**ISO/IEC 14496-12:2025/AMD 2:202x(E)**

ISO/IEC JTC1/SC 29

Secretariat: JISC

**Information technology — Coding of audio-visual objects — Part 12: ISO base media file format — Amendment 2: Carriage of depth and alpha**

CD

**Warning for WDs and CDs**

This document is not an ISO International Standard. It is distributed for review and comment. It is subject to change without notice and may not be referred to as an International Standard.

Recipients of this draft are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation.

*A model document of an International Standard (the Model International Standard) is available at:*[*https://www.iso.org/drafting-standards.html*](https://www.iso.org/drafting-standards.html)

© ISO 20XX

All rights reserved. Unless otherwise specified, or required in the context of its implementation, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO at the address below or ISO’s member body in the country of the requester.

ISO copyright office

CP 401 • Ch. de Blandonnet 8

CH-1214 Vernier, Geneva

Phone: +41 22 749 01 11

Email: copyright@iso.org

Website: www.iso.org

Published in Switzerland

Contents

[1 Clause 3.1, Terms and definitions 4](#_Toc204124389)

[2 Clause 8, Box structures 5](#_Toc204124390)

[2.1 Clause 8.7 Track data layout structures 5](#_Toc204124391)

[3 Clause 12, Media-specific definitions 7](#_Toc204124392)

[3.1 Clause 12.1.5 Colour information 7](#_Toc204124393)

[3.2 Clause 12.11 8](#_Toc204124394)

[12.11 Auxiliary maps 8](#_Toc204124395)

[12.11.1 General 8](#_Toc204124396)

[12.11.2 Media handler 8](#_Toc204124397)

[12.11.2.1 Depth media handler 8](#_Toc204124398)

[12.11.2.2 Alpha media handler 8](#_Toc204124399)

[12.11.3 Media header 8](#_Toc204124400)

[12.11.3.1 Depth media header 8](#_Toc204124401)

[12.11.3.1 Alpha media header 8](#_Toc204124402)

[12.11.4 Sample entry 8](#_Toc204124403)

[12.11.4.1 Depth media sample entry 8](#_Toc204124404)

[12.11.4.2 Alpha media sample entry 9](#_Toc204124405)

[12.11.5 Map Information box 9](#_Toc204124406)

[12.11.5.1 Depth information box 9](#_Toc204124407)

[12.11.5.1.1 Definition 9](#_Toc204124408)

[12.11.5.1.2 Syntax 9](#_Toc204124409)

[12.11.5.1.3 Semantics 9](#_Toc204124410)

[12.11.5.2 Alpha information box 10](#_Toc204124411)

[12.11.5.2.1 Definition 10](#_Toc204124412)

[12.11.6 Auxiliary Information box 11](#_Toc204124413)

[12.11.6.1 Invalid depth band box 11](#_Toc204124414)

[12.11.6.1.1 Definition 11](#_Toc204124415)

[12.11.6.1.2 Syntax 11](#_Toc204124416)

[12.11.6.1.3 Semantics 11](#_Toc204124417)

[4 Annex B Guidance on deriving from this document 11](#_Toc204124418)

[5 Annex L Depreciation of boxes and identifiers 12](#_Toc204124419)

[Annex L (normative) Depreciation of boxes and identifiers 13](#_Toc204124420)

[L.1 Overview 13](#_Toc204124421)

[L.2 Deprecated Boxes 13](#_Toc204124422)

[6 Annex M Handling of depth and alpha maps 13](#_Toc204124423)

[Annex M (informative) Handling of depth and alpha maps 14](#_Toc204124424)

[M.1 General 14](#_Toc204124425)

[M.2 Handling of parameters in SEI messages and container level 14](#_Toc204124426)

[M.2 Lossless conversion between SEI-based and IEEE 754 32-bit floating-point formats 14](#_Toc204124427)

[M.2.1 SEI-based floating point format 14](#_Toc204124428)

[M.2.2 IEEE 754 32-bit floating-point format 15](#_Toc204124429)

[M.2.3 Lossless conversion rules 15](#_Toc204124430)

[M.3 Interpretation of depth decoded sample values 15](#_Toc204124431)

[M.4 Recommended implementation of alpha blending composition 16](#_Toc204124432)

[7 Bibliography 16](#_Toc204124433)

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of ISO documents should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see [www.iso.org/directives](https://www.iso.org/directives-and-policies.html)).

ISO draws attention to the possibility that the implementation of this document may involve the use of (a) patent(s). ISO takes no position concerning the evidence, validity or applicability of any claimed patent rights in respect thereof. As of the date of publication of this document, ISO *[had/had not]* received notice of (a) patent(s) which may be required to implement this document. However, implementers are cautioned that this may not represent the latest information, which may be obtained from the patent database available at [www.iso.org/patents](http://www.iso.org/patents). ISO shall not be held responsible for identifying any or all such patent rights.

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT), see [www.iso.org/iso/foreword.html](https://www.iso.org/foreword-supplementary-information.html).

This document was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, SC 29, *Coding of audio, picture, multimedia and hypermedia information*.

A list of all parts in the ISO/IEC 14496 series can be found on the ISO website.

Any feedback or questions on this document should be directed to the user’s national standards body. A complete listing of these bodies can be found at [www.iso.org/members.html](https://www.iso.org/members.html).

**Information technology — Coding of audio-visual objects — Part 12: ISO base media file format — Amendment 2: Carriage of depth and alpha**

# Clause 3.1, Terms and definitions

Add the following definitions to clause 3.1:

**3.70   
alpha data**

a collection of alpha values (3.72)

**3.71  
alpha elementary stream**

an elementary stream (3.79) containing access units for coded alpha data (3.70)

**3.72  
alpha image**

an image representation of alpha data (3.70), with each sample associated with an alpha value (3.73)

**3.73  
alpha value**

transparency value determined relative to a minimum and a maximum value, with the minimum value indicating full transparency and the maximum value indicating full opacity

**3.74  
depth data**

a collection of depth values (3.78)

**3.75  
depth data range**

a range of depth values (3.78) defined by a minimum and a maximum depth value (3.78)

**3.76  
depth elementary stream**

an elementary stream (3.79) containing access units for coded depth data (3.74)

**3.77  
depth image**

an image representation of depth data (3.73), with each sample associated with a depth value (3.76)

**3.78  
depth value**

distance of a given point in 3D space relative to an origin, as measured along a certain direction

**3.79  
elementary stream**

a consecutive flow of mono-media data from a single source entity to a single destination entity on the compression layer

**3.80  
image representation**

a 2D array of samples, with each sample associated with an intensity value

# Clause 8, Box structures

## Clause 8.7 Track data layout structures

Add new clause 8.7.10 in track data layout structures (section 8.7)

**8.7.10 Compact Direct Sample References**

**8.7.10.1 Compact Direct Sample References Box**

**8.7.10.1.1 Definition**

Box Type: 'cdrf'  
Container: SampleTableBox or TrackFragmentBox  
Mandatory: No  
Quantity: Zero or one (per container)

CompactDirectSampleReferencesBox provides explicit coding dependencies of samples towards other samples. It associates with each sample of a track or track fragment:

* a sample identifier (sample ID) coded as an absolute sample ID or as a difference with the sample ID of a previous sample;
* a dependency list of sample references coded as a difference with the sample ID or as the identifier of another sample.

The listed dependencies should only contain the direct dependencies, i.e. if sample A depends on sample B which in turn depends on sample C, only sample B should be listed as a dependency to sample A.

CompactDirectSampleReferencesBox is structured as a list of entries for reconstructing the list of samples and their sample references, each entry defining either the sample identifier and the dependency list associated with a sample or a reference to a pattern of previously coded sample IDs and sample references in the reconstructed list of samples and their sample references.

The sample ID may be any identifier. It does not need to be unique, and the IDs used by the list of sample references refer to the previous sample defined with the given ID in the CompactDirectSampleReferencesBox. A sample ID used in a dependency list but not present in the track or past track fragments indicate a broken dependency, i.e. that the sample cannot be decoded.

NOTE For video track, the sample ID could be the Picture Order Count (POC) of a sample. Broken dependencies typically happen when tuning a stream on a stream access point sample of type 3 (SAP 3 sample), e.g. in an open-GOP case.

The no\_diff\_mode variable indicates if the dependencies and sample IDs are coded as direct values or as differential values.

The reconstruction process of the list of samples and their sample references shall produce the same result as the following model:

* initializing an empty list flat\_refs
* for each entry in the box,
  + if is\_ref is 0, appending to the flat\_refs list an entry containing {nb\_refs, is\_abs, sample\_ID\_code, ref\_IDs}
  + otherwise (is\_ref is 1), for each K ranging from 0 to num\_samples – 1 of the entry,
    - If no\_diff\_mode is 0, appending to flat\_refs the entry flat\_refs[offset + K%pattern\_length]
    - Otherwise, appending to flat\_refs a copy of flat\_refs[offset + K%pattern\_length] with sample\_ID\_code equal to sample\_IDs[K]
* validating that the number of entries in flat\_refs is the same as the number of samples in the track or track fragment
* for each sample J in the track or track run:
  + assigning sampleID as follows:
    - if flat\_refs[J].is\_abs is 1, setting sampleID to flat\_refs[J].sample\_ID\_code
    - otherwise, setting sampleID to flat\_refs[J].sample\_ID\_code + sample[J-1].sampleID
  + assigning sample reference IDs for each K in range [0, flat\_refs[J].nb\_refs] by:
    - If no\_diff\_mode is 0, removing from sample[J].sampleID the value flat\_refs[J].ref\_IDs[K], i.e. referenceSampleID = sample[J].sampleID - flat\_refs[J].ref\_IDs[K],
    - Otherwise, using the value flat\_refs[J].ref\_IDs[K]

#### 8.7.10.1.2 Syntax

aligned(8) class CompactDirectSampleReferencesBox extends Box('cdrf')

{

unsigned int(8) flags;

if (flags & 2) bits = 32;

else if (flags & 1) bits = 16;

else bits = 8

if (flags & 8) entry\_bits=32;

else if (flags & 4) entry\_bits=16;

else entry\_bits=8

unsigned int(entry\_bits) nb\_entries;

for (i = 0;i < nb\_entries; i++) {

bit(1) is\_ref;

if (is\_ref) {

unsigned int(bits-1) offset;

unsigned int(bits) pattern\_length;

unsigned int(bits) num\_samples;

if (no\_diff\_mode) {

signed int(bits) sample\_IDs[num\_samples];

} else {

unsigned int(bits-1) nb\_refs;

if (no\_diff\_mode) {

signed int(bits) sample\_ID\_code;

} else {

bit(1) is\_abs;

signed int(bits-1) sample\_ID\_code;  
 }

signed int(bits) ref\_diff\_IDs[nb\_refs];

}

}

}

#### 8.7.10.1.3 Semantics

nb\_entries indicates the number of entries in the loop for reconstructing the list of references.

no\_diff\_mode indicates, when set to 1, that the sample references in ref\_IDs[] and the sample IDs are coded as direct values, and when set to 0, that the sample references in ref\_IDs[] are coded as differential values and sample IDs may be coded as differential or direct values depending on is\_abs.

is\_ref indicates, if set to 1, that the entry is a reference to a pattern of previously coded sample references in the reconstructed list of references. Otherwise, if set to 0, an explicit list of sample references follows. The first sample in a track or track fragment shall have an associated is\_ref value of 0.

offset indicates the start of the pattern in the reconstructed list of references, value 0 designating the list of sample references of the first sample in the track or track fragment.

pattern\_length indicates the number of samples in the pattern starting from offset

num\_samples indicates the number of samples described. If this value is greater than pattern\_length, the pattern is looped over until all samples indicated by the number of samples num\_samples are described. It is not necessarily the case that num\_samples is a multiple of pattern\_length; the last repeated pattern may be truncated.

nb\_refs indicates the number of direct sample references for this sample. If 0, the sample has no direct references (i.e. the sample is a sync sample).

NOTE The number of direct sample references nb\_refs can represent a number of sample references possibly used for processing the sample, and does not necessarily represent a number of sample references actually used in a codec reference list for processing the sample.

is\_abs indicates, if set to 1, that sample\_ID\_code is the value of the sample identifier. If set to 0, it indicates that sample\_ID\_code is the difference between the sample identifier and the preceding sample identifier in the reconstructed list of references. The first sample in a track or track fragment shall have an associated is\_abs value of 1. When no\_diff\_mode is 1, the is\_abs field is not coded and takes the value 1.

sample\_ID\_code indicates the value of the sample identifier. If no\_diff\_mode is 1, it indicates the direct value of the sample identifier. If no\_diff\_mode is 0, it indicates the difference or direct value of the sample identifier coded as specified by is\_abs flag.

ref\_IDs is an array that indicates the sample identifiers of the direct sample references if any. If no\_diff\_mode is 0, the identifier is coded as a difference between the identifier of the sample being described by this entry and the identifier of the reference~~d~~ sample, i.e. *sampleID - referenceSampleID*. Otherwise (no\_diff\_mode is 1), the identifier is coded as the value of the sample identifier of the reference sample.

sample\_IDs is an array that indicate the list of values of the sample identifiers coded in a pattern.

When present in a SampleTableBox (respectively a TrackFragmentBox), the number of samples described by CompactDirectSampleReferencesBoxshall be equal to the number of samples present in the track (respectively in the track fragment).

# Clause 12, Media-specific definitions

## Clause 12.1.5 Colour information

Replace clause 12.1.5.2 with:

class ColourInformationBox extends Box('colr')

{

unsigned int(32) colour\_type;

if (colour\_type == 'nclx') /\* on-screen colours \*/

{

unsigned int(16) colour\_primaries;

unsigned int(16) transfer\_characteristics;

unsigned int(16) matrix\_coefficients;

unsigned int(1) full\_range\_flag;

unsigned int(7) reserved = 0;

}

else if (colour\_type == 'rICC')

{

ICC\_profile; // restricted ICC profile

}

else if (colour\_type == 'prof')

{

ICC\_profile; // unrestricted ICC profile

}

else if (colour\_type == 'bICC')

{

ICC\_profile; // Brotli compressed unrestricted ICC profile

}

}

## Clause 12.11

Add the following new clause 12.11:

## 12.11 Auxiliary maps

### 12.11.1 General

Auxiliary maps are image representations of non-visual elementary streams. Currently ISOBMFF supports carriage of depth data and alpha data as auxiliary maps.

### 12.11.2 Media handler

### 12.11.2.1 Depth media handler

Depth video tracks use the 'depv' handler type in the HandlerBox of the MediaBox, as defined in 8.4.3.

A depth video track is coded the same as a video track, but uses this different handler type, and is not intended to be visually displayed. Depth video tracks may be linked to a video track using a reference of type 'cdsc'.

### 12.11.2.2 Alpha media handler

Alpha video tracks use the 'alpv' handler type in the HandlerBox of the MediaBox, as defined in 8.4.3.

An alpha video track is coded the same as a video track, but uses this different handler type, and is not intended to be visually displayed. Alpha video tracks are linked to a video track using a reference of type 'cdsc'.

### 12.11.3 Media header

### 12.11.3.1 Depth media header

Depth video tracks use the NullMediaHeaderBox in the MediaInformationBox as defined in 8.4.5.

### 12.11.3.1 Alpha media header

Alpha video tracks use the NullMediaHeaderBox in the MediaInformationBox as defined in 8.4.5.

### 12.11.4 Sample entry

### 12.11.4.1 Depth media sample entry

Depth video tracks use VisualSampleEntry.

### 12.11.4.2 Alpha media sample entry

Alpha video tracks use VisualSampleEntry.

### 12.11.5 Map Information box

### 12.11.5.1 Depth information box

### 12.11.5.1.1 Definition

Box Type: 'depx'   
Container: VisualSampleEntry  
Mandatory: No  
Quantity: One

The DepthInformationBox may be used to provide information independent of the coding, to interpret the depth data.

When the DepthInformationBox is present, the decoded sample values are uniformly quantized into the range [0, maxVal], with maxVal = 2^bit\_depth – 1. Where bit\_depth is the number of bits used to represent a data point from the original component(s) of an image (e.g. in the case of monochrome image, the bit depth of the luma samples).

### 12.11.5.1.2 Syntax

class DepthInformationBox extends FullBox ('depx', version = 0, flags){  
 if (!is\_unit\_interval\_mode){  
 float(32) near\_plane;  
 float(32) far\_plane; unsigned int(6) units;  
 unsigned int(2) reserved;  
 }

}

### 12.11.5.1.3 Semantics

version is an integer that specifies the version of this box.

flags is a 24-bit integer with flags; the following values are defined:

is\_unit\_interval\_mode: Flag mask is 0x000001. The value 1 indicates that the near\_plane value is equal to 0, the far\_plane value is equal to 1, and the units value is equal to 0.

depth\_mapping\_type: Flag mask is 0x000006. The value indicates the type of mapping between decoded sample values and depth values, as specified in Table 15.

**Table 15 – Definition of depth\_mapping\_type.**

|  |  |
| --- | --- |
| depth\_mapping\_type | Interpretation |
| 0 | When depth\_mapping\_type is equal to 0, a linear relationship is defined between the decoded sample values and the depth values, where the decoded sample value equal to 0 corresponds to , and the decoded sample value equal to maxVal corresponds to . |
| 1 | When depth\_mapping\_type is equal to 1, an inverse relationship is defined between the decoded sample values and the depth values, where the decoded sample value equal to 0 corresponds to the inverse of , and the decoded sample value equal to maxVal corresponds to the inverse of . |
| 2 | When depth\_mapping\_type is equal to 2, a linear relationship is defined between the decoded sample values and the depth values, where the decoded sample value equal to 0 corresponds to , and the decoded sample value equal to maxVal corresponds to . |
| 3 | When depth\_mapping\_type is equal to 3, an inverse relationship is defined between the decoded sample values and the depth values, where the decoded sample value equal to 0 corresponds to the inverse of , and the decoded sample value equal to maxVal corresponds to the inverse of . |

The variable of Table 15 is specified by Equation 1:

(1)

The variable of Table 15 is specified by Equation 2:

(2)

near\_plane and far\_plane specify the nearest and the farthest depth values, respectively. The near\_plane value can be less than the far\_plane value or it can be greater than the far\_plane value. The near\_plane value and the far\_plane value shall have the same sign that can be either positive or negative.

units specifies the units of the depth values, as follows:

0: unspecified

1: the values are in meters

2: the values are in millimetres

3-63: reserved.

### 12.11.5.2 Alpha information box

### 12.11.5.2.1 Definition

Box Type: 'alpi'   
Container: VisualSampleEntry  
Mandatory: No  
Quantity: One

The AlphaInformationBox may be used to provide information independent of the coding, to interpret the alpha data.

The AlphaInformationBox is optional and if it is absent it is assumed that a value of 0 indicates full transparency and a value equal to 2^bit\_depth - 1 indicates full opacity. Where bit\_depth is the number of bits used to represent a data point from the original component(s) of an image (e.g. in the case of monochrome image, the bit depth of the luma samples).

**12.11.5.2.2 Syntax**

class AlphaInformationBox extends FullBox ('alpi', version = 0, flags){  
 unsigned int(16) opaque\_value;  
 unsigned int(16) transparent\_value;  
 unsigned int(6) reserved;  
}

**12.11.5.2.3 Semantics**

version is an integer that specifies the version of this box.

flags is a 24-bit integer with flags; the following values are defined:

is\_ premultiplied: Flag mask is 0x000001. Specifies if the frame values of the primary video stream comprised in the referenced video track alpha values are premultiplied by the alpha values. The value of 0 specifies that the frame values of the primary video stream are not premultiplied by the alpha values. The value of 1 specifies that the frame values of the primary video stream are premultiplied by the alpha values.

opaque\_value specifies the alpha value for which the referenced video track values are considered opaque for the purposes of alpha blending.

transparent\_value specifies the alpha value for which the referenced video track values are considered transparent for the purposes of alpha blending.

### 12.11.6 Auxiliary Information box

### 12.11.6.1 Invalid depth band box

### 12.11.6.1.1 Definition

Box Type: 'indb'   
Container: DepthInformationBox  
Mandatory: No  
Quantity: One

The InvalidDepthBandBox is used to provide information related to the presence of an invalid depth band in a depth image, independently of the coding.

### 12.11.6.1.2 Syntax

aligned(8) class InvalidDepthBandBox extends Box('indb') {  
 unsigned int(2) band\_side;

unsigned int(14) band\_length;  
}

### 12.11.6.1.3 Semantics

band\_sidespecifies the side of the depth image in which the invalid depth band is located. When the band\_side value is equal to 0, the band is located at the left side of the depth image. Analogously, when the band\_side value is equal to 1, 2, and 3, the band is located at the top, right and bottom side, respectively.

band\_lengthspecifies the length of the invalid depth band in the cropped depth image in pixel dimensions. When the band\_side value is equal to 0 or 2, the band\_length value indicates the width in pixel dimensions. When the band\_side value is equal to 1 or 3, the band\_length value indicates the height in pixel dimensions.

# Annex B Guidance on deriving from this document

Add the following clause B.2.3 to Annex B “Guidance on deriving from this document”:

**B.2.3 Deprecation of identifiers**

Derived specifications ought to maintain a list of boxes and other identifiers that were specified in earlier versions but are now deprecated.

When documenting deprecated identifiers, the specification ought to indicate the version in which the deprecation occurred or reference the last version where the element was valid. Deprecated identifiers ought not be reassigned or reused for new definitions in future versions of any specification. While these deprecated identifiers ought not be used in new content, maintaining their documentation supports compatibility with legacy content and assists implementers working with historical specifications. This documentation may be maintained in an informative annex.

# Annex L Depreciation of boxes and identifiers

Add the following new Annex L:

1. (normative)  
     
   Depreciation of boxes and identifiers
   1. Overview
   2. Deprecated Boxes

The following table L.1 provides an overview of all boxes which were defined by previous editions of this specification but are now deprecated. These box identifiers shall not get reassigned for future definitions.

Table L.1 — Previously defined boxes

|  |  |  |  |
| --- | --- | --- | --- |
| Four-CC | Description | Clause in last containing edition | Deprecated with edition-# |
| mere | Metabox Relation box | 8.11.8 | 6 |
| meco | Additional metadata container box | 8.11.7 | 6 |
| imif | IPMPInfoBox | 13.4.4 | 5 or earlier |
| ipmc | IPMP control box | 13.4.5 | 5 or earlier |
| stsl | Sample scale box | 8.5.4 | 5 or earlier |

# Annex M Handling of depth and alpha maps

Add the following new Annex M:

1. (informative)  
     
   Handling of depth and alpha maps
   1. General

ISOBMFF handles carriage of depth and alpha maps in a codec-agnostic way. This annex defines generic rules on how to use the encapsulated depth and alpha data.

# M.2 Handling of parameters in SEI messages and container level

NOTE Some of the following rules are the consequences of the file format constraints defined for the carriage of depth and alpha track (see Clause 12.11).

It is recommended:

1. If a Depth representation information SEI message, as specified in AVC [37], HEVC [38] or VVC/VSEI [36], with depth\_representation\_type equal to 1 or 3 is present, the DepthInformationBox should not be present. If present, the parameters of DepthInformationBox shall be ignored.
2. If a Depth representation information SEI message, as specified in AVC [37], HEVC [38] or VVC/VSEI [36], with depth\_representation\_type equal to 0 or 2, and a DepthInformationBox are both present, the parameters of the DepthInformationBox should be set as follows:
   1. If depth\_representation\_type = 0, then depth\_mapping\_type = 3.
   2. If depth\_representation\_type = 2, then depth\_mapping\_type = 2.
   3. If z\_far\_flag = 1, then far\_plane = ZFar and units = unspecified
   4. If z\_near\_flag = 1, then near\_plane = ZNear and units = unspecified
3. If an Alpha channel information SEI message, as specified in HEVC [38] or VVC/VSEI [36], and an AlphaInformationBox are both present, the parameters of the AlphaInformationBox should be set as follows:
   1. If alpha\_channel\_use\_idc = 0, then is\_premultiplied = 0
   2. If alpha\_channel\_use\_idc = 1, then is\_premultiplied = 1
   3. transparent\_value = alpha\_transparent\_value
   4. opaque\_value = alpha\_opaque\_value
   5. Lossless conversion between SEI-based and IEEE 754 32-bit floating-point formats
      1. SEI-based floating point format

The SEI-based refers to the floating-point format specified by the Depth representation information element syntax, provided in clause G.14.2.4.2 of HEVC [38] and given in Table M.1.

Table M.1 — Depth representation information element syntax, as defined in HEVC, clause G.14.2.4.2 [38]

|  |  |
| --- | --- |
| Depth\_rep\_info\_element( OutSign, OutExp, OutMantissa, OutManLen ) | Descriptor |
| da\_sign\_flag | u(1) |
| da\_exponent | u(7) |
| da\_mantissa\_len\_minus1 | u(5) |
| da\_mantissa | u(v) |

The three main fields of this floating-point format are defined as follows:

* **Sign**:1 bit
* **Exponent**: 7 bits **| Bias** = 31 **| Unbiased exponent values** = [-30, 95]
* **Mantissa:** 1-32 bits

The length of the mantissa field (i.e., 1 to 32 bits) is determined by the da\_mantissa\_len\_minus1 parameter.

* + 1. IEEE 754 32-bit floating-point format

The IEEE 754 32-bit refers to the single precision floating-point format specified in the IEEE 754 standard [39].

The three main fields of this floating-point format are defined as follows:

* **Sign**:1 bit
* **Exponent**: 8 bits **| Bias** = 127 **| Unbiased exponent values** = [-126, 127]
* **Mantissa:** 23 bits
  + 1. Lossless conversion rules

There are two rules that should be met for lossless conversion of the SEI-based floating point format () to the IEEE 754 32-bit floating-point format (), within the common range of real (representable) numbers ():

1. The mantissa length of must be set equal to 23, and
2. The unbiased exponent values of and must be equal, according to Equation 3

(3)

where represents the unsigned integer value of the exponent field of and takes a value in the range 0 to 127, represents the bias specified for and is equal to , represents the unsigned integer value of the exponent field of and takes a value in the range 0 to 255, represents the bias specified for and is equal to , and is determined by the smallest negative and the largest positive real (representable) number of X.

NOTE 1 Special cases, such as the is equal to 0 and 127, require special handling.

NOTE 2 To respect all aforementioned definitions, should take a value in the range 97 to 222.

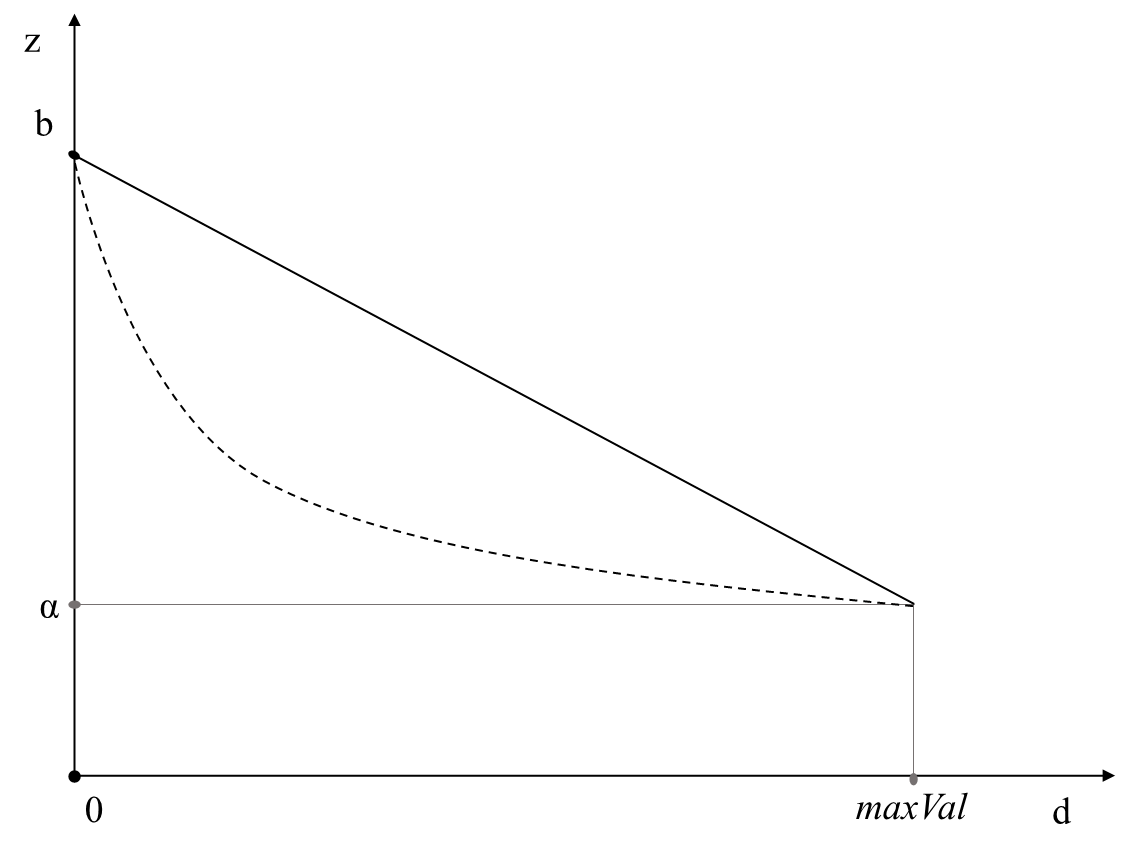
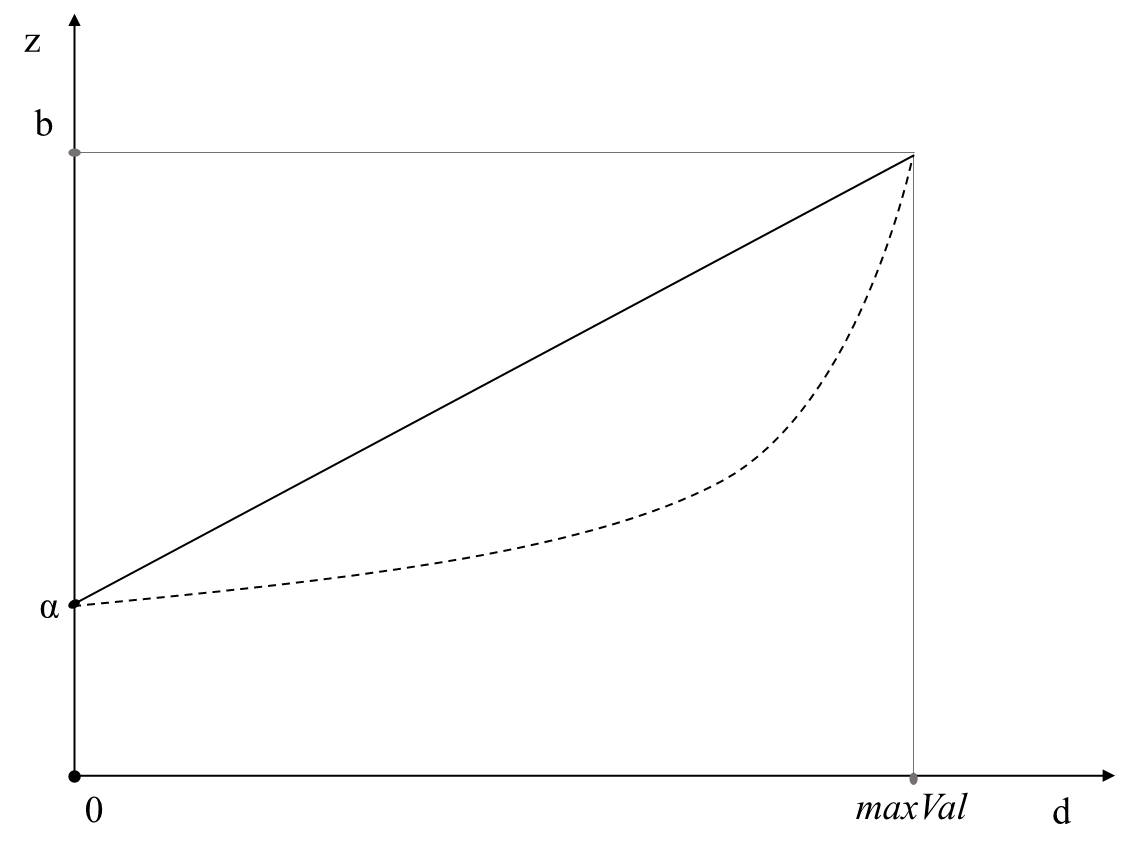
NOTE 3 The smallest negative number represented by X is , and the largest positive number represented by X is .

* 1. Interpretation of depth decoded sample values

Decoded sample values are interpreted as depth values that are computed based on the depth\_mapping\_type value, as specified in Table 15, and the variables and , as specified by Equations 1 and 2 (Clause12.11.5.1.3) , respectively, according to Equation 4:

(4)

Indicative graphs representing Equation 4 are shown in Figure M.2. The left plot corresponds to graphs obtained for depth\_mapping\_type value equal to 0 and 1. The right plot corresponds to graphs obtained for depth\_mapping\_type value equal to 2 and 3. A solid line is used to represent a linear relationship, specified with depth\_mapping\_type value equal to 0 and 2, and a dashed line is used to represent an inverse relationship, specified with depth\_mapping\_type value equal to 1 and 3, between the decoded sample values and the depth values .



**Figure M.2 —** **Graphs obtained for depth\_mapping\_type value equal to 0 and 1 (left) and for depth\_mapping\_type value equal to 2 and 3 (right)**

* 1. Recommended implementation of alpha blending composition

A recommended implementation of the alpha blending composition is provided in clause 8.23.2 of VSEI [36] specification. An identical implementation is obtained using the parameters specified in the AlphaInformationBox by applying the following changes in the corresponding VSEI [36] text:

* “alpha\_opaque\_value” is replaced by “opaque\_value”.
* “alpha\_transparent\_value” is replaced by “transparent\_value”.
* “alpha\_channel\_use\_idc equal to 0” is replaced by “is\_premultiplied equal to 0”.
* “alpha\_channel\_use\_idc equal to 1” is replaced by “is\_premultiplied equal to 1”.

# Bibliography

Add the following references in the bibliography:

[36] ISO/IEC 23002-7*, Information technology — MPEG video technologies — Part 7: Versatile supplemental enhancement information messages for coded video bitstreams*

[37] ISO/IEC 14496-10, *Information technology — Coding of audio-visual objects — Part 10: Advanced Video Coding (AVC)*

[38] ISO/IEC 23008-2, *Information technology — High efficiency coding and media delivery in heterogeneous environments — Part 2: High efficiency video coding (HEVC)*

[39] IEEE Computer Society (2019-07-22). *IEEE Standard for Floating-Point Arithmetic. IEEE STD 754-2019. IEEE*