|  |  |
| --- | --- |
| **Joint Collaborative Team on Video Coding (JCT-VC)**  **of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11**  17th Meeting: Valencia, ES, 27 March – 4 April 2014 | Document: JCTVC-Q0044 |

|  |  |  |  |
| --- | --- | --- | --- |
| *Title:* | **AhG18: On SAO quant-bits coding** | | |
| *Status:* | Input Document to JCT-VC | | |
| *Purpose:* | Proposal | | |
| *Author(s) or Contact(s):* | E.Alshina  A.Alshin | Tel: Email: | +82 10 3026 1305  elena\_a.alshina@samsung.com |
| *Source:* | Samsung Electronics Ltd. | | |

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

# Abstract

Quant bits for SAO parameters are signalled in bit-stream according to then JCTVC 16th meeting decision, but there is no recommendation about the choice for these parameters. This contribution suggests the settings for SAO quant bits depending on internal bit-depth and slice QP which improves SAO performance by 1.6% (LD) and 0.2%(AI) for 16 bits coding and 1.1%(LD) for 12 bits coding. At the same time this contribution suggests to move SAO Quant bit syntax form PPS to Slice Header.

# Introduction

Before JCTVC 16th meeting SAO quant bits depended on internal bit-depth and it was found that dependency is not optimal. In particular for high bit-depth and bit-rates quant bits for SAO should be zero. But for customer range of QPs (QP=17…38) tests presented in [1] show that non-zero quant bits for SAO provides visible performance improvement. It was decided to signal SAO quant bits in a bit-stream for Luma and Chroma separately and corresponding syntax elements are added to PPS extension [1]. The quant bits are allowed to be non zero for internal bit-depth higher than 10. Maximum magnitude for SAO offsets is limited by 31 (for any bit depth ≥10).

Fig. 1 presents typical dependency of SAO offsets magnitude from QP.



Fig. 1. SAO off-sets grow with QP.

Assuming that maximum magnitude for SAO offsets is 31 SAO quant bits should depend on QP as flows:

saoOffsetBitShift = Max(0,int((bitDepth-10) +A+ p\*QP-5 +0,5) ). (1)

Recommended control parameters values are A = 1,78 and p=0,165.

The default setting for SAO quant bits (SaoLumaOffsetBitShift and SaoChromaOffsetBitShift) is 0 for AhG18 test conditions. If one uses saoOffsetBitShift = 0 at customer range of QPs (test for QP=17…38 was requested during 16th JCTVC meeting) then there is very small gain or even performance drop from SAO (second line in Table 1). The goal of this contribution is to give an example for proper settings of SaoLumaOffsetBitShift and SaoChromaOffsetBitShift.

Formula (1) returns saoOffsetBitShift = 0 to most of QP posint used in AhG18 test conditions. So almost no performance impact is expected under AhG18 test conditions. The performance impact of settings (1) for saoOffsetBitShift at customer range of QPs is evaluated in this contribution. Tests sequences from AhG 18 test conditions were used for the test. All intra and low-delay B tests for internal bit-depths 12 and 16 were performed. Achieved performance improvement is higher for LD tests 1,1% (12 bits) and 1,6% (16 bits); but there is some improvement for all-intra test as well 0,2% (16 bits).

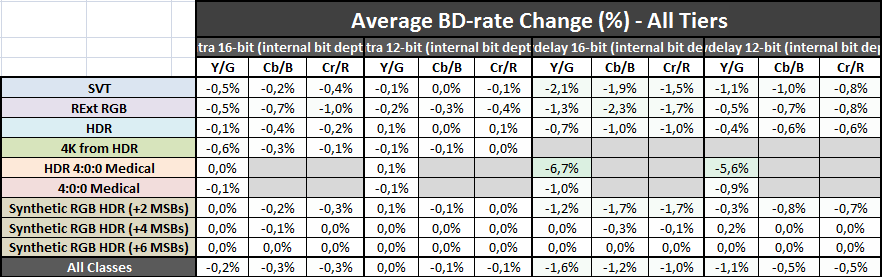
**Table 1.** Brief summary of test results Qp=17…38 (AhG18.

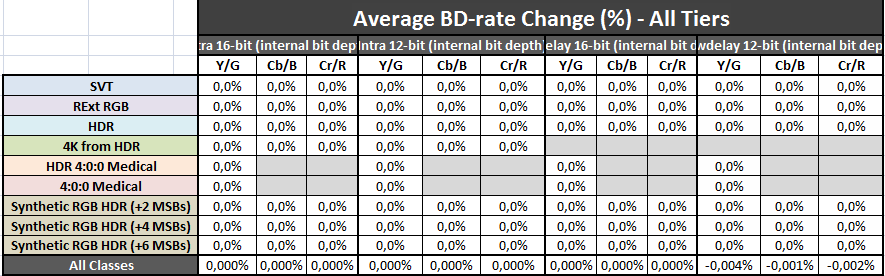
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test | AI 16 bits | AI 12 bits | LD 16 bits | LD 12 bits |
| saoOffsetBitShift = 0 | 0,0% | 0,0% | 0,0% | 0,0% |
| SAO off | -0,1% | 0,1% | -0,5% | 0,0% |
| suggested settings saoOffsetBitShift | -0,2% | 0,0% | -1,6% | -1,1% |

It is suggested to put (1) into the reference s/w as an example for saoOffsetBitShift settings. This would be encoder only change.

Tables 2 and 3 summarizes test results for QP=17..38 and AhG18 QPs correspondently.

**Table 2.** Test results for proposed encoder modification at QP=17…38.

**Table 3.** Test results for proposed encoder modification at AhG18 QPs.



# Syntax modification

Currently SAO quant bits information is encoded in PPS extension.

#### Picture parameter set RBSP syntax

|  |  |
| --- | --- |
| pic\_parameter\_set\_rbsp( ) { | Descriptor |
| **…** |  |
| if( pps\_extension1\_flag ) { |  |
| **…** |  |
| **sao\_luma\_bit\_shift** | ue(v) |
| **sao\_chroma\_bit\_shift** | ue(v) |
| **pps\_extension2\_flag** | u(1) |
| } |  |
| … |  |
| } |  |

So, **sao\_luma\_bit\_shift** and **sao\_chroma\_bit\_shift** are signalled even SAO is disabled. There is flag in SPS with enables SAO and there are 2 flags which enables SAO separately for Luma and Chroma in slice header.

Following syntax looks more reasonable: one flag in SPS extension which tells about presence of SAO quant bit information in a bit-stream (this flag is not encoded if SAO is disabled in SPS) and , **sao\_luma\_bit\_shift** and **sao\_chroma\_bit\_shift** are signalled in slice header (they are not encoded is SAO is disabled for current slice).

#### Sequence parameter set RBSP syntax

|  |  |
| --- | --- |
| seq\_parameter\_set\_rbsp( ) { | Descriptor |
| ….. |  |
| if( sps\_extension\_flag[ 0 ] ) { |  |
| **…** |  |
| if( sample\_adaptive\_offset\_enabled\_flag ) { |  |
| **adaptive\_sao\_bit\_shift\_active\_flag** | u(1) |
| } |  |
| } |  |
| ……. |  |
| **}** |  |

#### General slice segment header syntax

|  |  |
| --- | --- |
| slice\_segment\_header( ) { | Descriptor |
| ……. |  |
| if( sample\_adaptive\_offset\_enabled\_flag ) { |  |
| **slice\_sao\_luma\_flag** | u(1) |
| if( ChromaArrayType > 0 ) |  |
| **slice\_sao\_chroma\_flag** | u(1) |
| } |  |
| if( slice\_sao\_luma\_flag && adaptive\_sao\_bit\_shift\_active\_flag) |  |
| **sao\_luma\_bit\_shift** | ue(v) |
| if( slice\_sao\_chroma\_flag && adaptive\_sao\_bit\_shift\_active\_flag &&ChromaArrayType > 0) |  |
| **sao\_chroma\_bit\_shift** | ue(v) |
| ……. |  |
| } |  |

It is suggested to move SAO quant bit information to the slice header.

# Conclusion

SAO quant bit settings depending QP and bit-depth are recommended in this contribution. This encoder only modification leads to 1.1% (LD 12 bit) and 1.6% (LD 16 bits) performance improvement. It is apply these settings in reference s/w. Additionally it is suggested to move SAO quant bit syntax from PPS extension to slice header in order to avoid redundant information encoding if SAO is disabled.

# References

1. E. Alshina, A. Tourapis, W.-S. Kim, Y.-W. Huang, S.-T. Hsiang, “**AhG 18: On SAO performance at high bit-depth and high bit-rates**,” Document of Joint Collaborative Team on Video Coding, JCTVC-P0222, Jan. 2014.

# Patent rights declaration(s)

**Samsung Electronics Ltd. 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).**