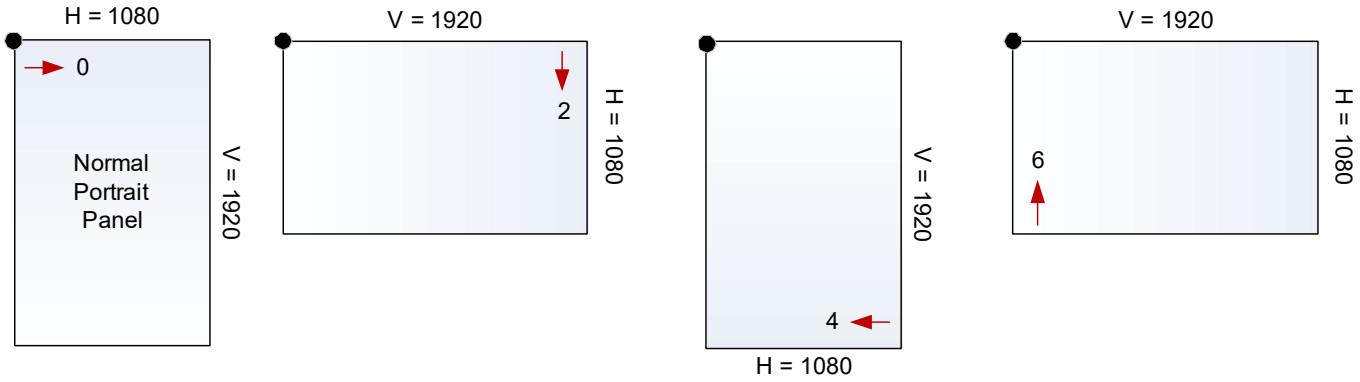


Figure C-2: Scan Orientation Settings for Example 1080x1920 Portrait Panel at Various Viewing Configurations

Note: *From the viewer's perspective, these examples represent only two display orientations. A landscape display can be implemented as shown in Figure C-1, Orientations 0 and 4, –or– as shown in Figure C-2, Orientations 2 and 6. A portrait display can be implemented as shown in Figure C-1, Orientations 2 and 6, –or– as shown in Figure C-2, Orientations 0 and 4.*

Figure C-3 and Figure C-4 illustrate all possible scan orientations for a landscape display device. Figure C-3 shows the orientations used with landscape timing (e.g., 1920x1080). Orientations 0 and 4 are "normal" left-to-right scan panels, while Orientations 1 and 5 are "flipped" right-to-left scan panels. Figure C-4 shows the orientations used with portrait timing (e.g., 1080x1920). Orientations 2 and 6 are "normal" left-to-right scan panels, while Orientations 3 and 7 are "flipped" right-to-left scan panels.

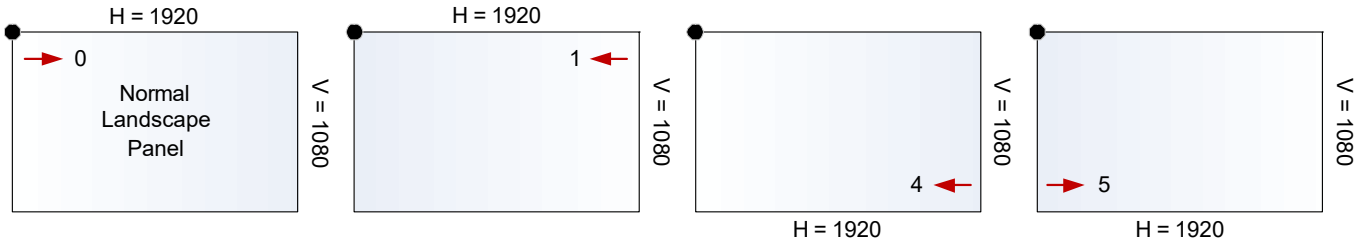


Figure C-3: Scan Orientation Settings Used with an Example Landscape 1920x1080 Timing to Implement Landscape Display Devices

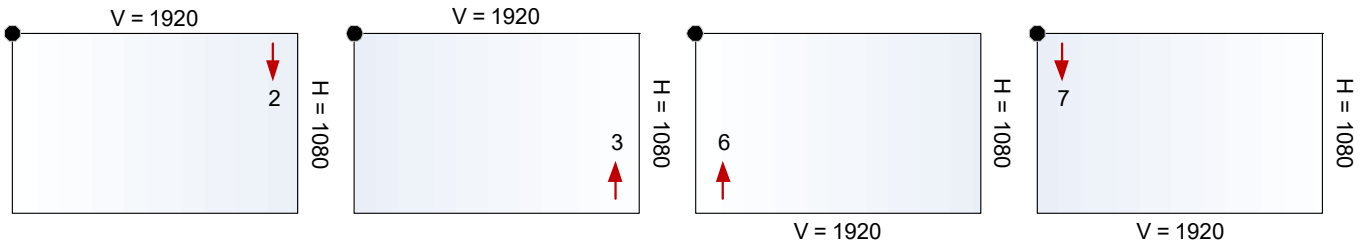


Figure C-4: Scan Orientation Settings Used with an Example Portrait 1080x1920 Timing to Implement Landscape Display Devices