|  |  |
| --- | --- |
| **Joint Collaborative Team on Video Coding (JCT-VC)**  **of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11**  21st Meeting: Warsaw, PL, 19–26 June 2015 | Document: JCTVC-U0133 |

|  |  |  |  |
| --- | --- | --- | --- |
| *Title:* | **Comment on signalling the palette\_transpose\_flag after last\_palette\_run\_type\_flag (JCTVC-U0090)** | | |
| *Status:* | Input Document to JCT-VC | | |
| *Purpose:* | Proposal | | |
| *Author(s) or Contact(s):* | Rajan Joshi, Vadim Seregin, Marta Karczewicz, Wei Pu, Feng Zou  5775 Morehouse Drive San Diego, CA 92121, USA | Tel: Email: | 1-858-658-4511 [rajanj@qti.qualcomm.com](mailto:rajanj@qti.qualcomm.com) |
| *Source:* | Qualcomm Incorporated | | |

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

# Abstract

In JCTVC-U0090, it is proposed that the **palette\_transpose\_flag** be signalled after **last\_palette\_run\_type\_flag** to group bypass bins together. It is asserted that when syntax elements related to palette delta QP or palette chroma QP offset are present, this does not result in grouping of more than one additional bypass bin. It is proposed that in addition to placing the **palette\_transpose\_flag** after **last\_palette\_run\_type\_flag**, the syntax elements related to palette delta QP and palette chroma QP offset be signalled after the **palette\_transpose\_flag**. It is asserted that due to this reordering, all the bins up to **last\_palette\_run\_type\_flag** are bypass-coded and grouped together.

# Introduction

In JCTVC-U0090, it is proposed that the **palette\_transpose\_fla**g be signalled after **last\_palette\_run\_type\_flag** to group bypass bins together. This is shown below.

|  |  |
| --- | --- |
| if( palette\_escape\_val\_present\_flag ) { |  |
| if( cu\_qp\_delta\_enabled\_flag && !IsCuQpDeltaCoded ) { |  |
| **cu\_qp\_delta\_palette\_abs** | ae(v) |
| if( cu\_qp\_delta\_palette\_abs ) |  |
| **cu\_qp\_delta\_palette\_sign\_flag** | ae(v) |
| } |  |
| if( cu\_chroma\_qp\_offset\_enabled\_flag && !IsCuChromaQpOffsetCoded ) { |  |
| **cu\_chroma\_qp\_palette\_offset\_flag** | ae(v) |
| if( cu\_chroma\_qp\_offset\_flag && chroma\_qp\_offset\_list\_len\_minus1 > 0 ) |  |
| **cu\_chroma\_qp\_palette\_offset\_idx** | ae(v) |
| } |  |
| } |  |
| if( MaxPaletteIndex > 0) { |  |
| **~~palette\_transpose\_flag~~** | ~~ae(v)~~ |
| **num\_palette\_indices\_idc** | ae(v) |
| for( i=0; i < NumPaletteIndices; i++ ) { |  |
| **palette\_index\_idc** | ae(v) |
| PaletteIndexIdc[ i ] = palette\_index\_idc |  |
| } |  |
| **last\_palette\_run\_type\_flag** | ae(v) |
| **palette\_transpose\_flag** | ae(v) |
| } |  |
| CurrNumIndices = 0 |  |

It should be noted that when **cu\_qp\_delta\_palette\_abs**, **cu\_chroma\_qp\_palette\_offset\_flag**, or **cu\_chroma\_qp\_palette\_offset\_idx** are present, this may result in grouping of only one additional bypass bin. This occurs when **cu\_qp\_delta\_palette\_sign\_flag** is present but chroma QP offset syntax elements are absent. In all other cases, there is no change in the grouping of bypass bins.

# Proposal

It is proposed that the position of syntax elements related to palette delta QP and palette chroma QP offset be moved after the new position of **palette\_transpose\_flag**. This is shown below:

|  |  |
| --- | --- |
| palette\_coding( x0, y0, nCbS ) { | Descriptor |
| … |  |
| if( CurrentPaletteSize != 0 ) |  |
| **palette\_escape\_val\_present\_flag** | ae(v) |
| ~~if( palette\_escape\_val\_present\_flag ) {~~ |  |
| ~~if( cu\_qp\_delta\_enabled\_flag && !IsCuQpDeltaCoded ) {~~ |  |
| **~~cu\_qp\_delta\_palette\_abs~~** | ~~ae(v)~~ |
| ~~if( cu\_qp\_delta\_palette\_abs )~~ |  |
| **~~cu\_qp\_delta\_palette\_sign\_flag~~** | ~~ae(v)~~ |
| ~~}~~ |  |
| ~~if( cu\_chroma\_qp\_offset\_enabled\_flag && !IsCuChromaQpOffsetCoded ) {~~ |  |
| **~~cu\_chroma\_qp\_palette\_offset\_flag~~** | ~~ae(v)~~ |
| ~~if( cu\_chroma\_qp\_offset\_flag && chroma\_qp\_offset\_list\_len\_minus1 > 0 )~~ |  |
| **~~cu\_chroma\_qp\_palette\_offset\_idx~~** | ~~ae(v)~~ |
| ~~}~~ |  |
| ~~}~~ |  |
| if( MaxPaletteIndex > 0) { |  |
| **~~palette\_transpose\_flag~~** | ~~ae(v)~~ |
| **num\_palette\_indices\_idc** | ae(v) |
| for( i=0; i < NumPaletteIndices; i++ ) { |  |
| **palette\_index\_idc** | ae(v) |
| PaletteIndexIdc[ i ] = palette\_index\_idc |  |
| } |  |
| **last\_palette\_run\_type\_flag** | ae(v) |
| palette\_transpose\_flag | ae(v) |
| } |  |
| if( palette\_escape\_val\_present\_flag ) { |  |
| if( cu\_qp\_delta\_enabled\_flag && !IsCuQpDeltaCoded ) { |  |
| **cu\_qp\_delta\_palette\_abs** | ae(v) |
| if( cu\_qp\_delta\_palette\_abs ) |  |
| **cu\_qp\_delta\_palette\_sign\_flag** | ae(v) |
| } |  |
| if( cu\_chroma\_qp\_offset\_enabled\_flag && !IsCuChromaQpOffsetCoded ) { |  |
| **cu\_chroma\_qp\_palette\_offset\_flag** | ae(v) |
| if( cu\_chroma\_qp\_offset\_flag && chroma\_qp\_offset\_list\_len\_minus1 > 0 ) |  |
| **cu\_chroma\_qp\_palette\_offset\_idx** | ae(v) |
| } |  |
| } |  |
| CurrNumIndices = 0 |  |
| PaletteScanPos = 0 |  |
| … |  |

Here the changes proposed in JCTVC-U0090 are highlighted in yellow and the changes proposed in the current proposal are highlighted in cyan. The effect of this is that all the bins up to **last\_palette\_run\_type\_flag** are bypass coded and grouped together.

# Patent rights declaration(s)

**Qualcomm Incorporated 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).**