Joint Collaborative Team on Video Coding (JCT-VC)
of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11 Document: JCTVC-N0291
14th Meeting: Vienna, AT, 25 July – 2 August 2013

*Title:* Best-effort decoding of 10-bit sequences: use-cases, requirements and specification methods

*Status:* Input Document to JCT-VC

*Purpose:* Proposal

*Author(s):* David Flynn | *dflynn@blackberry.com*
Gaëlle Martin-Cocher | *gmartincocher@blackberry.com*
Dake He | *dhe@blackberry.com*

*Source:* BlackBerry Ltd. / Research In Motion

## Abstract

HEVC, like other well-known codecs, contains profiles that accommodate the decoding of non-8-bit video sequences. In particular, HEVC contains a 10-bit profile that permits the carriage of 10-bit video sequences. In M0255, it was demonstrated that it is possible to re-purpose an existing 8-bit decoder design to decode 10-bit bitstreams. Moreover, with judicious application of rounding control, it is possible to reduce the accumulation of decoder error than might otherwise be expected.

This contribution provides a requirements basis and use case information, summarises the previous work, and suggests methods of text specification.

# 1 Use-cases and requirements

The guidance document [1] providing the vision, applications and requirements for the development of HEVC suggests that "complexity scalability in the encoder and decoder is desirable".

Currently the scope of decoder complexity scalability is extremely limited – there exist two profiles with identical tool sets with the only difference being the bit-depth at which processing occurs. In effect, the system scalability exists wholly with the encoder.

Adding a best-effort decoding solution that permits the decoding of Main-10 bitstreams with the resources of an 8-bit decoder would go some way to providing some balance to the receiving application.

Best-effort bit-depth decoding would permit grater flexibility in a number of application areas:

- To maintain compatibility between devices for user generated content, where different conformance points can lead to frustration at being unable to straightforwardly exchange content and the user would rather prefer to see a best effort decoding of their own generated content than to be unable to play it on their own device.
- To better manage the user experience based upon the capabilities of the decoder. A principal application of which is to permit devices that are subject to a plethora of different bitstream sources to provide a degraded display for bitstreams that exceed the bit-depth processing capabilities of the decoder, rather than being forced to offer no output while maintaining conformance to the specification.
- To provide more efficient creation of "browse quality" representations.
  In some environments (particularly in broadcast areas), it is common to maintain a repository of high quality master video content and to produce on-demand lower quality "browse" versions for use in research and rough-editing. Using a lower complexity decoding process can reduce the computational demands of these on-demand systems.
  Similarly, camera back viewing such as on consumer devices, allows the review of material using (often) a low resolution screen. Given the inevitable demand for 10-bit reproduction, it may be desirable to use a lower complexity decoder for decoding on the camera back for quick review so as to manage device costs.

- To permit hybrid transmission schemes, where a single bitstream is intentionally dual-purposed for both 8- and 10-bit decoders, allowing an encoder to control the drift encountered by the reduced complexity decoding scheme. Such techniques may be of interest when performing transitions to higher bit-depth delivery systems without having to implement fully bit-depth scalable delivery.

Mobile phones and other portable consumer equipment are often constrained in the type of decoder resource they can support where factors such as memory bandwidth can be of a particular concern (in both hardware and software). It is quite likely that some devices that do not support an HEVC hardware decoder will provide decoding through a software decoder. Reducing the memory bandwidth and word widths in such systems can be fundamental to achieving a real-time decoding constraint, especially where the effects of 10-bit video can severely impact software decoders due to the limited discrete word sizes available.

One notable effect of best-effort decoding techniques defined to apply to previously defined profiles is the departure of bit-exact decoding. Referring back to the guidance document, we do not believe that this is currently precluded, since, "the full decoding process should be specified, preferably with no mismatch".

It could also be said that the guidance also suggests, should such best-effort capabilities be provided by the standard, specifying them as fully as possible, preferably in a bit-exact manner.

### 1.1 Existing solutions

It has been suggested that scalability may be a suitable panacea for such applications. However, such systems tend not to be aligned to the same requirements principals, tending to require higher complexity decoders and transmission that is a burden for the more capable system, along with an associated bit-rate penalty. However, more fundamentally, they fail to address the practical problem of there existing single layer 10-bit bitstreams which would remain undecodable by an 8-bit decoder.

### 1.2 Limitation of scope

Currently HEVC only provides 10-bit and 8-bit profiles. The best-effort decoding techniques are equally applicable to other arbitrary bit-depth combinations. However, it is suggested to narrow the scope of initial work to focus on the 10-bit to 8-bit process, schemes which are of themselves both well understood in their processing and requirements.

## 2 Summary of previous work using rounding control

The contribution JCTVC-M0255 [2] provides two methods for decoding a 10-bit sequence using a modified 8-bit decoder. The first method is a straight forward adjustment that re-interprets the system bitdepth in the transform and subsequent stages to be 8-bit. The only extra burden that the decoder must handle is to decode any signalled SAO offsets using a truncated unary code configured for 10-bit operation. This method, while straightforward, is characterized by DC drift in intra pictures that accumulates towards the bottom right of the picture, and by drift in colour and/or saturation that evolves over many frames.

The second method proposed (see Figure 1) does not modify the transform behaviour, rather, it introduces a rounding stage after the residual is combined with the prediction and a consequential bit-depth increase during the prediction process. The rounding stage is implemented using round-to-even to maintain the mean expected signal value and thereby reduce the drift accumulation through the removal of biases. This design is particularly effective, even for 8-bit software decoders, since many calculations in the transform, prediction and combining processes must be calculated at a bit-depth higher than 8-bit. Ie, capabilities of the current design are exploited.

To examine the modified decoder behaviour in unfavourable conditions, the HM-10 low-delay-B main-10 sequences [3] were decoded and the PSNR measured against the original input sequences. Table 1 illustrates the method 2 per-sequence losses when compared against the PSNR of the normal decoder[1]. Figure 3 shows how the PSNR loss evolves as the sequence progresses. Visual inspection of the decoded video shows that the gross DC drifts observed using the first method are avoided, to the extent that under single stimulus conditions it can be difficult to observe a degradation.

[1] The Cactus sequence used in figure 2 is marked in black

Figure 1 – Illustrative signal coding path of hybrid 8-bit–10-bit decoder

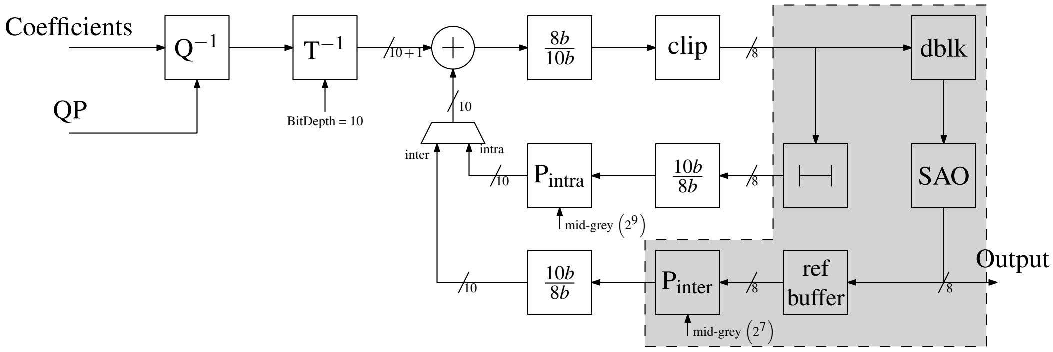



Figure 2 illustrates the typical drift effect observed when using the second method compared to the vanilla HM-10 decoder after decoding the ten-second 500-frame sequence Cactus encoded using the low-delay main-10 configuration at QP=27. The figure shows the first intra and the last inter frame of the sequence. Video sequences using both methods are available at the meeting for viewing.

Figure 2 – Typical distortion introduced by 8-bit decoding of 10-bit sequences. Sequence cactus QP=27, first frame (left), last frame (right). (a,b) Method–2, (c,d) HM-10.

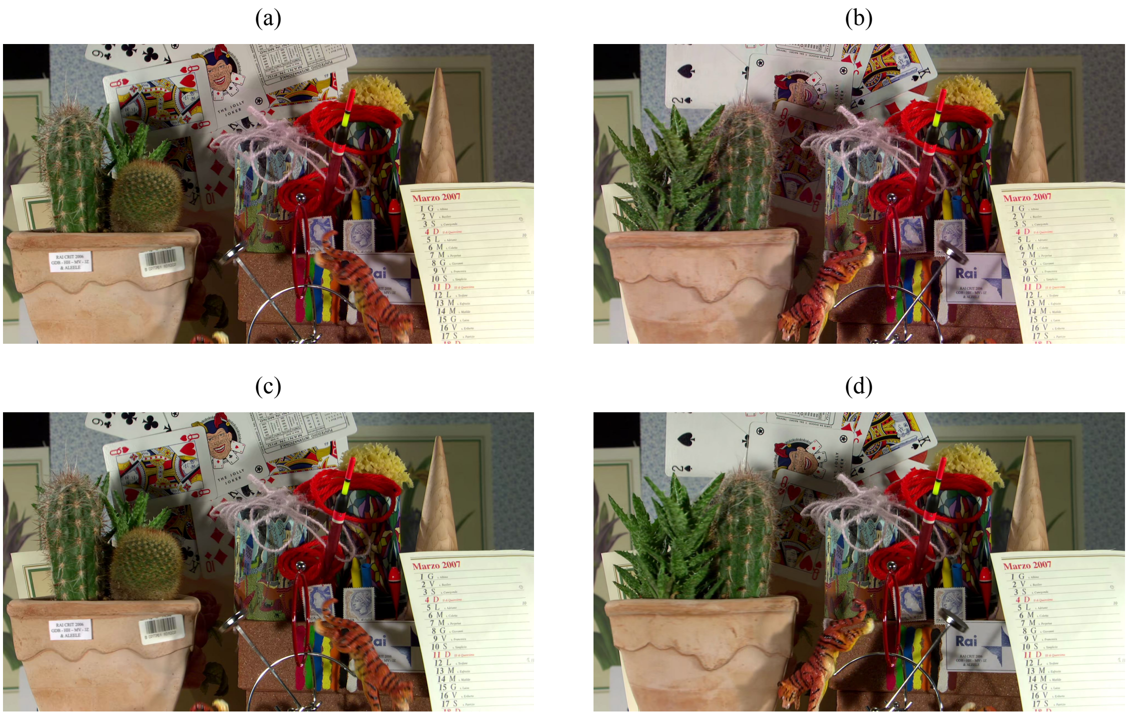
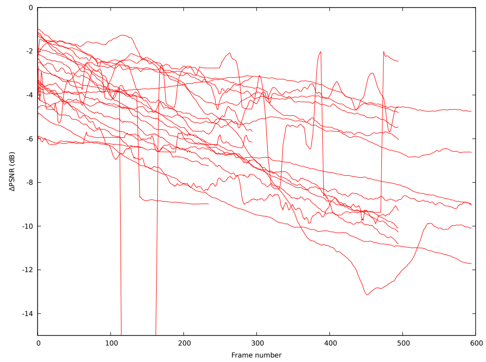
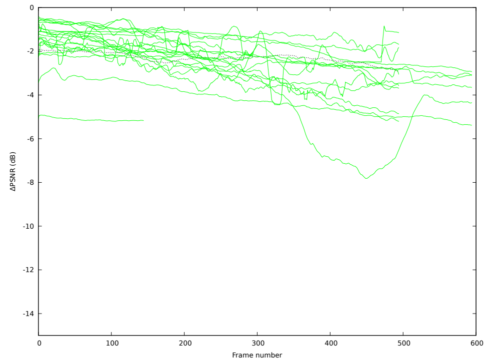


Figure 3 – Plot of frame-number versus ΔPSNR for all sequences at particular QPs using Method–2.

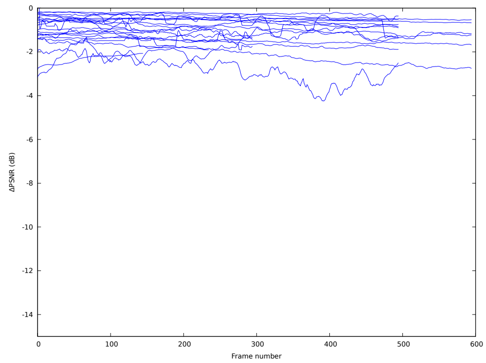
(a) QP 22




(b) QP 27




(c) QP 32




(d) QP 37

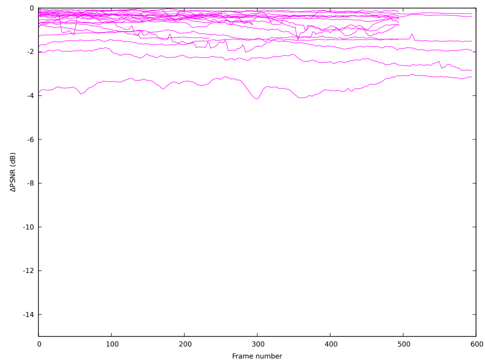



Table 1 – PSNR loss associated with method–2 of JCTVC-M0255, averaged over 10s sequences at QPs 22/27/32/37 using low-delay main-10 configuration

| Sequence | Luma PSNR | Chroma PSNR |
|---|---|---|
| BQMall | −7.1/−3.5/−1.1/−0.6 | −3.7/−2.7/−1.5/−1.2 |
| BQSquare | −3.9/−1.7/−0.3/−0.2 | −4.7/−2.0/−0.8/−0.6 |
| BQTerrace | −6.3/−2.6/−0.8/−1.3 | −4.2/−3.2/−1.7/−1.0 |
| BasketballDrillText | −7.2/−2.8/−0.6/−0.4 | −5.4/−2.9/−1.0/−0.4 |
| BasketballDrill | −6.5/−3.1/−0.5/−0.5 | −5.3/−4.1/−1.5/−0.3 |
| BasketballDrive | −3.9/−2.3/−1.7/−0.7 | −5.8/−2.6/−2.3/−0.9 |
| BasketballPass | −4.0/−1.3/−0.3/−0.2 | −2.8/−1.0/−0.4/−0.3 |
| BlowingBubbles | −3.5/−1.3/−0.5/−0.3 | −4.6/−1.7/−1.4/−0.8 |
| Cactus | −6.2/−2.3/−1.5/−1.5 | −5.1/−4.1/−1.5/−0.8 |
| ChinaSpeed | −7.7/−3.0/−2.8/−0.6 | −8.5/−4.0/−2.7/−1.2 |
| FourPeople | −8.9/−4.2/−1.5/−3.5 | −4.3/−4.3/−2.3/−0.8 |
| Johnny | −7.2/−2.4/−2.1/−1.4 | −7.8/−4.1/−2.3/−1.2 |
| Kimono1 | −5.6/−1.7/−1.4/−1.1 | −2.7/−1.8/−2.6/−1.0 |
| KristenAndSara | −5.4/−2.6/−0.9/−2.3 | −7.8/−3.4/−4.0/−1.4 |
| ParkScene | −5.6/−2.0/−1.0/−0.5 | −4.6/−2.8/−0.8/−0.7 |
| PartyScene | −3.3/−2.9/−0.6/−0.4 | −3.4/−1.7/−1.0/−0.9 |
| PeopleOnStreet | −3.6/−1.2/−1.2/−0.7 | −4.4/−3.2/−3.5/−1.3 |
| RaceHorses | −3.9/−1.4/−0.8/−0.3 | −5.1/−0.9/−0.5/−0.7 |
| RaceHorsesC | −2.4/−1.3/−0.8/−0.4 | −4.2/−2.9/−1.9/−0.7 |
| SlideEditing | −6.1/−2.0/−1.5/−0.3 | −1.9/−0.7/−0.3/−0.2 |
| SlideShow | −6.5/−1.9/−0.6/−0.3 | −4.4/−2.4/−1.1/−0.5 |
| Traffic | −6.2/−5.1/−2.3/−1.1 | −4.5/−3.0/−2.4/−0.8 |

# 3 Methods of specification

The problem of the specification of such a method is nuanced and tricky. On the one hand, doing nothing and leaving it to implementers may seem attractive, however, it creates problems of conformance and adherence to the specification.

Assuming that it is desirable to provide some form of specification or guidance on such techniques, there are three areas relating to the specification that need to be addressed. In no particular order, they are conformance, profiling and the decoding process.

## 3.1 Profiling

This section examines how to grant permission for a decoder to employ a best-effort decoding technique.

**Annex (normative part)**

As per JCTVC-M0255, a decoder can claim to implement the particular annex. The annex would describe the requirements for implementing the annex.

Figure 4 – Example preamble for annex text

**Annex F**

**Best-effort decoding of 10-bit sequences using an 8-bit decoder**

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

This annex specifies the recommended behaviour for decoders conformant with the Main profile that also provide a best-effort decoding capability for bit streams conformant to High profile that would otherwise not be decodable.

NOTE: The use of this recommended behaviour does not guarantee perfect reconstruction of the video sequence and an element of drift is likely.

**Profile (decoder)**

A virtual profile, that is to say, one without any assigned profile_idc, allows a decoder to follow the requirements of the profile and thereby claim implementation of it. To avoid any confusion with the Main-10 profile and to fairly represent its nature, the name of such a profile should be appropriately named. A suggested name with a suitable negative tone is the “Degraded Bitdepth Decoding profile”.

The example profile definition in figure 5 has a key characteristics, in that any decoder implementing the profile to use the modified decoding process for Main 10 bitstreams must also implement and use the Main profile decoding process for 8-bit bitstreams.

Figure 5 – Example profile text for a profile with only decoder conformance

**A.3.5 Degraded Bitdepth Decoding profile**

Decoders conforming to the Degraded Bitdepth Decoding profile at a specific level (identified by a specific value of general_level_idc) of a specific tier (identified by a specific value of general_tier_flag) shall be capable of decoding all bitstreams for which all of the following conditions apply:

- When using modified decoding process as specified in clause/annex XYZ:
  - general_profile_compatibility_flag[ 2 ] is equal to 1.
  - SPSs shall have at least on of bit_depth_luma_minus8 or bit_depth_chroma_minus8 greater than 0
  - general_level_idc represents a level lower than or equal to the specified level.
  - general_tier_flag represents a tier lower than or equal to the specified tier.
- Otherwise (when not using the modified decoding process):
  - general_profile_compatibility_flag[ 1 ] is equal to 1.
  - general_level_idc represents a level lower than or equal to the specified level.
  - general_tier_flag represents a tier lower than or equal to the specified tier.

**Profile (decoder + bitstream)**

It is possible to extend the previous description of the Degraded Bitdepth Decoding Profile to be assigned a particular compatable_profile_idc. Use of the compatibility flags in this way allows the encoder to signal to the decoder whether it is allowed to perform the modified decoding method.

A restriction is provided that requires Main 10 profile to be signalled when the Degraded Bitdepth Decoding Profile compatibility flag is signalled. As there is no sense in signalling the Degraded profile in conjunction with Main profile, this is also restricted.

Since the intent of general_profile_idc is used by the standard to provide a "best-viewed as" indication, a restriction is provided to prohibit general_profile_idc indicating that the Main 10 and Degraded Bitdepth Decoding are both signalled

Figure 6 – Example profile text for a profile with decoder and bitstream conformance

**A.3.5 Degraded Bitdepth Decoding profile**

Bitstreams conforming to the Degraded Bitdepth Decoding profile shall obey the following constraints:

- general_profile_compatibility_flag[ 2 ] is equal to 1.
- general_profile_idc is not equal to 4.
- SPSs shall have at least on of bit_depth_luma_minus8 or bit_depth_chroma_minus8 greater than 0

Conformance of a bitstream to the Degraded Bitdepth Decoding profile is indicated by general_profile_-compatibility_flag[ 4 ] being equal to 1.

NOTE - A bitstream conforming to the Degraded Bitdepth Decoding profile must also conform to the Main 10 profile. Since general_profile_idc should indicate the profile that provides the preferred decoded result or the preferred bitstream identification, this must not identify the Degraded Bitdepth Decoding Profile.

Decoders conforming to the Degraded Bitdepth Decoding profile at a specific level (identified by a specific value of general_level_idc) of a specific tier (identified by a specific value of general_tier_flag) shall be capable of decoding all bitstreams for which all of the following conditions apply:

- When using modified decoding process as specified in clause/annex XYZ:
  - general_profile_compatibility_flag[ 2 ] is equal to 1.
  - general_profile_compatibility_flag[ 4 ] is equal to 1.
  - SPSs shall have at least on of bit_depth_luma_minus8 or bit_depth_chroma_minus8 greater than 0
  - general_level_idc represents a level lower than or equal to the specified level.
  - general_tier_flag represents a tier lower than or equal to the specified tier.
- Otherwise (when not using the modified decoding process):
  - general_profile_compatibility_flag[ 1 ] is equal to 1.
  - general_level_idc represents a level lower than or equal to the specified level.
  - general_tier_flag represents a tier lower than or equal to the specified tier.

## 3.2 Conformance

Three methods of specifying conformance are proposed. These should be considered along with the profiling and decoding process specification. Normative conformance is expressed by the requirement to use a particular decoding process by way of profile specification, "*Decoders conforming to the Degraded Bitdepth Decoding profile ... using modified decoding process as specified in clause/annex XYZ"* (see figures 5 and 6).

**Normative – fixed**

The simplest conformance requirement would be to specify that decoders must follow a fixed decoding process. While simple, it does limit the flexibility of decoder designers to customize their best-effort decoding process to match their implementation constraints. On the other hand however, a fixed process does permit an encoder designer to implement encoding schemes that control the drift of both decoding points.

**Normative – fixed with optional components**

A method of providing some flexibility with reasonably well defined behaviour would be to describe a decoding process with optional components and that an implementation would have to demonstrate conformance to one of these.

## 3.3 Decoding process

The definition of the decoding process is strongly coupled with the method of handling conformance. At a minimum a description should be provided that aids the identification of certain issues that must be handled when changing the bitdepth (such as performing entropy decoding at the native bit-depth).

### Normative decoding process (fixed)

If the conformance requires a fixed decoding process then one may be introduced to describe the bit-exact behaviour expected of a decoder.

### Normative decoding process with optional elements

An interesting possibility is to provide a fairly rigid decoding process that includes some scope for flexibility by permitting certain steps to be omitted. It is not suggested to create a vast number of alternatives and combinations, rather to offer practical trade-offs that the decoder designer can consider.

## Example text – decoding process

**Annex F**

**Best-effort decoding of 10-bit sequences using an 8-bit decoder**

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

### F.1 General

This annex specifies the recommended behaviour for decoders conformant with the Main profile that also provide a best-effort decoding capability for bit streams conformant to High profile that would otherwise not be decodable.

NOTE: The use of this recommended behaviour does not guarantee perfect reconstruction of the video sequence and an element of drift is likely.

### F.2 Modified decoding process

A decoder implementing the modified decoding process as specified in this annex shall perform the ordinary decoding process as specified in this Recommendation | International Standard unless otherwise modified below.

The variables $\text{RestrictedBitDepth}_Y$ and $\text{RestrictedBitDepth}_C$ are each set to the value 8.

The following mathematical functions are defined:

$$\text{RClip1}_Y( x ) = \text{RClip1Idx}( x, 0 ) \qquad \text{(F-1)}$$

$$\text{RClip1}_C( x ) = \text{RClip1Idx}( x, 1 ) \qquad \text{(F-2)}$$

$$\text{RClip1Idx}( x, \text{cIdx} ) = \text{Clip3}( 0, ( 1 \ll \text{RestrictedBitDepthOf}( \text{cIdx} ) ) - 1, x ) \qquad \text{(F-3)}$$

$$\text{BitDepthOf}( \text{cIdx} ) = \begin{cases} \text{BitDepth}_Y & ; \quad \text{cIdx} = 0 \\ \text{BitDepth}_C & ; \quad \text{cIdx} = 1, 2 \end{cases} \qquad \text{(F-4)}$$

$$\text{RestrictedBitDepthOf}( \text{cIdx} ) = \begin{cases} \text{RestrictedBitDepth}_Y & ; \quad \text{cIdx} = 0 \\ \text{RestrictedBitDepth}_C & ; \quad \text{cIdx} = 1, 2 \end{cases} \qquad \text{(F-5)}$$

$$\text{RoundToEven}( x, \text{cIdx} ) = \text{RoundToEvenShift}( x, \text{BitDepthOf}( \text{cIdx} ) - \text{RestrictedBitDepthOf}( \text{cIdx} ) ) \qquad \text{(F-6)}$$

$$\text{RoundToEvenShift}( x, \text{shift} ) = x + \underline{(}1 \underline{\ll ( \text{shift} - 1 ) - 1} + \underline{(}( x \gg \text{shift} ) \mathbin{\&} 1\underline{)} \gg \text{shift} \qquad \text{(F-7)}$$

The sample adaptive offset semantics as specified in subclause 7.4.9.3 are modified as follows. Subsequent to the ordinary derivation of SaoOffsetVal, each value is modified as follows:

$$\text{SaoOffsetVal}[ \text{cIdx} ][ \text{rx} ][ \text{ry} ][ i ] = \text{SaoOffsetVal}[ \text{cIdx} ][ \text{rx} ][ \text{ry} ][ i ] \gg ( \text{BitDepthOf}( \text{cIdx} ) - \text{RestrictedBitDepthOf}( \text{cIdx} ) ) \qquad \text{(F-8)}$$

The generation of unavailable pictures as specified in subclause 8.3.3.2 shall be performed using $\text{RestrictedBitDepth}_Y$ in place of the variable $\text{BitDepth}_Y$ and $\text{RestrictedBitDepth}_C$ in place of the variable $\text{BitDepth}_C$.

The general decoding process for coding units coded in intra prediction mode as specified in subclause 8.4.1 when pcm_flag[ xCb ][ yCb ] is equal to 1, shall be performed using $\text{RestrictedBitDepth}_Y$ in place of the variable $\text{BitDepth}_Y$ and $\text{RestrictedBitDepth}_C$ in place of the variable $\text{BitDepth}_C$.

The general intra sample prediction process as specified in subclause 8.4.4.2.1 is modified as follows. Subsequent to the ordinary derivation of p[ x ][ y ] when the sample p[ x ][ y ] is marked as "available for intra prediction", the value of p[ x ][ y ] is modified as follows:

$$p[ x ][ y ] = p[ x ][ y ] \ll ( \text{BitDepthOf}( \text{cIdx} ) - \text{RestrictedBitDepthOf}( \text{cIdx} ) ) \qquad \text{(F-9)}$$

The general decoding process for coding units coded in inter prediction mode as specified in subclause 8.5.1 is modified as follows. Subsequent to the invocation of the inter prediction process as specified in subclause 8.5.2, the three arrays $predSamples_L$, $predSamples_{Cb}$ and $predSamples_{Cr}$ are modified as follows:

$$predSamples_L[\, x \,][\, y \,] = predSamples_L[\, x \,][\, y \,] \;<<\; (\, BitDepth_Y - RestrictedBitDepth_Y \,) \quad \text{with } x,y = 0..nCbS_L \tag{F-10}$$

$$predSamples_{Cb}[\, x \,][\, y \,] = predSamples_{Cb}[\, x \,][\, y \,] \;<<\; (\, BitDepth_C - RestrictedBitDepth_C \,) \quad \text{with } x,y = 0..nCbS_C \tag{F-11}$$

$$predSamples_{Cr}[\, x \,][\, y \,] = predSamples_{Cr}[\, x \,][\, y \,] \;<<\; (\, BitDepth_C - RestrictedBitDepth_C \,) \quad \text{with } x,y = 0..nCbS_C \tag{F-12}$$

The luma sample interpolation process as specified in subclause 8.5.3.3.3.2 shall be performed using $RestrictedBitDepth_Y$ in place of the variable $BitDepth_Y$.

The chroma sample interpolation process as specified in subclause 8.5.3.3.3.3 shall be performed using $RestrictedBitDepth_C$ in place of the variable $BitDepth_C$.

The weighted sample prediction process as specified in subclause 8.5.3.3.4.1 shall be performed using $RestrictedBitDepth_Y$ in place of the variable $BitDepth_Y$ and $RestrictedBitDepth_C$ in place of the variable $BitDepth_C$.

The picture construction process prior to the in-loop filter process as specified in subclause 8.6.5 is modified as follows. The derivation of recSamples[][] using equation (8-280) is replaced by the following:

$$\begin{aligned} &recSamples[\, xCurr + i \,][\, yCurr + j \,] = \\ &\quad RClip1Idx(\, RoundToEven(\, predSamples[\, i \,][\, j \,] + resSamples[\, i \,][\, j \,], cIdx \,), cIdx \,) \\ &\quad \text{with } i = 0..nCurrS - 1,\ j = 0..nCurrS - 1 \end{aligned} \tag{F-13}$$

The decision process for luma block edges as specified in subclause 8.7.2.5.3 shall be performed using $RestrictedBitDepth_Y$ in place of the variable $BitDepth_Y$.

The decision process for chroma block edges as specified in subclause 8.7.2.5.5 shall be performed using $RestrictedBitDepth_C$ in place of the variable $BitDepth_C$

The filtering process for a luma sample as specified in subclause 8.7.2.5.7 shall be performed using $RClip1_Y$ in place of the function $Clip1_Y$.

The filtering process for a chroma sample as specified in subclause 8.7.2.5.8 shall be performed using $RClip1_C$ in place of the function $Clip1_C$.

The sample adaptive offset coding tree block modification process as specified in subclause 8.7.3.2 shall be performed using $RestrictedBitDepth_Y$ in place of $BitDepth_Y$ and $RestrictedBitDepth_C$ in place of $BitDepth_C$.

### F.3 Interpretation of SEI message semantics

TBD

### F.4 Interpretation of VUI semantics

TBD

**BlackBerry Limited may have current or pending patent rights relating to the technology described in this contribution and, conditioned on reciprocity, is prepared to grant licenses under reasonable and non-discriminatory terms as necessary for implementation of the resulting ITU-T Recommendation | ISO/IEC International Standard (per box 2 of the ITU-T/ITU-R/ISO/IEC patent statement and licensing declaration form).**

# References

[1] “Vision, applications and requirements for high efficiency video coding (hevc),” W11872, ISO/IEC JCT1/SC29/WG11, Jan. 2011.

[2] D. Flynn, G. Martin-Cocher, and D. He, “Decoding a 10-bit hevc sequence using an 8-bit decoder,” JCTVC-M0255, JCT-VC, Apr. 2013.

[3] F. Bossen, “Common hm test conditions and software reference configurations,” JCTVC-L1100, JCT-VC, Jan. 2013.