Annex E

Video usability information

(This annex forms an integral part of this Recommendation | International Standard.)

E.1 General

This annex specifies syntax and semantics of the VUI parameters of the SPSs.

VUI parameters are not required for constructing the luma or chroma samples by the decoding process. Conforming decoders are not required to process this information for output order conformance to this Specification (see Annex C and clause F.13 for the specification of output order conformance). Some VUI parameters are required to check bitstream conformance and for output timing decoder conformance.

In this annex, specification for presence of VUI parameters is also satisfied when those parameters (or some subset of them) are conveyed to decoders (or to the HRD) by other means not specified in this Specification. When present in the bitstream, VUI parameters shall follow the syntax and semantics specified in this annex. When the content of VUI parameters is conveyed for the application by some means other than presence within the bitstream, the representation of the content of the VUI parameters is not required to use the same syntax specified in this annex. For the purpose of counting bits, only the appropriate bits that are actually present in the bitstream are counted.

E.2 VUI syntax

E.2.1 VUI parameters syntax

vui_parameters( ) {	**Descriptor**
aspect_ratio_info_present_flag	u(1)
if(aspect_ratio_info_present_flag) {	
aspect_ratio_idc	u(8)
if(aspect_ratio_idc = = EXTENDED_SAR) {	
sar_width	u(16)
sar_height	u(16)
}	
}	
overscan_info_present_flag	u(1)
if(overscan_info_present_flag)	
overscan_appropriate_flag	u(1)
video_signal_type_present_flag	u(1)
if(video_signal_type_present_flag) {	
video_format	u(3)
video_full_range_flag	u(1)
colour_description_present_flag	u(1)
if(colour_description_present_flag) {	
colour_primaries	u(8)
transfer_characteristics	u(8)
matrix_coeffs	u(8)
}	
}	
chroma_loc_info_present_flag	u(1)
if(chroma_loc_info_present_flag) {	
chroma_sample_loc_type_top_field	ue(v)
chroma_sample_loc_type_bottom_field	ue(v)

}	
neutral_chroma_indication_flag	u(1)
field_seq_flag	u(1)
frame_field_info_present_flag	u(1)
default_display_window_flag	u(1)
if(default_display_window_flag) {	
def_disp_win_left_offset	ue(v)
def_disp_win_right_offset	ue(v)
def_disp_win_top_offset	ue(v)
def_disp_win_bottom_offset	ue(v)
}	
vui_timing_info_present_flag	u(1)
if(vui_timing_info_present_flag) {	
vui_num_units_in_tick	u(32)
vui_time_scale	u(32)
vui_poc_proportional_to_timing_flag	u(1)
if(vui_poc_proportional_to_timing_flag)	
vui_num_ticks_poc_diff_one_minus1	ue(v)
vui_hrd_parameters_present_flag	u(1)
if(vui_hrd_parameters_present_flag)	
hrd_parameters(1, sps_max_sub_layers_minus1)	
}	
bitstream_restriction_flag	u(1)
if(bitstream_restriction_flag) {	
tiles_fixed_structure_flag	u(1)
motion_vectors_over_pic_boundaries_flag	u(1)
restricted_ref_pic_lists_flag	u(1)
min_spatial_segmentation_idc	ue(v)
max_bytes_per_pic_denom	ue(v)
max_bits_per_min_cu_denom	ue(v)
log2_max_mv_length_horizontal	ue(v)
log2_max_mv_length_vertical	ue(v)
}	
}	

E.2.2 HRD parameters syntax

hrd_parameters(commonInfPresentFlag, maxNumSubLayersMinus1) {	**Descriptor**
if(commonInfPresentFlag) {	
nal_hrd_parameters_present_flag	u(1)
vcl_hrd_parameters_present_flag	u(1)
if(nal_hrd_parameters_present_flag || vcl_hrd_parameters_present_flag){	
sub_pic_hrd_params_present_flag	u(1)
if(sub_pic_hrd_params_present_flag) {	
tick_divisor_minus2	u(8)
du_cpb_removal_delay_increment_length_minus1	u(5)
sub_pic_cpb_params_in_pic_timing_sei_flag	u(1)
dpb_output_delay_du_length_minus1	u(5)
}	
bit_rate_scale	u(4)
cpb_size_scale	u(4)
if(sub_pic_hrd_params_present_flag)	
cpb_size_du_scale	u(4)
initial_cpb_removal_delay_length_minus1	u(5)
au_cpb_removal_delay_length_minus1	u(5)
dpb_output_delay_length_minus1	u(5)
}	
}	
for(i = 0; i <= maxNumSubLayersMinus1; i++) {	
fixed_pic_rate_general_flag[i]	u(1)
if(!fixed_pic_rate_general_flag[i])	
fixed_pic_rate_within_cvs_flag[i]	u(1)
if(fixed_pic_rate_within_cvs_flag[i])	
elemental_duration_in_tc_minus1[i]	ue(v)
else	
low_delay_hrd_flag[i]	u(1)
if(!low_delay_hrd_flag[i])	
cpb_cnt_minus1[i]	ue(v)
if(nal_hrd_parameters_present_flag)	
sub_layer_hrd_parameters(i)	
if(vcl_hrd_parameters_present_flag)	
sub_layer_hrd_parameters(i)	
}	
}	

E.2.3 Sub-layer HRD parameters syntax

sub_layer_hrd_parameters(subLayerId) {	**Descriptor**
for(i = 0; i <= CpbCnt; i++) {	
bit_rate_value_minus1[i]	ue(v)
cpb_size_value_minus1[i]	ue(v)
if(sub_pic_hrd_params_present_flag) {	
cpb_size_du_value_minus1[i]	ue(v)
bit_rate_du_value_minus1[i]	ue(v)
}	
cbr_flag[i]	u(1)
}	
}	

E.3 VUI semantics

E.3.1 VUI parameters semantics

aspect_ratio_info_present_flag equal to 1 specifies that aspect_ratio_idc is present. aspect_ratio_info_present_flag equal to 0 specifies that aspect_ratio_idc is not present.

aspect_ratio_idc specifies the value of the sample aspect ratio of the luma samples. Table E.1 shows the meaning of the code. When aspect_ratio_idc indicates EXTENDED_SAR, the sample aspect ratio is represented by sar_width : sar_height. When the aspect_ratio_idc syntax element is not present, aspect_ratio_idc value is inferred to be equal to 0. Values of aspect_ratio_idc in the range of 17 to 254, inclusive, are reserved for future use by ITU-T | ISO/IEC and shall not be present in bitstreams conforming to this version of this Specification. Decoders shall interpret values of aspect_ratio_idc in the range of 17 to 254, inclusive, as equivalent to the value 0.

Table E.1 – Interpretation of sample aspect ratio indicator

aspect_ratio_idc	Sample aspect ratio	Examples of use (informative)
0	Unspecified	
1	1:1 ("square")	7680x4320 16:9 frame without horizontal overscan 3840x2160 16:9 frame without horizontal overscan 1280x720 16:9 frame without horizontal overscan 1920x1080 16:9 frame without horizontal overscan (cropped from 1920x1088) 640x480 4:3 frame without horizontal overscan
2	12:11	720x576 4:3 frame with horizontal overscan 352x288 4:3 frame without horizontal overscan
3	10:11	720x480 4:3 frame with horizontal overscan 352x240 4:3 frame without horizontal overscan
4	16:11	720x576 16:9 frame with horizontal overscan 528x576 4:3 frame without horizontal overscan
5	40:33	720x480 16:9 frame with horizontal overscan 528x480 4:3 frame without horizontal overscan
6	24:11	352x576 4:3 frame without horizontal overscan 480x576 16:9 frame with horizontal overscan
7	20:11	352x480 4:3 frame without horizontal overscan 480x480 16:9 frame with horizontal overscan
8	32:11	352x576 16:9 frame without horizontal overscan
9	80:33	352x480 16:9 frame without horizontal overscan
10	18:11	480x576 4:3 frame with horizontal overscan
11	15:11	480x480 4:3 frame with horizontal overscan
12	64:33	528x576 16:9 frame without horizontal overscan
13	160:99	528x480 16:9 frame without horizontal overscan
14	4:3	1440x1080 16:9 frame without horizontal overscan
15	3:2	1280x1080 16:9 frame without horizontal overscan
16	2:1	960x1080 16:9 frame without horizontal overscan
17..254	Reserved	
255	EXTENDED_SAR	

NOTE 1 – For the examples in Table E.1, the term "without horizontal overscan" refers to display processes in which the display area matches the area of the cropped decoded pictures and the term "with horizontal overscan" refers to display processes in which some parts near the left or right border of the cropped decoded pictures are not visible in the display area. As an example, the entry "720x576 4:3 frame with horizontal overscan" for aspect_ratio_idc equal to 2 refers to having an area of 704x576 luma samples (which has an aspect ratio of 4:3) of the cropped decoded frame (720x576 luma samples) that is visible in the display area.

NOTE 2 – For the examples in Table E.1, the frame spatial resolutions shown as examples of use would be the dimensions of the conformance cropping window when field_seq_flag is equal to 0 and would have twice the height of the dimensions of the conformance cropping window when field_seq_flag is equal to 1.

sar_width indicates the horizontal size of the sample aspect ratio (in arbitrary units).

sar_height indicates the vertical size of the sample aspect ratio (in the same arbitrary units as sar_width).

sar_width and sar_height shall be relatively prime or equal to 0. When aspect_ratio_idc is equal to 0 or sar_width is equal to 0 or sar_height is equal to 0, the sample aspect ratio is unspecified in this Specification.

overscan_info_present_flag equal to 1 specifies that the overscan_appropriate_flag is present. When overscan_info_present_flag is equal to 0 or is not present, the preferred display method for the video signal is unspecified.

overscan_appropriate_flag equal to 1 indicates that the cropped decoded pictures output are suitable for display using overscan. overscan_appropriate_flag equal to 0 indicates that the cropped decoded pictures output contain visually important information in the entire region out to the edges of the conformance cropping window of the picture, such that the cropped decoded pictures output should not be displayed using overscan. Instead, they should be displayed using either an exact match between the display area and the conformance cropping window, or using underscan. As used in this paragraph, the term "overscan" refers to display processes in which some parts near the borders of the cropped decoded pictures are not visible in the display area. The term "underscan" describes display processes in which the entire cropped decoded pictures are visible in the display area, but they do not cover the entire display area. For display processes that neither use overscan nor underscan, the display area exactly matches the area of the cropped decoded pictures.

NOTE 3 – For example, overscan_appropriate_flag equal to 1 might be used for entertainment television programming, or for a live view of people in a videoconference and overscan_appropriate_flag equal to 0 might be used for computer screen capture or security camera content.

video_signal_type_present_flag equal to 1 specifies that video_format, video_full_range_flag and colour_description_present_flag are present. video_signal_type_present_flag equal to 0, specify that video_format, video_full_range_flag and colour_description_present_flag are not present.

video_format indicates the representation of the pictures as specified in Table E.2, before being coded in accordance with this Specification. When the video_format syntax element is not present, video_format value is inferred to be equal to 5. The values 6 and 7 for video_format are reserved for future use by ITU-T | ISO/IEC and shall not be present in bitstreams conforming to this version of this Specification. Decoders shall interpret the values 6 and 7 for video_format as equivalent to the value 5.

Table E.2 – Meaning of video_format

video_format	Meaning
0	Component
1	PAL
2	NTSC
3	SECAM
4	MAC
5	Unspecified video format

video_full_range_flag indicates the black level and range of the luma and chroma signals as derived from E'_Y, E'_{PB}, and E'_{PR} or E'_R, E'_G and E'_B real-valued component signals.

When the video_full_range_flag syntax element is not present, the value of video_full_range_flag is inferred to be equal to 0.

colour_description_present_flag equal to 1 specifies that colour_primaries, transfer_characteristics and matrix_coeffs are present. colour_description_present_flag equal to 0 specifies that colour_primaries, transfer_characteristics and matrix_coeffs are not present.

colour_primaries indicates the chromaticity coordinates of the source primaries as specified in Table E.3 in terms of the CIE 1931 definition of x and y as specified in ISO 11664-1.

When the colour_primaries syntax element is not present, the value of colour_primaries is inferred to be equal to 2 (the chromaticity is unspecified or is determined by the application). Values of colour_primaries that are identified as reserved in Table E.3 are reserved for future use by ITU-T | ISO/IEC and shall not be present in bitstreams conforming to this version of this Specification. Decoders shall interpret reserved values of colour_primaries as equivalent to the value 2.

Table E.3 – Colour primaries

Value	Primaries			Informative remark
0	Reserved			For future use by ITU-T | ISO/IEC
1	primary	x	y	Rec. ITU-R BT.709-5 Rec. ITU-R BT.1361 conventional colour gamut system and extended colour gamut system IEC 61966-2-1 (sRGB or sYCC) IEC 61966-2-4 Annex B of SMPTE RP 177 (1993)
	green	0.300	0.600	
	blue	0.150	0.060	
	red	0.640	0.330	
	white D65	0.3127	0.3290	
2	Unspecified			Image characteristics are unknown or are determined by the application.
3	Reserved			For future use by ITU-T | ISO/IEC
4	primary	x	y	Rec. ITU-R BT.470-6 System M (historical) United States National Television System Committee (1953), Recommendation for transmission standards for colour television United States Federal Communications Commission (2003), Title 47 Code of Federal Regulations 73.682 (a) (20)
	green	0.21	0.71	
	blue	0.14	0.08	
	red	0.67	0.33	
	white C	0.310	0.316	
5	primary	x	y	Rec. ITU-R BT.470-6 System B, G (historical) Rec. ITU-R BT.601-6 625 Rec. ITU-R BT.1358 625 Rec. ITU-R BT.1700 625 PAL and 625 SECAM
	green	0.29	0.60	
	blue	0.15	0.06	
	red	0.64	0.33	
	white D65	0.3127	0.3290	
6	primary	x	y	Rec. ITU-R BT.601-6 525 Rec. ITU-R BT.1358 525 Rec. ITU-R BT.1700 NTSC SMPTE 170M (2004) (functionally the same as the value 7)
	green	0.310	0.595	
	blue	0.155	0.070	
	red	0.630	0.340	
	white D65	0.3127	0.3290	
7	primary	x	y	SMPTE 240M (1999) (functionally the same as the value 6)
	green	0.310	0.595	
	blue	0.155	0.070	
	red	0.630	0.340	
	white D65	0.3127	0.3290	
8	primary	x	y	Generic film (colour filters using Illuminant C)
	green	0.243	0.692 (Wratten 58)	
	blue	0.145	0.049 (Wratten 47)	
	red	0.681	0.319 (Wratten 25)	
	white C	0.310	0.316	
9	primary	x	y	Rec. ITU-R BT.2020
	green	0.170	0.797	
	blue	0.131	0.046	
	red	0.708	0.292	
	white D65	0.3127	0.3290	
10	primary	x	y	SMPTE ST 428-1 (CIE 1931 XYZ)
	Y	0.0	1.0	
	Z	0.0	0.0	
	X	1.0	0.0	
	centre white	1 ÷ 3	1 ÷ 3	
11..255	Reserved			For future use by ITU-T | ISO/IEC

transfer_characteristics indicates the opto-electronic transfer characteristic of the source picture as specified in Table E.4 as a function of a linear optical intensity input Lc with a nominal real-valued range of 0 to 1. For interpretation of entries in Table E.4 that are expressed in terms of multiple curve segments parameterized by the variable α over a region bounded by the variable β or by the variables β and γ, the values of α and β are defined to be the positive constants necessary for the curve segments that meet at the value β to have continuity of value and continuity of slope at the value β, and the value of γ, when applicable, is defined to be the positive constant necessary for the associated curve segments to meet at the value γ. For example, for transfer_characteristics equal to 1, 6, 14 or 15, α has the value $1 + 5.5 * \beta = 1.099\,296\,826\,809\,442...$ and β has the value 0.018 053 968 510 807....

When the transfer_characteristics syntax element is not present, the value of transfer_characteristics is inferred to be equal to 2 (the transfer characteristics are unspecified or are determined by the application). Values of transfer_characteristics that are identified as reserved in Table E.4 are reserved for future use by ITU-T | ISO/IEC and shall not be present in bitstreams conforming to this version of this Specification. Decoders shall interpret reserved values of transfer_characteristics as equivalent to the value 2.

Table E.4 – Transfer characteristics

Value	Transfer characteristic		Informative remark
0	Reserved		For future use by ITU-T | ISO/IEC
1	$V = \alpha * L_c^{0.45} - (\alpha - 1)$ $V = 4.500 * L_c$	for $1 >= L_c >= \beta$ for $\beta > L_c >= 0$	Rec. ITU-R BT.709-5 Rec. ITU-R BT.1361 conventional colour gamut system (functionally the same as the values 6, 14 and 15)
2	Unspecified		Image characteristics are unknown or are determined by the application.
3	Reserved		For future use by ITU-T | ISO/IEC
4	Assumed display gamma 2.2		Rec. ITU-R BT.470-6 System M (historical) United States National Television System Committee (1953), Recommendation for transmission standards for colour television United States Federal Communications Commission (2003), Title 47 Code of Federal Regulations 73.682 (a) (20) Rec. ITU-R BT.1700 (2007 revision) 625 PAL and 625 SECAM
5	Assumed display gamma 2.8		Rec. ITU-R BT.470-6 System B, G (historical)
6	$V = \alpha * L_c^{0.45} - (\alpha - 1)$ $V = 4.500 * L_c$	for $1 >= L_c >= \beta$ for $\beta > L_c >= 0$	Rec. ITU-R BT.601-6 525 or 625 Rec. ITU-R BT.1358 525 or 625 Rec. ITU-R BT.1700 NTSC SMPTE 170M (2004) (functionally the same as the values 1, 14 and 15)
7	$V = \alpha * L_c^{0.45} - (\alpha - 1)$ $V = 4.0 * L_c$	for $1 >= L_c >= \beta$ for $\beta > L_c >= 0$	SMPTE 240M (1999)
8	$V = L_c$	for all values of L_c	Linear transfer characteristics
9	$V = 1.0 + \mathrm{Log10}(L_c) \div 2$ $V = 0.0$	for $1 >= L_c >= 0.01$ for $0.01 > L_c >= 0$	Logarithmic transfer characteristic (100:1 range)
10	$V = 1.0 + \mathrm{Log10}(L_c) \div 2.5$ $V = 0.0$	for $1 >= L_c >= \mathrm{Sqrt}(10) \div 1000$ for $\mathrm{Sqrt}(10) \div 1000 > L_c >= 0$	Logarithmic transfer characteristic (100 * Sqrt(10) : 1 range)
11	$V = \alpha * L_c^{0.45} - (\alpha - 1)$ $V = 4.500 * L_c$ $V = -\alpha * (-L_c)^{0.45} + (\alpha - 1)$	for $L_c >= \beta$ for $\beta > L_c > -\beta$ for $-\beta >= L_c$	IEC 61966-2-4
12	$V = \alpha * L_c^{0.45} - (\alpha - 1)$ $V = 4.500 * L_c$ $V = -(\alpha * (-4 * L_c)^{0.45} - (\alpha - 1)) \div 4$	for $1.33 > L_c >= \beta$ for $\beta > L_c >= -\gamma$ for $-\gamma > L_c >= -0.25$	Rec. ITU-R BT.1361 extended colour gamut system
13	$V = \alpha * L_c^{(1 \div 2.4)} - (\alpha - 1)$ $V = 12.92 * L_c$	for $1 >= L_c >= \beta$ for $\beta > L_c >= 0$	IEC 61966-2-1 (sRGB or sYCC)

Value	Transfer characteristic		Informative remark
14	$V = \alpha * L_c^{0.45} - (\alpha - 1)$ $V = 4.500 * L_c$	for $1 >= L_c >= \beta$ for $\beta > L_c >= 0$	Rec. ITU-R BT.2020 (functionally the same as the values 1, 6 and 15)
15	$V = \alpha * L_c^{0.45} - (\alpha - 1)$ $V = 4.500 * L_c$	for $1 >= L_c >= \beta$ for $\beta > L_c >= 0$	Rec. ITU-R BT.2020 (functionally the same as the values 1, 6 and 14)
16	$V = ((c_1 + c_2 * L_c^n) \div (1 + c_3 * L_c^n))^m$ $c_1 = c_3 - c_2 + 1 = 3424 \div 4096 = 0.8359375$ $c_2 = 32 * 2413 \div 4096 = 18.8515625$ $c_3 = 32 * 2392 \div 4096 = 18.6875$ $m = 128 * 2523 \div 4096 = 78.84375$ $n = 0.25 * 2610 \div 4096 = 0.1593017578125$ for which L_c equal to 1 for peak white is ordinarily intended to correspond to a display luminance level of 10 000 candelas per square metre	for all values of L_c	SMPTE ST 2084 for 10, 12, 14 and 16-bit systems.
17	$V = (48 * L_c \div 52.37)^{(1 \div 2.6)}$ for which L_c equal to 1 for peak white is ordinarily intended to correspond to a display luminance level of 48 candelas per square metre	for all values of L_c	SMPTE ST 428-1
18..255	Reserved		For future use by ITU-T | ISO/IEC

matrix_coeffs describes the matrix coefficients used in deriving luma and chroma signals from the green, blue and red or Y, Z and X primaries, as specified in Table E.5.

matrix_coeffs shall not be equal to 0 unless one or more of the following conditions are true:

– BitDepth_C is equal to BitDepth_Y.

– chroma_format_idc is equal to 3 (4:4:4).

The specification of the use of matrix_coeffs equal to 0 under all other conditions is reserved for future use by ITU-T | ISO/IEC.

matrix_coeffs shall not be equal to 8 unless one of the following conditions is true:

– BitDepth_C is equal to BitDepth_Y,

– BitDepth_C is equal to $\text{BitDepth}_Y + 1$ and chroma_format_idc is equal to 3 (4:4:4).

The specification of the use of matrix_coeffs equal to 8 under all other conditions is reserved for future use by ITU-T | ISO/IEC.

When the matrix_coeffs syntax element is not present, the value of matrix_coeffs is inferred to be equal to 2 (unspecified).

The interpretation of matrix_coeffs, together with colour_primaries and transfer_characteristics, is specified by the equations below.

NOTE 4 – For purposes of YZX representation when matrix_coeffs is equal to 0, the symbols R, G and B are substituted for X, Y and Z, respectively, in the following descriptions of Equations E-1 to E-3, E-7 to E-9, E-13 to E-15 and E-19 to E-21.

E_R, E_G and E_B are defined as "linear-domain" real-valued signals based on the indicated colour primaries before application of the transfer characteristics function. The application of the transfer characteristics function is denoted by $(x)'$ for an argument x. The signals E'_R, E'_G and E'_B are determined by application of the transfer characteristics function as follows:

$$E'_R = (E_R)' \qquad \text{(E-1)}$$

$$E'_G = (E_G)' \qquad \text{(E-2)}$$

$$E'_B = (E_B)' \qquad \text{(E-3)}$$

The range of E'_R, E'_G and E'_B is specified as follows:

– If transfer_characteristics is not equal to 11 or 12, E'_R, E'_G and E'_B are real numbers with values in the range of 0 to 1 inclusive.

– Otherwise, (transfer_characteristics is equal to 11 (IEC 61966-2-4) or 12 (Rec. ITU-R BT.1361 extended colour gamut system)), E'_R, E'_G and E'_B are real numbers with a larger range not specified in this Specification.

Nominal white is specified as having E'_R equal to 1, E'_G equal to 1 and E'_B equal to 1.

Nominal black is specified as having E'_R equal to 0, E'_G equal to 0 and E'_B equal to 0.

The interpretation of matrix_coeffs is specified as follows:

– If video_full_range_flag is equal to 0, the following applies:
  – If matrix_coeffs is equal to 1, 4, 5, 6, 7, 9 or 10, the following equations apply:

$$Y = \text{Clip1}_Y(\text{ Round}(\ (\ 1\ <<\ (\ \text{BitDepth}_Y - 8\ )\ )\ *\ (\ 219\ *\ E'_Y + 16\ )\ )\ ) \tag{E-4}$$

$$Cb = \text{Clip1}_C(\text{ Round}(\ (\ 1\ <<\ (\ \text{BitDepth}_C - 8\ )\ )\ *\ (\ 224\ *\ E'_{PB} + 128\ )\ )\ ) \tag{E-5}$$

$$Cr = \text{Clip1}_C(\text{ Round}(\ (\ 1\ <<\ (\ \text{BitDepth}_C - 8\ )\ )\ *\ (\ 224\ *\ E'_{PR} + 128\ )\ )\ ) \tag{E-6}$$

  – Otherwise, if matrix_coeffs is equal to 0 or 8, the following equations apply:

$$R = \text{Clip1}_Y(\ (\ 1\ <<\ (\ \text{BitDepth}_Y - 8\ )\ )\ *\ (\ 219\ *\ E'_R + 16\ )\ ) \tag{E-7}$$

$$G = \text{Clip1}_Y(\ (\ 1\ <<\ (\ \text{BitDepth}_Y - 8\ )\ )\ *\ (\ 219\ *\ E'_G + 16\ )\ ) \tag{E-8}$$

$$B = \text{Clip1}_Y(\ (\ 1\ <<\ (\ \text{BitDepth}_Y - 8\ )\ )\ *\ (\ 219\ *\ E'_B + 16\ )\ ) \tag{E-9}$$

  – Otherwise, if matrix_coeffs is equal to 2, the interpretation of the matrix_coeffs syntax element is unknown or is determined by the application.
  – Otherwise (matrix_coeffs is not equal to 0, 1, 2, 4, 5, 6, 7, 8, 9 or 10), the interpretation of the matrix_coeffs syntax element is reserved for future definition by ITU-T | ISO/IEC.

– Otherwise (video_full_range_flag is equal to 1), the following applies:
  – If matrix_coeffs is equal to 1, 4, 5, 6, 7, 9 or 10 the following equations apply:

$$Y = \text{Clip1}_Y(\text{ Round}(\ (\ (\ 1\ <<\ \text{BitDepth}_Y\ ) - 1\ )\ *\ E'_Y\ )\ ) \tag{E-10}$$

$$Cb = \text{Clip1}_C(\text{ Round}(\ (\ (\ 1\ <<\ \text{BitDepth}_C\ ) - 1\ )\ *\ E'_{PB} + (\ 1\ <<\ (\ \text{BitDepth}_C - 1\ )\ )\ )\ ) \tag{E-11}$$

$$Cr = \text{Clip1}_C(\text{ Round}(\ (\ (\ 1\ <<\ \text{BitDepth}_C\ ) - 1\ )\ *\ E'_{PR} + (\ 1\ <<\ (\ \text{BitDepth}_C - 1\ )\ )\ )\ ) \tag{E-12}$$

  – Otherwise, if matrix_coeffs is equal to 0 or 8, the following equations apply:

$$R = \text{Clip1}_Y(\ (\ (\ 1\ <<\ \text{BitDepth}_Y\ ) - 1\ )\ *\ E'_R\ ) \tag{E-13}$$

$$G = \text{Clip1}_Y(\ (\ (\ 1\ <<\ \text{BitDepth}_Y\ ) - 1\ )\ *\ E'_G\ ) \tag{E-14}$$

$$B = \text{Clip1}_Y(\ (\ (\ 1\ <<\ \text{BitDepth}_Y\ ) - 1\ )\ *\ E'_B\ ) \tag{E-15}$$

  – Otherwise, if matrix_coeffs is equal to 2, the interpretation of the matrix_coeffs syntax element is unknown or is determined by the application.
  – Otherwise (matrix_coeffs is not equal to 0, 1, 2, 4, 5, 6, 7, 8, 9 or 10), the interpretation of the matrix_coeffs syntax element is reserved for future definition by ITU-T | ISO/IEC. Reserved values for matrix_coeffs shall not be present in bitstreams conforming to this version of this Specification. Decoders shall interpret reserved values of matrix_coeffs as equivalent to the value 2.

The variables E'_Y, E'_{PB} and E'_{PR} (for matrix_coeffs not equal to 0 or 8) or Y, Cb and Cr (for matrix_coeffs equal to 0 or 8) are specified as follows:

– If matrix_coeffs is not equal to 0, 8 or 10, the following equations apply:

$$E'_Y = K_R * E'_R + (\ 1 - K_R - K_B\ ) * E'_G + K_B * E'_B \tag{E-16}$$

$$E'_{PB} = 0.5 * (\ E'_B - E'_Y\ ) \div (\ 1 - K_B\ ) \tag{E-17}$$

$$E'_{PR} = 0.5 * (\ E'_R - E'_Y\ ) \div (\ 1 - K_R\ ) \tag{E-18}$$

NOTE 5 – E'_Y is a real number with the value 0 associated with nominal black and the value 1 associated with nominal white. E'_{PB} and E'_{PR} are real numbers with the value 0 associated with both nominal black and nominal white. When transfer_characteristics is not equal to 11 or 12, E'_Y is a real number with values in the range of 0 to 1 inclusive. When transfer_characteristics is not equal to 11 or 12, E'_{PB} and E'_{PR} are real numbers with values in the range of −0.5 to 0.5 inclusive. When transfer_characteristics is equal to 11 (IEC 61966-2-4), or 12 (ITU-R BT.1361 extended colour gamut system), E'_Y, E'_{PB} and E'_{PR} are real numbers with a larger range not specified in this Specification.

– Otherwise, if matrix_coeffs is equal to 0, the following equations apply:

$$Y \;= \mathrm{Round}(\ G\ ) \tag{E-19}$$

$$Cb = \mathrm{Round}(\ B\ ) \tag{E-20}$$

$$Cr = \mathrm{Round}(\ R\ ) \tag{E-21}$$

– Otherwise, if matrix_coeffs is equal to 8, the following applies:

  – If $BitDepth_C$ is equal to $BitDepth_Y$, the following equations apply:

$$Y \;= \mathrm{Round}(\ 0.5 * G + 0.25 * (\ R + B\ )\ ) \tag{E-22}$$

$$Cb = \mathrm{Round}(\ 0.5 * G - 0.25 * (\ R + B\ )\ ) + (\ 1\ \ll\ (\ BitDepth_C - 1\ )\ ) \tag{E-23}$$

$$Cr = \mathrm{Round}(\ 0.5 * (R - B\ )\ ) + (\ 1\ \ll\ (\ BitDepth_C - 1\ )\ ) \tag{E-24}$$

NOTE 6 – For purposes of the YCgCo nomenclature used in Table E.5, Cb and Cr of Equations E-23 and E-24 may be referred to as Cg and Co, respectively. The inverse conversion for the above three equations should be computed as:

$$t \;= Y - (\ Cb - (\ 1\ \ll\ (\ BitDepth_C - 1\ )\ )\ ) \tag{E-25}$$

$$G = \mathrm{Clip1}_Y(\ Y + (\ Cb - (\ 1\ \ll\ (\ BitDepth_C - 1\ )\ )\ )\ ) \tag{E-26}$$

$$B = \mathrm{Clip1}_Y(\ t - (\ Cr - (\ 1\ \ll\ (\ BitDepth_C - 1\ )\ )\ )\ ) \tag{E-27}$$

$$R = \mathrm{Clip1}_Y(\ t + (\ Cr - (\ 1\ \ll\ (\ BitDepth_C - 1\ )\ )\ )\ ) \tag{E-28}$$

  – Otherwise ($BitDepth_C$ is not equal to $BitDepth_Y$), the following equations apply:

$$Cr = \mathrm{Round}(\ R\ ) - \mathrm{Round}(\ B\ ) + (\ 1\ \ll\ (\ BitDepth_C - 1\ )\ ) \tag{E-29}$$

$$t = \mathrm{Round}(\ B\ ) + (\ (\ Cr - (\ 1\ \ll\ (\ BitDepth_C - 1\ )\ )\ )\ \gg\ 1\ ) \tag{E-30}$$

$$Cb = \mathrm{Round}(\ G\ ) - t + (\ 1\ \ll\ (\ BitDepth_C - 1\ )\ ) \tag{E-31}$$

$$Y = t + (\ (\ Cb - (\ 1\ \ll\ (\ BitDepth_C - 1\ )\ )\ )\ \gg\ 1\ ) \tag{E-32}$$

NOTE 7 – For purposes of the YCgCo nomenclature used in Table E.5, Cb and Cr of Equations E-31 and E-29 may be referred to as Cg and Co, respectively. The inverse conversion for the above four equations should be computed as.

$$t \;= Y - (\ (\ Cb - (\ 1\ \ll\ (\ BitDepth_C - 1\ )\ )\ )\ \gg\ 1\ ) \tag{E-33}$$

$$G = \mathrm{Clip1}_Y(\ t + (\ Cb - (\ 1\ \ll\ (\ BitDepth_C - 1\ )\ )\ )\ ) \tag{E-34}$$

$$B = \mathrm{Clip1}_Y(\ t - (\ (\ Cr - (\ 1\ \ll\ (\ BitDepth_C - 1\ )\ )\ )\ \gg\ 1\ )\ ) \tag{E-35}$$

$$R = \mathrm{Clip1}_Y(\ B + (\ Cr - (\ 1\ \ll\ (\ BitDepth_C - 1\ )\ )\ )\ ) \tag{E-36}$$

– Otherwise (matrix_coeffs is equal to 10), the signal E'_Y is determined by application of the transfer characteristics function as follows and Equations E-39 to E-46 apply for specification of the signals E'_{PB} and E'_{PR}:

$$E_Y = K_R * E_R + (\ 1 - K_R - K_B\ ) * E_G + K_B * E_B \tag{E-37}$$

$$E'_Y = (\ E_Y\ )' \tag{E-38}$$

NOTE 8 – In this case, E_Y is defined from the "linear-domain" signals for E_R, E_G and E_B, prior to application of the transfer characteristics function, which is then applied to produce the signal E'_Y. E_Y and E'_Y are analogue with the value 0 associated with nominal black and the value 1 associated with nominal white.

$$E'_{PB} = (\ E'_B - E'_Y\ ) \div (\ 2 * N_B\ ) \qquad \text{for } - N_B\ \le\ E'_B - E'_Y\ \le\ 0 \tag{E-39}$$

$$E'_{PB} = (E'_B - E'_Y) \div (2 * P_B) \qquad \text{for } 0 < E'_B - E'_Y \ <= \ P_B \tag{E-40}$$

$$E'_{PR} = (E'_R - E'_Y) \div (2 * N_R) \qquad \text{for } - N_R \ <= \ E'_R - E'_Y \ <= \ 0 \tag{E-41}$$

$$E'_{PR} = (E'_R - E'_Y) \div (2 * P_R) \qquad \text{for } 0 < E'_R - E'_Y \ <= \ P_R \tag{E-42}$$

where the constants N_B, P_B, N_R and P_R are determined by application of the transfer characteristics function to expressions involving the constants K_B and K_R as follows:

$$N_B = (1 - K_B)' \tag{E-43}$$

$$P_B = 1 - (K_B)' \tag{E-44}$$

$$N_R = (1 - K_R)' \tag{E-45}$$

$$P_R = 1 - (K_R)' \tag{E-46}$$

Table E.5 – Matrix coefficients

Value	Matrix	Informative remark
0	Identity	The identity matrix. Typically used for GBR (often referred to as RGB); however, may also be used for YZX (often referred to as XYZ); see Equations E-19 to E-21 IEC 61966-2-1 (sRGB) SMPTE ST 428-1
1	$K_R = 0.2126$; $K_B = 0.0722$	ITU-R Rec. BT.709-5 ITU-R Rec. BT.1361 conventional colour gamut system and extended colour gamut system IEC 61966-2-1 (sYCC) IEC 61966-2-4 $xvYCC_{709}$ Annex B of SMPTE RP 177 (1993)
2	Unspecified	Image characteristics are unknown or are determined by the application.
3	Reserved	For future use by ITU-T | ISO/IEC
4	$K_R = 0.30$; $K_B = 0.11$	United States Federal Communications Commission Title 47 Code of Federal Regulations (2003) 73.682 (a) (20)
5	$K_R = 0.299$; $K_B = 0.114$	ITU-R Rec. BT.470-6 System B, G (historical) ITU-R Rec. BT.601-6 625 ITU-R Rec. BT.1358 625 ITU-R Rec. BT.1700 625 PAL and 625 SECAM IEC 61966-2-4 $xvYCC_{601}$ (functionally the same as the value 6)
6	$K_R = 0.299$; $K_B = 0.114$	ITU-R Rec. BT.601-6 525 ITU-R Rec. BT.1358 525 ITU-R Rec. BT.1700 NTSC SMPTE 170M (2004) (functionally the same as the value 5)
7	$K_R = 0.212$; $K_B = 0.087$	SMPTE 240M (1999)
8	YCgCo	See Equations E-22 to E-36
9	$K_R = 0.2627$; $K_B = 0.0593$	Rec. ITU-R BT.2020 non-constant luminance system See Equations E-16 to E-18
10	$K_R = 0.2627$; $K_B = 0.0593$	Rec. ITU-R BT.2020 constant luminance system See Equations E-37 to E-46
11..255	Reserved	For future use by ITU-T | ISO/IEC

chroma_loc_info_present_flag equal to 1 specifies that chroma_sample_loc_type_top_field and chroma_sample_loc_type_bottom_field are present. chroma_loc_info_present_flag equal to 0 specifies that chroma_sample_loc_type_top_field and chroma_sample_loc_type_bottom_field are not present.

When chroma_format_idc is not equal to 1, chroma_loc_info_present_flag should be equal to 0.

chroma_sample_loc_type_top_field and **chroma_sample_loc_type_bottom_field** specify the location of chroma samples as follows:

– If chroma_format_idc is equal to 1 (4:2:0 chroma format), chroma_sample_loc_type_top_field and chroma_sample_loc_type_bottom_field specify the location of chroma samples for the top field and the bottom field, respectively, as shown in Figure E.1.

– Otherwise (chroma_format_idc is not equal to 1), the values of the syntax elements chroma_sample_loc_type_top_field and chroma_sample_loc_type_bottom_field shall be ignored. When chroma_format_idc is equal to 2 (4:2:2 chroma format) or 3 (4:4:4 chroma format), the location of chroma samples is specified in clause 6.2. When chroma_format_idc is equal to 0, there is no chroma sample array.

The value of chroma_sample_loc_type_top_field and chroma_sample_loc_type_bottom_field shall be in the range of 0 to 5, inclusive. When the chroma_sample_loc_type_top_field and chroma_sample_loc_type_bottom_field are not present, the values of chroma_sample_loc_type_top_field and chroma_sample_loc_type_bottom_field are inferred to be equal to 0.

NOTE 9 – When coding progressive source material, chroma_sample_loc_type_top_field and chroma_sample_loc_type_bottom_field should have the same value.

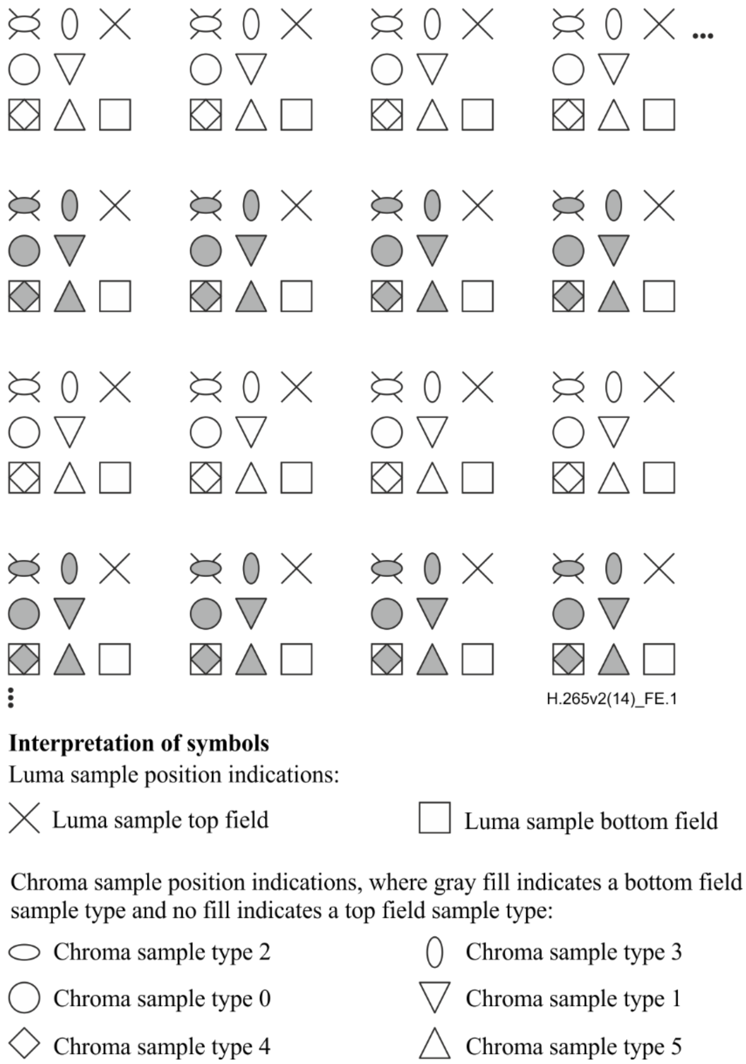

Figure E.1 – Location of chroma samples for top and bottom fields for chroma_format_idc equal to 1 (4:2:0 chroma format) as a function of chroma_sample_loc_type_top_field and chroma_sample_loc_type_bottom_field

neutral_chroma_indication_flag equal to 1 indicates that the value of all decoded chroma samples is equal to $1 << (\text{BitDepth}_C - 1)$. neutral_chroma_indication_flag equal to 0 provides no indication of decoded chroma sample values. When neutral_chroma_indication_flag is equal to 1, it is a requirement of bitstream conformance that the value of all decoded chroma samples produced by the decoding process shall be equal to $1 << (\text{BitDepth}_C - 1)$. When neutral_chroma_indication_flag is not present, it is inferred to be equal to 0.

NOTE 10 – When neutral_chroma_indication_flag is equal to 1, it is not necessary for the decoder to apply the specified decoding process in order to determine the value of the decoded chroma samples.

field_seq_flag equal to 1 indicates that the CVS conveys pictures that represent fields, and specifies that a picture timing SEI message shall be present in every access unit of the current CVS. field_seq_flag equal to 0 indicates that the CVS conveys pictures that represent frames and that a picture timing SEI message may or may not be present in any access unit of the current CVS. When field_seq_flag is not present, it is inferred to be equal to 0. When general_frame_only_constraint_flag is equal to 1, the value of field_seq_flag shall be equal to 0.

NOTE 11 – The specified decoding process does not treat access units conveying pictures that represent fields or frames differently. A sequence of pictures that represent fields would therefore be coded with the picture dimensions of an individual field. For example, access units containing pictures that represent 1080i fields would commonly have cropped output dimensions of 1920x540, while the sequence picture rate would commonly express the rate of the source fields (typically between 50 and 60 Hz), instead of the source frame rate (typically between 25 and 30 Hz).

frame_field_info_present_flag equal to 1 specifies that picture timing SEI messages are present for every picture and include the pic_struct, source_scan_type and duplicate_flag syntax elements. frame_field_info_present_flag equal to 0 specifies that the pic_struct syntax element is not present in picture timing SEI messages.

When frame_field_info_present_flag is present and either or both of the following conditions are true, frame_field_info_present_flag shall be equal to 1:

– field_seq_flag is equal to 1.

– general_progressive_source_flag is equal to 1 and general_interlaced_source_flag is equal to 1.

When frame_field_info_present_flag is not present, its value is inferred as follows:

– If general_progressive_source_flag is equal to 1 and general_interlaced_source_flag is equal to 1, frame_field_info_present_flag is inferred to be equal to 1.

– Otherwise, frame_field_info_present_flag is inferred to be equal to 0.

default_display_window_flag equal to 1 indicates that the default display window parameters follow next in the VUI. default_display_window_flag equal to 0 indicates that the default display window parameters are not present. The default display window parameters identify the area that is within the conformance cropping window and that is suggested to be displayed in the absence of any alternative indication (provided within the bitstream or by external means not specified in this Specification) of preferred display characteristics.

def_disp_win_left_offset, **def_disp_win_right_offset**, **def_disp_win_top_offset** and **def_disp_win_bottom_offset** specify the samples of the pictures in the CVS that are within the default display window, in terms of a rectangular region specified in picture coordinates for display. When default_display_window_flag is equal to 0, the values of def_disp_win_left_offset, def_disp_win_right_offset, def_disp_win_top_offset and def_disp_win_bottom_offset are inferred to be equal to 0.

The following variables are derived from the default display window parameters:

$$\text{leftOffset} = \text{conf\_win\_left\_offset} + \text{def\_disp\_win\_left\_offset} \tag{E-47}$$

$$\text{rightOffset} = \text{conf\_win\_right\_offset} + \text{def\_disp\_win\_right\_offset} \tag{E-48}$$

$$\text{topOffset} = \text{conf\_win\_top\_offset} + \text{def\_disp\_win\_top\_offset} \tag{E-49}$$

$$\text{bottomOffset} = \text{conf\_win\_bottom\_offset} + \text{def\_disp\_win\_bottom\_offset} \tag{E-50}$$

The default display window contains the luma samples with horizontal picture coordinates from SubWidthC * leftOffset to pic_width_in_luma_samples − (SubWidthC * rightOffset + 1) and vertical picture coordinates from SubHeightC * topOffset to pic_height_in_luma_samples − (SubHeightC * bottomOffset + 1), inclusive.

The value of SubWidthC * (leftOffset + rightOffset) shall be less than pic_width_in_luma_samples and the value of SubHeightC * (topOffset + bottomOffset) shall be less than pic_height_in_luma_samples.

When ChromaArrayType is not equal to 0, the corresponding specified samples of the two chroma arrays are the samples having picture coordinates (x / SubWidthC, y / SubHeightC), where (x, y) are the picture coordinates of the specified luma samples.

vui_timing_info_present_flag equal to 1 specifies that vui_num_units_in_tick, vui_time_scale, vui_poc_proportional_to_timing_flag and vui_hrd_parameters_present_flag are present in the vui_parameters( ) syntax structure. vui_timing_info_present_flag equal to 0 specifies that vui_num_units_in_tick, vui_time_scale, vui_poc_proportional_to_timing_flag and vui_hrd_parameters_present_flag are not present in the vui_parameters( ) syntax structure.

vui_num_units_in_tick is the number of time units of a clock operating at the frequency vui_time_scale Hz that corresponds to one increment (called a clock tick) of a clock tick counter. vui_num_units_in_tick shall be greater than 0. A clock tick, in units of seconds, is equal to the quotient of vui_num_units_in_tick divided by vui_time_scale. For example, when the picture rate of a video signal is 25 Hz, vui_time_scale may be equal to 27 000 000 and vui_num_units_in_tick may be equal to 1 080 000 and consequently a clock tick may be equal to 0.04 seconds.

When vps_num_units_in_tick is present in the VPS referred to by the SPS, vui_num_units_in_tick, when present, shall be equal to vps_num_units_in_tick, and when not present, is inferred to be equal to vps_num_units_in_tick.

vui_time_scale is the number of time units that pass in one second. For example, a time coordinate system that measures time using a 27 MHz clock has a vui_time_scale of 27 000 000. The value of vui_time_scale shall be greater than 0.

When vps_time_scale is present in the VPS referred to by the SPS, vui_time_scale, when present, shall be equal to vps_time_scale, and when not present, is inferred to be equal to vps_time_scale.

vui_poc_proportional_to_timing_flag equal to 1 indicates that the picture order count value for each picture in the CVS that is not the first picture in the CVS, in decoding order, is proportional to the output time of the picture relative to the output time of the first picture in the CVS. vui_poc_proportional_to_timing_flag equal to 0 indicates that the picture order count value for each picture in the CVS that is not the first picture in the CVS, in decoding order, may or may not be proportional to the output time of the picture relative to the output time of the first picture in the CVS.

When vps_poc_proportional_to_timing_flag is present in the VPS referred to by the SPS and the value is equal to 1, vui_poc_proportional_to_timing_flag, when present, shall be equal to 1.

vui_num_ticks_poc_diff_one_minus1 plus 1 specifies the number of clock ticks corresponding to a difference of picture order count values equal to 1. The value of vui_num_ticks_poc_diff_one_minus1 shall be in the range of 0 to $2^{32} - 2$, inclusive.

When vps_num_ticks_poc_diff_one_minus1 is present in the VPS referred to by the SPS, vui_num_ticks_poc_diff_one_minus1, when present, shall be equal to vps_num_ticks_poc_diff_one_minus1.

vui_hrd_parameters_present_flag equal to 1 specifies that the syntax structure hrd_parameters( ) is present in the vui_parameters( ) syntax structure. vui_hrd_parameters_present_flag equal to 0 specifies that the syntax structure hrd_parameters( ) is not present in the vui_parameters( ) syntax structure.

bitstream_restriction_flag equal to 1, specifies that the bitstream restriction parameters for the CVS are present. bitstream_restriction_flag equal to 0, specifies that the bitstream restriction parameters for the CVS are not present.

tiles_fixed_structure_flag equal to 1 indicates that each PPS that is active in the CVS has the same value of the syntax elements num_tile_columns_minus1, num_tile_rows_minus1, uniform_spacing_flag, column_width_minus1[i], row_height_minus1[i] and loop_filter_across_tiles_enabled_flag, when present. tiles_fixed_structure_flag equal to 0 indicates that tiles syntax elements in different PPSs may or may not have the same value. When the tiles_fixed_structure_flag syntax element is not present, it is inferred to be equal to 0.

NOTE 12 – The signalling of tiles_fixed_structure_flag equal to 1 is a guarantee to a decoder that each picture in the CVS has the same number of tiles distributed in the same way which might be useful for workload allocation in the case of multi-threaded decoding.

motion_vectors_over_pic_boundaries_flag equal to 0 indicates that no sample outside the picture boundaries and no sample at a fractional sample position for which the sample value is derived using one or more samples outside the picture boundaries is used for inter prediction of any sample. motion_vectors_over_pic_boundaries_flag equal to 1 indicates that one or more samples outside the picture boundaries may be used in inter prediction. When the motion_vectors_over_pic_boundaries_flag syntax element is not present, motion_vectors_over_pic_boundaries_flag value is inferred to be equal to 1.

restricted_ref_pic_lists_flag equal to 1 indicates that all P and B slices (when present) that belong to the same picture have an identical reference picture list 0 and that all B slices (when present) that belong to the same picture have an identical reference picture list 1.

min_spatial_segmentation_idc, when not equal to 0, establishes a bound on the maximum possible size of distinct coded spatial segmentation regions in the pictures of the CVS. When min_spatial_segmentation_idc is not present, it is inferred to be equal to 0. The value of min_spatial_segmentation_idc shall be in the range of 0 to 4095, inclusive.

The variable minSpatialSegmentationTimes4 is derived from min_spatial_segmentation_idc as follows:

$$\text{minSpatialSegmentationTimes4} = \text{min\_spatial\_segmentation\_idc} + 4 \tag{E-51}$$

A slice is said to contain a specific luma sample when the coding block that contains the luma sample is contained in the slice. Correspondingly, a tile is said to contain a specific luma sample when the coding block that contains the luma sample is contained in the tile.

Depending on the value of min_spatial_segmentation_idc, the following applies:

– If min_spatial_segmentation_idc is equal to 0, no limit on the maximum size of spatial segments is indicated.

– Otherwise (min_spatial_segmentation_idc is not equal to 0), it is a requirement of bitstream conformance that exactly one of the following conditions shall be true:

  – In each PPS that is activated within the CVS, tiles_enabled_flag is equal to 0 and entropy_coding_sync_enabled_flag is equal to 0 and there is no slice in the CVS that contains more than (4 * PicSizeInSamplesY) / minSpatialSegmentationTimes4 luma samples.

  – In each PPS that is activated within the CVS, tiles_enabled_flag is equal to 1 and entropy_coding_sync_enabled_flag is equal to 0 and there is no tile in the CVS that contains more than (4 * PicSizeInSamplesY) / minSpatialSegmentationTimes4 luma samples.

  – In each PPS that is activated within the CVS, tiles_enabled_flag is equal to 0 and entropy_coding_sync_enabled_flag is equal to 1 and the syntax elements pic_width_in_luma_samples, pic_height_in_luma_samples and the variable CtbSizeY obey the following constraint:

$$\begin{aligned}&(2 * \text{pic\_height\_in\_luma\_samples} + \text{pic\_width\_in\_luma\_samples}) * \text{CtbSizeY}\\ &\quad <= (4 * \text{PicSizeInSamplesY}) / \text{minSpatialSegmentationTimes4}\end{aligned} \tag{E-52}$$

NOTE 13 – The syntax element min_spatial_segmentation_idc can be used by a decoder to calculate the maximum number of luma samples to be processed by one processing thread, making the assumption that the decoder maximally utilizes the parallel decoding information. However, it is important to be aware that there may be some inter-dependencies between the different threads – e.g., due to entropy coding synchronization or deblocking filtering across tile or slice boundaries. To aid decoders in planning the decoding workload distribution, encoders are encouraged to set the value of min_spatial_segmentation_idc to the highest possible value for which one of the above three conditions is true. For example, for the case when tiles_enabled_flag is equal to 0 and entropy_coding_sync_enabled_flag is equal to 1, encoders should set min_spatial_segmentation_idc equal to 4 * PicSizeInSamplesY / ((2 * pic_height_in_luma_samples + pic_width_in_luma_samples) * CtbSizeY) − 4.

max_bytes_per_pic_denom indicates a number of bytes not exceeded by the sum of the sizes of the VCL NAL units associated with any coded picture in the CVS.

The number of bytes that represent a picture in the NAL unit stream is specified for this purpose as the total number of bytes of VCL NAL unit data (i.e., the total of the NumBytesInNalUnit variables for the VCL NAL units) for the picture. The value of max_bytes_per_pic_denom shall be in the range of 0 to 16, inclusive.

Depending on the value of max_bytes_per_pic_denom the following applies:

– If max_bytes_per_pic_denom is equal to 0, no limits are indicated.

– Otherwise (max_bytes_per_pic_denom is not equal to 0), it is a requirement of bitstream conformance that no coded picture shall be represented in the CVS by more than the following number of bytes.

$$(\text{PicSizeInMinCbsY} * \text{RawMinCuBits}) \div (8 * \text{max\_bytes\_per\_pic\_denom}) \tag{E-53}$$

When the max_bytes_per_pic_denom syntax element is not present, the value of max_bytes_per_pic_denom is inferred to be equal to 2.

max_bits_per_min_cu_denom indicates an upper bound for the number of coded bits of coding_unit( ) data for any coding block in any picture of the CVS. The value of max_bits_per_min_cu_denom shall be in the range of 0 to 16, inclusive.

Depending on the value of max_bits_per_min_cu_denom, the following applies:

– If max_bits_per_min_cu_denom is equal to 0, no limit is specified by this syntax element.

– Otherwise (max_bits_per_min_cu_denom is not equal to 0), it is a requirement of bitstream conformance that no coded coding_unit( ) shall be represented in the bitstream by more than the following number of bits:

$$(128 + \text{RawMinCuBits}) \div \text{max\_bits\_per\_min\_cu\_denom} \\ * (1 \ll (2 * (\text{log2CbSize} - \text{MinCbLog2SizeY}))) \quad \text{(E-54)}$$

where log2CbSize is the value of log2CbSize for the given coding block and the number of bits of coding_unit( ) data for the same coding block is given by the number of times read_bits(1) is called in clauses 9.3.4.3.3 and 9.3.4.3.4.

When the max_bits_per_min_cu_denom is not present, the value of max_bits_per_min_cu_denom is inferred to be equal to 1.

log2_max_mv_length_horizontal and **log2_max_mv_length_vertical** indicate the maximum absolute value of a decoded horizontal and vertical motion vector component, respectively, in quarter luma sample units, for all pictures in the CVS. A value of n asserts that no value of a motion vector component is outside the range of -2^n to $2^n - 1$, inclusive, in units of quarter luma sample displacement. The value of log2_max_mv_length_horizontal shall be in the range of 0 to 16, inclusive. The value of log2_max_mv_length_vertical shall be in the range of 0 to 15, inclusive. When log2_max_mv_length_horizontal is not present, the values of log2_max_mv_length_horizontal and log2_max_mv_length_vertical is inferred to be equal to 15.

E.3.2 HRD parameters semantics

The hrd_parameters( ) syntax structure provides HRD parameters used in the HRD operations for a layer set. When the hrd_parameters( ) syntax structure is included in a VPS, the applicable layer set to which the hrd_parameters( ) syntax structure applies is specified by the corresponding hrd_layer_set_idx[i] syntax element in the VPS. When the hrd_parameters( ) syntax structure is included in an SPS, the layer set to which the hrd_parameters( ) syntax structure applies is the layer set for which the associated layer identifier list contains all nuh_layer_id values present in the CVS.

For interpretation of the following semantics, the bitstream (or a part thereof) refers to the bitstream subset (or a part thereof) associated with the layer set to which the hrd_parameters( ) syntax structure applies.

nal_hrd_parameters_present_flag equal to 1 specifies that NAL HRD parameters (pertaining to Type II bitstream conformance) are present in the hrd_parameters( ) syntax structure. nal_hrd_parameters_present_flag equal to 0 specifies that NAL HRD parameters are not present in the hrd_parameters( ) syntax structure.

NOTE 1 – When nal_hrd_parameters_present_flag is equal to 0, the conformance of the bitstream cannot be verified without provision of the NAL HRD parameters and all buffering period and picture timing SEI messages, by some means not specified in this Specification.

The variable NalHrdBpPresentFlag is derived as follows:

– If one or more of the following conditions are true, the value of NalHrdBpPresentFlag is set equal to 1:
  – nal_hrd_parameters_present_flag is present in the bitstream and is equal to 1.
  – The need for presence of buffering periods for NAL HRD operation to be present in the bitstream in buffering period SEI messages is determined by the application, by some means not specified in this Specification.
– Otherwise, the value of NalHrdBpPresentFlag is set equal to 0.

vcl_hrd_parameters_present_flag equal to 1 specifies that VCL HRD parameters (pertaining to all bitstream conformance) are present in the hrd_parameters( ) syntax structure. vcl_hrd_parameters_present_flag equal to 0 specifies that VCL HRD parameters are not present in the hrd_parameters( ) syntax structure.

NOTE 2 – When vcl_hrd_parameters_present_flag is equal to 0, the conformance of the bitstream cannot be verified without provision of the VCL HRD parameters and all buffering period and picture timing SEI messages, by some means not specified in this Specification.

The variable VclHrdBpPresentFlag is derived as follows:

– If one or more of the following conditions are true, the value of VclHrdBpPresentFlag is set equal to 1:
  – vcl_hrd_parameters_present_flag is present in the bitstream and is equal to 1.
  – The need for presence of buffering periods for VCL HRD operation to be present in the bitstream in buffering period SEI messages is determined by the application, by some means not specified in this Specification.
– Otherwise, the value of VclHrdBpPresentFlag is set equal to 0.

The variable CpbDpbDelaysPresentFlag is derived as follows:

– If one or more of the following conditions are true, the value of CpbDpbDelaysPresentFlag is set equal to 1:

– nal_hrd_parameters_present_flag is present in the bitstream and is equal to 1.

– vcl_hrd_parameters_present_flag is present in the bitstream and is equal to 1.

– The need for presence of CPB and DPB output delays to be present in the bitstream in picture timing SEI messages is determined by the application, by some means not specified in this Specification.

– Otherwise, the value of CpbDpbDelaysPresentFlag is set equal to 0.

When NalHrdBpPresentFlag and VclHrdBpPresentFlag are both equal to 0, the value of CpbDpbDelaysPresentFlag shall be equal to 0.

sub_pic_hrd_params_present_flag equal to 1 specifies that sub-picture level HRD parameters are present and the HRD may operate at access unit level or sub-picture level. sub_pic_hrd_params_present_flag equal to 0 specifies that sub-picture level HRD parameters are not present and the HRD operates at access unit level. When sub_pic_hrd_params_present_flag is not present, its value is inferred to be equal to 0.

tick_divisor_minus2 is used to specify the clock sub-tick. A clock sub-tick is the minimum interval of time that can be represented in the coded data when sub_pic_hrd_params_present_flag is equal to 1.

du_cpb_removal_delay_increment_length_minus1 plus 1 specifies the length, in bits, of the du_cpb_removal_delay_increment_minus1[i] and du_common_cpb_removal_delay_increment_minus1 syntax elements of the picture timing SEI message and the du_spt_cpb_removal_delay_increment syntax element in the decoding unit information SEI message.

sub_pic_cpb_params_in_pic_timing_sei_flag equal to 1 specifies that sub-picture level CPB removal delay parameters are present in picture timing SEI messages and no decoding unit information SEI message is available (in the CVS or provided through external means not specified in this Specification). sub_pic_cpb_params_in_pic_timing_sei_flag equal to 0 specifies that sub-picture level CPB removal delay parameters are present in decoding unit information SEI messages and picture timing SEI messages do not include sub-picture level CPB removal delay parameters. When the sub_pic_cpb_params_in_pic_timing_sei_flag syntax element is not present, it is inferred to be equal to 0.

dpb_output_delay_du_length_minus1 plus 1 specifies the length, in bits, of the pic_dpb_output_du_delay syntax element in the picture timing SEI message and the pic_spt_dpb_output_du_delay syntax element in the decoding unit information SEI message.

bit_rate_scale (together with bit_rate_value_minus1[i]) specifies the maximum input bit rate of the i-th CPB.

cpb_size_scale (together with cpb_size_value_minus1[i]) specifies the CPB size of the i-th CPB when the CPB operates at the access unit level.

cpb_size_du_scale (together with cpb_size_du_value_minus1[i]) specifies the CPB size of the i-th CPB when the CPB operates at sub-picture level.

initial_cpb_removal_delay_length_minus1 plus 1 specifies the length, in bits, of the nal_initial_cpb_removal_delay[i], nal_initial_cpb_removal_offset[i], vcl_initial_cpb_removal_delay[i] and vcl_initial_cpb_removal_offset[i] syntax elements of the buffering period SEI message. When the initial_cpb_removal_delay_length_minus1 syntax element is not present, it is inferred to be equal to 23.

au_cpb_removal_delay_length_minus1 plus 1 specifies the length, in bits, of the cpb_delay_offset syntax element in the buffering period SEI message and the au_cpb_removal_delay_minus1 syntax element in the picture timing SEI message. When the au_cpb_removal_delay_length_minus1 syntax element is not present, it is inferred to be equal to 23.

dpb_output_delay_length_minus1 plus 1 specifies the length, in bits, of the dpb_delay_offset syntax element in the buffering period SEI message and the pic_dpb_output_delay syntax element in the picture timing SEI message. When the dpb_output_delay_length_minus1 syntax element is not present, it is inferred to be equal to 23.

fixed_pic_rate_general_flag[i] equal to 1 indicates that, when HighestTid is equal to i, the temporal distance between the HRD output times of consecutive pictures in output order is constrained as specified below. fixed_pic_rate_general_flag[i] equal to 0 indicates that this constraint may not apply.

When fixed_pic_rate_general_flag[i] is not present, it is inferred to be equal to 0.

fixed_pic_rate_within_cvs_flag[i] equal to 1 indicates that, when HighestTid is equal to i, the temporal distance between the HRD output times of consecutive pictures in output order is constrained as specified below. fixed_pic_rate_within_cvs_flag[i] equal to 0 indicates that this constraint may not apply.

When fixed_pic_rate_general_flag[i] is equal to 1, the value of fixed_pic_rate_within_cvs_flag[i] is inferred to be equal to 1.

elemental_duration_in_tc_minus1[i] plus 1 (when present) specifies, when HighestTid is equal to i, the temporal distance, in clock ticks, between the elemental units that specify the HRD output times of consecutive pictures in output order as specified below. The value of elemental_duration_in_tc_minus1[i] shall be in the range of 0 to 2 047, inclusive.

For each picture n that is output and not the last picture in the bitstream (in output order) that is output, the value of the variable DpbOutputElementalInterval[n] is specified by:

$$\text{DpbOutputElementalInterval[n]} = \text{DpbOutputInterval[n]} \div \text{DeltaToDivisor} \quad \text{(E-55)}$$

where DpbOutputInterval[n] is specified in Equation C-18 and DeltaToDivisor is specified in Table E.6 based on the value of frame_field_info_present_flag and pic_struct for the CVS containing picture n. Entries marked "-" in Table E.6 indicate a lack of dependence of DeltaToDivisor on the corresponding syntax element.

When HighestTid is equal to i and fixed_pic_rate_general_flag[i] is equal to 1 for a CVS containing picture n, the value computed for DpbOutputElementalInterval[n] shall be equal to ClockTick * (elemental_duration_in_tc_minus1[i] + 1), wherein ClockTick is as specified in Equation C-3 (using the value of ClockTick for the CVS containing picture n) when one of the following conditions is true for the following picture in output order nextPicInOutputOrder that is specified for use in Equation C-18:

– picture nextPicInOutputOrder is in the same CVS as picture n.

– picture nextPicInOutputOrder is in a different CVS and fixed_pic_rate_general_flag[i] is equal to 1 in the CVS containing picture nextPicInOutputOrder, the value of ClockTick is the same for both CVSs and the value of elemental_duration_in_tc_minus1[i] is the same for both CVSs.

When HighestTid is equal to i and fixed_pic_rate_within_cvs_flag[i] is equal to 1 for a CVS containing picture n, the value computed for DpbOutputElementalInterval[n] shall be equal to ClockTick * (elemental_duration_in_tc_minus1[i] + 1), wherein ClockTick is as specified in Equation C-3 (using the value of ClockTick for the CVS containing picture n) when the following picture in output order nextPicInOutputOrder that is specified for use in Equation C-18 is in the same CVS as picture n.

Table E.6 – Divisor for computation of DpbOutputElementalInterval[n]

frame_field_info_present_flag	pic_struct	DeltaToDivisor
0	-	1
1	1	1
1	2	1
1	0	1
1	3	2
1	4	2
1	5	3
1	6	3
1	7	2
1	8	3
1	9	1
1	10	1
1	11	1
1	12	1

low_delay_hrd_flag[i] specifies the HRD operational mode, when HighestTid is equal to i, as specified in Annex C or clause F.13. When not present, the value of low_delay_hrd_flag[i] is inferred to be equal to 0.

NOTE 3 – When low_delay_hrd_flag[i] is equal to 1, "big pictures" that violate the nominal CPB removal times due to the number of bits used by an access unit are permitted. It is expected, but not required, that such "big pictures" occur only occasionally.

cpb_cnt_minus1[i] plus 1 specifies the number of alternative CPB specifications in the bitstream of the CVS when HighestTid is equal to i. The value of cpb_cnt_minus1[i] shall be in the range of 0 to 31, inclusive. When not present, the value of cpb_cnt_minus1[i] is inferred to be equal to 0.

E.3.3 Sub-layer HRD parameters semantics

The variable CpbCnt is set equal to cpb_cnt_minus1[subLayerId].

bit_rate_value_minus1[i] (together with bit_rate_scale) specifies the maximum input bit rate for the i-th CPB when the CPB operates at the access unit level. bit_rate_value_minus1[i] shall be in the range of 0 to $2^{32} - 2$, inclusive. For any i > 0, bit_rate_value_minus1[i] shall be greater than bit_rate_value_minus1[i − 1].

When SubPicHrdFlag is equal to 0, the bit rate in bits per second is given by:

$$\text{BitRate[i]} = (\text{ bit\_rate\_value\_minus1[i]} + 1\ ) * 2^{(\ 6\ +\ \text{bit\_rate\_scale}\ )} \qquad \text{(E-56)}$$

When SubPicHrdFlag is equal to 0 and the bit_rate_value_minus1[i] syntax element is not present, the value of BitRate[i] is inferred to be equal to BrVclFactor * MaxBR for VCL HRD parameters and to be equal to BrNalFactor * MaxBR for NAL HRD parameters, where MaxBR, BrVclFactor and BrNalFactor are specified in clause A.4.

cpb_size_value_minus1[i] is used together with cpb_size_scale to specify the i-th CPB size when the CPB operates at the access unit level. cpb_size_value_minus1[i] shall be in the range of 0 to $2^{32} - 2$, inclusive. For any i greater than 0, cpb_size_value_minus1[i] shall be less than or equal to cpb_size_value_minus1[i − 1].

When SubPicHrdFlag is equal to 0, the CPB size in bits is given by:

$$\text{CpbSize[i]} = (\text{ cpb\_size\_value\_minus1[i]} + 1\ ) * 2^{(\ 4\ +\ \text{cpb\_size\_scale}\ )} \qquad \text{(E-57)}$$

When SubPicHrdFlag is equal to 0 and the cpb_size_value_minus1[i] syntax element is not present, the value of CpbSize[i] is inferred to be equal to CpbVclFactor * MaxCPB for VCL HRD parameters and to be equal to CpbNalFactor * MaxCPB for NAL HRD parameters, where MaxCPB, CpbVclFactor and CpbNalFactor are specified in clause A.4.

cpb_size_du_value_minus1[i] is used together with cpb_size_du_scale to specify the i-th CPB size when the CPB operates at sub-picture level. cpb_size_du_value_minus1[i] shall be in the range of 0 to $2^{32} - 2$, inclusive. For any i greater than 0, cpb_size_du_value_minus1[i] shall be less than or equal to cpb_size_du_value_minus1[i − 1].

When SubPicHrdFlag is equal to 1, the CPB size in bits is given by:

$$\text{CpbSize[i]} = (\text{ cpb\_size\_du\_value\_minus1[i]} + 1\ ) * 2^{(\ 4\ +\ \text{cpb\_size\_du\_scale}\ )} \qquad \text{(E-58)}$$

When SubPicHrdFlag is equal to 1 and the cpb_size_du_value_minus1[i] syntax element is not present, the value of CpbSize[i] is inferred to be equal to CpbVclFactor * MaxCPB for VCL HRD parameters and to be equal to CpbNalFactor * MaxCPB for NAL HRD parameters, where MaxCPB, CpbVclFactor and CpbNalFactor are specified in clause A.4.

bit_rate_du_value_minus1[i] (together with bit_rate_scale) specifies the maximum input bit rate for the i-th CPB when the CPB operates at the sub-picture level. bit_rate_du_value_minus1[i] shall be in the range of 0 to $2^{32} - 2$, inclusive. For any i > 0, bit_rate_du_value_minus1[i] shall be greater than bit_rate_du_value_minus1[i − 1].

When SubPicHrdFlag is equal to 1, the bit rate in bits per second is given by:

$$\text{BitRate[i]} = (\text{ bit\_rate\_du\_value\_minus1[i]} + 1\ ) * 2^{(\ 6\ +\ \text{bit\_rate\_scale}\ )} \qquad \text{(E-59)}$$

When SubPicHrdFlag is equal to 1 and the bit_rate_du_value_minus1[i] syntax element is not present, the value of BitRate[i] is inferred to be equal to BrVclFactor * MaxBR for VCL HRD parameters and to be equal to BrNalFactor * MaxBR for NAL HRD parameters, where MaxBR, BrVclFactor and BrNalFactor are specified in clause A.4.

cbr_flag[i] equal to 0 specifies that to decode this CVS by the HRD using the i-th CPB specification, the hypothetical stream scheduler (HSS) operates in an intermittent bit rate mode. cbr_flag[i] equal to 1 specifies that the HSS operates in a constant bit rate (CBR) mode. When not present, the value of cbr_flag[i] is inferred to be equal to 0.