|  |  |
| --- | --- |
| **Joint Collaborative Team on 3D Video Coding Extension Development**  **of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11**  3rd Meeting: Geneva, CH, 17-23 Jan. 2013 | Document: JCT3V-C0054\_proposed text |

|  |  |  |  |
| --- | --- | --- | --- |
| *Title:* | **Proposed text for JCT3V-C0054 based on 3D-AVC working draft 4** | | |
| *Status:* | Input Document | | |
| *Purpose:* | Proposal | | |
| *Author(s) or Contact(s):* | Jewon Kang Ying Chen Li Zhang Marta Karczewicz  5775 Morehouse Drive San Diego, CA 92121 USA | Tel: Email: | 1-858-651-8457 [jewonk@qti.qualcomm.com](mailto:jewonk@qti.qualcomm.com)  1-858-845-6589  [cheny@qti.qualcomm.com](mailto:cheny@qti.qualcomm.com)  +1-858-651-6660 [lizhang@qti.qualcomm.com](mailto:lizhang@qti.qualcomm.com) |
| *Source:* | Qualcomm Incorporated | | |

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

# Abstract

As an attachment of JCT3V-C0054, this document provides detailed syntax specification for the proposed method, and text changes to the current working draft 4 are in red texts.

J.8.3.5.1 Defining of coordinates and sizes of a luma block to be predicted

Let ( xM, yM ) be equal to the output of the subclause 6.4.1 (the location of upper-left luma sample for the current macroblock with address mbAddr relative to the upper-left sample of the picture).

Let ( xP, yP ) be equal to the output of the subclause 6.4.2.1 (the location of upper-left luma sample for the macroblock partition mbPartIdx).

Let ( xB, yB ) be equal to the output of the subclause 6.4.2.2 ~~6.4.3~~ (the location of upper-left luma sample for the 4x4 luma block relative to the top-left sample of the sub-macroblock ~~defined by Luma4x4BlkIdx that can be 0...15~~).

~~Let ( xB8, yB8 ) be equal to the output of the process 6.4.5 (the location of the upper-left luma sample for the 8x8 luma block where Luma8x8BlkIdx =(Luma4x4BlkIdx>>2) is given as an input).~~

~~The variables xCT, yCT, CBX, CBY are set acooding to table J-8-X: [Ed.(MH): CBX and CBY are not proper variable names as they have no camelCasing.]~~

**~~Table J-8-X – Specification of parameters for deriving LRec, URec, LRef, URef blocks’ sample values~~**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **~~mb\_type~~** | **~~transform\_size\_8x8\_flag equal to 1~~** | | | | **~~transform\_size\_8x8\_flag equal to 0~~** | | | |
| **~~xCT~~** | **~~yCT~~** | **~~CBX~~** | **~~CBY~~** | **~~xCT~~** | **~~yCT~~** | **~~CBX~~** | **~~CBY~~** |
| ~~P\_L0\_16x16~~ | ~~xCT = xM~~ | ~~yCT = yM~~ | ~~16~~ | ~~16~~ | ~~xCT = xM + xB8~~ | ~~yCT = yM + yB8~~ | ~~8~~ | ~~8~~ |
| ~~P\_L0\_L0\_16x8~~ | ~~xCT = xM + xP~~ | ~~yCT = yM + yP~~ | ~~16~~ | ~~8~~ | ~~xCT = xM + xB~~ | ~~yCT = yM + yB~~ | ~~4~~ | ~~4~~ |
| ~~P\_L0\_L0\_8x16~~ | ~~xCT = xM + xP~~ | ~~yCT = yM + yP~~ | ~~8~~ | ~~16~~ | ~~xCT = xM + xB~~ | ~~yCT = yM + yB~~ | ~~4~~ | ~~4~~ |
| ~~P\_8x8, P\_8x8ref0~~ | ~~xCT = xM + xP~~ | ~~yCT = yM + yP~~ | ~~8~~ | ~~8~~ | ~~xCT = xM + xB~~ | ~~yCT = yM + yB~~ | ~~4~~ | ~~4~~ |
| ~~P\_Skip~~ | ~~Undefined~~ | | | | ~~xCT = xM~~ | ~~yCT = yM~~ | ~~16~~ | ~~16~~ |

The vairables xCT, yCT, CBX, CBY are set as follows:

* If the mb\_type is equal to P\_8x8 or P\_8x8ref() and sub\_my\_type is not P\_L0\_8x8,
  + xCT is set equal to xM + xP + xB
  + yCT is set equal to yM +yP + yB
  + CBX and CBY are set equal to 4.
* Otherwise (the mb\_type is equal to P\_L0\_16x16, P\_L0\_L0\_16x8, or P\_L0\_L0\_8x16, or P\_8x8 and P\_8x8ref() whose sub\_my\_type are P\_L0\_8x8)
  + xCT is set equal to xM + xP
  + yCT is set equal to yM +yP
  + CBX is set equal to MbPartWidth( mb\_type )
  + CBY is set equal to MbPartHeight( mb\_type ).

If one or more of the following conditions are true, W0C is set to 1 and logWDC is set to 15.

– xCT is smaller than (xL +1) for xL = 3;

– yCT is smaller than (yL + 1) for yL = 3;

– (mvL0[ 0 ] +((xCT – xL – 1)<<2)) is smaller than 0, for xL = 3;

– (mvL0[ 1 ] +((yCT – yL – 1)<<2)) is smaller than 0 for yL = 3;

– (mvL0[ 0 ] +((xCT + CBX)<<2)) is larger or equal to (PicWidthInSamplesL<<2);

– (mvL0[ 1 ] +((yCT + CBY)<<2)) is larger or equal to (PicHeightInSamplesL<<2).

Otherwise LRef, URef, LRec, URec sample values are derived as it is specified in J.8.3.6.2 and J.8.3.6.3 followed by calculation of variables NeighborRefSum, NeighborSum and W0L, O0L specified in the subclause J.8.3.6.4 and J.8.3.6.5 correspondently.

J.8.3.7 Inter prediction process with adaptive luminance compensation for mb\_type equal to P\_8x8 ~~4x4 transform~~

This subclause is invoked for a macroblock in a texture view component only when ~~transform\_size\_8x8\_flag is equal to 0 and~~ mb\_alc\_flag is equal to 1, mb\_type is equal to P\_8x8, and the both SubMbPartWidth and SubMbPartHeight are less than 8..

The Inter prediction process for a macroblock consists of the following steps in order.

1. Subclause J.8.3.5 is invoked to derive motion vector components, reference indices and prediction weights as follows.
2. ~~The variable MvCnt is incremented by subMvCnt.~~
3. For each ~~macroblock partition mbPartIdx and a~~ sub-macroblock partition subMbPartIdx the following assignments are made. [Ed.(MH): Variable name conventions are not obeyed below, rewriting needed.]

~~– sub\_mb\_type\_Normal[ mbPartIdx ] = sub\_mb\_type[ mbPartIdx ];~~

~~– mb\_type\_Normal = mb\_type;~~

~~– If mb\_type is equal to P\_L0\_16x16, the following apply in order.~~

~~– mb\_type is set to be equal to P\_8x8;~~

~~– For mbPartIdx from 0 to NumMbPart( mb\_type ) – 1, inclusive, sub\_mb\_type[ mbPartIdx ] is set to be equal to P\_L0\_8x8.~~

~~– Otherwise, the following apply apply in order.~~

~~– mb\_type is set to be equal to P\_8x8;~~

~~– For mbPartIdx from 0 to NumMbPart( mb\_type ) – 1, inclusive, and sub\_mb\_type[ mbPartIdx ] is set to be equal to P\_L0\_4x4.~~

For i from 0 to NumSubMbPart -1, sub\_mb\_type[ i ] is stored to sub\_mb\_type\_temp [ i ] and is set equal to P\_L0\_4x4.

1. The decoding process for Inter prediction samples for each ~~macroblock partition mbPartIdx and a~~ sub-macroblock partition subMbPartIdx consists of the following in ordered.

– Invoke the subclause J.8.3.6 to derive the prediction weights in adaptive luminance compensation.

– Invoke subclause 8.4.2 to derive the inter prediction samples.

– Invoke subclause 8.5 to derive the transform coefficients and construct the picture prior to deblocking

filter process.

1. For each ~~macroblock partition mbPartIdx and a~~ sub-macroblock partition i~~subMbPartIdx~~, sub\_mb\_type\_temp [ i ] is restored to sub\_mb\_type[ i ]. ~~the following assignments are made.~~

~~– mb\_type = mb\_type\_Normal~~

~~– sub\_mb\_type[ mbPartIdx ] = sub\_mb\_type\_Normal[ mbPartIdx ]~~