ETSI GS CCM 001 V1.1.1 (2017-02)

Compound Content Management Specification

Reference

DGS/CCM-001

Keywords

CCM system, HDR, HDR composer, HDR Metadata Embedder, UHDTV, WCG

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Contents

Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (https://ipr.etsi.org/).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Foreword

This Group Specification (GS) has been produced by ETSI Industry Specification Group (ISG) on intelligent Compound Content Management (CCM).

Modal verbs terminology

In the present document "**shall**", "**shall not**", "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the ETSI Drafting Rules (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

Introduction

The present document defines and specifies the additional functionality required in consumer devices to enable the accurate recreation of both today's television signal, i.e. SDR signal, and the HDR/WCG signal from content created using the relevant production techniques standardized in SMPTE.

The requirement not to compromise the quality of the legacy signal using today's displays but yet to provide the highest quality HDR/WCG signal for a new generation of televisions is onerous and requires analysis to derive an optimal solution.

However, given the requirement to achieve backwards compatibility with existing receivers, some additional technology will be required in the next generation of HDR/WCG receivers to accurately recreate the HDR/WCG signal.

The full HDR system as shown in the figure 1 describes an end-to-end technology suite that enables the creation and distribution of content mastered with a high dynamic range and wide colour gamut. Display management may be used to more accurately match the capability of a given television to the HDR/WCG signal source. This creates seamless video experience for the viewer. The display management process may be guided by metadata generated in the HDR encoder.

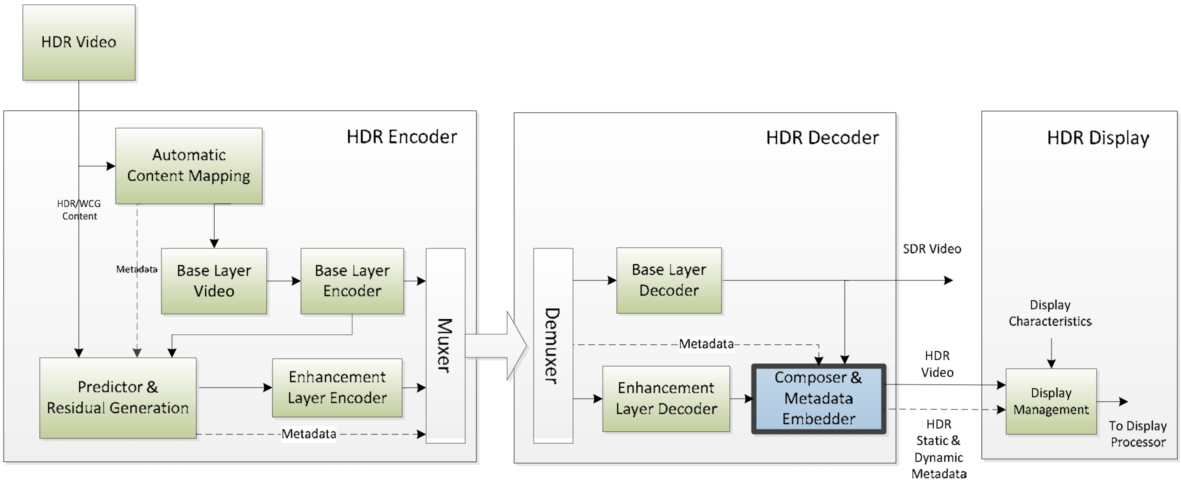


Figure 1: Overview of a generic HDR system and the additional HDR Compound Content Management functionality

1 Scope

The present document specifies the additional functionality required in Consumer devices for the reconstruction of an HDR/WCG signal using the "intelligent Compound Content Management" processing defined. Additionally it defines a method whereby this HDR/WCG signal and its associated metadata is transported over existing commonly used consumer baseband interfaces.

This process is one element of an end-to-end HDR video system. The definition of the overall end-to-end HDR video system is out of scope of the present document.

2 References

2.1 Normative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

Referenced documents which are not found to be publicly available in the expected location might be found at http://docbox.etsi.org/Reference.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are necessary for the application of the present document.

[1] SMPTE ST 2084:2014: "Dynamic Range Electro-Optical Transfer Function of Mastering Reference Displays".

[2] SMPTE ST 2086:2014: "Mastering Display Color Volume Metadata Supporting High Luminance And Wide Color Gamut Images".

[3] SMPTE ST 2094-1:2016: "Mapping VC-3 Coding Units into the MXF Generic Container".

[4] SMPTE ST 2094-10:2016: "Dynamic Metadata for Color Volume Transform - Application #1".

[5] Recommendation ITU-R BT.1886-0: "Reference electro-optical transfer function for flat panel displays used in HDTV studio production".

[6] Recommendation ITU-R BT.2100-0: "Image parameter values for high dynamic range television for use in production and international programme exchange".

[7] Recommendation ITU-R BT.709-6: "Parameter values for the HDTV standards for production and international programme exchange".

[8] Recommendation ITU-R BT.2020-2: "Parameter values for ultra-high definition television systems for production and international programme exchange".

[9] Recommendation ITU-T H.265 | ISO/IEC 23008-2: "Information technology - High efficiency coding and media delivery in heterogeneous environments - Part 2: High efficiency video coding".

[10] ISO/IEC 13818-1: "Information technology - Generic coding of moving pictures and associated audio information - Part 1: Systems".

2.2 Informative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are not necessary for the application of the present document but they assist the user with regard to a particular subject area.

[i.1] SMPTE 319M-2000: "For Television - Transporting MPEG-2 Recoding Information through 4:2:2 Component Digital Interfaces".

[i.2] Recommendation ITU-R BT.2246-2: "The present state of ultra-high definition television".

3 Definitions and abbreviations

3.1 Definitions

For the purpose of the present document, the following terms and definitions apply:

base layer: carries the video signal of the compound content which is used for the prediction of the HDR/WCG signal

block: MxN (M-column by N-row) array of samples

Compound Content Management (CCM): process deriving a picture with a high dynamic range

NOTE: The high dynamic range picture is derived from a BL, metadata and potentially an EL. This method can be part of the encoding / decoding process or not.

composer: component deriving a picture with a high dynamic range

NOTE: The composer is controlled by metadata and combines two baseband pictures derived from decodable compressed streams.

display management: process adapting the signal to the dynamic range and the colour gamut of the target display

dynamic range: ratio of the maximum light intensity to the minimum light intensity [i.2]

NOTE: In digital cameras the dynamic range is normally measured in terms of stops, which describe the total light range by power of 2.

enhancement layer: carries the additional information required to accurately recreate the HDR/WCG signal

high dynamic range: typically a dynamic range of more than 10 stops is referred to as high dynamic range

inverse mapping: colour mapping of the BL signal to form a predictor for the EL signal

inverse quantization: non-linearly scales the EL signal from a code word with a bit depth equal to EL_bit_depth to a code word with a higher bit depth

residual: EL signal which is the difference of the inverse mapped BL signal and the high dynamic range output signal

standard dynamic range: typically a dynamic range of up to 10 stops is referred to as standard dynamic range

3.2 Symbols

3.2.1 Arithmetic operators

For the purposes of the present document, the following arithmetic operators apply:

+	Addition
-	Subtraction (as a two-argument operator) or negation (as a unary prefix operator)
*	Multiplication, including matrix multiplication
÷	Used to denote division in mathematical equations where no truncation or rounding is intended.
/	Integer division with truncation of the result toward zero. For example, 7/4 and - 7/- 4 are truncated to 1 and - 7/4 and 7/- 4 are truncated to - 1.

3.2.2 Bit-wise operators

For the purposes of the present document, the following bit-wise operators apply:

x >> y	Arithmetic right shift of a two's complement integer representation of x by y binary digits. This function is defined only for non-negative integer values y. Bits shifted into the most significant bits (MSBs) as a result of the right shift have a value equal to the MSB of x prior to the shift operation.
x << y	Arithmetic left shift of a two's complement integer representation of x by y binary digits. This function is defined only for non-negative integer values y. Bits shifted into the least significant bits as a result of the left shift have a value equal to 0.

3.2.3 Relational operators

For the purposes of the present document, the following relational operators apply:

>	Greater than
>=	Greater than or equal to
<	Less than
<=	Less than or equal to
==	Equal to
!=	Not equal to

3.2.4 Assignment operators

For the purposes of the present document, the following assignment operators apply:

=	Assignment operator
++	Increment, i.e. *x* is equivalent to $x = x + 1$; when used in an array index, evaluates to the value of the variable prior to the increment operation.
--	Decrement, i.e. *x*- - is equivalent to $x = x - 1$; when used in an array index, evaluates to the value of the variable prior to the decrement operation.
+=	Increment by amount specified, i.e. x += 4 is equivalent to x = x + 4, and x += (-4) is equivalent to x = x + (-4).
-=	Decrement by amount specified, i.e. x -= 4 is equivalent to x = x - 4, and x -= (-4) is equivalent to x = x - (-4).

3.2.5 Mathematical functions

For the purposes of the present document, the following mathematical functions apply:

$$\mathrm{Abs}(x) = \begin{cases} x & ; \quad x \geq 0 \\ -x & ; \quad x < 0 \end{cases}$$

$$\mathrm{Clip3}(x, y, z) = \begin{cases} x & ; \quad z < x \\ y & ; \quad z > y \\ z & ; \quad otherwise \end{cases}$$

Floor(x) the largest integer less than or equal to x.

$$\text{Max(x,y)} = \begin{cases} x & ; \quad x \geq y \\ y & ; \quad x < y \end{cases}$$

Round(x)=Sign(x)*Floor(Abs(x)+0,5)

$$\text{Sign(x)} = \begin{cases} 1 & ; \quad \text{x} > 0 \\ 0 & ; \quad x = 0 \\ -1 & ; \quad x < 0 \end{cases}$$

$$\text{Short(x)} = \begin{cases} x & ; \quad x < 2^{15} \\ x - 2^{16} & ; \quad x \geq 2^{15} \end{cases}$$, where x is a 16 bit integer value.

3.2.6 Order of operation precedence

When order of precedence in an expression is not indicated explicitly by use of parentheses, operations are evaluated sequentially from left to right.

3.3 Abbreviations

For the purposes of the present document, the following abbreviations apply:

BL Base Layer
CCM Compound Content Management
CM Composing Metadata
CRC Cyclic Redundancy Check
DM Display Management
DPB Decoded Picture Buffer
EL Enhancement Layer
EOS End Of stream
EOTF Electro-Optical Transfer Function
HDR High Dynamic Range
ITU-R International Telecommunications Union - Radiocommunications standardization sector
ITU-T International Telecommunications Union - Telecommunications standardization sector
LSB Least Significant Bit
MMR Multivariate Multiple Regression
NLQ Non-Linear Quantization
PPS Picture Parameter Set
PQ Perceptual Quantizer

NOTE: As defined in SMPTE ST 2084 [1].

SDR Standard Dynamic Range
SHVC Scalable High Efficiency Video Codec
SMPTE Society of Motion Pictures and Television Engineers
ST STandard
UHD Ultra High Definition
UHDTV Ultra High Definition Television
WCG Wide Colour Gamut

4 CCM system model

The CCM system consists of two main components, the Composer and the HDR Display Management Metadata Embedder - see figure 2.

The composer reconstructs the HDR signal from the base layer (BL) image, the associated enhancement layer (EL) image and the related metadata information. First the EL signal is inverse quantized by applying inverse quantization coefficients delivered in the composing metadata (CM). Based on the BL signal and the related mapping coefficients of the CM, the predicted HDR signal is calculated. Finally the inverse quantized residual is added to the predicted HDR signal in order to recreate the HDR signal. If the characteristics of the BL input signal are different from the ones defined in SMPTE ST 2084 [1], a conversion of the recreated HDR signal according to SMPTE ST 2084 is performed to achieve the HDR output signal. The composer is defined in clause 5.

NOTE 1: Implementers can use different resolutions for the EL or the BL input pictures, thus requiring resampling of the input data before HDR composition in order to adjust either image resolution accordingly. Resampling is not a normative part of the present document and is assumed to be performed prior to the HDR reconstruction step defined in the present document. A reference implementation for the spatial resampling is provided in annex B.

Additional HDR, named HDR display management (DM) metadata, may be present at the input of the CCM system. The HDR DM metadata combines SMPTE ST 2086 static metadata [2] and SMPTE ST 2094-1/10 dynamic metadata [3] and [4], and is used to maintain the artistic intent when the content is mapped to the display capabilities. This HDR DM metadata should be passed-through to the output of the composer or the metadata embedder inserts the HDR DM metadata into the HDR picture for transport over baseband interfaces. The HDR DM metadata embedder is defined in clause 6.

NOTE 2: The composer metadata and functionality is independent of the type of SMPTE ST 2094 metadata carried.

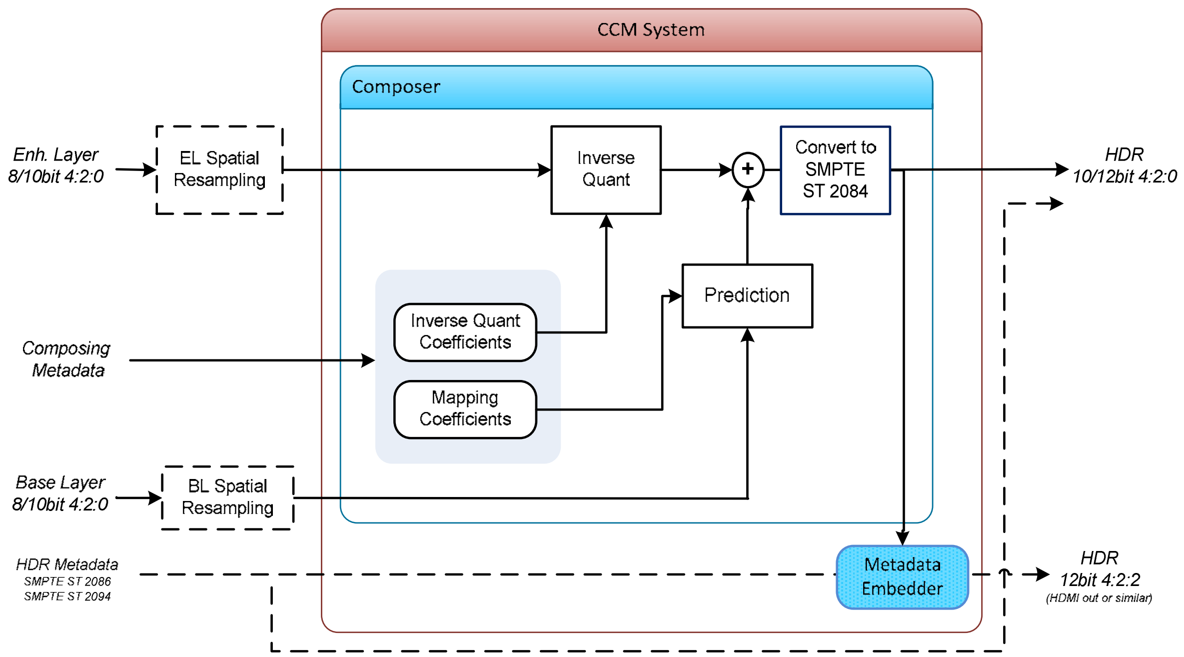


Figure 2: HDR Compound Content system high-level block diagram

5 Composer definition

5.1 Introduction

This clause defines the guidelines for combining the BL with the EL information and the composer metadata to recreate the HDR picture.

Clause 5.2 defines the input and output picture formats of the composer.

Clause 5.3 defines the CM, which is needed for the recreation of the HDR output picture.

Clause 5.4 defines the decoding process, which is performed to recreate the HDR output picture. Clause 5.4.2 applies to the BL mapping process. Clause 5.4.3 defines the EL processing and the reconstruction of the HDR signal.

Clause 5.5 defines the conversion of the recreated HDR picture to the transfer characteristic as defined in SMPTE ST 2084 [1].

5.2 Input and output picture format

5.2.1 General

The composer defines two input pictures, named BL and EL picture, and one output picture, named HDR picture.

Input and ouput pictures consist of three sample arrays of one of the following colour representation methods:

- Luma and two Chroma (YCbCr).
- Intensity and two Chroma (ICtCp) as defined in Recommendation ITU-R BT.2100-0 [6].

The dimension of a luma array is specified as (PicWidth)x(PicHeight) and the dimension of a chroma array is specified as (PicWidthC)x(PicHeightC). PicWidth specifies the width of each luma array in units of luma samples. PicHeight specifies the height of each luma array in units of luma samples. PicWidthC specifies the width of each chroma array in units of chroma samples. PicHeightC specifies the height of each chroma array in units of chroma samples.

NOTE: For convenience of notation and terminology in the present document, the variables and terms associated with these arrays are referred to as luma (or Y) and chroma, where the two chroma arrays are referred to as Cb and Cr; regardless of the actual colour representation method in use.

5.2.2 Input picture format

BL picture shall obey the following constraints:

a) Transfer characteristics of the BL picture shall be as defined in SMPTE ST 2084 [1] or as defined in Recommendation ITU-R BT.1886-0 [5].

b) Colour primaries shall be as defined in Recommendation ITU-R BT.2100-0 [6], Recommendation ITU-R BT.709-6 [7] or Recommendation ITU-R BT.2020-2 [8].

BL picture and EL picture shall obey the following constraints:

a) BL picture and EL picture shall have the same vertical and horizontal resolution (see also note 1 in clause 4).

b) For BL and EL picture, each of the two chroma arrays shall have half the height and half the width of the luma array.

c) BL and EL shall have the same framerate.

d) Samples in each array of BL picture and EL picture shall have the same bit depth.

e) The luma sample phase of the BL and EL shall be identical.

f) The number of bits necessary for the representation of each of the samples in the luma and chroma array shall be equal to 8 or 10.

g) Locations of chroma samples relative to corresponding luma samples shall be the same for BL picture and EL picture.

5.2.3 Output picture format

Output pictures have the same size and same colour representation method as the BL pictures.

Output pictures shall obey to the following constraints:

a) Transfer characteristic of the output picture shall be as defined in SMPTE ST 2084 [1].

b) The number of bits necessary for the representation of each of the samples in the luma and chroma array shall be equal to 10 or 12.

5.3 Composing metadata (CM) definitions

5.3.1 Introduction

Clause 5.3 defines the composing metadata parameters which are needed to recreate the HDR output picture. These parameters shall be derived from the parsing process of the BL bitstream, the EL bitstream or both, or by other means.

Clause 5.3.2 defines the composing metadata parameters which are needed for the composing process as described in clause 5.4.

Clause 5.3.3 defines the composing metadata parameters which are needed for the inverse mapping process as described in clause 5.4.2.

Clause 5.3.4 defines the composing metadata parameters which are needed for the non-linear inverse quantization process as described in clause 5.4.3.2.

5.3.2 CM header parameter definitions

ccm_profile specifies the profile to which the CM process conforms as specified in annex A. CM Metadata shall not contain values of ccm_profile other than those specified in Annex A. Other values of ccm_profile are reserved for future use.

ccm_level indicates a level to which the CCM process conforms as specified in annex A. CCM Metadata shall not contain values of ccm_level other than those specified in Annex A. Other values of cm_level are reserved for future use.

coefficient_log2_denom specifies the number of fractional bits for the EL decoding coefficients. Coefficient_log2_denom shall be in the range of (EL_bit_depth + 5) to 32, inclusive.

BL_bit_depth_minus8 is used to derive the bit depth of BL signal BL_bit_depth = BL_bit_depth_minus8 + 8. The value of BL_bit_depth_minus8 shall be equal to 0 or 2.

EL_bit_depth_minus8 is used to derive the bit depth of EL signal EL_bit_depth = EL_bit_depth_minus8 + 8. The value of EL_bit_depth_minus8 shall be equal to 0 or 2.

hdr_bit_depth_minus8 is used to derive the bit depth of the reconstructed HDR signal. hdr_bit_depth = hdr_bit_depth_minus8 + 8. The value of hdr_bit_depth_minus8 shall be equal to 2 or 4.

disable_residual_flag specifies whether the EL is added to inverse-mapped BL for HDR reconstruction. If disable_residual_flag is equal to 1, the EL signal will be ignored. If disable_residual_flag is equal to 0, the EL signal is inverse quantized and added to the inverse-mapped BL signal to reconstruct the HDR signal.

If the EL input picture is not present, the value of disable_residual_flag derived from the CCM input data shall be ignored and it shall be set to 1.

num_pivots_minus2[cmp] plus 2 specifies the number of pivot points used to perform piecewise mapping process for the colour component cmp. The value of num_pivots_minus2 shall be in the range of 0 to 15, inclusive.

NOTE: For example, if one mapping model is applied for the entire BL signal's dynamic range, then the value of num_pivots_minus2 is equal to 0.

pred_pivot_value[cmp][i] is used to derive the value pivot_value[cmp][i] of the (i)-th pivot point for the colour component cmp. These pivot points collectively define the boundaries of the piecewise mapping process. The values of pivot_value[cmp][i] is derived by invoking the assign_pivot_values process, with the array pred_pivot_value[ ][ ] as input and the array pivot_value[ ][ ] as output. The value of i shall be in the range of 0 to (num_pivots_minus2 + 1), inclusive. The number of bits used for the representation of the pred_pivot_value[cmp][i] syntax element is BL_bit_depth bits.

```
Assign_pivot_values(pred_pivot_value[ ][ ],num_pivots_minus2 )
{
    pivot_value[ 3 ][ num_pivots_minus_2+2 ];

    for( cmp = 0; cmp < 3; cmp ++ ) {
        pivot_value[ cmp ][ 0 ] = pred_pivot_value[ cmp ][ 0 ];
        for( i = 1; i < num_pivots_minus2[ cmp ] + 2; i ++)
            pivot_value[ cmp ][ i ] = pivot_value[ cmp ][ i - 1 ] + pred_pivot_value[ cmp ][ i ];
    }
    return pivot_value;
}
```

max_display_mastering_luminance specifies the nominal maximum display luminance value of the mastering display in units of 1 cd/m^2. The maximum value of the syntax element max_display_mastering_luminace shall not be greater than 10 000 cd/m^2.

min_display_mastering_luminance specifies the nominal minimum display luminance value of the mastering display in units of 0,0001 cd/m^2. The value of min_display_mastering_luminance shall be less than the value of max_display_mastering_luminance.

5.3.3 CM mapping parameter definitions

5.3.3.1 CM general mapping parameter definitions

mapping_idc[cmp][pivot_idx] specifies the mapping method used for the colour component cmp, and the component value identified by pivot_idx. The value of mapping_idc[cmp][pivot_idx] with the index cmp equal to 0 shall be set to 0. The value of mapping_idc[cmp][pivot_idx] with the index cmp equal to 1 or 2 shall be set to 0 or 1. The values of mapping_idc[cmp][pivot_idx] and corresponding mapping method are specified in table 1.

Table 1: Definition of inter-layer mapping methods

mapping_idc	Mapping name	Mapping method
0	MAPPING_POLYNOMIAL	Nth order polynomial mapping (N>=1)
1	MAPPING_MMR	MMR mapping

5.3.3.2 CM polynomial mapping parameter definitions

poly_order_minus1[cmp][pivot_idx] plus 1 specifies the order of the polynomial mapping method identified by mapping_idc[cmp][pivot_idx]. The value of poly_order_minus1[cmp][pivot_idx] shall be in the range of 0 to 1, inclusive.

poly_coef_int[cmp][pivot_idx][i] specifies the integer portion of fp_poly_coef[cmp][pivot_idx][i]. The value of poly_coeff_int[cmp][pivot_idx][i] shall be in the range of -64 to 63, inclusive.

NOTE: fp_poly_coef[cmp][pivot_idx][i] is used to derive the value of the i-th order polynomial coefficient associated with mapping_idc[cmp][pivot_idx].

poly_coef[cmp][pivot_idx][i] specifies the fractional portion of fp_poly_coef[cmp][pivot_idx][i]. The length of the poly_coef[cmp][pivot_idx][i] syntax element is coefficient_log2_denom bits. The value of the i-th order polynomial coefficient associated with mapping_idc[cmp][pivot_idx] is derived as follows:

fp_poly_coef[cmp][pivot_idx][i] = (poly_coef_int[cmp][pivot_idx][i] << coefficient_log2_denom) + poly_coef[cmp][pivot_idx][i].

5.3.3.3 CM MMR mapping parameter definitions

mmr_order_minus1[cmp][pivot_idx] plus 1 specifies the order of the MMR mapping method identified by mapping_idc[cmp][pivot_idx]. The value of mmr_order_minus1[cmp][pivot_idx] shall be in the range of 0 to 2, inclusive.

mmr_constant_int[cmp][pivot_idx] specifies the integer portion of fp_mmr_constant[cmp][pivot_idx]. The value of mmr_constant_int[cmp][pivot_idx] shall be in the range of -65 536 to 65 535, inclusive.

NOTE 1: fp_mmr_constant[cmp][pivot_idx] is used to derive the value of the MMR constant coefficient associated with mapping_idc[cmp][pivot_idx].

mmr_constant[cmp][pivot_idx] specifies the fractional portion of fp_mmr_constant[cmp][pivot_idx]. The length of the mmr_constant[cmp][pivot_idx] syntax element is coefficient_log2_denom bits. The value of MMR constant coefficient associated with mapping_idc[cmp][pivot_idx] is derived as follows:

fp_mmr_constant[cmp][pivot_idx] = (mmr_constant_int[cmp][pivot_idx] << coefficient_log2_denom) + mmr_constant[cmp][pivot_idx].

mmr_coef_int[cmp][pivot_idx][i][j] specifies the integer portion of fp_mmr_coef[cmp][pivot_idx][i] [j].]. The value of mmr_coeff_int[cmp][pivot_idx][i][j] shall be in the range of -65 536 to 65 535, inclusive.

NOTE 2: fp_mmr_coef[cmp][pivot_idx][i][j] is used to derive the value of the j-th MMR coefficient at the i-th order associated with mapping_idc[cmp][pivot_idx].

mmr_coef[cmp][pivot_idx][i][j] specifies the fractional portion of fp_mmr_coef[cmp][pivot_idx][i][j]. The length of the mmr_coef[cmp][pivot_idx][i][j] syntax element is coefficient_log2_denom bits. The value of the j-th MMR coefficient at the i-th order associated with mapping_idc[cmp][pivot_idx] is derived as follows:

fp_mmr_coef[cmp][pivot_idx][i][j] = (mmr_coef_int[cmp][pivot_idx][i][j] << coefficient_log2_denom) + mmr_coef[cmp][pivot_idx][i][j].

5.3.4 CM quantization parameter definitions

nlq_offset[cmp] specifies the non-linear quantization offset coefficient associated with the component cmp. The value of nlq_offset[cmp] is in the range of 0 to $2^{EL\_bit\_depth}$ - 1, inclusive.

hdr_in_max_int[cmp] specifies the integer portion of fp_hdr_in_max[cmp]. The value of hdr_in_max_int[cmp] shall be in the range of 0 to 1, inclusive.

NOTE 1: fp_hdr_in_max[cmp] is used to derive the value of non-linear quantization maximum EL signal value coefficient associated with the component cmp.

hdr_in_max[cmp] specifies the fractional portion of fp_hdr_in_max[cmp]. The length of the hdr_in_max[cmp] syntax element is coefficient_log2_denom bits. The value of non-linear quantization maximum EL signal value coefficient associated with the component cmp is derived as follows:

fp_hdr_in_max[cmp] = (hdr_in_max_int[cmp] << coefficient_log2_denom) + hdr_in_max[cmp].

linear_deadzone_slope_int[cmp] specifies the integer portion of fp_linear_deadzone_slope[cmp]. The value of linear_deadzone_slope_int[cmp] shall be in the range of 0 to 1, inclusive.

NOTE 2: fp_linear_deadzone_slope[cmp] is used to derive the value of linear dead zone slope coefficient associated with the component cmp.

linear_deadzone_slope[cmp] specifies the fractional portion of fp_linear_deadzone_slope[cmp]. The length of the linear_deadzone_slope[pivot_idx][cmp] syntax element is coefficient_log2_denom bits. The value of linear dead zone slope coefficient associated with the component cmp is derived as follows:

fp_linear_deadzone_slope[cmp] = (linear_deadzone_slope_int[cmp] << coefficient_log2_denom) + linear_deadzone_slope[cmp].

linear_deadzone_threshold_int[cmp] specifies the integer portion of fp_linear_deadzone_threshold[cmp]. The value of linear_deadzone_threshold_int[cmp] shall be in the range of 0 to 1, inclusive.

NOTE 3: fp_linear_deadzone_threshold[cmp] is used to derive the value of linear dead zone threshold coefficient associated with the component cmp.

linear_deadzone_threshold[cmp] specifies the fractional portion of fp_linear_deadzone_threshold[cmp]. The length of the linear_deadzone_threshold[pivot_idx][cmp] syntax element is coefficient_log2_denom bits. The value of linear dead zone threshold coefficient associated with the component cmp is derived as follows:

fp_linear_deadzone_threshold[cmp] = (linear_deadzone_threshold_int[cmp] << coefficient_log2_denom) + linear_deadzone_threshold[cmp].

5.4 Composing process

5.4.1 Introduction

The inverse mapping process depicted in figure 3 applies to the input BL samples. The output samples form the prediction signal, which is needed for the reconstruction of the HDR signal.

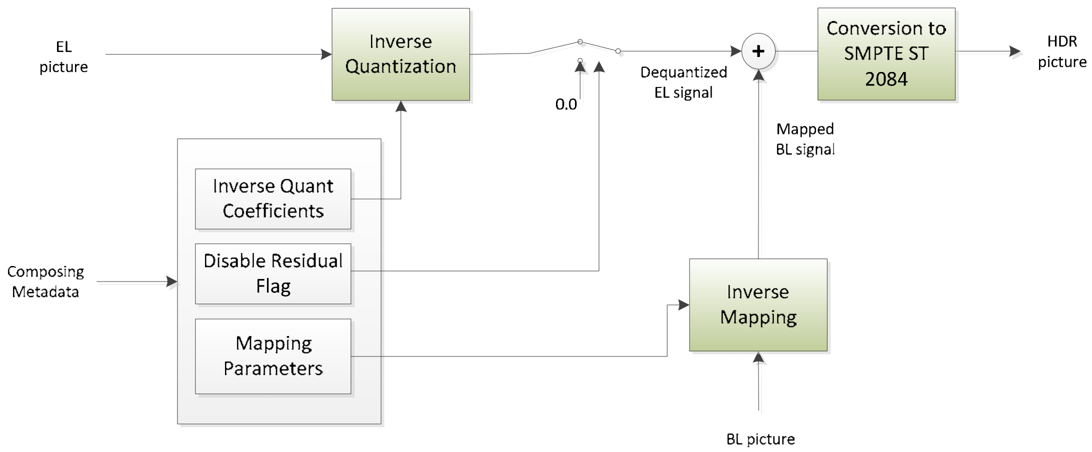


Figure 3: Block diagram of the reconstruction of the HDR signal

The BL signal is mapped to the EL signal domain using pre-selected mapping coefficients signalled in the CM. The prediction signal is added to the inverse quantized EL signal to reconstruct the HDR signal as shown in figure 3.

NOTE: Inverse mapping is done by analysing the correlations of the corresponding components between the BL signal and the original HDR signal by colour mapping analysis in the HDR encoder.

5.4.2 Inverse mapping of the base layer signal

5.4.2.1 General

The inverse mapping process for each prediction method is described in clause 5.4.2.3.

The input of this process is a BL picture; the output is the mapped BL picture, which is needed for the reconstruction of the HDR picture. The bit depth of the input samples is equal to BL_bit_depth; the bit depth of the unsigned samples of the mapped BL picture shall be equal to 16 bit.

Inputs to this process are:

- A BL picture S with its three sample arrays S_{cmp} (cmp=0,1,2) with elements $s_{i,j,cmp}$, which is a (PicWidth)x(PicHeight) luma block of the BL picture, when cmp is equal to 0, and a (PicWidthC)x(PicHeightC) chroma block of the BL picture, when cmp is not equal to 0.

- The composing metadata as defined in clause 5.3.

Output of this process is:

- A mapped BL picture V with its three sample arrays V_{cmp} (cmp=0,1,2) with elements $v_{i,j,cmp}$, which is a (PicWidth)x(PicHeight) luma block of the mapped BL picture, when cmp is equal to 0, and a (PicWidthC)x(PicHeightC) chroma block of the mapped BL picture, when cmp is not equal to 0.

The inverse mapping is defined by the two steps for each sample $s=s_{i,j,cmp}$.

1) The process described in clause 5.4.2.2 is invoked to derive a pivot index pivot_idx.

2) The process described in clause 5.4.2.3 is invoked to compute the sample $v=v_{i,j,cmp}$.

5.4.2.2 Derivation of the pivot index

The mapping process for the samples of the BL picture is defined piece-wise depending on the value of the corresponding BL input sample. This clause describes how the pivot index pivot_idx is determined for a sample from the BL input picture.

The following process is invoked for each sample of each of the three components indexed by cmp = 0,1,2.

Inputs to this process are:

- An element s = $s_{i,j}$ of the sample array S_{cmp} ,which is the luma component of the BL input picture, if cmp is equal to 0, or a chroma component of the BL input picture, if cmp is not equal to 0.

- The variables num_pivots_minus2[cmp] and the array pivot_value[cmp][num_pivots_minus2[cmp]+2].

Output of this process is:

- The pivot index pivot_idx=pivot_idx[i][j][cmp].

The value of pivot_idx is derived as specified in the following pseudo-code:

```
pivot_idx = num_pivots_minus2[cmp]+1;

for(idx = 0; idx < num_pivots_minus2[ cmp ] + 1; idx++) {
    if(s < pivot_value[cmp][idx + 1]) {
        pivot_idx = idx;
        break;
    }
}
```

5.4.2.3 BL input sample mapping

5.4.2.3.1 General

For the reconstruction of the HDR picture each BL picture is mapped to the HDR domain. The mapping shall be derived individually for each sample of each of the three colour components.

First the value of mapping_idc is read depending on the pivot_idx derived for the element $s_{i,j}$ of the sample array S_{cmp}. Then, depending on the value of mapping_idc the process described in clauses 5.4.2.3.2 and 5.4.2.3.3 are invoked.

If mapping_idc is equal to 0, the mapping method specified in clause 5.4.2.3.2 is invoked with coefficients indexed by cmp and pivot_idx.

If mapping_idc is equal to 1, the mapping method specified in clause 5.4.2.3.3 is invoked with coefficients indexed by cmp and pivot_idx.

5.4.2.3.2 Polynomial mapping

The polynomial based mapping process for a sample of the colour components indexed by cmp is derived as follows.

Inputs to this process are:

- The sample $s=s_{i,j}$ of the sample array S_{cmp} ,which is the luma component of the BL input picture, if cmp is equal to 0, or a chroma component of the BL input picture, if cmp is not equal to 0.
- The variables num_pivots_minus2= num_pivots_minus2[cmp].
- The array pivot_value [num_pivots_minus2+2]= array pivot_value[cmp][num_pivots_minus2+2].
- The pivot index pivot_idx derived as described in clause 5.4.2.2.
- poly_order _minus1=poly_order_minus1[cmp][pivot_idx].
- poly_coef[i]=fp_poly_coef[cmp][pivot_idx][i] with i=0,.., poly_order _minus1+1.

Output of this process is:

- The sample $v= v_{i,j}$ of the sample array V_{cmp} ,which is the luma component of the mapped BL picture, if cmp is equal to 0, or a chroma component of the mapped BL picture, if cmp is not equal to 0.

The following pseudo-code specifies how the value of v is derived for each sample:

```
if(s < pivot_value[cmp][0])
    s = pivot_value[cmp][0];
if(s > pivot_value[cmp][num_pivots_minus2[cmp]+1])
    s = pivot_value[cmp][num_pivots_minus2[cmp]+1];

// compute polynom at s in fixed point arithmetic
ss=1;
shift=20;  // 2*(maximum BL_bit_depth)
vv=0;
for(i=0;i<=poly_order_minus1+1;i++)
{
    vv += poly_coef[i] * (ss<<shift);
    ss *= s;
    shift -= BL_bit_depth;
}
vv = (vv < 0)? 0 : vv;
v = vv >> (4+coefficient_log2_denom);
v =(v > 0xffff)? 0xffff : v;
```

5.4.2.3.3 MMR mapping

The MMR based mapping process for a sample of the color components indexed by cmp with values equal to 1 or 2 is derived as follows.

First the BL luma (colour component 0) shall be down-sampled, to align with chroma samples, before applying the MMR mapping process.

The input to the down-sampling process is a sample array S_0 with elements $s_{i,j,0}$, which is a (PicWidth)x(PicHeight) luma block of the BL picture.

The output is a sample array S_0^* with the elements $s_{i,j,0}^*$, which is a (PicWidthC)x(PicHeightC) down-sampled luma block of the BL picture. The value of each sample $s_{i,j,0}^*$ is defined as follows:

$$s_{i,j}^* = \left(\left(\left(s_{2i-1,2j} + 2 * s_{2i,2j} + s_{2i+1,2j} + 2\right) \gg 2 + \left(s_{2i-1,2j+1} + 2 * s_{2i,2j+1} + s_{2i+1,2j+1} + 2\right) \gg 2\right) + 1\right) \gg 1$$

If the sample used is out of picture boundary, it is set to the value of the sample at the picture boundary. That is, image edges are handled by pixel replication.

Then the MMR based mapping is applied as follows.

The inputs to this process are:

- The samples $sX=sX_{i,j}$ (X=0,1,2) of the sample array S_X, where S_0 is the down-sampled luma component S_0^* of the BL input picture and S_1 and S_2 are the chroma components of the BL input picture.
- Let s=sN, where N=cmp is the color component index.
- The variables num_pivots_minus2= num_pivots_minus2[cmp].
- The array pivot_value[cmp][num_pivots_minus2+2].
- The pivot index pivot_idx derived as described in clause 5.4.2.2.
- The order of the MMR mapping MMR_order, which is one of 0,1,2,3.
- A two-dimensional array mmr_const=fp_mmr_const[cmp][pivot_idx[cmp]].
- A two-dimensional array mmr_coef [][]=fp_mmr_coef[cmp][pivot_idx[cmp]][][].

Output of this process is:

- The sample $v=v_{i,j}$ of the sample array V_{cmp} ,which is a chroma component of the mapped BL picture, with cmp equal to 1 or 2.

The following pseudo-code specifies how the value of v is derived for each sample:

```
if(s0 < pivot_value[0][0])
    s0 = pivot_value[0][0];
if(s0 > pivot_value[0][num_pivots_minus2[0]+1])
    s0 = pivot_value[0][num_pivots_minus2[0]+1];
if(s1 < pivot_value[1][0])
    s1 = pivot_value[1][0];
if(s1 > pivot_value[1][num_pivots_minus2[1]+1])
    s1 = pivot_value[1][num_pivots_minus2[1]+1];

if(s2 < pivot_value[2][0])
    s2 = pivot_value[2][0];
if(s2 > pivot_value[2][num_pivots_minus2[2]+1])
    s2 = pivot_value[2][num_pivots_minus2[2]+1];

    // constant
tt[0] = 1<<20;
num_coeff = 1;

// first order
if(MMR_order >= 1) {
    tt[1] = s0 << (20-BL_bit_depth);
    tt[2] = s1 << (20-BL_bit_depth);
    tt[3] = s2 << (20-BL_bit_depth);
    tt[4] = (s0*s1) << (20-2*BL_bit_depth);
    tt[5] = (s0*s2) << (20-2*BL_bit_depth);
    tt[6] = (s1*s2) << (20-2*BL_bit_depth);
    tt[7] = (tt[4]*tt[3]) >> 20;
}

// second order
if(MMR_order >= 2){
    tt[8]  = (s0*s0) << (20-2*BL_bit_depth);
    tt[9]  = (s1*s1) << (20-2*BL_bit_depth);
    tt[10] = (s2*s2) << (20-2*BL_bit_depth);
    tt[11] = (tt[4]*tt[4]) >> 20;
    tt[12] = (tt[5]*tt[5]) >> 20;
    tt[13] = (tt[6]*tt[6]) >> 20;
    tt[14] = (tt[7]*tt[7]) >> 20;
}

// third order
if(MMR_order >= 3){
    tt[15] = (tt[1]*tt[8]) >> 20;
    tt[16] = (tt[2]*tt[9]) >> 20;
    tt[17] = (tt[3]*tt[10]) >> 20;
    tt[18] = (tt[4]*tt[11]) >> 20;
    tt[19] = (tt[5]*tt[12]) >> 20;
    tt[20] = (tt[6]*tt[13]) >> 20;
```

```
        tt[21] = (tt[7]*tt[14]) >> 20;
    }

    rr = mmr_constant*tt[0];
    cnt = 1;
    for(i = 1; i <= MMR_order; i++ ){
        for(j = 0; j < 7; j++){
            rr += mmr_coeff[i][j]*tt[cnt];
            cnt++;
        }
    }

    rr = rr < 0 ? 0 : rr;
    v = (rr >> (4+coefficient_log2_denom);
    v = v > 0xffff ? 0xffff : v;
```

5.4.3 Enhancement layer processing

5.4.3.1 General

For each BL picture, there is a corresponding EL picture. In the EL picture, the sample values carry the three quantized components of the EL signal. The EL post processing is defined by the following two steps:

1) A non-linear inverse quantization process is applied to the EL signal as described in clause 5.4.3.2. The output of this process is the inverse quantized EL signal.

2) The HDR signal reconstruction process adds the inverse quantized EL signal to the mapped BL signal as described in clause 5.4.3.3.

If disable_residual_flag is equal to 1, the composer shall take the mapping from the BL signal as the reconstructed HDR signal. If disable_residual_flag is equal to 0, the composer shall add the processed EL signal to the mapping from the BL signal to reconstruct the HDR signal.

5.4.3.2 Enhancement layer non-linear inverse quantization *(NLQ_LINEAR_DZ)*

This process is invoked for each colour component indexed by cmp with cmp equal to 0, 1 or 2.

Inputs to this process are:

- A sample array E_{cmp} with elements $e = e_{i,j}$ which is a (PicWidth)x(PicHeight) luma block of the EL input signal, when cmp is equal to 0, and a (PicWidthC)x(PicHeightC) chroma block of the EL input signal, when cmp is not equal to 0.
- The variables fp_linear_deadzone_threshold[cmp], fp_linear_deadzone_slope[cmp] and fp_hdr_in_max[cmp].
- The variable nlq_offset[cmp].
- The variable EL_bit_depth.
- The variable coefficient_log2_denom.

Output of this process is:

- A sample array R_{cmp} with elements $r = r_{i,j}$ which is a (PicWidth)x(PicHeight) luma block of the inverse quantized EL signal, when cmp is equal to 0, and a (PicWidthC)x(PicHeightC) chroma block of the inverse quantized EL signal, when cmp is not equal to 0. The bit depth of the signed samples r shall be equal to 17 including the sign bit.

The value of each sample r of the array R_{cmp} is derived as specified in the following pseudo-code:

```
// coefficients
T = fp_linear_deadzone_threshold[cmp];
S = fp_linear_deadzone_slope[cmp];
R = fp_hdr_in_max[cmp];
// input data
rr = e - nlq_offset[cmp];
if(rr == 0){
    r = 0;
```

```
    return;
}
else{
    sign = rr < 0? -1 : 1;
    rr <<= 1;
    rr -= sign;
    rr <<= (10-EL_bit_depth);
    // output data
    dq = rr * S;
    TT = (T << (10-EL_bit_depth + 1))*sign;
    dq += TT;
    RR = (R << (10-EL_bit_depth + 1);
    if( dq > RR )
        dq = RR;
    else if(dq < -RR)
        dq = -RR;
    r = (dq >> (coefficient_log2_denom - 5 - EL_bit_depth));
}
```

5.4.3.3 HDR signal reconstruction

For the reconstruction of the HDR output picture this process is invoked for each component with cmp equal to 0, 1 or 2.

Inputs to this process are:

- A sample array V_{cmp} with elements v = $v_{i,j}$ which is a (PicWidth)x(PicHeight) luma block of the predicted BL signal, when cmp is equal to 0, and a (PicWidthC)x(PicHeightC)chroma block of the predicted BL signal, when cmp is not equal to 0.
- A sample array R_{cmp} with elements r = $r_{i,j}$ which is a (PicWidth)x(PicHeight) luma block of the inverse quantized EL signal, when cmp is equal to 0, and a (PicWidthC)x(PicHeightC) chroma block of the inverse quantized EL signal, when cmp is not equal to 0.
- The variables disable_residual_flag and hdr_bit_depth.

Output of this process is:

- A sample array H_{cmp} with elements h = $h_{i,j}$ which is a (PicWidth)x(PicHeight) luma block of the reconstructed HDR signal, when cmp is equal to 0, and a (PicWidthC)x(PicHeightC) chroma block of the reconstructed HDR signal, when cmp is not equal to 0.

Initially the variable out_bit_depth which defines the bit depth of the reconstructed HDR signal is derived as follows:

- If the transfer characteristics of the BL input picture are equal to the transfer characteristics as defined in Recommendation ITU-R BT.1886-0 [5], out_bit_depth shall be equal to 14.
- Otherwise (the transfer characteristics of the BL input picture are equal to the transfer characteristics as defined in SMPTE ST 2084 [1]), out_bit_depth shall be equal to hdr_bit_depth.

Then, the value of each sample h is derived as specified in the following pseudo-code:

```
MAXOUT = (1 << out_bit_depth) - 1;
h = v;
if(!disable_residual_flag)
    h += r;

h += (1 << (15 - out_bit_depth));
h >>= (16 - out_bit_depth);

h = h < 0 ? 0 : h;
h = h > MAXOUT ? MAXOUT : h;
```

5.5 Conversion to SMPTE ST 2084 transfer characteristics

If the transfer characteristics of the BL input picture are equal to the transfer characteristics as defined in SMPTE ST 2084 [1], this conversion process shall be omitted, otherwise, if the transfer characteristics are equal to the transfer characteristics defined in Recommendation ITU-R BT.1886-0 [5], the following process shall apply.

The input to this process shall be the sample array H_{cmp} of the reconstructed 14 bit HDR signal from clause 5.4.3.3. The output of this process shall be an HDR signal with a bit depth equal to hdr_bit_depth and transfer characteristics as defined in SMPTE ST 2084 [1]. This conversion process invokes the following steps:

- Chroma-upsampling and colour conversion of the input array H_{cmp} from the original colour space to the RGB colour space as defined in Recommendation ITU-R BT.2020-2 [8].
- Conversion of the signal from the gamma-adjusted RGB domain to the linear light domain using the equations as defined in Recommendation ITU-R BT.1886-0 [5], annex 1. Thereby L_w shall be set to the value of the syntax element max_display_mastering_luminance and L_B shall be set to the value of the syntax element min_display_mastering_luminance.
- Conversion of the signal from linear light domain using the reference Inverse-EOTF as defined in clause 5 of SMPTE ST 2084 [1].
- Colour conversion of the signal from the RGB colour space as defined in Recommendation ITU-R BT.2020-2 [8] to the original colour space, and chroma-downsampling.

NOTE: The method described in annex C of the present document may be used as a guideline for the implementation of the conversion process defined in this clause.

6 HDR display management (DM) metadata embedder definition

6.1 Introduction

This clause describes the guidelines for inserting HDR DM metadata into an HDR picture with transfer characteristics as defined in SMPTE ST 2084 [1] for transport over baseband interfaces (see figure 2).

As the HDR signal is likely to provide a dynamic range and a colour gamut out of the HDR sink's capability range, display management may be performed in order to adapt the HDR signal to this range. Display management operations are image-content dependent in order to achieve the maximum performance. This image-dependent mapping is controlled by HDR DM metadata which is generated from the original HDR content.

In general, a group of HDR pictures share one HDR DM metadata package, or in worst case, every individual HDR picture has its own HDR DM metadata. So it is critical to make sure that the HDR DM metadata is synchronized to its corresponding HDR picture(s).

The data structure of HDR DM metadata is defined in clause 6.2.

The packetization of HDR DM metadata transmission is provided in clause 6.3.

A mechanism of embedding HDR DM metadata transmission packet in the HDR signal is introduced in clause 6.4, that is similar to the MPEG-2 recoding information transport mechanism as defined in SMPTE 319M-2000 standard [i.1].

NOTE 1: The metadata elements are defined according to the SMPTE standards ST 2086 [2], ST 2094-1 [3] and ST 2094 -10 [4] or set to the values defined below and according to the characteristics of the associated HDR picture.

NOTE 2: The composer described in clause 5 is independent of the type of SMPTE ST 2094 metadata carried. This clause only defines the method to embed SMPTE ST 2094 -10 [4] elements.

6.2 HDR display management metadata definition

6.2.1 HDR display management metadata syntax

The parsing process of each syntax element by the descriptor u(n) is described in Recommendation ITU-T H.265 [9].

Table 2: HDR Display management metadata syntax

hdr_dm_metadata() {	**Descriptor**	**Default Value**
if(metadata_version == 0) {		
dm_metadata()		
}		
}		

Table 3: Display management metadata syntax

dm_metadata() {	**Descriptor**	**Default Value**
reserved_0x00_8bits	u(8)	0x00
scene_refresh_flag	u(8)	0x00
for(c = 0; c < 3; c++) {		
for(i = 0; i < 3; i++) {		
YCCtoRGB_coef_hi[c][i]	u(8)	refer to 6.2.2
YCCtoRGB_coef_lo[c][i]	u(8)	refer to 6.2.2
}		
}		
for(c = 0; c < 3; c++) {		
YCCto_RGB_offset_byte3[c]	u(8)	refer to 6.2.2
YCCto_RGB_offset_byte2[c]	u(8)	refer to 6.2.2
YCCto_RGB_offset_byte1[c]	u(8)	refer to 6.2.2
YCCto_RGB_offset_byte0[c]	u(8)	refer to 6.2.2
}		
for(c = 0; c < 3; c++) {		
for(i = 0; i < 3; i++) {		
RGBtoLMS_coef_hi[c][i]	u(8)	refer to 6.2.2
RGBtoLMS_coef_lo[c][i]	u(8)	refer to 6.2.2
}		
}		
reserved_0xff_8bits	u(8)	0xFF
reserved_0xff_8bits	u(8)	0xFF
for(i = 0; i < 8; i++) {		
reserved_0x00_8bits	u(8)	0x00
}		
signal_bit_depth	u(8)	0x0C
signal_color_space	u(8)	0x00
reserved_0x01_8bits	u(8)	0x01
reserved_0x01_8bits	u(8)	0x01
source_min_PQ_hi	u(8)	0x00
source_min_PQ_lo	u(8)	0x3E
source_max_PQ_hi	u(8)	0x0E
source_max_PQ_lo	u(8)	0x70
reserved_0x00_8bits	u(8)	0x00
reserved_0x2a_8bits	u(8)	0x2A
num_ext_blocks	u(8)	0x00
if(num_ext_blocks > 0) {		
for(i = 0; i < num_ext_blocks; i ++) {		
ext_metadata_block()		
}		
}		
}		

Table 4: Extension metadata block syntax

ext_metadata_block() {	**Descriptor**	**Default Value**
ext_block_length_byte3	u(8)	0x00
ext_block_length_byte2	u(8)	0x00
ext_block_length_byte1	u(8)	0x00
ext_block_length_byte0	u(8)	0x00
ext_block_length = ext_block_length_byte3 << 24 | ext_block_length_byte2 << 16 | ext_block_length_byte1 << 8 | ext_block_length_byte0		
ext_block_level	u(8)	0x00
ext_metadata_payload(ext_block_length, ext_block_level)		
}		

Table 5: Extension metadata payload syntax

ext_metadata_payload(ext_block_length, ext_block_level) {	**Descriptor**	**Default Value**
if(ext_block_level == 1) {		
min_PQ_hi	u(8)	refer to 6.2.2
min_PQ_lo	u(8)	refer to 6.2.2
max_PQ_hi	u(8)	refer to 6.2.2
max_PQ_lo	u(8)	refer to 6.2.2
avg_PQ_hi	u(8)	refer to 6.2.2
avg_PQ_lo	u(8)	refer to 6.2.2
}		
if(ext_block_level == 2) {		
target_max_PQ_hi	u(8)	refer to 6.2.2
target_max_PQ_lo	u(8)	refer to 6.2.2
trim_slope_hi	u(8)	0x08
trim_slope_lo	u(8)	0x00
trim_offset_hi	u(8)	0x08
trim_offset_lo	u(8)	0x00
trim_power_hi	u(8)	0x08
trim_power_lo	u(8)	0x00
trim_chroma_weight_hi	u(8)	0x08
trim_chroma_weight_lo	u(8)	0x00
trim_saturation_gain_hi	u(8)	0x08
trim_saturation_gain_lo	u(8)	0x00
ms_weight_hi	u(8)	0x0F
ms_weight_lo	u(8)	0xFF
}		
if(ext_block_level == 5) {		
active_area_left_offset_hi	u(8)	0x00
active_area_left_offset_lo	u(8)	0x00
active_area_right_offset_hi	u(8)	0x00
active_area_right_offset_lo	u(8)	0x00
active_area_top_offset_hi	u(8)	0x00
active_area_top_offset_lo	u(8)	0x00
active_area_bottom_offset_hi	u(8)	0x00
active_area_bottom_offset_lo	u(8)	0x00
}		
}		

6.2.2 HDR display management metadata semantics

This clause defines the semantics of the DM metadata syntax defined in clause 6.2.1. If a metadata parameter is not present its value shall be set to the default value as defined in table 3.

reserved_0xff_8bits shall be equal to 0xff in HDR display management metadata conforming to this version of the present document. Other values for reserved_0xff_8bits are reserved for future use.

reserved_0x00_8bits shall be equal to 0x00 in HDR display management metadata conforming to this version of the present document. Other values for reserved_0x00_8bits are reserved for future use.

reserved_0x01_8bits shall be equal to 0x01 in HDR display management metadata conforming to this version of the present document. Other values for reserved_0x01_8bits are reserved for future use.

reserved_0x2a_8bits shall be equal to 0x2a in HDR display management metadata conforming to this version of the present document. Other values for reserved_0x2a_8bits are reserved for future use.

A scene is herein defined as a set of consecutive HDR output pictures in output order.

scene_refresh_flag equal to 1 indicates that the picture, which DM metadata affects, is the first picture of a scene. scene_refresh_flag equal to 0 indicates that picture, which DM metadata affects, is a picture of the same scene as the previous picture in output order. See also TimeIntervalStart and TimeInterValDuration definitions in SMPTE ST 2094-1 [3].

The following three-by-three matrix parameters YCCtoRGB_coef[c][i] with the row index c=0..2 and the column index i=0..2 and three-by-one vector parameters YCCtoRGB_offset[c] with c=0..2 represent data used for the conversion from the opponent colour representation (YCbCr, ICtCp) to the non-linear trichromatic colour representation (RGB, LMS) of the HDR ouput picture. See also the definition of Display Primaries in SMPTE ST 2086 [2].

The parameters are specified as follows:

YCCtoRGB_coef[c][i] specifies the value of three-by-three coefficients of the YCCtoRGB transform matrix. The value of YCCtoRGB_coef[c][i] is defined as Short(YCCtoRGB_coef_hi[c][i]<<8 | YCCtoRGB_coef_lo[c][i]). The value shall be in the range of -32 768 to 32 767, inclusive.

If information about YCCtoRGB_coef[c][i] is not available, the three-by-three coefficients of the YCCtoRGB transform matrix shall be set equal to [9 575, 0, 14 742; 9 575, -1 754, -4 383; 9 575, 17 372, 0].

YCCtoRGB_offset[c] specifies the value of three-by-one offset of the YCCtoRGB transform matrix. The value of YCCtoRGB_offset[c] is defined as (YCCtoRGB_offset_byte3[c]<<24 | YCCtoRGB_offset_byte2[c]<<16 | YCCtoRGB_offset_byte1[c]<<8 | YCCtoRGB_offset_byte0[c]).The value shall be in the range of 0 to $2^{32}-1$, inclusive.

If information about YCCtoRGB_offset[c] is not available, the three-by-one YCCtoRGB transform matrix offsets shall be set equal to [67 108 864, 536 870 912, 536 870 912].

The following three-by-three parameters RGBtoLMS_coef[c][i] with the row index c=0..2 and the column index i=0..2 represent data used for the conversion from the linear trichromatic colour representation (RGB, LMS) of the HDR ouput picture to the linear LMS colour representation.

The parameters are specified as follows:

RGBtoLMS_coef[c][i] specifies the value of three-by-three coefficients of RGBtoLMS transform matrix. The value of RGBtoLMS_coef[c][i] is defined as Short(RGBtoLMS_coef_hi[c][i] <<8 | RGBtoLMS_coef_lo[c][i]). The value shall be in the range of -32 768 to 32 767, inclusive.

If information about RGBtoLMS_coef[c][i] is not available, the three-by-three coefficients of the RGBtoLMS transform matrix shall be set equal to [5 845, 9 702, 837; 2 568, 12 256, 1 561; 0, 679, 15 705].

signal_bit_depth indicates the bit depth of the associated HDR picture.

If the HDR picture is the output picture of the composing process as defined in clause 5.4, the value of the syntax element signal_bit_depth shall be set equal to the value of the syntax element hdr_bit_depth (see clause 5.3.2).

signal_color_space specifies the HDR picture colour space. The value shall be in the range of 0 to 3, inclusive. The corresponding colour spaces are defined in table 6.

If the HDR picture is the output picture of the composing process as defined in clause 5.4, the value shall be set equal to the corresponding colour representation method of the associated HDR output picture.

Table 6: Definition of HDR signal color space

signal_color_space	colour space representation
0	YCbCr
1	Reserved
2	ICtCp
3	Reserved

source_min_PQ specifies the minimum luminance value of source display in 12-bit encoding with an EOTF as defined in SMPTE ST 2084 [1]. The value of source_min_PQ is defined by (source_min_PQ_hi<<8 | source_min_PQ_lo). The value shall be in the range of 0 to 4 095, inclusive. See also the definition of Minimum Display Mastering Luminance in SMPTE ST 2086 [2].

source_max_PQ specifies the maximum luminance value of source display in 12-bit encoding with an EOTF as defined in SMPTE ST 2084 [1]. The value of source_max_PQ is defined by (source_max_PQ_hi<<8 | source_max_PQ_lo). The value shall be in the range of 0 to 4 095, inclusive. See also the definition of Maximum Display Mastering Luminance in SMPTE ST 2086 [2].

num_ext_blocks specifies the number of extended metadata blocks. The value shall be in the range of 0 to 254, inclusive.

ext_block_length is used to derive the size of current extended metadata block payload in bytes. The value shall be in the range of 0 to 2^{32}-1, inclusive. Ext_block_length is not present if num_ext_blocks is equal to 0.

The value of ext_block_length is defined as (ext_block_length_byte3<<24 | ext_block_length_byte2<<16 | ext_block_length_byte1<<8 | ext_block_length_byte0).

The value of ext_block_length shall be set equal to 6, when the value ext_block_level is set equal to 1.

The value of ext_block_length shall be set equal to 14, when the value ext_block_level is set equal to 2.

The value of ext_block_length shall be set equal to 8, when the value ext_block_level is set equal to 5.

ext_block_level specifies the level of payload contained in the current extended metadata block. The value shall be in the range of 0 to 255, inclusive. The corresponding extended metadata block types are defined in table 7. If ext_block_level is not present, it shall be inferred to be equal to 0. Values of ext_block_level that are reserved shall not be present in the bitstreams conforming to this version of the specification. Blocks using values of ext_block_level that are reserved shall be ignored.

Table 7: Definition of extended metadata block type

ext_block_level	extended metadata block type
0	Reserved
1	Level 1 Metadata - Content Range
2	Level 2 Metadata - Trim Pass
3	Reserved
4	Reserved
5	Level 5 Metadata - Active Area
6 …255	Reserved

When an extended display mapping metadata block with ext_block_level equal to 5 is present, the following constraints shall apply:

- An extended display mapping metadata block with ext_block_level equal to 5 shall be preceded by at least one extended display mapping metadata block with ext_block_level equal to 1 or 2.
- Between any two extended display mapping metadata blocks with ext_block_level equal to 5, there shall be at least one extended display mapping metadata block with ext_block_level equal to 1 or 2.
- No extended display mapping metadata block with ext_block_level equal to 1 or 2 shall be present after the last extended display mapping metadata block with ext_block_level equal to 5.

- The metadata of an extended display mapping metadata block with ext_block_level equal to 1 or 2 shall be applied to the active area specified by the first extended display mapping metadata block with ext_block_level equal to 5 following this block.
- When the active area defined by the current extended display mapping metadata block with ext_block_level equal to 5 overlaps with the active area defined by preceding extended display mapping metadata blocks with ext_block_level equal to 5, all metadata of the extended display mapping metadata blocks with ext_block_level equal to 1 or 2 associated with the current extended display mapping metadata block with ext_block_level equal to 5 shall be applied to the pixel values of the overlapping area.

min_PQ specifies the minimum luminance value of current scene in 12-bit encoding with an EOTF as defined in SMPTE ST 2084 [1]. The value of min_PQ is defined as (min_PQ_hi<<8 | min_PQ_lo). The value shall be in the range of 0 to 4 095, inclusive. If min_PQ is not present, it shall be inferred to be equal to the value of source_min_PQ. Note that the 12-bit min_PQ value is calculated as follows:

$$\mathrm{min\_PQ} = \mathrm{Clip3}(0, 4\,095, \mathrm{Round}(\mathrm{Min} * 4\,096)$$

where Min is MinimumPqencodedMaxrgb as defined in clause 6.1.3 of SMPTE ST 2094-10 [4].

max_PQ specifies the maximum luminance value of current scene in 12-bit encoding with an EOTF as defined in SMPTE ST 2084 [1]. The value of max_PQ is defined by (max_PQ_hi<<8 | max_PQ_lo). The value shall be in the range of 0 to 4 095, inclusive. If max_PQ is not present, it shall be inferred to be equal to the value of source_max_PQ. Note that the 12-bit max_PQ value is calculated as follows:

$$\mathrm{max\_PQ} = \mathrm{Clip3}(0, 4\,095, \mathrm{Round}(\mathrm{Max} * 4\,096)$$

where Max is MaximumPqencodedMaxrgb as defined in clause 6.1.5 of SMPTE ST 2094-10 [4].

avg_PQ specifies the midpoint luminance value of current scene in 12-bit encoding with an EOTF as defined in SMPTE ST 2084 [1]. The value of avg_PQ is defined as (avg_PQ_hi<<8 | avg_PQ_lo). The value shall be in the range of 0 to 4 095, inclusive. If avg_PQ is not present, it shall be inferred to be equal to the value of (source_min_PQ + source_max_PQ)/2. Note that the 12-bit avg_PQ value is calculated as follows:

$$\mathrm{avg\_PQ} = \mathrm{Clip3}(0, 4\,095, \mathrm{Round}(\mathrm{Avg} * 4\,096)$$

where Avg is AveragePqencodedMaxrgb as defined in clause 6.1.4 of SMPTE ST 2094-10 [4].

target_max_PQ specifies the maximum luminance value of a target display in 12-bit encoding with an EOTF as defined in [1]. The value of target_max_PQ is defined as (target_max_PQ_hi<<8 | target_max_PQ_lo). The value shall be in the range of 0 to 4 095, inclusive. If target_max_PQ is not present, it shall be inferred to be equal to the value of source_max_PQ. The target_max_PQ is the PQ encoded value of TargetedSystemDisplayMaximumLuminance as defined in clause 10.4 of SMPTE ST 2094-1 [3].

NOTE 1: If there is more than one extension block with ext_block_level equal to 2, those blocks will not have a duplicated target_max_PQ.

trim_slope specifies the slope metadata. The value shall be in the range of 0 to 4 095, inclusive. The value of trim_slope is defined as (trim_slope_hi<<8 | trim_slope_lo). Note that the 12-bit slope value is calculated as follows:

$$trim\_slope = \mathrm{Clip3}(0, 4\,095, \mathrm{Round}((S\text{-}0{,}5) * 4\,096))$$

where *S* is the ToneMappingGain as defined in clause 6.2.3 of SMPTE ST 2094-10 [4].

trim_offset specifies the offset metadata. The value shall be in the range of 0 to 4 095, inclusive. The value of trim_offset is defined by (trim_offset_hi<<8 | trim_offset_lo. Note that the 12-bit offset value is calculated as follows:

$$trim\_offset = \mathrm{Clip3}(0, 4\,095, \mathrm{Round}((O+0{,}5) * 4\,096))$$

where *O* is the ToneMappingOffset as defined in clause 6.2.2 of SMPTE ST 2094-10 [4].

trim_power specifies the power metadata. The value of trim_power is defined as (trim_power_hi<<8 | trim_power_lo). The value shall be in the range of 0 to 4 095, inclusive. Note that the 12-bit power value is calculated as follows:

$$trim\_power = \mathrm{Clip3}(0, 4\,095, \mathrm{Round}((P\text{-}0{,}5) * 4\,096))$$

where *P* is the ToneMappingGamma as defined in clause 6.2.4 of SMPTE ST 2094-10 [4].

trim_chroma_weight specifies the chroma weight metadata. The value of trim_chroma_weight is defined as (trim_chroma_weight_hi<<8 | trim_chroma_weight_lo). The value shall be in the range of 0 to 4 095, inclusive. Note that the 12-bit chroma weight value is calculated as follows:

$$trim\_chroma\_weight = \mathrm{Clip3}(0, 4\,095, \mathrm{Round}((CW + 0{,}5) * 4\,096))$$

where *CW* is the ChromaCompensationWeight as defined in clause 6.3.1 of SMPTE ST 2094-10 [4].

trim_saturation_gain specifies the saturation gain metadata. The value of trim_saturation_gain is defined as (trim_saturation_gain_hi<<8 | trim_saturation_gain_lo). The value shall be in the range of 0 to 4 095, inclusive. Note that the 12-bit saturation gain value is calculated as follows:

$$trim\_saturation\_gain = \mathrm{Clip3}(0, 4\,095, \mathrm{Round}((SG + 0{,}5) * 4\,096))$$

where *SG* is the SaturationGain as defined in clause 6.3.2 of SMPTE ST 2094-10 [4].

ms_weight specifies the multiscale weight metadata. If the most significant bit of ms_weight_hi is equal to 1, then the value of ms_weight is unspecified and all bits of ms_weight_hi and ms_weight_lo shall be set equal to 1.

Otherwise, if the most significant bit of ms_weight_hi is equal to 0, the value of ms_weight is defined as Short(ms_weight_hi<<8 | ms_weight_lo) and shall be in the range of 0 to 4 095, inclusive. Note that the 12-bit multiscale weight value is calculated as follows:

$$ms\_weight = \mathrm{Clip3}(0, 4\,095, \mathrm{Round}(MS * 4\,096))$$

where *MS* is the ToneDetailFactor as defined in clause 6.4.2 of SMPTE ST 2094-10 [4].

active_area_left_offset, active_area_right_offset, active_area_top_offset, active_area_bottom_offset specify the active area of current picture, in terms of a rectangular region specified in picture coordinates for active area. The values shall be in the range of 0 to 8 191, inclusive. See also UpperLeftCorner and LowerRightCorner definitions in SMPTE ST 2094-1 [3].

The value of active_area_left_offset is defined as (active_area_left_offset_hi<<8 | active_area_left_offset_lo).

The value of active_area_right_offset is defined as (active_area_right_offset_hi<<8 | active_area_right_offset_lo).

The value of active_area_top_offset is defined as (active_area_top_offset_hi<<8 | active_area_top_offset_lo).

The value of active_area_bottom_offset is defined as (active_area_bottom_offset_hi<<8 | active_area_bottom_offset_lo).

The coordinates of top left active pixel is derived as follows:

$$X_{top\_left} = \text{active\_area\_left\_offset}$$

$$Y_{top\_left} = \text{active\_area\_top\_offset}$$

The coordinates of top left active pixel are defined as the UpperLeftCorner in clause 9.2 of SMPTE ST 2094-1 [3].

With Xsize is the horizontal resolution of the current picture and Ysize is the vertical resolution of current picture, the coordinates of bottom right active pixel are derived as follows:

$$X_{bottom\_right} = \text{Xsize} - 1 - \text{active\_area\_right\_offset}$$

$$Y_{bottom\_right} = \text{Ysize} - 1 - \text{active\_area\_bottom\_offset}$$

The coordinates of bottom right active pixel are defined as the LowerRightCorner in clause 9.3 of SMPTE ST 2094-1 [3].

Where X_{bottom_right} shall be no smaller than X_{top_left} and Y_{bottom_right} shall be no smaller than Y_{top_left}.

NOTE 2: The syntax elements TargetedSystemDisplayMinimumLuminance, TargtedSystemDisplayPrimaries and TargetedSystemDisplayWhitePontChromaticity, which are defined in SMPTE ST 2094-1 [3] and contained in the Metadata Set defined in clause 7.1 of SMPTE ST 2094-10 [4] and are not explicitly defined in the HDR display management metadata syntax as defined in clause 6.2.1, may be set as follows:

The value of TargetedSystemDisplayMinimumLuminance as defined in clause 10.5 of SMPTE ST 2094-1 [3] may be set equal to the value of min_display_mastering_luminance, which is defined in clause 5.3.2.

The values of TargtedSystemDisplayPrimaries as defined in clause 10.2 of SMPTE ST 2094-1 [3], and TargetedSystemDisplayWhitePontChromaticity as defined in clause 10.3 of SMPTE ST 2094-1 [3] may be inferred by target_max_PQ as follows:

- If target_max_PQ is smaller than or equal to 100 cd/m², the value of TargtedSystemDisplayPrimaries and TargetedSystemDisplayWhitePointChromaticity may be set to the same corresponding values as specified for colour_primaries equal to 1 in table E.3 of Recommendation ITU-T H.265 [9].
- Otherwise, if target_max_PQ is larger than 100 cd/m², the value of TargtedSystemDisplayPrimaries and TargetedSystemDisplayWhitePointChromaticity may be set to the same corresponding values as specified for colour_primaries equal to 9 in table E.3 of Recommendation ITU-T H.265 [9].

6.3 HDR DM metadata transmission

6.3.1 HDR DM metadata transmission packetization

An HDR DM metadata structure is carried using one or more HDR DM metadata transmission packets. Each HDR DM metadata transmission packet has 128 bytes (1 024 bits). The first three bytes of an HDR DM metadata transmission packet are the packet header, the last four bytes are the packet tail, and the intermediate 121 bytes are the packet body. The packet body holds the HDR DM metadata structure described in clause 6.2. Table 8 gives an overview of the HDR DM metadata transmission packetization.

Table 8: HDR DM metadata transmission packetization overview

Byte \ Bit #	7	6	5	4	3	2	1	0
Header Byte 0	packet_type		metadata_type		metadata_version			no_md
Header Byte 1	affected_metadata_id				current_metadata_id			
Header Byte 2	reserved (=0)							EOS
Body Byte 0	The 1st byte of metadata packet body							
Body Byte 1	The 2nd byte of metadata packet body							
Body Byte 2	The 3rd byte of metadata packet body							
…	…							
Body Byte 120	The last byte of metadata packet body							
Tail Byte 0	CRC32[31:24]							
Tail Byte 1	CRC32[23:16]							
Tail Byte 2	CRC32[15:8]							
Tail Byte 3	CRC32[7:0]							

The HDR DM metadata transmission packet shall carry the metadata for the currently transmitted HDR picture or for the upcoming HDR picture. Each time the metadata of the HDR picture changes in a way that requires transmission of new HDR DM metadata, a metadata_id is incremented. Each HDR DM metadata transmission packet includes two fields, affected_metadata_id and current_metadata_id, that together indicate, whether the metadata refer to the current HDR picture or to the upcoming HDR picture. All HDR DM metadata transmission packets within a single HDR picture shall have the same current_metadata_id field value.

6.3.2 HDR DM metadata transmission packet header

Table 9: HDR DM metadata transmission packet header

Byte \ Bit #	7	6	5	4	3	2	1	0
Header Byte 0	packet_type		metadata_type		metadata_version			no_md
Header Byte 1	affected_metadata_id				current_metadata_id			
Header Byte 2	reserved (=0)							EOS

no_md set to indicate that there is no HDR DM metadata carried in the currently transmitted HDR picture. When no_md is equal to 1, the HDR sink shall ignore the current packet body and the following HDR DM metadata transmission packet(s) transmitted within the picture.

metadata_version specifies the revision number of the specification.

Metadata_version = 0b000: current version.

metadata_version = 0b001~0b111: reserved for future revisions.

metadata_type specifies the type of the metadata.

Metadata_type = 0b00: HDR DM metadata as defined in clause 6.2.

Metadata_type = 0b01~0b11: reserved for future use.

packet_type specifies the type of HDR DM metadata transmission packet.

Packet_type = 0b00: a single packet carries an entire HDR DM metadata structure.

packet_type = 0b01: the packet is the first packet, if multiple packets carry one HDR DM metadata structure.

packet_type = 0b10: the packet is not the first and not the last packet, if multiple packets carry one HDR DM metadata structure.

packet_type = 0b11: the packet is the last packet, if multiple packets carry one HDR DM metadata structure.

affected_metadata_id indicates the id number of the HDR picture, which is referred by this HDR DM metadata structure. The value shall be in the range of 0 and 15, inclusive.

current_metadata_id indicates the id number of the currently transmitted HDR picture. The value shall be in the range of 0 and 15, inclusive.

If the HDR DM metadata structure refers to the current HDR picture, affected_metadata_id shall be equal to current_metadata_id. If the HDR DM metadata structure refers to the immediate next HDR picture, affected_metadata_id shall be set to (current_metadata_id + 1) (mod 16). Affected_metadata_id shall be set to either current_metadata_id or (current_metadata_id + 1) (mod 16), therefore, only the metadata of the current and the immediate next HDR picture may be described. If metadata_version is equal to 0b000, affected_metadata_id and current_metadata_id shall be set to the identical value of affected_dm_metadata_id and current_dm_metadata_id, respectively.

EOS set to indicate that the immediate next picture contains non-HDR signal. When EOS is equal to 1, HDR sink shall switch to SDR operation mode for the immediate next picture.

6.3.3 HDR DM metadata transmission packet body

6.3.3.1 HDR DM metadata transmission packet body (packet_type = 0b00)

Table 10: HDR DM metadata transmission single packet body

Byte \ Bit #	7	6	5	4	3	2	1	0
Body Byte 0	metadata_length_hi							
Body Byte 1	metadata_length_lo							
Body Byte 2 to (N+1)	metadata_structure							
Body Byte (N+2) to 120	trailing_byte							

metadata_length(_hi, _lo) specifies the length of HDR DM metadata structure in bytes, not including metadata_length_hi, metadata_length_lo and trailing_byte. The values of metadata_length(_hi, _lo) shall satisfy the following condition:

N = (metadata_length_hi<<8 | metadata_length_lo) ≤ 119

metadata_structure is defined in clause 6.2 of the present document.

trailing_byte is equal to 0x00. Trailing_byte is not present if N is equal to 119.

6.3.3.2 HDR DM metadata transmission packet body (packet_type = 0b01)

Table 11: HDR DM metadata transmission first packet body

Byte \ Bit #	7	6	5	4	3	2	1	0
Body Byte 0	metadata_length_hi							
Body Byte 1	metadata_length_lo							
Body Byte 2 to 120	partial_metadata_structure							

metadata_length(_hi, _lo) specifies the length of HDR DM metadata structure in bytes, not including metadata_length_hi, metadata_length_lo. The values of metadata_length(_hi,_lo) shall satisfy the condition that 119 < L = (metadata_length_hi<<8 | metadata_length_lo) ≤ 0x2F00.

partial_metadata_structure is the first portion of metadata structure if multiple packets carry one metadata structure. The size of partial_metadata_structure is equal to 119 bytes.

6.3.3.3 HDR DM metadata transmission packet body (packet_type = 0b10 or 0b11)

Table 12: HDR DM metadata transmission last packet body

Byte \ Bit #	7	6	5	4	3	2	1	0
Body Byte 0 to (M-1)	remaining_metadata_structure							
Body Byte (M+1) to 120	trailing_byte							

remaining_metadata_structure is the remaining portion of metadata structure if multiple packets carry one metadata structure. The size of remaining_metadata_structure is equal to M bytes, where:

M = 121, if packet_type is equal to 0b10.

M = 121, if packet_type is equal to 0b11 and (L - 119) (mod 121) is equal to 0.

M = (L - 119) (mod 121), if packet_type is equal to 0b11 and (L - 119) (mod 121) ≠ 0.

trailing_byte is equal to 0x00. Trailing_byte is not present, if packet_type is equal to 0b10 or M is equal to 121.

If metadata_version is equal to 0b000, the HDR source/sink device shall be able to transmit/receive HDR DM metadata structures with a size of 512 bytes.

NOTE: An HDR DM metadata structure of size of 512 bytes requires one HDR DM metadata transmission packet with packet_type = 0x01, three HDR DM metadata transmission packets with packet_type = 0x10, and one HDR DM metadata transmission packet with packet_type = 0x11.

6.3.4 HDR DM metadata transmission packet tail

Table 13: HDR DM metadata transmission packet header

Byte \ Bit #	7	6	5	4	3	2	1	0
Tail Byte 0	CRC32[31:24]							
Tail Byte 1	CRC32[23:16]							
Tail Byte 2	CRC32[15:8]							
Tail Byte 3	CRC32[7:0]							

CRC32[31:0] is a 32-bit CRC to allow error detection for each HDR DM metadata transmission packet. A model for generating this 32-bit CRC is defined in annex A of ISO/IEC 13818-1 [10]. The CRC32 value is calculated on the first 992 bits of one HDR DM metadata transmission packet, using the generator polynomial below. HDR sink devices shall look into a packet only after CRC passes.

$$X^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^{8} + x^{7} + x^{5} + x^{4} + x^{2} + x + 1$$

6.4 Embedding HDR DM metadata transmission packets in an HDR signal

6.4.1 Embedding a single HDR DM metadata transmission packet (packet_type = 0b00)

If the size of the HDR DM metadata structure is less than or equal to 119 bytes, HDR source shall send one single HDR DM metadata transmission packet. The 128-byte HDR DM metadata transmission packet shall be placed bit by bit onto the least significant bits of the 12-bit chroma samples in the HDR YCbCr 4:2:2 signal.

The mapping relationship between a bit position in the HDR DM metadata transmission packet and the pixel sample position in HDR YCbCr 4:2:2 signal buffer is described below.

When HDR DM metadata transmission packet is represented in a sequence of bytes, the first byte goes first with the most significant bit of that byte first.

Example: For byte 0x01, the seven bits '0' go first followed by the one bit '1'. The i-th bit of an HDR DM metadata transmission packet is at bit[m] ($0 \leq m \leq 7$) of byte[n] ($0 \leq n \leq 127$) where i = (n*8+(7-m)).

Assume the HDR picture has a resolution of W × H and a pixel sample can be represented using coordinates (y, x), where $0 \leq y \leq H$ and $0 \leq x \leq W$. For each pixel, there are one luma sample and one chroma sample. The chroma sample represents the Cb component for even pixels and the Cr component for odd pixels. In raster scan order the i-th pixel is at (y, x) with y = i / W and x = i % W, i.e. i = W*y + x.

The i-th bit of an HDR DM metadata transmission packet shall be placed onto the least significant bit of the chroma sample of the i-th pixel in raster scan order in the HDR picture.

6.4.2 Bit scrambling

To increase the imperceptibility of embedding HDR DM metadata in HDR signal, a form of parity scrambling is applied to the data before being placed in LSB of the 12-bit chroma samples. The parity scrambling scheme is depicted in figure 4. The parity used to scramble HDR DM metadata is derived by combining the parity of bit[11:1] of the chroma sample with the parity of bit[11:0] of the subsequent luma sample in the video stream.

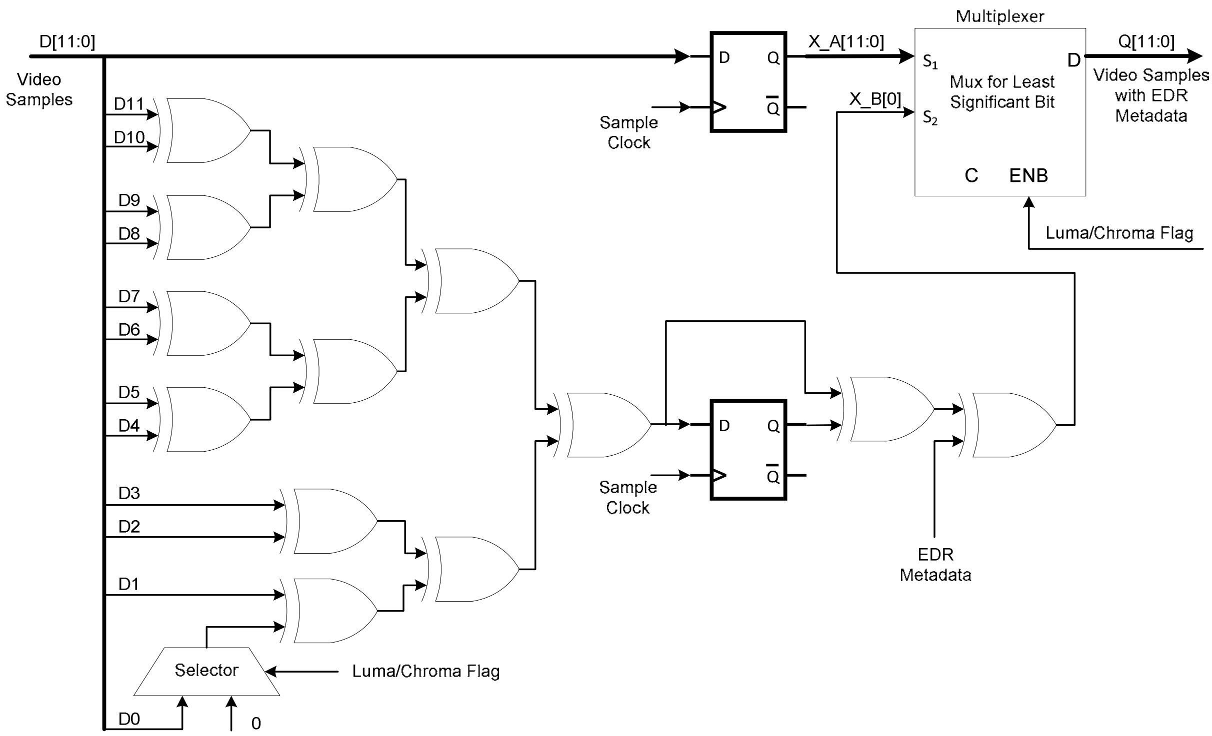


Figure 4: Scrambling HDR DM metadata

A detailed example of the metadata bits embedded into the samples for the first four luma and associated chroma samples is shown in table 14.

Table 14: Layout of HDR DM metadata embedded in 12-bit HDR YCbCr 4:2:2 video data

D11	Cb[0][11]	Y[0][11]	Cr[0][11]	Y[1][11]	Cb[1][11]	Y[2][11]	Cr[1][11]	Y[3][11]
D10	Cb[0][10]	Y[0][10]	Cr[0][10]	Y[1][10]	Cb[1][10]	Y[2][10]	Cr[1][10]	Y[3][10]
D9	Cb[0][9]	Y[0][9]	Cr[0][9]	Y[1][9]	Cb[1][9]	Y[2][9]	Cr[1][9]	Y[3][9]
D8	Cb[0][8]	Y[0][8]	Cr[0][8]	Y[1][8]	Cb[1][8]	Y[2][8]	Cr[1][8]	Y[3][8]
D7	Cb[0][7]	Y[0][7]	Cr[0][7]	Y[1][7]	Cb[1][7]	Y[2][7]	Cr[1][7]	Y[3][7]
D6	Cb[0][6]	Y[0][6]	Cr[0][6]	Y[1][6]	Cb[1][6]	Y[2][6]	Cr[1][6]	Y[3][6]
D5	Cb[0][5]	Y[0][5]	Cr[0][5]	Y[1][5]	Cb[1][5]	Y[2][5]	Cr[1][5]	Y[3][5]
D4	Cb[0][4]	Y[0][4]	Cr[0][4]	Y[1][4]	Cb[1][4]	Y[2][4]	Cr[1][4]	Y[3][4]
D3	Cb[0][3]	Y[0][3]	Cr[0][3]	Y[1][3]	Cb[1][3]	Y[2][3]	Cr[1][3]	Y[3][3]
D2	Cb[0][2]	Y[0][2]	Cr[0][2]	Y[1][2]	Cb[1][2]	Y[2][2]	Cr[1][2]	Y[3][2]
D1	Cb[0][1]	Y[0][1]	Cr[0][1]	Y[1][1]	Cb[1][1]	Y[2][1]	Cr[1][1]	Y[3][1]
D0	Metadata	Y[0][0]	Metadata	Y[1][0]	Metadata	Y[2][0]	Metadata	Y[3][0]

6.4.3 Repetition of HDR DM metadata transmission packets

To increase the robustness of the HDR DM metadata transmission, each HDR DM metadata transmission packet shall be sent three times consecutively. For instance, to transmit a packet with packet_type = 0b00, HDR source shall embed three copies of this packet in an HDR picture.

The first bit of 1st copy is placed at the 0th pixel's chroma sample's least significant bit.

The last bit of 1st copy is placed at the 1 023rd pixel's chroma sample's least significant bit.

The first bit of 2nd copy is placed at the 1 024th pixel's chroma sample's least significant bit.

The last bit of 2nd copy is placed at the 2 047th pixel's chroma sample's least significant bit.

The first bit of 3rd copy is placed at the 2 048th pixel's chroma sample's least significant bit.

The last bit of 3rd copy is placed at the 3 071st pixel's chroma sample's least significant bit.

After extracting a copy of the HDR DM metadata transmission packet, an HDR sink device shall calculate the CRC on 128 bytes. If the CRC output is zero, the HDR sink device shall keep the current copy and skip the other copies. If CRC output is non-zero, the HDR sink device shall extract the next copy until the CRC output of the copy is zero. If the CRC output of all three copies is non-zero, the CRC-32 check fails on the current frame.

6.4.4 Embedding multiple HDR DM metadata transmission packets (packet_type ≠ 0b00)

If the size of HDR DM metadata structure is larger than 119 bytes, HDR source shall split the HDR DM metadata structure into multiple HDR DM metadata transmission packets including one packet with packet_type = 0b01 and one packet with packet_type = 0b11 and zero or more packets with packet_type = 0b10. All of the packets shall have the same affected_metadata_id and shall be transmitted within a single picture.

The HDR DM metadata transmission packets carrying parts of the same HDR DM metadata structure shall be sent consecutively after the scrambling defined in clause 6.3.2 and packet repetition defined in clause 6.3.3 are applied to each of the multiple HDR DM metadata transmission packets. The bit[m] of byte[n] of the P-th HDR DM metadata transmission packet shall be placed onto the least significant bit of chroma sample of the i-th pixel of raster order in the HDR picture, where i = P*3 072 + (n*8 + (7 - m)).

Annex A (normative): Profiles and Levels

A.1 Introduction

Profiles and levels specify restrictions on CCM metadata and hence put limits on the capabilities of the CCM. Profiles and levels may also be used to indicate interoperability points between individual CCM implementations.

Each profile specifies a subset of algorithmic features and limits that shall be supported by all decoders conforming to that profile.

Each level specifies a set of limits on the values that may be taken by the syntax elements of the present document. The same set of level definitions is used with all profiles, but individual implementations may support a different level for each supported profile. For any given profile, a level generally corresponds to a particular CCM processing load and memory capability.

A.2 Profiles

A.2.1 Main profile / ETSI profile 1

A CCM conforming to Main profile / ETSI profile 1 shall obey the following constraints:

a) mapping_idc[cmp][pivot_idx] with cmp=0 shall be equal to 0.

b) mapping_idc[cmp][pivot_idx] with cmp=1..2 shall be 0 or 1.

c) BL_bit_depth_minus8 shall be equal to 0 or 2.

d) EL_bit_depth_minus8 shall be equal to 0 or 2.

e) disable_residual_flag shall be equal to 0 or 1.

f) coefficient_log2_denom shall be in the range of 0 to 23, inclusive.

Conformance of CCM metadata to the Main profile / ETSI profile 1 is indicated by ccm_profile being equal to 1.

A.2.2 ETSI profile 2

A CCM conforming to ETSI profile 2 shall obey the following constraints:

a) mapping_idc[cmp][pivot_idx] with cmp=0..2 shall be equal to 0.

b) BL_bit_depth_minus8 shall be equal to 2.

c) disable_residual_flag shall be equal to 1.

d) coefficient_log2_denom shall be in the range of 0 to 23, inclusive.

Conformance of CCM metadata to the ETSI profile 2 is indicated by ccm_profile being equal to 3.

A.2.3 ETSI profile 3

A CCM conforming to ETSI profile 3 shall obey the following constraints:

a) mapping_idc[cmp][pivot_idx] with cmp=0..2 shall be equal to 0.

b) BL_bit_depth_minus8 shall be equal to 0.

c) EL_bit_depth_minus8 shall be equal to 0.

d) disable_residual_flag shall be equal to 0 or 1.

e) coefficient_log2_denom shall be in the range of 0 to 23, inclusive.

Conformance of CCM metadata to the ETSI profile 3 is indicated by ccm_profile being equal to 4.

A.3 Levels

A CCM conforming to the Main profile / ETSI profile 1, ETSI profile 2, or ETSI profile 3 at a specific level as defined in table A.1 shall obey the constraints specified in table A.2.

Table A.1: Definition of levels

ccm_level	CCM decoder level definitions
0	1

Table A.2: Definition of level limits

Value limit of syntax elements	CCM level limits
	Level 1
num_pivots_minus2[0]	7 (Polynomial)
num_pivots_minus2[1] num_pivots_minus2[2]	3 (Polynomial) 0 (MMR)

Annex B (informative): EL and BL spatial resampling

B.1 Introduction

Annex B describes a reference implementation of the spatial up-sampling process of the EL and the BL.

B.2 Base layer spatial resampling

For the spatial resampling of the BL signal a vertical up-sampling is applied first, followed by a horizontal spatial up-sampling. If the pixel used in the BL spatial resampling filter is out of the picture boundary, it is set to the value of pixel at the picture boundary. That is, image edges are handled by pixel replication.

The horizontal resampling filter SPATIAL_RESAMPLING_FILTER_H is defined as follows:

r[2n] = y[n]

r[2n+1] = Clip3(0, 0xFFFF, (22 * y[n-3] + 94 * y[n-2] - 524 * y[n-1] + 2 456 * y[n] + 2 456 * y[n+1] - 524 * y[n+2] + 94 * y[n+3] + 22 * y[n+4] + 2 048) >> 12)

where n is the index of the horizontal position of a sample y.

The vertical resampling filter SPATIAL_RESAMPLING_FILTER_V is defined as follows:

r[2n] = Clip3(0, 0xFFFF, (2 * y[n-3] - 12 * y[n-2] + 65 * y[n-1] + 222 * y[n] - 25 * y[n+1] + 4 * y[n+2] + 128) >> 8)

r[2n+1] = Clip3(0, 0xFFFF, (4 * y[n-2] - 25 * y[n-1] + 222 * y[n] + 65 * y[n+1] - 12 * y[n+2] + 2 * y[n+3] + 128) >> 8)

where n is the index of the vertical position of a sample y.

B.3 Enhancement layer spatial resampling

For the spatial resampling of the EL signal a vertical upsampling is applied first, followed by a horizontal spatial up-sampling. The input is an EL signal with bit depth equal to EL_bit_depth and chroma format equal to EL_chroma_format_idc. The output is a spatially up-sampled reconstructed EL signal with the same data type as the input signals. If the pixel used in the EL spatial resampling filter is out of picture boundary, it is set to the value of pixel at the picture boundary. That is, image edges are handled by pixel replication.

For the vertical EL spatial resampling of a colour component with cmp equal to 0 the following filter is applied:

r[2n] = Clip3(0, 0xFFFF,(- 3 * y[n-2] + 29 * y[n-1] + 111 * y[n] - 9 * y[n+1] + 64) >> 7)

r[2n+1] = Clip3(0, 0xFFFF,(- 9 * y[n-1] + 111 * y[n] + 29 * y[n+1] - 3 * y[n+2] + 64) >> 7)

where n is the index of the vertical position of a sample y.

For the vertical EL spatial resampling of a colour component with cmp equal to 1 or 2 the following filter is applied:

r[2n] = Clip3(0, 0xFFFF, (64 * y[n-1] + 192 * y[n] + 128) >> 8)

r[2n+1] = Clip3(0, 0xFFFF, (192 * y[n] + 64 * y[n+1] + 128) >> 8)

where n is the index of the vertical position of a sample y.

The same filter SPATIAL_RESAMPLING_FILTER_H as defined in clause B.2 is applied to all components cmp of the EL signal.

Annex C (informative): Method for the conversion of transfer characteristics from Recommendation ITU-R BT.1886 to SMPTE ST 2084

Annex C describes a method for the conversion process of a signal with transfer characteristics as defined in Recommendation ITU-R BT.1886-0 [5] to transfer characteristics as defined in SMPTE ST 2084 [1], and may be used by implementers as a guideline for their implementation and conformance testing of clause 5.5 of the present document.

Figure C.1 shows a block diagram of the proposed conversion method. Input is the reconstructed 14 bit HDR signal with transfer characteristics as defined in Recommendation ITU-R BT.1886-0 [5] and colour primaries as defined in Recommendation ITU-R BT.2020-2 [8]. Output is the reconstructed HDR signal with a bit depth equal to the value of hdr_bit_depth, transfer characteristics as defined in SMPTE ST 2084 [1] and colour primaries as defined in Recommendation ITU-R BT.2020-2 [8].

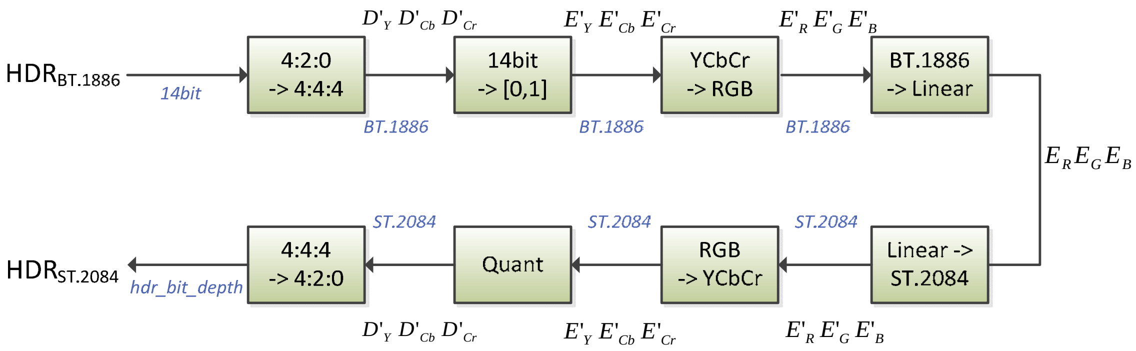
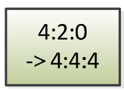


Figure C.1: Block diagram of conversion from HDR signal with BT.1886 transfer characteristics to SMPTE ST.2084 transfer characteristics

The functions and equations of each block in figure C.1 are as follows:



Horizontally and vertically up-sampling by a factor of two with respect to the D'_Y component are invoked for the D'_{Cb} and D'_{Cr} component. The vertical up-sampling filter is applied first, followed by the horizontal up-sampling filter. If a pixel used in the up-sampling filters is out of the picture boundary, it is set to the value of the pixel at the picture boundary. That is, image edges are handled by pixel replication.

The vertical up-sampling filter is defined as follows:

$$f[2n] = 64 * s[n]$$

$$f[2n+1] = -4 * s[n-1] + 36 * s[n] + 36 * s[n+1] - 4 * s[n+2]$$

where n is the index of the vertical position of a sample *s*.

The horizontal up-sampling filter is defined as follows:

$$r[2n] = (f[n] + 32) \gg 6$$

$$r[2n+1] = (-4 * f[n-1] + 36 * f[n] + 36 * f[n+1] - 4 * f[n+2] + 2\,048) \gg 12$$

where n is the index of the horizontal position of a sample *f*.

14bit ->[0,1]

Normalization of the 14bit represented signal with the components D'_Y, D'_{Cb} and D'_{Cr}, i.e. the chroma-upsampled reconstructed HDR signal, to the normalized signal with the components E'_Y, E'_{Cb} and E'_{Cr}:

$$E'_Y = Clip3\left(0{,}1{,}0, \left(\left(\frac{D'_Y}{64} - 16\right) \div 219\right)\right)$$

$$E'_{Cb} = Clip3\left(-0{,}5{,}0{,}5, \left(\left(\frac{D'_{Cb}}{64} - 128\right) \div 224\right)\right)$$

$$E'_{Cr} = Clip3\left(-0{,}5{,}0{,}5, \left(\left(\frac{D'_{Cr}}{64} - 128\right) \div 224\right)\right)$$

YCbCr -> RGB

Conversion from normalized luminance and colour-difference signals $E'_Y\ E'_{Cb}\ E'_{Cr}$ (BT.2020) to normalized $R'\ G'B'$ colour signals $E'_R\ E'_G\ E'_B$ (BT.2020):

$$E'_R = Clip3\left(0{,}1, (E'_Y + 1{,}47460 * E'_{Cr})\right)$$

$$E'_G = Clip3\left(0{,}1, (E'_Y - 0{,}16455 * E'_{Cb} - 0{,}57135 * E'_{Cr})\right)$$

$$E'_B = Clip3\left(0{,}1, (E'_Y + 1{,}88140 * E'_{Cb})\right)$$

BT.1886 ->Linear

Non-linear Recommendation ITU-R BT.1886-0 [5] to linear conversion from normalized $R'G'B'$ colour signals $E'_R E'_G E'_B$ to linearly represented RGB colour signals $L_R L_G L_B$ is defined by:

$$L_R = Clip3\left(\min\_display\_mastering\_luminance, \max\_display\_mastering\_luminance, BT1886\_EOTF(E'_R)\right)$$

$$L_G = Clip3\left(\min\_display\_mastering\_luminance, \max\_display\_mastering\_luminance, BT1886\_EOTF(E'_G)\right)$$

$$L_B = Clip3\left(\min\_display\_mastering\_luminance, \max\_display\_mastering\_luminance, BT1886\_EOTF(E'_B)\right)$$

with:

$$BT1886\_EOTF(V) = a * Max\left(0, (V + b)\right)^{2.4}$$

$$a = \left(\max\_display\_mastering\_luminance^{\frac{1}{2.4}} - \min\_display\_mastering\_luminance^{\frac{1}{2.4}}\right)^{2.4}$$

$$b = \frac{\min\_display\_mastering\_luminance^{\frac{1}{2.4}}}{\max\_display\_mastering\_luminance^{\frac{1}{2.4}} - \min\_display\_mastering\_luminance^{\frac{1}{2.4}}}$$

Linear -> ST.2084

Linear to non-linear SMPTE ST 2084 conversion from linearly represented RGB colour signals $L_R L_G L_B$ to normalized $R'G'B'$ colour signals $E'_R E'_G E'_B$ is defined by the following equations:

$$E'_R = Clip3\left(0{,}1, ST2084\_InvEOTF(L_R \div 10\ 000)\right)$$

$$E'_G = Clip3\left(0{,}1, ST2084\_InvEOTF(L_G \div 10\ 000)\right)$$

$$E'_B = Clip3\big(0{,}1, ST2084\_InvEOTF(L_B \div 10\,000)\big)$$

with:

$$ST2084\_InvEOTF(L) = \left(\frac{c_1 + c_2 L^{m_1}}{1 + c_3 L^{m_1}}\right)^{m_2}$$

$$m_1 = \frac{2\,610}{4\,096} * \frac{1}{4}$$

$$m_2 = \frac{2\,523}{4\,096} * 128$$

$$c_1 = \frac{3\,424}{4\,096} = c_3 - c_2 + 1$$

$$c_2 = \frac{2\,413}{4\,096} * 32$$

$$c_3 = \frac{2\,392}{4\,096} * 32$$

RGB
-> YCbCr

Conversion from normalized $R'\,G'B'$ colour signals $E'_R\,E'_G\,E'_B$ (BT.2020) to normalized luminance and colour-difference signals $E'_Y\,E'_{Cb}\,E'_{Cr}$ (BT.2020):

$$E'_Y = 0{,}2627 * E'_R + 0{,}6780 * E'_G + 0{,}0593 * E'_B$$

$$E'_{Cb} = \frac{E'_B - E'_Y}{1{,}8814}$$

$$E'_{Cr} = \frac{E'_R - E'_Y}{1{,}4746}$$

Quant

Quantization of the normalized colour signals E'_Y, E'_{Cb} and E'_{Cr} to the digitally represented colour signals D'_Y, D'_{Cb} and D'_{Cr} with the bitdepth of hdr_bit_depth bits:

$$D'_Y = Clip3\left(0, 2^{hdr\_bit\_depth-1}, Round\left(2^{hdr\_bit\_depth-8} * (219 * E'_Y + 16)\right)\right)$$

$$D'_{Cb} = Clip3\left(0, 2^{hdr\_bit\_depth-1}, Round\left(2^{hdr\_bit\_depth-8} * (224 * E'_{Cb} + 128)\right)\right)$$

$$D'_{Cr} = Clip3\left(0, 2^{hdr\_bit\_depth-1}, Round\left(2^{hdr\_bit\_depth-8} * (224 * E'_{Cr} + 128)\right)\right)$$

4:4:4
-> 4:2:0

Horizontally and vertically subsampling by a factor of two with respect to the D'_Y component are invoked for the D'_{Cb} and D'_{Cr} component. The horizontal down-sampling filter is applied first, followed by the vertical down-sampling filter. If a pixel used in the down-sampling filters is out of the picture boundary, it is set to the value of the pixel at the picture boundary. That is, image edges are handled by pixel replication.

The horizontal down-sampling filter is defined as follows:

$$f[n] = s[2n-1] + 6 * s[2n] + s[2n+1]$$

where n is the index of the horizontal position of a sample *s*.

The vertical down-sampling filter is defined as follows:

$$r[n] = (f[2n-1] + 6 * f[2n] + f[2n+1] + 32) \gg 6$$

where n is the index of the vertical position of a sample *f*.

History

Document history		
V1.1.1	February 2017	Publication