|  |  |
| --- | --- |
| **Joint Collaborative Team on Video Coding (JCT-VC)**  **of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11**  9th Meeting: Geneva, CH, 27 April – 7 May 2012 | Document: JCTVC-I0220 |

|  |  |  |  |
| --- | --- | --- | --- |
| *Title:* | **AHG15: Clarification of mapping process for reference picture lists combination in B slices** | | |
| *Status:* | Input Document to JCT-VC | | |
| *Purpose:* | Proposal | | |
| *Author(s) or Contact(s):* | Yong He, Yan Ye, Jie Dong  9710 Scranton Rd, #250 San Diego, CA 92121 | Tel: Email: | +1-858-210-4803 [Yong.he@interdigital.com](mailto:Yong.he@interdigital.com)  [yan.ye@interdigital.com](mailto:yan.ye@interdigital.com)  [jie.dong@interdigital.com](mailto:jie.dong@interdigital.com) |
| *Source:* | InterDigital Communications, LLC | | |

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

# Abstract

The current mapping process for reference picture lists combinations in B slice in subclause 8.3.4.3 in [1] lacks clarity and does not match the software implementation. This contribution proposes to clarify the process by establishing a mapping index array between reference picture lists and the corresponding reference pictures in the DPB. Updated WD text and HM software implementation are provided.

# Problem Statement

The reference picture list combination syntax in HEVC CD [1] is copied below in Table 1. The decoding process of reference picture list combination in B slices is described in subclause 8.3.4.3 and also copied below. The flow chart in Figure 1 shows that the default reference picture list combination aggregates all unique reference pictures (that is, with unique POC values) from L0 and L1 into LC. In order to detect whether a reference picture from L0 or L1 is already present in LC (shaded blocks in Figure 1), the implementation in HM6.x adopts the pseudo code in Figure 2; however, this is not described in the decoding process in 8.3.4.3 in [1], and the highlight parts of the text spec lack clarity.



Figure 1. Flow chart of default reference picture lists combination

Bool bTempRefIdxInLC = true;

for ( Int iRefIdxLC = 0; iRefIdxLC < m\_aiNumRefIdx[REF\_PIC\_LIST\_C]; iRefIdxLC++ )

{

if ( m\_apcRefPicList[REF\_PIC\_LIST\_0][iNumRefIdx]->getPOC() == m\_apcRefPicList[m\_eListIdFromIdxOfLC[iRefIdxLC]][m\_iRefIdxFromIdxOfLC[iRefIdxLC]]->getPOC() )

{

bTempRefIdxInLC = false;

break;

}

}

Figure 2. Pseudo code to detect unique pictures (shaded blocks in Figure 1)

A strict description of the decoding process in 8.3.4.3 that can also align with reference software implementation is proposed in Section 2.

Table 1. Reference picture list combination syntax in [1]

|  |  |
| --- | --- |
| ref\_pic\_list\_combination( ) { | Descriptor |
| if( slice\_type = = B ) { |  |
| **ref\_pic\_list\_combination\_flag** | u(1) |
| if( ref\_pic\_list\_combination\_flag ) { |  |
| **num\_ref\_idx\_lc\_active\_minus1** | ue(v) |
| **ref\_pic\_list\_modification\_flag\_lc** | u(1) |
| if( ref\_pic\_list\_modification\_flag\_lc) |  |
| for( i =0; i <= num\_ref\_idx\_lc\_active\_minus1; i++ ) { |  |
| **pic\_from\_list\_0\_flag** | u(1) |
| if( ( pic\_from\_list\_0\_flag &&  num\_ref\_idx\_l0\_active\_minus1 > 0 ) | |  ( !pic\_from\_list\_0\_flag &&  num\_ref\_idx\_l1\_active\_minus1 > 0 ) |  |
| **ref\_idx\_list\_curr** | ue(v) |
| } |  |
| } |  |
| } |  |
| } |  |

#### 8.3.4.3. Mapping process for reference picture lists combination in B slices

[Ed.: (WJ) needs to be checked once again. Try to find better way to represent]

This initialization process is invoked when decoding a B slice header.

Input to this process are the reference picture list RefPicListX and num\_ref\_idx\_lX\_active\_minus1 with X being 0 or 1.

Outputs of this process are arrays PredLCToPredLx and RefIdxLCToRefIdxLx.

When the current slice is a B slice and ref\_pic\_list\_modification\_flag\_lc is equal to 0, the following ordered steps apply:

Let refIdxL0 and refIdxL1 be indices into the reference picture lists RefPicListL0 and RefPicListL1. They are initially set equal to 0.

Let refIdxLC be an index into PredLCToPredLx and RefIdxLCToRefIdxLx. It is initially set equal to 0.

The following process is repeated until refIdxL0 and refIdxL1 are both greater than num\_ref\_idx\_l0\_active\_minus1 and num\_ref\_idx\_l1\_active\_minus1, respectively:

– If refIdxL0 is less than or equal to num\_ref\_idx\_l0\_active\_minus1,

– If the entry RefPicListL0[ refIdxL0 ] is the first occurance of the reference picture,

PredLCToPredLx[ refIdxLC ] = Pred\_L0, (8‑12)  
 RefIdxLCToRefIdxLx[ refIdxLC++ ] = refIdxL0.

– refIdxL0++.

– If refIdxL1 is less than or equal to num\_ref\_idx\_l1\_active\_minus1 and ref\_pic\_list\_combination\_flag equal to 1,

– If the entry RefPicListL1[ refIdxL1 ] is the first occurance of the reference picture,

PredLCToPredLx[ refIdxLC ] = Pred\_L1, (8‑13) RefIdxLCToRefIdxLx[ refIdxLC++ ] = refIdxL1.

– refIdxL1++.

When the current slice is a B slice and ref\_pic\_list\_modification\_flag\_lc is equal to 1, the following ordered steps apply:

Let refIdxLC be an index into the reference picture list PredLCToPredLx and RefIdxLCToRefIdxLx. It is initially set equal to 0.

The corresponding syntax elements pic\_from\_list\_0\_flag and ref\_idx\_list\_curr are processed in the order they occur in the bitstream. For each of these syntax elements pairs, the following applies.

– If pic\_from\_list\_0\_flag is equal to 1,

PredLCToPredLx[ refIdxLC ] = Pred\_L0, (8‑14)

– Otherwise,

PredLCToPredLx[ refIdxLC ] = Pred\_L1 (8‑15)

– RefIdxLCToRefIdxLx[ refIdxLC++ ] = ref\_idx\_list\_curr

When refIdxLC is greater than num\_com\_ref\_list\_active\_minus1+ 1, the extra entries past position num\_com\_ref\_list\_active\_minus1 are discarded from PredLCToPredLx and RefIdxLCToRefIdxLx.

When refIdxLC is less than num\_com\_ref\_list\_active\_minus1 + 1, the remaining entries in PredLCToPredLx and RefIdxLCToRefIdxLx are set equal to Pred\_L0 and 0, respectively.

# Proposed WD text change

At the 8th JCT-VC meeting, JCTVC-H0138 established two arrays RefPicList0Temp and RefPicList1Temp to store reference pictures in the DPB to perform L0 and L1 modification. Such mapping relationship between pictures in the DPB and those on the lists can also be used to easily identify if any pictures are the same pictures (that is, same POC values), and if so, which ones. Figure 3 gives an example. The DPB has 5 reference pictures that can be used by the current picture, and the mapping indices between entries in List 0 and List 1 and these 5 pictures can be easily identified.



Figure 3. Mapping relationship between L0/L1 entries and the current DPB

With this mapping, a marking array RefPicSetMarked can be established for the picture in the DPB. Every time an entry from L0 or L1 is added to LC, the corresponding position in RefPicSetMarked is set to 1. It is therefore proposed to clarify the decoding process in 8.3.4.3 with the proposed WD text given below, where changes are highlighted.

**Changes to decoding process in 8.3.4.3:**

When the current slice is a B slice and ref\_pic\_list\_modification\_flag\_lc is equal to 0, the following ordered steps apply:

Let refIdxL0 and refIdxL1 be indices into the reference picture lists RefPicListL0 and RefPicListL1. They are initially set equal to 0.

Let refIdxLC be an index into PredLCToPredLx and RefIdxLCToRefIdxLx. It is initially set equal to 0.

Let RefPicSetMarked be an array of size NumPocTotalCurr. It is initially set to be all 0.

Let RefPicList0ToRPS be an array of size num\_ref\_idx\_l0\_active\_minus1+1. It is initialized to be

for( cIdx = 0; cIdx ≤ num\_ref\_idx\_l0\_active\_minus1; cIdx++)

RefPicList0ToRPS[ cIdx ] = ref\_pic\_list\_modification\_flag\_l0 ? list\_entry\_l0[ cIdx ] : cIdx

Let RefPicList1ToRPS be an array of size num\_ref\_idx\_l1\_active\_minus1+1. It is initialized to be

for( cIdx = 0; cIdx ≤ num\_ref\_idx\_l1\_active\_minus1; cIdx++) {

tempIdx = ref\_pic\_list\_modification\_flag\_l1 ? list\_entry\_l1[ cIdx ] : cIdx

RefPicList1ToRPS[cIdx] = tempIdx < NumPocStCurrAfter ? tempIdx +NumPocStCurrBefore :

tempIdx < NumPocStCurrAfter + NumPocStCurrBefore ?

tempIdx – NumPocStCurrAfter: tempIdx

}

The following process is repeated until refIdxL0 and refIdxL1 are both greater than num\_ref\_idx\_l0\_active\_minus1 and num\_ref\_idx\_l1\_active\_minus1, respectively:

– If refIdxL0 is less than or equal to num\_ref\_idx\_l0\_active\_minus1,

– ~~If the entry RefPicListL0[ refIdxL0 ] is the first occurance of the reference picture,~~ If RefPicSetMarked[RefPicList0ToRPS[refIdxL0]] is equal to 0

PredLCToPredLx[ refIdxLC ] = Pred\_L0, (8‑12)  
 RefIdxLCToRefIdxLx[ refIdxLC++ ] = refIdxL0.

RefPicSetMarked[RefPicList0ToRPS[refIdxL0]] = 1

– refIdxL0++.

– If refIdxL1 is less than or equal to num\_ref\_idx\_l1\_active\_minus1 ~~and ref\_pic\_list\_combination\_flag equal to 1~~,

– ~~If the entry RefPicListL1[ refIdxL1 ] is the first occurance of the reference picture,~~ If RefPicSetMarked[RefPicList1ToRPS[refIdxL1]] is equal to 0

PredLCToPredLx[ refIdxLC ] = Pred\_L1, (8‑13)

RefIdxLCToRefIdxLx[ refIdxLC++ ] = refIdxL1.

RefPicSetMarked[RefPicList1ToRPS[refIdxL1]] = 1

– refIdxL1++.

# References

1. [B. Bross](mailto:benjamin.bross@hhi.fraunhofer.de), [W.-J. Han](mailto:wjhan.han@samsung.com), [J.-R. Ohm](mailto:ohm@ient.rwth-aachen.de), [G. J. Sullivan](mailto:garysull@microsoft.com), [T. Wiegand](mailto:thomas.wiegand@hhi.fraunhofer.de). WD6: Working Draft 6 of High-Efficiency Video Coding. Document no JCTVC-H1003. February 2012.
2. F. Bossen. Common HM test conditions and software reference configurations. Document no JCTVC-G1200. November 2011.
3. Y. Suzuki, et al, Extension of uni-prediction simplification in B slices. Document no JCTVC-D421, January 2011.
4. Y. He and Y. Ye, AHG21: Unification of reference picture list modification processes. Document no JCTVC-H0138, Feb 2012.

# Patent rights declaration(s)

**InterDigital Communications, LLC 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).**