**COMMITTEE DRAFT AMENDMENT****© ISO/IEC 2021 – All rights reserved****Text of ISO/IEC 23090-3:2021/CDAM 1** **63****Part 3: Versatile video coding, AMENDMENT 1: Operation range extensions****Information technology — Coded representation of immersive media****Élément introductif — Élément central — Partie 3: Titre de la partie****Information technology — Coded representation of immersive media — Part 3: Versatile video coding, AMENDMENT 1: Operation range extensions****E****2021-04-28****(30) Committee****ISO/IEC****ISO/IEC J****2021****1****Amendment****International Standard****202x****52****ISO/IEC 23090‑****ISO/IEC 23090‑3****ISO/IEC 23090-3:2021/CDAM 1****JISC****Coding of audio, picture, multimedia and hypermedia information****Information technology****5****29****1** **2****見出し 2****見出し 1****0****2****STD Version 2.1c2****30** **4** **ISO/IEC JTC 1/SC 29 /WG 5 N 52**

Date: **2021-04-28**

**Text of ISO/IEC 23090-3:2021/CDAM 1**

ISO/IEC JTC 1/SC 29/WG 5

Secretariat:  JISC

**Information technology — Coded representation of immersive media — Part 3: Versatile video coding, AMENDMENT 1: Operation range extensions**

*Élément introductif — Élément central — Partie 3: Titre de la partie*

|  |
| --- |
| **Warning**  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 |

**Copyright notice**

This ISO document is a working draft or committee draft and is copyright-protected by ISO. While the reproduction of working drafts or committee drafts in any form for use by participants in the ISO standards development process is permitted without prior permission from ISO, neither this document nor any extract from it may be reproduced, stored or transmitted in any form for any other purpose without prior written permission from ISO.

Requests for permission to reproduce this document for the purpose of selling it should be addressed as shown below or to ISO's member body in the country of the requester:

[Indicate the full address, telephone number, fax number, telex number, and electronic mail address, as appropriate, of the Copyright Manger of the ISO member body responsible for the secretariat of the TC or SC within the framework of which the working document has been prepared.]

Reproduction for sales purposes may be subject to royalty payments or a licensing agreement.

Violators may be prosecuted.

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

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 document 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)).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see [www.iso.org/patents](https://www.iso.org/iso-standards-and-patents.html)) or the IEC list of patent declarations received (see [http://patents.iec.ch](http://patents.iec.ch/)).

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*, Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information*, in collaboration with ITU-T. Technically aligned twin text is published as Rec. ITU-T H.266.

A list of all parts in the ISO/IEC 23090 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 — Coded representation of immersive media — Part 3: Versatile video coding, AMENDMENT 1: Operation range extensions

*7.3.2.4*

In subclause 7.3.2.4, replace the rows of the syntax table that begin with the row that contains “if( sps\_extension\_flag )”, through the remainder of the syntax table, with the following:

|  |  |
| --- | --- |
| if( sps\_extension\_flag ) { |  |
| **sps\_range\_extension\_flag** | u(1) |
| **sps\_extension\_7bits** | u(7) |
| if( sps\_range\_extension\_flag ) |  |
| sps\_range\_extension( ) |  |
| if( sps\_extension\_7bits ) |  |
| while( more\_rbsp\_data( ) ) |  |
| **sps\_extension\_data\_flag** | u(1) |
| } |  |
| rbsp\_trailing\_bits( ) |  |
| } |  |

*7.3.2.21*

Replace the contents of subclause 7.3.2.21 with the following:

**7.3.2.21** **VUI payload syntax**

|  |  |
| --- | --- |
| vui\_payload( payloadSize ) { | **Descriptor** |
| VuiExtensionBitsPresentFlag = 0 |  | |
| vui\_parameters( payloadSize ) /\* Specified in Rec. ITU-T H.274 | ISO/IEC 23002-7 \*/ |  |
| if( VuiExtensionBitsPresentFlag  | |  more\_data\_in\_payload( ) ) { |  | |
| if( payload\_extension\_present( ) ) |  | |
| **vui\_reserved\_payload\_extension\_data** | u(v) | |
| **vui\_payload\_bit\_equal\_to\_one** /\* equal to 1 \*/ | f(1) | |
| while( !byte\_aligned( ) ) |  | |
| **vui\_payload\_bit\_equal\_to\_zero** /\* equal to 0 \*/ | f(1) | |
| } |  | |
| } |  |

*New subclause 7.3.2.22*

Add subclause 7.3.2.22 as follows:

**7.3.2.22 Sequence parameter set range extension syntax**

|  |  |
| --- | --- |
| sps\_range\_extension( ) { | **Descriptor** |
| **extended\_precision\_processing\_flag** | u(1) |
| **sps\_ts\_residual\_coding\_rice\_present\_in\_sh\_flag** | u(1) |
| **sps\_rrc\_rice\_extension\_flag** | u(1) |
| **sps\_persistent\_rice\_adaptation\_enabled\_flag** | u(1) |
| } |  |

*7.3.7*

In subclause 7.3.7, insert the following additional two rows in the syntax table, after the row that contains “sh\_ts\_residual\_coding\_disabled\_flag”:

|  |  |
| --- | --- |
| if( !sh\_ts\_residual\_coding\_disabled\_flag  &&  sps\_ts\_residual\_coding\_rice\_present\_in\_sh\_flag ) |  |
| **sh\_ts\_residual\_coding\_rice\_idx\_minus1** | u(3) |

*7.3.11.11*

In subclause 7.3.11.11, insert the following additional two rows in the syntax table, after the consecutive rows that start with “lastScanPos” and “lastSubBlock”:

|  |  |
| --- | --- |
| HistValue = sps\_persistent\_rice\_adaptation\_enabled\_flag ? ( 1  <<  StatCoeff[ cIdx ] ) : 0 |  |
| updateHist = sps\_persistent\_rice\_adaptation\_enabled\_flag ? 1 : 0 |  |

And replace the following two rows of the syntax table:

|  |  |
| --- | --- |
| if( abs\_level\_gtx\_flag[ n ][ 1 ] ) |  |
| **abs\_remainder**[ n ] | ae(v) |

with the following:

|  |  |
| --- | --- |
| if( abs\_level\_gtx\_flag[ n ][ 1 ] ) { |  |
| **abs\_remainder**[ n ] | ae(v) |
| if( updateHist  &&  abs\_remainder[ n ] > 0 ) { |  |
| StatCoeff[ cIdx ] = ( StatCoeff[ cIdx ] +  Floor( Log2( abs\_remainder[ n ] ) ) + 2 ) >> 1 |  |
| updateHist = 0 |  |
| } |  |
| } |  |

And replace the following two rows of the syntax table:

|  |  |
| --- | --- |
| if( sb\_coded\_flag[ xS ][ yS ] ) |  |
| **dec\_abs\_level**[ n ] | ae(v) |

with the following:

|  |  |
| --- | --- |
| if( sb\_coded\_flag[ xS ][ yS ] ) { |  |
| **dec\_abs\_level**[ n ] | ae(v) |
| if( updateHist  &&  dec\_abs\_level[ n ] > 0 ) { |  |
| StatCoeff[ cIdx ] = ( StatCoeff[ cIdx ] +  Floor( Log2( dec\_abs\_level[ n ] ) ) ) >> 1 |  |
| updateHist = 0 |  |
| } |  |

*7.4.3.3*

In subclause 7.4.3.3, replace the following:

A VPS RBSP shall be available to the decoding process prior to it being referenced, included in at least one AU with TemporalId equal to 0 or provided through external means.

All VPS NAL units with a particular value of vps\_video\_parameter\_set\_id in a CVS shall have the same content.

**vps\_video\_parameter\_set\_id** provides an identifier for the VPS for reference by other syntax elements. The value of vps\_video\_parameter\_set\_id shall be greater than 0.

with the following:

A VPS RBSP shall be available to the decoding process, by inclusion in at least one AU with TemporalId equal to 0 or provided through external means, prior to it being referenced by either of the following:

* a PH NAL unit having a ph\_pic\_parameter\_set\_id that refers to a PPS with pps\_seq\_parameter\_set\_id equal to the value of sps\_seq\_parameter\_set\_id in an SPS NAL unit with sps\_video\_parameter\_set\_id equal to the value of vps\_video\_parameter\_set\_id in the VPS RBSP,
* a coded slice NAL unit having sh\_picture\_header\_in\_slice\_header\_flag equal to 1 with a ph\_pic\_parameter\_set\_id that refers to a PPS with pps\_seq\_parameter\_set\_id equal to the value of sps\_seq\_parameter\_set\_id in an SPS NAL unit with sps\_video\_parameter\_set\_id equal to the value of vps\_video\_parameter\_set\_id in the VPS RBSP.

Such a PH NAL unit or coded slice NAL unit references the previous VPS RBSP in decoding order (relative to the position of the PH NAL unit or coded slice NAL unit in decoding order) with that value of vps\_video\_parameter\_set\_id.

All VPS NAL units with a particular value of vps\_video\_parameter\_set\_id in a CVS shall have the same content.

**vps\_video\_parameter\_set\_id** provides an identifier for the VPS for reference by other syntax elements. The value of vps\_video\_parameter\_set\_id shall be greater than 0.

VPS NAL units, regardless of the nuh\_layer\_id values, share the same value space of vps\_video\_parameter\_set\_id.

*7.4.3.4*

In subclause 7.4.3.4, replace the following:

An SPS RBSP shall be available to the decoding process prior to it being referenced, included in at least one AU with TemporalId equal to 0 or provided through external means.

with the following:

An SPS RBSP shall be available to the decoding process, by inclusion in at least one AU with TemporalId equal to 0 or provided through external means, prior to it being referenced by either of the following:

* a PH NAL unit having a ph\_pic\_parameter\_set\_id that refers to a PPS with pps\_seq\_parameter\_set\_id equal to the value of sps\_seq\_parameter\_set\_id in the SPS RBSP,
* a coded slice NAL unit having sh\_picture\_header\_in\_slice\_header\_flag equal to 1 with a ph\_pic\_parameter\_set\_id that refers to a PPS with pps\_seq\_parameter\_set\_id equal to the value of sps\_seq\_parameter\_set\_id in the SPS RBSP.

Such a PH NAL unit or coded slice NAL unit references the previous SPS RBSP in decoding order (relative to the position of the PH NAL unit or coded slice NAL unit in decoding order) with that value of sps\_seq\_parameter\_set\_id.

And replace the following:

**sps\_extension\_flag** equal to 0 specifies that no sps\_extension\_data\_flag syntax elements are present in the SPS RBSP syntax structure. sps\_extension\_flag equal to 1 specifies that sps\_extension\_data\_flag syntax elements might be present in the SPS RBSP syntax structure. sps\_extension\_flag shall be equal to 0 in bitstreams conforming to this version of this Specification. However, some use of sps\_extension\_flag equal to 1 could be specified in some future version of this Specification, and decoders conforming to this version of this Specification shall allow the value of sps\_extension\_flag equal to 1 to appear in the syntax.

**sps\_extension\_data\_flag** could have any value. Its presence and value do not affect the decoding process specified in this version of this Specification. Decoders conforming to this version of this Specification shall ignore all sps\_extension\_data\_flag syntax elements.

with the following:

**sps\_extension\_flag** equal to 1 specifies that the syntax elements sps\_range\_extension\_flag and sps\_extension\_7bits are present in the SPS RBSP syntax structure. sps\_extension\_flag equal to 0 specifies that these syntax elements are not present.

**sps\_range\_extension\_flag** equal to 1 specifies that the sps\_range\_extension( ) syntax structure is present in the SPS RBSP syntax structure. sps\_range\_extension\_flag equal to 0 specifies that this syntax structure is not present. When not present, the value of sps\_range\_extension\_flag is inferred to be equal to 0.

**sps\_extension\_7bits** equal to 0 specifies that no sps\_extension\_data\_flag syntax elements are present in the SPS RBSP syntax structure. When present, sps\_extension\_7bits shall be equal to 0 in bitstreams conforming to this version of this Specification. Values of sps\_extension\_7bits not equal to 0 are reserved for future use by ITU-T | ISO/IEC. Decoders shall allow the value of sps\_extension\_7bits not equal to 0 to appear in the syntax. When not present, the value of sps\_extension\_7bits is inferred to be equal to 0.

**sps\_extension\_data\_flag** may have any value. Its presence and value do not affect the decoding process specified in this version of this Specification. Decoders conforming to this version of this Specification shall ignore all sps\_extension\_data\_flag syntax elements.

*7.4.3.5*

In subclause 7.4.3.5, replace the following:

A PPS RBSP shall be available to the decoding process prior to it being referenced, included in at least one AU with TemporalId less than or equal to the TemporalId of the PPS NAL unit or provided through external means**.**

with the following:

A PPS RBSP shall be available to the decoding process, by inclusion in at least one AU with TemporalId less than or equal to the TemporalId of the PPS NAL unit or provided through external means, prior to it being referenced by either of the following:

* a PH NAL unit having ph\_pic\_parameter\_set\_id equal to the value of pps\_pic\_parameter\_set\_id in the PPS RBSP,
* a coded slice NAL unit having sh\_picture\_header\_in\_slice\_header\_flag equal to 1 with ph\_pic\_parameter\_set\_id equal to the value of pps\_pic\_parameter\_set\_id in the PPS RBSP.

Such a PH NAL unit or coded slice NAL unit references the previous PPS RBSP in decoding order (relative to the position of the PH NAL unit or coded slice NAL unit in decoding order) with that value of pps\_pic\_parameter\_set\_id.

*7.4.3.6*

In subclause 7.4.3.6, replace the following:

Each APS RBSP shall be available to the decoding process prior to it being referenced, included in at least one AU with TemporalId less than or equal to the TemporalId of the coded slice NAL unit that refers it or provided through external means.

All APS NAL units with a particular value of nal\_unit\_type, a particular value of aps\_adaptation\_parameter\_set\_id, and a particular value of aps\_params\_type within a PU shall have the same content.

with the following:

Each APS RBSP shall be available to the decoding process, by inclusion in at least one AU with TemporalId less than or equal to the TemporalId of the coded slice NAL unit that refers it or provided through external means, prior to it being referenced by any of the following:

* a coded slice NAL unit in a PU with a PH NAL unit having a ph\_alf\_aps\_id\_luma[ i ] or ph\_alf\_aps\_id\_chroma or ph\_alf\_cc\_cb\_aps\_id or ph\_alf\_cc\_cr\_aps\_id syntax element that is present and equal to the aps\_adaptation\_parameter\_set\_id of an APS RBSP with aps\_params\_type equal to ALF\_APS,
* a coded slice NAL unit in a PU having a PH NAL unit with a ph\_lmcs\_aps\_id syntax element that is present and equal to the aps\_adaptation\_parameter\_set\_id of an APS RBSP with aps\_params\_type equal to LMCS\_APS,
* a coded slice NAL unit in a PU having a PH NAL unit with a ph\_scaling\_list\_aps\_id syntax element that is present and equal to the aps\_adaptation\_parameter\_set\_id of an APS RBSP with aps\_params\_type equal to SCALING\_APS,
* a coded slice NAL unit having a sh\_alf\_aps\_id\_luma[ i ] or sh\_alf\_aps\_id\_chroma or sh\_alf\_cc\_cb\_aps\_id or sh\_alf\_cc\_cr\_aps\_id syntax element that is present and equal to the aps\_adaptation\_parameter\_set\_id of an APS RBSP with aps\_params\_type equal to ALF\_APS,
* a coded slice NAL unit having sh\_picture\_header\_in\_slice\_header\_flag equal to 1 with a ph\_alf\_aps\_id\_luma[ i ] or ph\_alf\_aps\_id\_chroma or ph\_alf\_cc\_cb\_aps\_id or ph\_alf\_cc\_cr\_aps\_id syntax element that is present and equal to the aps\_adaptation\_parameter\_set\_id of an APS RBSP with aps\_params\_type equal to ALF\_APS,
* a coded slice NAL unit having sh\_picture\_header\_in\_slice\_header\_flag equal to 1 with a ph\_lmcs\_aps\_id syntax element that is present and equal to the aps\_adaptation\_parameter\_set\_id of an APS RBSP with aps\_params\_type equal to LMCS\_APS,
* a coded slice NAL unit having sh\_picture\_header\_in\_slice\_header\_flag equal to 1 with a ph\_scaling\_list\_aps\_id syntax element that is present and equal to the aps\_adaptation\_parameter\_set\_id of an APS RBSP with aps\_params\_type equal to SCALING\_APS.

Such a coded slice NAL unit references the previous APS RBSP in decoding order (relative to the position of the coded slice NAL unit in decoding order) with the corresponding values of aps\_adaptation\_parameter\_set\_id and aps\_params\_type.

All APS NAL units with a particular value of nal\_unit\_type, a particular value of aps\_adaptation\_parameter\_set\_id, and a particular value of aps\_params\_type within a PU shall have the same content.

NOTE 1 The content of an APS RBSP in a suffix APS NAL unit and the content of a prefix APS NAL unit with the same values of aps\_adaptation\_parameter\_set\_id (and aps\_params\_type) in the same PU can be different. When a suffix APS NAL unit is present in a PU, its APS RBSP cannot be referenced by the decoding process of that PU, since the suffix APS NAL unit cannot precede the PH NAL unit or any coded slice NAL units of that PU (see subclause 7.4.2.4.4). However, the APS RBSP in a suffix APS NAL unit can be referenced by the decoding process of subsequent PUs in the bitstream (if any).

And replace the three existing NOTEs with the following:

NOTE 2 In a CVS that contains only one layer, the nuh\_layer\_id of referenced APSs is equal to the nuh\_layer\_id of the VCL NAL units.

NOTE 3 An APS NAL unit (with a particular value of nal\_unit\_type, a particular value of aps\_adaptation\_parameter\_set\_id, and a particular value of aps\_params\_type) could be shared across pictures, and different slices within a picture can refer to different ALF APSs.

NOTE 4 A suffix APS NAL unit associated with a particular VCL NAL unit (a VCL NAL unit that precedes the suffix APS NAL unit in decoding order and is the last VCL NAL unit of the PU containing that VCL NAL unit) is not for use in the decoding process of that particular VCL NAL unit or any other VCL NAL unit of the PU containing that particular VCL NAL unit, but rather is for use in the decoding process of VCL NAL units of PUs that follow the suffix APS NAL unit in decoding order (if any).

*New subclause 7.4.3.22*

Add new subclause 7.4.3.22 as follows:

**7.4.3.22 Sequence parameter set range extension semantics**

**extended\_precision\_processing\_flag** equal to 1 specifies that an extended dynamic range may be used for transform coefficients and transform processing. extended\_precision\_processing\_flag equal to 0 specifies that the extended dynamic range is not used. When not present, the value of extended\_precision\_processing\_flag is inferred to be equal to 0.

The variable ExtendedPrecisionFlag is derived as follows:

– If extended\_precision\_processing\_flag is equal to 1 and BitDepth is greater than 10, ExtendedPrecisionFlag is set equal to 1.

– Otherwise (extended\_precision\_processing\_flag is equal to 0 or BitDepth is less than or equal to 10), ExtendedPrecisionFlag is set equal to 0.

The variable Log2TransformRange is derived as follows:

Log2TransformRange = ExtendedPrecisionFlag ? Max( 15, Min( 20, BitDepth + 6 ) ) : 15 (106)

**sps\_ts\_residual\_coding\_rice\_present\_in\_sh\_flag** equal to 1 specifies that sh\_ts\_residual\_coding\_rice\_idx\_minus1 may be present in slice\_header( ) syntax structures referring to the SPS. sps\_ts\_residual\_coding\_rice\_present\_in\_sh\_flag equal to 0 specifies that sh\_ts\_residual\_coding\_rice\_idx\_minus1 is not present in slice\_header( ) syntax structures referring to the SPS. When not present, the value of sps\_ts\_residual\_coding\_rice\_present\_in\_sh\_flag is inferred to be equal to 0.

**sps\_rrc\_rice\_extension\_flag** equal to 1 specifies that an extension of Rice parameter derivation for the binarization of abs\_remaining[ ] and dec\_abs\_level[ ] is enabled. sps\_rrc\_rice\_extension\_flag equal to 0 specifies that the extension of Rice parameter derivation for the binarization of abs\_remaining[ ] and dec\_abs\_level[ ] is disabled. When not present, the value of sps\_rrc\_rice\_extension\_flag is inferred to be equal to 0.

**sps\_persistent\_rice\_adaptation\_enabled\_flag** equal to 1 specifies that Rice parameter derivation for the binarization of abs\_remainder[ ] and dec\_abs\_level[ ] is initialized at the start of each TU using statistics accumulated from previous TUs. sps\_persistent\_rice\_adaptation\_enabled\_flag equal to 0 specifies that no previous TU state is used in Rice parameter derivation. When not present, the value of sps\_persistent\_rice\_adaptation\_enabled\_flag is inferred to be equal to 0.

And renumber the prior Equations (106) to (1509) in the document as Equations (107) to (1510), and update the corresponding cross-references accordingly.

*7.4.8*

In subclause 7.4.8, add the following after the semantics of sh\_ts\_residual\_coding\_disabled\_flag:

**sh\_ts\_residual\_coding\_rice\_idx\_minus1** plus 1 specifies the Rice parameter used for the residual\_ts\_coding( ) syntax structure. When not present, the value of sh\_ts\_residual\_coding\_rice\_idx\_minus1 is inferred to be equal to 0.

*7.4.12.11*

In subclause 7.4.12.11, replace the following (with renumbering as shown):

CoeffMin = −( 1 << 15) ) (189)

CoeffMax = ( 1 << 15) ) − 1 (190)

with the following:

CoeffMin = −( 1 << ( ExtendedPrecisionFlag ? Max( 15, Min( 20, BitDepth + 6 ) ) : 15 ) ) (190)

CoeffMax = ( 1 << ( ExtendedPrecisionFlag ? Max( 15, Min( 20, BitDepth + 6 ) ) : 15 ) ) − 1 (191)

*8.1.2*

In subclause 8.1.2, replace the paragraphs numbered as steps 3 and 4 with the following:

1. The processes in subclauses 8.4, 8.5, 8.6, 8.7, and 8.8 specify decoding processes using syntax elements in all syntax structure layers. When any NAL units are present in a PU that follow the last VCL NAL unit of the picture in decoding order, their decoding is deferred until after all slices of the current picture have been decoded and the in-loop filter processes of subclause 8.8 have been applied. It is a requirement of bitstream conformance that the coded slices of the picture shall contain slice data for every CTU of the picture, such that the division of the picture into slices, and the division of the slices into CTUs each forms a partitioning of the picture.
2. After all slices of the current picture have been decoded, the in-loop filter processes of subclause 8.8 have been applied, and all remaining NAL units of the PU have been decoded, the current decoded picture is marked as "used for short-term reference", the picture referred to by each ILRP entry, when present, in RefPicList[ 0 ] or RefPicList[ 1 ] is marked as "used for short-term reference", and the variable PictureOutputFlag of the current picture is derived as follows:

*8.7.3*

In subclause 8.7.3, replace the following (with renumbering as shown):

bdShift = BitDepth + rectNonTsFlag + (1133)  
 ( ( Log2( nTbW ) + Log2( nTbH ) ) / 2 ) − 5 + sh\_dep\_quant\_used\_flag

with the following:

bdShift = BitDepth + rectNonTsFlag + (1134)  
 ( ( Log2( nTbW ) + Log2( nTbH ) ) / 2 ) + 10 − Log2TransformRange + sh\_dep\_quant\_used\_flag

*8.7.4*

In subclause 8.7.4, replace the following (with renumbering as shown):

bdShift = ( nTbH > 1  &&  nTbW > 1 ) ? ( 20 − BitDepth ) : ( 21 − BitDepth ) (1164)

with the following:

bdShift = ( nTbH > 1  &&  nTbW > 1 ) ? ( 5 + Log2TransformRange − BitDepth ) :  
 ( 6 + Log2TransformRange − BitDepth ) (1165)

*9.3.2.1*

In subclause 9.3.2.1, replace the following:

* If the CTU is the first CTU in a slice or tile, the initialization process for context variables is invoked as specified in subclause 9.3.2.2 and the array PredictorPaletteSize[ chType ], with chType = 0, 1, is initialized to 0.

with the following:

The context variables of the arithmetic decoding engine are initialized as follows:

* If the CTU is the first CTU in a slice or tile, the initialization process for context variables is invoked as specified in subclause 9.3.2.2 and the array PredictorPaletteSize[ chType ], with chType = 0, 1, is initialized to 0, and the following applies:

– If sps\_persistent\_rice\_adaptation\_enabled\_flag is equal to 0, the values of StatCoeff[ idx ] for idx ranging from 0 to 2, inclusive, are initialized to be equal to 0.

– Otherwise the following applies:

StatCoeff[ idx ] = ( bitDepth > 10 ) ? ( 2 \* Floor( Log2( bitDepth − 10 ) ) : 0 (1511)

And renumber the prior Equations (1510) to (1517) in the document as Equations (1512) to (1519), and update the corresponding cross-references accordingly.

*9.3.3.2*

Replace the contents of subclause 9.3.3.2 with the following:

**9.3.3.2 Rice parameter derivation process for abs\_remainder[ ] and dec\_abs\_level[ ]**

Inputs to this process are the base level baseLevel, the colour component index cIdx, the luma location ( x0, y0 ) specifying the top-left sample of the current transform block relative to the top-left sample of the current picture, the current coefficient scan location ( xC, yC ), the binary logarithm of the transform block width log2TbWidth, and the binary logarithm of the transform block height log2TbHeight.

Output of this process is the Rice parameter cRiceParam.

Given the array AbsLevel[ x ][ y ] for the transform block with component index cIdx and the top-left luma location ( x0, y0 ), the variable locSumAbs is derived as specified by the following pseudo-code process:

locSumAbs = 0  
if( xC < ( 1 << log2TbWidth ) − 1 ) {  
 locSumAbs += AbsLevel[ xC + 1 ][ yC ]  
 if( xC < ( 1 << log2TbWidth ) − 2 )  
 locSumAbs += AbsLevel[ xC + 2 ][ yC ]  
 else  
 locSumAbs += HistValue  
 if( yC < ( 1 << log2TbHeight ) − 1 )  
 locSumAbs += AbsLevel[ xC + 1 ][ yC + 1 ] (1520)  
 else  
 locSumAbs += HistValue  
} else  
 locSumAbs += 2 \* HistValue  
if( yC < ( 1 << log2TbHeight ) − 1 ) {  
 locSumAbs += AbsLevel[ xC ][ yC + 1 ]  
 if( yC < ( 1 << log2TbHeight ) − 2 )  
 locSumAbs += AbsLevel[ xC ][ yC + 2 ]  
 else  
 locSumAbs += HistValue  
} else  
 locSumAbs += HistValue

The lists Tx[ ] and Rx[ ] are specifed as follows:

Tx[ ] = { 32, 128, 512, 2048 } (1521)  
Rx[ ] = { 0, 2, 4, 6, 8 } (1522)

The value of the variable shiftVal is derived as follows :

if( !sps\_rrc\_rice\_extension\_flag )  
 shiftVal = 0 (1523)  
else  
 shiftVal = ( localSumAbs < Tx[ 0 ] ) ? Rx[ 0 ] : ( ( localSumAbs < Tx[ 1 ] ) ? Rx[ 1 ] :  
 ( ( localSumAbs < Tx[ 2 ] ) ? Rx[ 2 ] : ( ( localSumAbs < Tx[ 3 ] ) ? Rx[ 3 ] : Rx[4] ) ) )

The value of locSumAbs is updated as follows:

locSumAbs = Clip3( 0, 31, ( locSumAbs >> shiftVal ) − baseLevel \* 5 ) (1524)

Given the variable locSumAbs, the Rice parameter cRiceParam is first derived as specified in Table 128, and then updated as follows:

cRiceParam = cRiceParam + shiftVal (1525)

When baseLevel is equal to 0, the variable ZeroPos[ n ] is derived as follows:

ZeroPos[ n ] = ( QState < 2 ? 1 : 2 )  <<  cRiceParam (1526)

**Table 128 – Specification of cRiceParam based on locSumAbs**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **locSumAbs** | **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** |
| cRiceParam | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 2 | 2 |
| **locSumAbs** | **16** | **17** | **18** | **19** | **20** | **21** | **22** | **23** | **24** | **25** | **26** | **27** | **28** | **29** | **30** | **31** |
| cRiceParam | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 2 | 3 | 3 | 3 | 3 |

And renumber the prior Equations (1520) to (1526) in the document as Equations (1527) to (1533), and update the corresponding cross-references accordingly.

*9.3.3.11*

In subclause 9.3.3.11, replace the following:

* If transform\_skip\_flag[ x0 ][ y0 ][ cIdx ] is equal to 1 and sh\_ts\_residual\_coding\_disabled\_flag is equal to 0, the Rice parameter cRiceParam is set equal to 1.

with the following:

* If transform\_skip\_flag[ x0 ][ y0 ][ cIdx ] is equal to 1 and sh\_ts\_residual\_coding\_disabled\_flag is equal to 0, the Rice parameter cRiceParam is set equal to sh\_ts\_residual\_coding\_rice\_idx\_minus1 + 1.

*9.3.3.11*

In subclause 9.3.3.11, replace the following:

The rice parameter cRiceParam is derived as follows:

* If transform\_skip\_flag[ x0 ][ y0 ][ cIdx ] is equal to 1 and sh\_ts\_residual\_coding\_disabled\_flag is equal to 0, the Rice parameter cRiceParam is set equal to 1.
* Otherwise, the rice parameter cRiceParam is derived by invoking the rice parameter derivation process for abs\_remainder[ ] as specified in subclause 9.3.3.2 with the variable baseLevel set equal to 4, the colour component index cIdx, the luma location ( x0, y0 ), the current coefficient scan location ( xC, yC ), the binary logarithm of the transform block width log2TbWidth, and the binary logarithm of the transform block height log2TbHeight as inputs.

with the following:

The variable baseLevel is derived as follows:

* If sps\_rrc\_rice\_extension\_flag is equal to 0, baseLevel is set equal to 4.
* Otherwise (sps\_rrc\_rice\_extension\_flag is equal to 1), the following applies:

baseLevel = ( BitDepth > 12 ) ? ( ( sh\_slice\_type  = =  I ) ? 1 : 2 ) : ( ( sh\_slice\_type  = =  I ) ? 2 : 3 )  
 (1534)

The Rice parameter cRiceParam is derived as follows:

* If transform\_skip\_flag[ x0 ][ y0 ][ cIdx ] is equal to 1 and sh\_ts\_residual\_coding\_disabled\_flag is equal to 0, the Rice parameter cRiceParam is set equal to 1.
* Otherwise, the Rice parameter cRiceParam is derived by invoking the Rice parameter derivation process for abs\_remainder[ ] as specified in subclause 9.3.3.2 with baseLevel, the colour component index cIdx, the luma location ( x0, y0 ), the current coefficient scan location ( xC, yC ), the binary logarithm of the transform block width log2TbWidth, and the binary logarithm of the transform block height log2TbHeight as inputs.

And renumber the prior Equations (1527) onwards in the document as Equations (1535) onwards, and update the corresponding cross-references accordingly.

And replace the following:

* The suffix bin string is specified by invoking the limited k-th order EGk binarization process as specified in subclause 9.3.3.6 for the binarization of suffixVal with the Exp-Golomb order k set equal to cRiceParam + 1, variable maxPreExtLen set equal to 11 and variable truncSuffixLen set equal to 15 as input.

with the following:

* The suffix bin string is specified by invoking the limited k-th order EGk binarization process as specified in subclause 9.3.3.6 for the binarization of suffixVal with the Exp-Golomb order k set equal to cRiceParam + 1, variable maxPreExtLen set equal to 26 − Log2TransformRange and variable truncSuffixLen set equal to Log2TransformRange as input.

*9.3.3.12*

In subclause 9.3.3.12, replace the following:

* The suffix bin string is specified by invoking the limited k-th order EGk binarization process as specified in subclause 9.3.3.6 for the binarization of suffixVal with the Exp-Golomb order k set equal to cRiceParam + 1, variable maxPreExtLen set equal to 11 and variable truncSuffixLen set equal to 15 as input.

with the following:

* The suffix bin string is specified by invoking the limited k-th order EGk binarization process as specified in subclause 9.3.3.6 for the binarization of suffixVal with the Exp-Golomb order k set equal to cRiceParam + 1, variable maxPreExtLen set equal to 26 − Log2TransformRange and variable truncSuffixLen set equal to Log2TransformRange as input.

*A.4.1*

In Table A.1 of subclause A.4.1, insert the following row after the row for level 6.2:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **6.3** | 105 | 80 216 064 | 240 000 | 800 000 | 1 000 | 990 | 30 |

*A.4.2*

In Table A.2 of subclause A.4.2, insert the following row after the row for level 6.2:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **6.3** | 4 812 963 840 | 320 000 | 800 000 | 8 | 4 |

*C.3.3*

In subclause C.3.3, delete the “NOTE”.

*C.6*

In subclause C.6, step 4, replace the following:

* nal\_unit\_type is equal to APS\_NUT, TRAIL\_NUT, STSA\_NUT, RADL\_NUT, or RASL\_NUT, or nal\_unit\_type is equal to GDR\_NUT and the associated ph\_recovery\_poc\_cnt is greater than 0.

with the following:

– nal\_unit\_type is equal to PREFIX\_APS\_NUT, SUFFIX\_APS\_NUT, TRAIL\_NUT, STSA\_NUT, RADL\_NUT, or RASL\_NUT, or nal\_unit\_type is equal to GDR\_NUT and the associated ph\_recovery\_poc\_cnt is greater than 0.

*D.2.1*

Replace the contents of subclause D.2.1 with the following:

**D.2.1** **General SEI payload syntax**

|  |  |
| --- | --- |
| sei\_payload( payloadType, payloadSize ) { | **Descriptor** |
| SeiExtensionBitsPresentFlag = 0 |  |
| if( nal\_unit\_type = = PREFIX\_SEI\_NUT ) |  |
| if( payloadType = = 0 ) |  |
| buffering\_period( payloadSize ) |  |
| else if( payloadType = = 1 ) |  |
| pic\_timing( payloadSize ) |  |
| else if( payloadType = = 3 ) |  |
| filler\_payload( payloadSize ) /\* Specified in Rec. ITU-T H.274 | ISO/IEC 23002-7 \*/ |  |
| else if( payloadType = = 4 ) /\* Specified in Rec. ITU-T H.274 | ISO/IEC 23002-7 \*/ |  |
| user\_data\_registered\_itu\_t\_t35( payloadSize ) |  |
| else if( payloadType = = 5 ) /\* Specified in Rec. ITU-T H.274 | ISO/IEC 23002-7 \*/ |  |
| user\_data\_unregistered( payloadSize ) |  |
| else if( payloadType = = 19 ) /\* Specified in Rec. ITU-T H.274 | ISO/IEC 23002-7 \*/ |  |
| film\_grain\_characteristics( payloadSize ) |  |
| else if( payloadType = = 45 ) /\* Specified in Rec. ITU-T H.274 | ISO/IEC 23002-7 \*/ |  |
| frame\_packing\_arrangement( payloadSize ) |  |
| else if( payloadType = = 47 ) /\* Specified in Rec. ITU-T H.274 | ISO/IEC 23002-7 \*/ |  |
| display\_orientation( payloadSize ) |  |
| else if( payloadType = = 129 ) /\* Specified in Rec. ITU-T H.274 | ISO/IEC 23002-7 \*/ |  |
| parameter\_sets\_inclusion\_indication( payloadSize ) |  |
| else if( payloadType = = 130 ) |  |
| decoding\_unit\_info( payloadSize ) |  |
| else if( payloadType = = 133 ) |  |
| scalable\_nesting( payloadSize ) |  |
| else if( payloadType = = 137 ) /\* Specified in Rec. ITU-T H.274 | ISO/IEC 23002-7 \*/ |  |
| mastering\_display\_colour\_volume( payloadSize ) |  |
| else if( payloadType = = 142 ) /\* Specified in Rec. ITU-T H.274 | ISO/IEC 23002-7 \*/ |  |
| colour\_transform\_info( payloadSize ) |  |
| else if( payloadType = = 144 ) /\* Specified in Rec. ITU-T H.274 | ISO/IEC 23002-7 \*/ |  |
| content\_light\_level\_info( payloadSize ) |  |
| else if( payloadType = = 145 ) /\* Specified in Rec. ITU-T H.274 | ISO/IEC 23002-7 \*/ |  |
| dependent\_rap\_indication( payloadSize ) |  |
| else if( payloadType = = 147 ) /\* Specified in Rec. ITU-T H.274 | ISO/IEC 23002-7 \*/ |  |
| alternative\_transfer\_characteristics( payloadSize ) |  |
| else if( payloadType = = 148 ) /\* Specified in Rec. ITU-T H.274 | ISO/IEC 23002-7 \*/ |  |
| ambient\_viewing\_environment( payloadSize ) |  |
| else if( payloadType = = 149 ) /\* Specified in Rec. ITU-T H.274 | ISO/IEC 23002-7 \*/ |  |
| content\_colour\_volume( payloadSize ) |  |
| else if( payloadType = = 150 ) /\* Specified in Rec. ITU-T H.274 | ISO/IEC 23002-7 \*/ |  |
| equirectangular\_projection( payloadSize ) |  |
| else if( payloadType = = 153 ) /\* Specified in Rec. ITU-T H.274 | ISO/IEC 23002-7 \*/ |  |
| generalized\_cubemap\_projection( payloadSize ) |  |
| else if( payloadType = = 154 ) /\* Specified in Rec. ITU-T H.274 | ISO/IEC 23002-7 \*/ |  |
| sphere\_rotation( payloadSize ) |  |
| else if( payloadType = = 155 ) /\* Specified in Rec. ITU-T H.274 | ISO/IEC 23002-7 \*/ |  |
| regionwise\_packing( payloadSize ) |  |
| else if( payloadType = = 156 ) /\* Specified in Rec. ITU-T H.274 | ISO/IEC 23002-7 \*/ |  |
| omni\_viewport( payloadSize ) |  |
| else if( payloadType = = 165 ) /\* Specified in Rec. ITU-T H.274 | ISO/IEC 23002-7 \*/ |  |
| alpha\_channel\_info( payloadSize ) |  |
| else if( payloadType = = 168 ) /\* Specified in Rec. ITU-T H.274 | ISO/IEC 23002-7 \*/ |  |
| frame\_field\_info( payloadSize ) |  |
| else if( payloadType = = 177 ) /\* Specified in Rec. ITU-T H.274 | ISO/IEC 23002-7 \*/ |  |
| depth\_representation\_info( payloadSize ) |  |
| else if( payloadType = = 179 ) /\* Specified in Rec. ITU-T H.274 | ISO/IEC 23002-7 \*/ |  |
| multiview\_acquisition\_info( payloadSize ) |  |
| else if( payloadType = = 200 ) |  |
| sei\_manifest( payloadSize ) |  |
| else if( payloadType = = 201 ) |  |
| sei\_prefix\_indication( payloadSize ) |  |
| else if( payloadType = = 202 ) /\* Specified in Rec. ITU-T H.274 | ISO/IEC 23002-7 \*/ |  |
| annotated\_regions( payloadSize ) |  |
| else if( payloadType = = 203 ) |  |
| subpic\_level\_info( payloadSize ) |  |
| else if( payloadType = = 204 ) /\* Specified in Rec. ITU-T H.274 | ISO/IEC 23002-7 \*/ |  |
| sample\_aspect\_ratio\_info( payloadSize ) |  |
| else if( payloadType = = 205 ) /\* Specified in Rec. ITU-T H.274 | ISO/IEC 23002-7 \*/ |  |
| scalability\_dimension\_info( payloadSize ) |  |
| else if( payloadType = = 206 ) /\* Specified in Rec. ITU-T H.274 | ISO/IEC 23002-7 \*/ |  |
| extended\_drap\_indication( payloadSize ) |  |
| else /\* Specified in Rec. ITU-T H.274 | ISO/IEC 23002-7 \*/ |  |
| reserved\_message( payloadSize ) |  |
| else /\* nal\_unit\_type = = SUFFIX\_SEI\_NUT \*/ |  |
| if( payloadType = = 3 ) /\* Specified in Rec. ITU-T H.274 | ISO/IEC 23002-7 \*/ |  |
| filler\_payload( payloadSize ) |  |
| else if( payloadType = = 132 ) /\* Specified in Rec. ITU-T H.274 | ISO/IEC 23002-7 \*/ |  |
| decoded\_picture\_hash( payloadSize ) |  |
| else if( payloadType = = 133 ) |  |
| scalable\_nesting( payloadSize ) |  |
| else /\* Specified in Rec. ITU-T H.274 | ISO/IEC 23002-7 \*/ |  |
| reserved\_message( payloadSize ) |  |
| if( SeiExtensionBitsPresentFlag | | more\_data\_in\_payload( ) ) { |  |
| if( payload\_extension\_present( ) ) |  |
| **sei\_reserved\_payload\_extension\_data** | u(v) |
| **sei\_payload\_bit\_equal\_to\_one** /\* equal to 1 \*/ | f(1) |
| while( !byte\_aligned( ) ) |  |
| **sei\_payload\_bit\_equal\_to\_zero** /\* equal to 0 \*/ | f(1) |
| } |  |
| } |  |

*D.2.2*

In subclause D.2.2, in Table D.1, insert the following rows in the table after the row for “Scalable nesting”:

|  |  |
| --- | --- |
| SEI manifest | The CVS containing the SEI message |
| SEI prefix indication | The CVS containing the SEI message |

And replace the following paragraphs

The list VclAssociatedSeiList is set to consist of the payloadType values 3, 19, 45, 129, 137, 144, 145, 147 to 150, inclusive, 153 to 156, inclusive, 168, and 204.

The list PicUnitRepConSeiList is set to consist of the payloadType values 0, 1, 19, 45, 129, 133, 137, 147 to 150, inclusive, 153 to 156, inclusive, 168, 203, and 204.

with the following:

Only the filler payload, decoded picture hash, and scalable nesting SEI messages may be included in a suffix SEI NAL unit; all other SEI messages are not allowed to be included in a suffix SEI NAL unit. When a scalable nesting SEI message is included in a suffix SEI NAL unit, it shall contain only those SEI messages that are allowed to be included in a suffix NAL unit. When a scalable nesting SEI message is included in a prefix SEI NAL unit, it shall contain only those SEI messages that are allowed to be included in a prefix NAL unit.

The list VclAssociatedSeiList is set to consist of the payloadType values 3, 19, 45, 47, 129, 132, 137, 142, 144, 145, 147 to 150, inclusive, 153 to 156, inclusive, 165, 168, 177, 179, 200 to 202, inclusive, and 204 to 206, inclusive.

The list PicUnitRepConSeiList is set to consist of the payloadType values 0, 1, 19, 45, 129, 133, 137, 147 to 150, inclusive, 153 to 156, inclusive, 165, 168, 177, 179, and 200 to 206, inclusive.

And replace the following paragraph:

– When there are multiple SEI messages with a particular value of payloadType not equal to 133 that are associated with a particular AU or DU and apply to a particular OLS or layer, regardless of whether some or all of these SEI messages are scalable-nested, the SEI messages shall have the same SEI payload content.

with the following:

– When there are multiple SEI messages with a particular value of payloadType not equal to 4, 5, or 133 that are associated with a particular AU or DU and apply to a particular OLS or layer, regardless of whether some or all of these SEI messages are scalable-nested, the SEI messages shall have the same SEI payload content.

*D.5.2*

In subclause D.5.2, replace the following three paragraphs:

**dui\_dpb\_output\_du\_delay\_present\_flag** equal to 1 specifies the presence of the dui\_dpb\_output\_du\_delay syntax element in the DUI SEI message. dui\_dpb\_output\_du\_delay\_present\_flag equal to 0 specifies the absence of the dui\_dpb\_output\_du\_delay syntax element in the DUI SEI message. When not present, the value of dui\_dpb\_output\_du\_delay\_present\_flag is inferred to be equal to 0.

**dui\_dpb\_output\_du\_delay** is used to compute the DPB output time of the AU when DecodingUnitHrdFlag is equal to 1 and bp\_du\_dpb\_params\_in\_pic\_timing\_sei\_flag is equal to 0. It specifies how many sub clock ticks to wait after removal of the last DU in an AU from the CPB before the decoded pictures of the AU are output from the DPB. When not present, the value of dui\_dpb\_output\_du\_delay is inferred to be equal to pt\_dpb\_output\_du\_delay. The length of the syntax element dui\_dpb\_output\_du\_delay is given in bits by bp\_dpb\_output\_delay\_du\_length\_minus1 + 1.

It is a requirement of bitstream conformance that all DUI SEI messages that are associated with the same AU, apply to the same operation point, and have bp\_du\_dpb\_params\_in\_pic\_timing\_sei\_flag equal to 0 shall have the same value of dui\_dpb\_output\_du\_delay.

with the following:

**dui\_dpb\_output\_du\_delay\_present\_flag** equal to 1 specifies the presence of the dui\_dpb\_output\_du\_delay syntax element in the DUI SEI message. dui\_dpb\_output\_du\_delay\_present\_flag equal to 0 specifies the absence of the dui\_dpb\_output\_du\_delay syntax element in the DUI SEI message. When not present, the value of dui\_dpb\_output\_du\_delay\_present\_flag is inferred to be equal to 0. When bp\_du\_dpb\_params\_in\_pic\_timing\_sei\_flag is equal to 0, at least one DUI SEI message associated with the DUs of an AU shall have dui\_dpb\_output\_du\_delay\_present\_flag equal to 1.

**dui\_dpb\_output\_du\_delay** is used to compute the DPB output time of the AU when DecodingUnitHrdFlag is equal to 1 and bp\_du\_dpb\_params\_in\_pic\_timing\_sei\_flag is equal to 0. It specifies how many sub clock ticks to wait after removal of the last DU in an AU from the CPB before the decoded pictures of the AU are output from the DPB. When not present and when bp\_du\_dpb\_params\_in\_pic\_timing\_sei\_flag is equal to 0, the value of dui\_dpb\_output\_du\_delay is inferred to be equal to dui\_dpb\_output\_du\_delay from any DU belonging to the same AU for which dui\_dpb\_output\_du\_delay\_present\_flag is equal to 1. The length of the syntax element dui\_dpb\_output\_du\_delay is given in bits by bp\_dpb\_output\_delay\_du\_length\_minus1 + 1.

It is a requirement of bitstream conformance that all DUI SEI messages that are associated with the same AU, apply to the same operation point, and have bp\_du\_dpb\_params\_in\_pic\_timing\_sei\_flag equal to 0 and dui\_dpb\_output\_du\_delay\_present\_flag equal to 1 shall have the same value of dui\_dpb\_output\_du\_delay.

*D.8 and D.9*

Renumber subclauses D.8 and D.9 as D.10 and D.11, respectively.

*New subclauses D.8 and D.9*

Add subclauses D.8 and D.9 and subordinate subclauses as follows:

***D.8 SEI manifest SEI message***

**D.8.1 SEI manifest SEI message syntax**

|  |  |
| --- | --- |
| sei\_manifest( payloadSize ) { | **Descriptor** |
| **manifest\_num\_sei\_msg\_types** | u(16) |
| for( i = 0; i < manifest\_num\_sei\_msg\_types; i++ ) { |  |
| **manifest\_sei\_payload\_type**[ i ] | u(16) |
| **manifest\_sei\_description**[ i ] | u(8) |
| } |  |
| } |  |

**D.8.2 SEI manifest SEI message semantics**

The SEI manifest SEI message conveys information on SEI messages that are indicated as expected (i.e., likely) to be present or not present. Such information may include the following:

– The indication that certain types of SEI messages are expected (i.e., likely) to be present (although not guaranteed to be present) in the CVS.

– For each type of SEI message that is indicated as expected (i.e., likely) to be present in the CVS, the degree of expressed necessity of interpretation of the SEI messages of this type, as follows:

* The degree of necessity of interpretation of an SEI message type may be indicated as "necessary", "unnecessary", or "undetermined".
* An SEI message is indicated by the encoder (i.e., the content producer) as being "necessary" when the information conveyed by the SEI message is considered as necessary for interpretation by the decoder or receiving system in order to properly process the content and enable an adequate user experience; it does not mean that the bitstream is required to contain the SEI message in order to be a conforming bitstream. It is at the discretion of the encoder to determine which SEI messages are to be considered as necessary in a particular CVS. However, it is suggested that some SEI messages, such as the frame packing arrangement, segmented rectangular frame packing arrangement, and omnidirectional projection indication SEI messages, should typically be considered as necessary.

– The indication that certain types of SEI messages are expected (i.e., likely) not to be present (although not guaranteed not to be present) in the CVS.

NOTE – An example of such a usage of an SEI manifest SEI message is to express the expectation that there are no frame packing arrangement SEI messages or omnidirectional projection indication SEI messages in the CVS, and therefore that the rendering of the decoded video pictures for display purposes would not need any of the additional post-processing that is commonly associated with the interpretation of these SEI messages.

The content of an SEI manifest SEI message may, for example, be used by transport-layer or systems-layer processing elements to determine whether the CVS is suitable for delivery to a receiving and decoding system, based on whether the receiving system can properly process the CVS to enable an adequate user experience or whether the CVS satisfies the application needs.

When an SEI manifest SEI message is present in any access unit of a CVS, an SEI manifest SEI message shall be present in the first access unit of the CVS. The SEI manifest SEI message persists in decoding order from the current access unit until the end of the CVS. When there are multiple SEI manifest SEI messages present in a CVS, they shall have the same content.

An SEI NAL unit containing an SEI manifest SEI message shall not contain any other SEI messages other than SEI prefix indication SEI messages. When present in an SEI NAL unit, the SEI manifest SEI message shall be the first SEI message in the SEI NAL unit.

**manifest\_num\_sei\_msg\_types** specifies the number of types of SEI messages for which information is provided in the SEI manifest SEI message.

**manifest\_sei\_payload\_type**[ i ] indicates the payloadType value of the i-th type of SEI message for which information is provided in the SEI manifest SEI message. The values of manifest\_sei\_payload\_type[ m ] and manifest\_sei\_payload\_type[ n ] shall not be identical when m is not equal to n.

**manifest\_sei\_description**[ i ] provides information on SEI messages with payloadType equal to manifest\_sei\_payload\_type[ i ] as specified in Table 143.

**Table 143 – Interpretation of manifest\_sei\_description[ i ]**

|  |  |
| --- | --- |
| **Value** | **Description** |
| 0 | Indicates that there is no SEI message with payloadType equal to manifest\_sei\_payload\_type[ i ] expected to be present in the CVS. |
| 1 | Indicates that there are SEI messages with payloadType equal to manifest\_sei\_payload\_type[ i ] expected to be present in the CVS, and these SEI messages are considered as necessary. |
| 2 | Indicates that there are SEI messages with payloadType equal to manifest\_sei\_payload\_type[ i ] expected to be present in the CVS, and these SEI messages are considered as unnecessary. |
| 3 | Indicates that there are SEI messages with payloadType equal to manifest\_sei\_payload\_type[ i ] expected to be present in the CVS, and the necessity of these SEI messages is undetermined. |
| 4..255 | Reserved |

The value of manifest\_sei\_description[ i ] shall be in the range of 0 to 3, inclusive, in bitstreams conforming to this version of this Specification. Other values for manifest\_sei\_description[ i ] are reserved for future use by ITU-T | ISO/IEC. Decoders shall allow the value of manifest\_sei\_description[ i ] greater than or equal to 4 to appear in the syntax and shall ignore all information for payloadType equal to manifest\_sei\_payload\_type[ i ] signalled in the SEI manifest SEI message and shall ignore all SEI prefix indication SEI messages with prefix\_sei\_payload\_type equal to manifest\_sei\_payload\_type[ i ] when manifest\_sei\_description[ i ] is greater than or equal to 4.

***D.9 SEI prefix indication SEI message***

**D.9.2 SEI prefix indication SEI message syntax**

|  |  |
| --- | --- |
| sei\_prefix\_indication( payloadSize ) { | **Descriptor** |
| **prefix\_sei\_payload\_type** | u(16) |
| **num\_sei\_prefix\_indications\_minus1** | u(8) |
| for( i = 0; i <= num\_sei\_prefix\_indications\_minus1; i++ ) { |  |
| **num\_bits\_in\_prefix\_indication\_minus1**[ i ] | u(16) |
| for( j = 0; j <= num\_bits\_in\_prefix\_indication\_minus1[ i ]; j++ ) |  |
| **sei\_prefix\_data\_bit**[ i ][ j ] | u(1) |
| while( !byte\_aligned( ) ) |  |
| **byte\_alignment\_bit\_equal\_to\_one** /\* equal to 1 \*/ | f(1) |
| } |  |
| } |  |

**D.9.2 SEI prefix indication SEI message semantics**

The SEI prefix indication SEI message carries one or more SEI prefix indications for SEI messages of a particular value of payloadType. Each SEI prefix indication is a bit string that follows the SEI payload syntax of that value of payloadType and contains a number of complete syntax elements starting from the first syntax element in the SEI payload.

Each SEI prefix indication for an SEI message of a particular value of payloadType indicates that one or more SEI messages of this value of payloadType are expected (i.e., likely) to be present in the CVS and to start with the provided bit string. A starting bit string would typically contain only a true subset of an SEI payload of the type of SEI message indicated by the payloadType, may contain a complete SEI payload, and shall not contain more than a complete SEI payload. It is not prohibited for SEI messages of the indicated value of payloadType to be present that do not start with any of the indicated bit strings.

These SEI prefix indications should provide sufficient information for indicating what type of processing is needed or what type of content is included. The former (type of processing) indicates decoder-side processing capability, e.g., whether some type of frame unpacking is needed. The latter (type of content) indicates, for example, whether the bitstream contains subtitle captions in a particular language.

The content of an SEI prefix indication SEI message may, for example, be used by transport-layer or systems-layer processing elements to determine whether the CVS is suitable for delivery to a receiving and decoding system, based on whether the receiving system can properly process the CVS to enable an adequate user experience or whether the CVS satisfies the application needs (as determined in some manner by external means outside the scope of this Specification).

In one example, when the payloadType indicates the frame packing arrangement SEI message, an SEI prefix indication should include up to at least the syntax element frame\_packing\_arrangement\_type; and when the payloadType indicates the omnidirectional projection indication SEI message, an SEI prefix indication should include up to at least the syntax element projection\_type.

In another example, for user data registered SEI messages that are used to carry captioning information, an SEI prefix indication should include up to at least the language code; and for user data unregistered SEI messages extended for private use, an SEI prefix indication should include up to at least the UUID.

When an SEI prefix indication SEI message is present in any access unit of a CVS, an SEI prefix indication SEI message shall be present in the first access unit of the CVS. The SEI prefix indication SEI message persists in decoding order from the current access unit until the end of the CVS. When there are multiple SEI prefix indication SEI messages present in a CVS for a particular value of payloadType, they shall have the same content.

An SEI NAL unit containing an SEI prefix indication SEI message for a particular value of payloadType shall not contain any other SEI messages other than an SEI manifest SEI message and SEI prefix indication SEI messages for other values of payloadType.

**prefix\_sei\_payload\_type** indicates the payloadType value of the SEI messages for which one or more SEI prefix indications are provided in the SEI prefix indication SEI message. When an SEI manifest SEI message is also present for the CVS, the value of prefix\_sei\_payload\_type shall be equal to one of the manifest\_sei\_payload\_type[ m ] values for which manifest\_sei\_description[ m ] is equal to 1 to 3, inclusive, as indicated by an SEI manifest SEI message that applies to the CVS.

**num\_sei\_prefix\_indications\_minus1** plus 1 specifies the number of SEI prefix indications.

**num\_bits\_in\_prefix\_indication\_minus1**[ i ] plus 1 specifies the number of bits in the i-th SEI prefix indication.

**sei\_prefix\_data\_bit**[ i ][ j ] specifies the j-th bit of the i-th SEI prefix indication.

The bits sei\_prefix\_data\_bit[ i ][ j ] for j ranging from 0 to num\_bits\_in\_prefix\_indication\_minus1[ i ], inclusive, follow the syntax of the SEI payload with payloadType equal to prefix\_sei\_payload\_type, and contain a number of complete syntax elements starting from the first syntax element in the SEI payload syntax, and may or may not contain all the syntax elements in the SEI payload syntax. The last bit of these bits (i.e., the bit sei\_prefix\_data\_bit[ i ][ num\_bits\_in\_prefix\_indication\_minus1[ i ] ]) shall be the last bit of a syntax element in the SEI payload syntax, unless it is a bit within an itu\_t\_t35\_payload\_byte or user\_data\_payload\_byte.

NOTE – The exception for itu\_t\_t35\_payload\_byte and user\_data\_payload\_byte is provided because these syntax elements may contain externally-specified syntax elements, and the determination of the boundaries of such externally-specified syntax elements is a matter outside the scope of this Specification.

**byte\_alignment\_bit\_equal\_to\_one** shall be equal to 1.

*New subclauses D.11.7 and D.11.8*

*Add subclauses D.11.7 and D.11.8 as follows:*

**D.11.7 Use of the annotated regions SEI message**

For purposes of interpretation of the annotated regions SEI message, the following variables are specified:

– CroppedWidth is set equal to pps\_pic\_width\_in\_luma\_samples − SubWidthC \* ( pps\_conf\_win\_right\_offset + pps\_conf\_win\_left\_offset ).

– CroppedHeight is set equal to pps\_pic\_height\_in\_luma\_samples − SubHeightC \* ( pps\_conf\_win\_bottom\_offset + pps\_conf\_win\_top\_offset ).

– ConfWinLeftOffset is set equal to pps\_conf\_win\_left\_offset.

– ConfWinTopOffset is set equal to pps\_conf\_win\_top\_offset.

**D.11.8 Use of the extended dependent random access point (EDRAP) indication SEI message**

A picture that is associated with an EDRAP indication SEI message is referred to as an EDRAP picture.

The following constraints apply to an EDRAP picture:

– The VCL NAL units of the EDRAP picture shall have nal\_unit\_type equal to TRAIL\_NUT.

– The EDRAP picture shall have TemporalId equal to 0.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_