|  |  |
| --- | --- |
| **Joint Collaborative Team on Video Coding (JCT-VC)**  **of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11**  8th Meeting: San José, CA, USA, 1–10 February, 2012 | Document: JCTVC-H0100r5 |

|  |  |  |  |
| --- | --- | --- | --- |
| *Title:* | Non-CE11.1: Context reduction of significance map coding | | |
| *Status:* | Input Document to JCT-VC | | |
| *Purpose:* | Proposal | | |
| *Author(s) or Contact(s):* | Takeshi Tsukuba\*  Tomohiro Ikai\*  Vadim Seregin\*\*  Joel Sole\*\*  Marta Karczewicz\*\*  \*1-9-2 Nakase, Mihama-ku, Chiba-shi, Chiba 261-8520  JAPAN  \*\*5775 Morehouse Drive San Diego, CA 92121, USA | Tel: Email: | +81-43-299-8526 [tsukuba.takeshi@sharp.co.jp](mailto:tsukuba.takeshi@sharp.co.jp)  [vseregin@qualcomm.comm](mailto:vseregin@qualcomm.comm)  [joels@qualcomm.com](mailto:joels@qualcomm.com)  [martak@qualcomm.com](mailto:martak@qualcomm.com) |
| *Source:* | SHARP Corporation, Qualcomm | | |

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

# Abstract

In this contribution, a simplified context derivation for both 4x4 and 8x8 blocks in the significant \_coeff\_flag coding is proposed. The proposed method uses a common lookup table for both 4x4 and 8x8 and reduces 11 contexts. It is reported that average BD-rate change is 0.01% to 0.07% on common condition ( QP = 22, 27, 32, 37) and -0.02% to 0.05% on low QP condition ( QP = 12, 17, 22, 27).

It is also proposed to share DC context with all blocks sizes from 4x4 to 32x32 for both luma and chroma significance map in order to further reduce the number of contexts. The average BD-rage change is -0.05% to 0.05% on common condition and -0.04% to 0.03% on low QP condition.

# Introduction

At significant\_coeff\_flag coding, HM-5.0 uses a total of 37 contexts for 4x4 and 8x8 blocks: 9 for 4x4 luma , 6 for 4x4 chroma and 11 for 8x8 luma and chroma as shown in Figure 1[2]. Context index for 4x4 and 8x8 blocks is derived based on coefficient position (xC, yC) and the lookup tables as follows:

* If log2TrafoWidth is equal to log2TrafoHeight and log2TrafoWidth is equal to 2, sigCtx is derived using ctxIdxMap4x4[ ] specified in Table 9‑39 as follows..

sigCtx = ctxIdxMap4x4[ ((cIdx > 0) ? 15 : 0) + (yC << 2) + xC ] (9‑55)

* Otherwise if log2TrafoWidth is equal to log2TrafoHeight and log2TrafoWidth is equal to 3, sigCtx is derived using ctxIdxMap8x8[ ] specified in Table 9‑40 as follows.

sigCtx = ((xC + yC) = = 0) ? 10 : ctxIdxMap8x8[ ((yC >> 1 ) << 2) + (xC >> 1) ] (9‑56)

sigCtx += ( cIdx > 0) ? 6: 9 (9‑57)

Table ‑ – Specifcation of ctxIdxMap4x4[ i ]

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **I** | **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** |
| **ctxIdxMap4x4[ i ]** | 0 | 1 | 4 | 5 | 2 | 3 | 4 | 5 | 6 | 6 | 8 | 8 | 7 | 7 | 8 |
| **I** | **15** | **16** | **17** | **18** | **19** | **20** | **21** | **22** | **23** | **24** | **25** | **26** | **27** | **28** | **29** |
| **ctxIdxMap4x4[ i ]** | 0 | 1 | 2 | 4 | 1 | 1 | 2 | 4 | 3 | 3 | 5 | 5 | 4 | 4 | 5 |

Table ‑ – Specifcation of ctxIdxMap8x8[ i ]

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **I** | **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** |
| **ctxIdxMap8x8[ i ]** | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 3 | 8 | 6 | 6 | 7 | 9 | 9 | 7 | 7 |

In the case of 8x8 block, as shown in eq.(9-56), context index for DC is derived based on the coefficient position, while the others are derived as the position of 2x2 sub-block and a lookup table. Therefore, it requires one conditional branch process for classifying the lower 4 frequency components into a DC component and AC components.

In the current design, there exist the following issues;

* The sizes of the look-up tables are large.
* In 8x8 block, it is complicated to classify the lower 4 frequency into a DC component and AC components.
* The number of contexts can be reduced.

In order to solve those issues, the proposed method is presented in the next section.

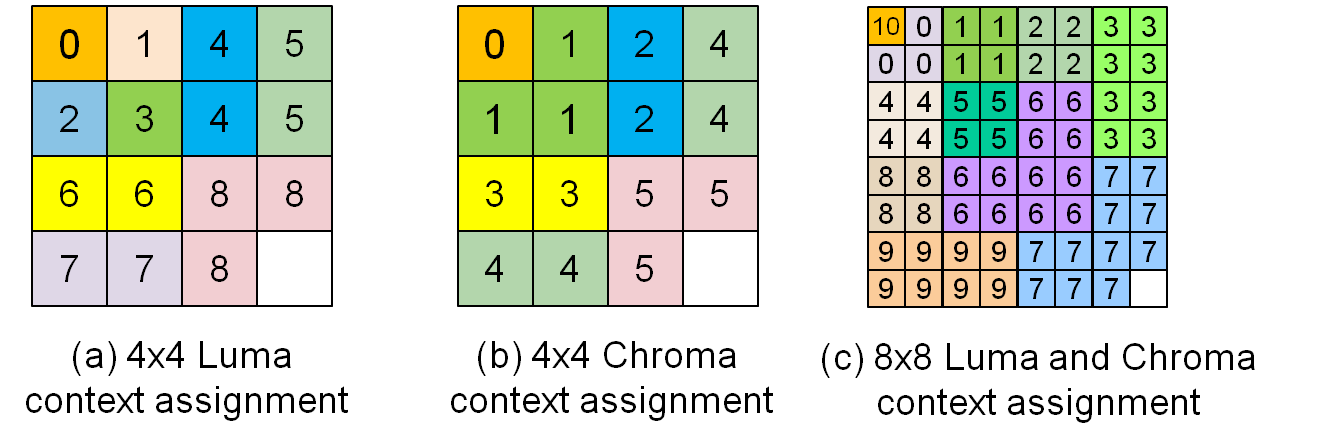


Figure 1: Context assignment of 4x4 and 8x8 significance map coding in HM-5.0

# Technical Description

## Proposal 1

The proposal 1 is to unify context derivation for 4x4 and 8x8 blocks of each color component by using a similar context assignment in 4x4 and 8x8 blocks as shown in Figure 2.



Figure 2: The Proposed Context assignment of 4x4 and 8x8 significance map coding

In the proposal 1, context index for 4x4 and 8x8 blocks is derived based on coefficient position and lookup tables as follows:

* If log2TrafoWidth is equal to log2TrafoHeight and log2TrafoWidth is less than or equal to 3 and cIdx is equal to 0, sigCtx is derived using ctxIdxMapLuma [ ] specified in Table 9‑39 as follows..

Index = log2TrafoWidth==2 ? (yC<<2) + xC: ( ( yC>>1) <<2) + (xC>>1)

sigCtxOffset = log2TrafoWidth==2 ? 0 : 7

~~sigCtx = ((xC + yC) = = 0) ? 10 : ctxIdxMap8x8[ ((yC >> 1 ) << 2) + (xC >> 1) ] (9‑56)~~

sigCtx = sigCtxOffset + ctxIdxMapLuma[ Index ] (9‑55)

* Otherwise if log2TrafoWidth is equal to log2TrafoHeight and log2TrafoWidth is less than or equal to 3 and cIdx is equal to 1, sigCtx is derived using ctxIdxMapChroma [ ] specified in Table 9‑40 as follows.

Index = log2TrafoWidth==2 ? (yC<<2) + xC: ( ( yC>>1) <<2) + (xC>>1)

sigCtxOffset = logTrafoWidth==2 ? 0 : 6

sigCtx = sigCtxOffset + ctxIdxMapChroma [ Index ] (9‑56)

Table 9‑39 – Specifcation of ctxIdxMapLuma [ i ]

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **I** | **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** |
| **ctxIdxMapLuma[ I ] [ i ]** | 0 | 1 | 2 | 3 | 1 | 1 | 2 | 3 | 4 | 4 | 6 | 6 | 5 | 5 | 6 | 6 |

Table ‑ – Specifcation of ctxIdxMapChroma[ i ]

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **I** | **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** |
| **ctxIdxMapChroma[ i ]** | 0 | 1 | 2 | 3 | 1 | 1 | 2 | 3 | 4 | 4 | 5 | 5 | 3 | 3 | 5 | 5 |

In the proposed method, the lookup table for 8x8 is removed so that the table for 4x4 block is used for both 4x4 and 8x8 blocks. The DC classification of 8x8 block is also removed.

The benefits of the proposal 1 are summarized as follows:

* The number of contexts is reduced by 11
* DC classification of 8x8 block is removed.
* One lookup table ctxIdxMap8x8 is removed

## Proposal 2

An extension of the method is proposed to further reduce the number of contexts. Currently DC context is shared among 16x16 and 32x32 blocks. It is proposed to share DC context with all blocks sizes from 4x4 to 32x32 for both luma and chroma significant map as shown in Figure 3. As a result the number of context further reduced by 2 and in total context reduction reached 13 contexts. In the case of applying proposed DC context sharing alone, the number of context reduced by 4.



Figure 3 : Proposed DC context sharing

# Simulation results

The proposal 1 was implemented on HM-5.0. The simulations were performed based on common condition and low QP condition. Table 1 and 2 summarize the results of the experiments.

**Table 1: Performance of the proposal 1 (common condition)**



**Table 2: Performance of the proposal 1 (QP = 12, 17, 22, 27)**



DC context sharing, proposal 2, was implemented on top of new context assignments (proposal 1) and experimental results are summarized in Table 3 and Table 4. In the case of applying proposed DC context sharing alone, experimental results are summarized in Table 5 and Table 6.

**Table 3: Performance of the proposal 2 (proposal 1 with DC context sharing) (common condition)**



**Table 4: Performance of the proposal 2 (proposal 1 with DC context sharing) (QP = 12, 17, 22, 27)**



**Table 5: Performance of the proposal 2 (Only DC context sharing) (QP = 22, 27, 32, 37)**



**Table 6: Performance of the proposal 2 (Only DC context sharing) (QP = 12, 17, 22, 27)**



As additional information, the combination with JCTVC-H-0443[3] and proposal 1 was tested. In JCTVC-H0443, it is proposed to use fixed-position based context for 16x4/4x16 blocks and share contexts and tables with 8x8 blocks. In the combination method, the table for deriving context index is accessed in a similar way to JCTVC-H0443 as follows:

For the 4x16 case,

ctxIdxMapLuma[ ( ( yC>>1) << 2) + (xC>>1) ] or ctxIdxMapChroma[ ( ( yC>>1) << 2) + (xC>>1) ]

For the 16x4 case,

ctxIdxMapLuma[ ( ( xC>>1) << 2) + (yC>>1) ] or ctxIdxMapChroma[ ( ( xC>>1) << 2) + (yC>>1) ]

Experimental results are summarized in Table 7 and Table 8. Average BD-rate change is 0.01% to 0.09% on common condition and 0.01% to 0.15% on low QP condition ( QP = 12, 17, 22, 27).

**Table 7: Additional information on performance of the combination with H0443 and proposal 1 (common condition )**



**Table 8: Additional information on performance of the combination with H0443 and proposal 1 ( QP = 12, 17, 22, 27)**



# Conclusion

In this document, a modification of significance map coding for 4x4 and 8x8 blocks is proposed. In this proposal, the number of context for 4x4 and 8x8 blocks is reduced by up to 13 (about 35%, 13/37). The performance loss is negligible (up to 0.1 %). We recommend the proposal to be adopted as a part of the next HM.

# References

1. Vivienne Sze, “Description of Core Experiment (CE11): Coefficient scanning and coding,” JCTVC-G1211, 7th Meeting: Geneva, CH, 21-30 November, 2011.
2. B. Bross, W-J Han, J-R Ohm, G. J. Sullivan, and T. Wiegand, “WD5: Working Draft 5 of High-Efficiency Video Coding,” JCT-VC of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11, 7th Meeting, Geneva, November, 2011.
3. J. Kim, B. Jeon, J Sole and M. Karczewicz, “CE11: Hamonization of residual coding and NSQT,” JCTVC-H0443, 8th Meeting: San José, CA, USA, 1–10 February, 2012.

# Proposed WD text

The changes of WD text are highlighted in yellow. The base text is JCTVC-G1103[2].

Table 9‑27 – Values of variable initValue for significant\_coeff\_flag ctxIdx

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Initialisation variable** | **significant\_coeff\_flag ctxIdx** | | | | | | | | | | | | | | | |
| **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** |
| **initValue** | 74 | 72 | 72 | 55 | 71 | 54 | 71 | 88 | 134 | 71 | 53 | 70 | 68 | 84 | 90 | 84 |
|  | **16** | **17** | **18** | **19** | **20** | **21** | **22** | **23** | **24** | **25** | **26** | **27** | **28** | **29** | **30** | **31** |
| **initValue** | 88 | 74 | 130 | 118 | 88 | 120 | 87 | 149 | 52 | 70 | 118 | 133 | 162 | 114 | 129 | 115 |
|  | **32** | **33** | **34** | **35** | **36** | **37** | **38** | **39** | **40** | **41** | **42** | **43** | **44** | **45** | **46** | **47** |
| **initValue** | 51 | 74 | 115 | 87 | 89 | 152 | 118 | 87 | 70 | 70 | 53 | 118 | 134 | 101 | 101 | 68 |
|  | **48** | **49** | **50** | **51** | **52** | **53** | **54** | **55** | **56** | **57** | **58** | **59** | **60** | **61** | **62** | **63** |
| **initValue** | 68 | 67 | 100 | 168 | 147 | 150 | 120 | 115 | 118 | 119 | 136 | 102 | 70 | 67 | 53 | 117 |
|  | **64** | **65** | **66** | **67** | **68** | **69** | **70** | **71** | **72** | **73** | **74** | **75** | **76** | **77** | **78** | **79** |
| **initValue** | 102 | 115 | 115 | 114 | 83 | 100 | 168 | 131 | 150 | 120 | 152 | 118 | 87 | 70 | 70 | 53 |
|  | **80** | **81** | **82** | **83** | **84** | **85** | **86** | **87** | **88** | **89** | **90** | **91** | **92** | **93** | **94** | **95** |
| **initValue** | 71 | 103 | 101 | 101 | 68 | 68 | 67 | 116 | 168 | 147 | 150 | 120 | 115 | 118 | 119 | 136 |
|  | **96** | **97** | **98** | **99** | **100** | **101** | **102** | **103** | **104** | **105** | **106** | **107** | **108** | **109** | **110** |  |
| **initValue** | 102 | 86 | 67 | 84 | 117 | 102 | 115 | 115 | 99 | 83 | 100 | 152 | 131 | 150 | 120 |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **initValue** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **initValue** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

## Changes of “9.2.3.1.1.5 Derivation process of ctxIdxInc for the syntax element significant\_coeff\_flag”

Inputs to this process are the color component index cIdx, the current coefficient scan position ( xC , yC ), the transform block width log2TrafoWidth and the transform block height log2TrafoHeight.

Output of this process is ctxIdxInc.

The variable sigCtx depends on the current position ( xC, yC ), the color component index cIdx, the transform block size and previsously decoded bins of the syntax element significant\_coeff\_flag. For the derivation of sigCtx, the following applies.

// process for proposal2

* if xC + yC is equal to 0, sigCtx is derived as follows.

sigCtx = 0 (9‑54)

// common process for proposal1 and proposal2

* If log2TrafoWidth is equal to log2TrafoHeight and log2TrafoWidth is less than or equal to 3 and cIdx is equal to 0, sigCtx is derived using ctxIdxMapLuma [ ] specified in as follows.

Index = log2TrafoWidth==2 ? (yC<<2) + xC: ( ( yC>>1) <<2) + (xC>>1)

sigCtxOffset = log2TrafoWidth==2 ? 0 : 7

sigCtx = sigCtxOffset + ctxIdxMapLuma[Index] (9‑55)

* Otherwise if log2TrafoWidth is equal to log2TrafoHeight and log2TrafoWidth is less than or equal to 3 and cIdx is equal to 1, sigCtx is derived using ctxIdxMapChroma [ ] specified in as follows.

Index = log2TrafoWidth==2 ? (yC<<2) + xC: ( ( yC>>1) <<2) + (xC>>1)

sigCtxOffset = logTrafoWidth==2 ? 0 : 6

sigCtx = sigCtxOffset + ctxIdxMapChroma [ Index ] (9‑56)

* Otherwise if xC + yC is equal to 0, sigCtx is derived as follows.

sigCtx = ( cIdx > 0) ? 14: 12 (9‑57)

* Otherwise (xC + yC is greater than 0), sigCtx is derived using previously decoded bins of the syntax element significant\_coeff\_flag as follows.
* The variable sigCtx is initialized as follows.

sigCtx = 0 (9‑58)

* When xC is less than ( 1 << log2TrafoWidth ) − 1, the following applies.

sigCtx = sigCtx + significant\_coeff\_flag[ xC + 1 ][ yC ] (9‑59)

* When xC is less than ( 1 << log2TrafoWidth ) − 1 and yC is less than ( 1 << log2TrafoHeight ) − 1, the following applies.

sigCtx = sigCtx + significant\_coeff\_flag[ xC + 1 ][ yC + 1 ] (9‑60)

* When xC is less than ( 1 << log2Width ) − 2, the following applies.

sigCtx = sigCtx + significant\_coeff\_flag[ xC + 2 ][ yC ] (9‑61)

* When all of the following conditions are true,
  + yC is less than ( 1 << log2TrafoHeight ) − 1,
  + xC % 4 is not equal to 0 or yC % 4 is not equal to 0,
  + xC % 4 is not equal to 3 or yC % 4 is not equal to 2,

the following applies.

sigCtx = sigCtx + significant\_coeff\_flag[ xC ][ yC + 1 ] (9‑62)

* When yC is less than ( 1 << log2TrafoHeight ) − 2 and sigCtx is less than 4, the following applies.

sigCtx = sigCtx + significant\_coeff\_flag[ xC ][ yC + 2 ] (9‑63)

* The variable sigCtx is modified as follows.
  + If cIdx is equal to 0 and xC + yC are greater than (1 << (max(log2TrafoWidth, log2TrafoHeight) − 2)) − 1, the following applies.

sigCtx = ( (sigCtx + 1) >> 1 ) + 18 (9‑63)

* + Otherwise, the following applies.

sigCtx = ( (sigCtx + 1) >> 1 ) + ( (cIdx > 0) ? 13 : 15 ) (9‑63)

The context index increment ctxIdxInc is derived using the color component index cIdx and sigCtx as follows.

* If cIdx is equal to 0, ctxIdxInc is derived as follows.

ctxIdxInc = sigCtx (9‑64)

* Otherwise (cIdx is greater than 0), ctxIdxInc is derived as follows.

ctxIdxInc = 21 + sigCtx (9‑65)

Table 9‑39 – Specifcation of ctxIdxMapLuma [ i ]

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **I** | **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** |
| **ctxIdxMapLuma[ i ] [ i ]** | 0 | 1 | 2 | 3 | 1 | 1 | 2 | 3 | 4 | 4 | 6 | 6 | 5 | 5 | 6 | 6 |

Table 9‑40 – Specifcation of ctxIdxMapChroma[ i ]

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **I** | **0** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** | **13** | **14** | **15** |
| **ctxIdxMapChroma[ i ]** | 0 | 1 | 2 | 3 | 1 | 1 | 2 | 3 | 4 | 4 | 5 | 5 | 3 | 3 | 5 | 5 |

# Patent rights declaration(s)

**Sharp Corporation 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).**

**Qualcomm 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).**