|  |  |
| --- | --- |
| **Joint Collaborative Team on Video Coding (JCT-VC)**  **of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11**  10th Meeting: Stockholm, SE, 11–20 July 2012 | Document: JCTVC-J0116r1 |

|  |  |  |  |
| --- | --- | --- | --- |
| *Title:* | **AHG13: Signalling of long-term reference pictures in the SPS** | | |
| *Status:* | Input Document to JCT-VC | | |
| *Purpose:* | Proposal | | |
| *Author(s) or Contact(s):* | Adarsh K. Ramasubramonian Ye-Kui Wang Ying Chen  5775 Morehouse Drive San Diego, CA 92121, USA  Chong Soon Lim (Panasonic)  Sachin Deshpande (Sharp)  Hendry (LG Electronics) | Tel: Email: | 1-858-658-5804 [aramasub@qualcomm.com](mailto:aramasub@qualcomm.com)  1-858-651-8345 [yekuiw@qualcomm.com](mailto:yekuiw@qualcomm.com)  1-858-845-6589 [cheny@qualcomm.com](mailto:cheny@qualcomm.com)  [chongsoon.lim@sg.panasonic.com](mailto:chongsoon.lim@sg.panasonic.com)  +1 360 817 8486 [sdeshpande@sharplabs.com](mailto:sdeshpande@sharplabs.com)  [hendry.hendry@lge.com](mailto:hendry.hendry@lge.com) |
| *Source:* | Qualcomm Incorporated, Sharp, Panasonic, LG Electronics | | |

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

# Abstract

This document proposes to enable the inclusion of candidate long-term reference pictures, as part of the reference picture set signalling in the sequence parameter set. Two slightly different options are provided. The document reports that, for test condition 2.6 in JCTVC-H0725 [1], the proposed method uses 36% and 42% fewer bits, respectively for the two proposed options, to signal the syntax elements related to long-term reference pictures in the sequence parameter set and the slice header, when compared to the signalling in HEVC text specification draft 7.

# Introduction

The current long-term reference picture (LTRP) signalling is present only in the slice header. Thus, for any particular LTRP, the POC information used to identify the LTRP must be included in all slice headers, as part of the reference picture set (RPS) information, since the LTRP is firstly identified as a long term reference until the LTRP becomes no longer needed for inter prediction reference. However, an LTRP may stay in the DPB for a long period, thus the current LTRP signalling requires the POC information of the LTRP to be repeated in all slice headers of pictures within the long period.

In many scenarios wherein LTRPs are pre-selected or the bitstream is multi-pass encoded, reference pictures that would be used for LTRPs and would stay in the DPB for long periods are known at the beginning of the bitstream. In these scenarios, signalling of candidate LTRPs (or more specifically, the LSB bits of the POC values of the LTRPs and used\_by\_curr\_pic\_lt flag when using option 2 method) in the SPS can save a lot of bits to be transmitted, because in this case, instead of transmitting directly and repeatedly the POC LSB information of the LTRPs in the slice header, only the indices to the candidate LTRPs included in the SPS need to be transmitted.

Thus we propose to enable the inclusion of candidate long-term reference pictures in the SPS for reference picture set signalling. For this we describe two slightly different syntax options. We have slight preference for the option 2 as it has similar complexity as option 1 but provides higher bit savings. The proposed syntax and semantics changes for the two options are provided in the remainder of this document.

# Proposal

We first describe option 1 syntax and semantics in section 2.1-2.3, followed by a slight variant option 2 syntax and semantics in section 2.4-2.6.

## Option 1

### SPS changes

|  |  |
| --- | --- |
| seq\_parameter\_set\_rbsp( ) { | Descriptor |
| **...** |  |
| **num\_short\_term\_ref\_pic\_sets** | ue(v) |
| for( i = 0; i < num\_short\_term\_ref\_pic\_sets; i++) |  |
| short\_term\_ref\_pic\_set( i ) |  |
| **long\_term\_ref\_pics\_present\_flag** | u(1) |
| if( long\_term\_ref\_pics\_present\_flag ) { |  |
| **num\_long\_term\_ref\_pics\_sps** | ue(v) |
| for( i = 0; i < num\_long\_term\_ref\_pics\_sps; i++ ) |  |
| **lt\_ref\_pic\_poc\_lsb\_sps**[ i ] | u(v) |
| } |  |
| ... |  |
| } |  |

**num\_long\_term\_ref\_pics\_sps** specifies the number of long-term reference pictures that are specified in the sequence parameter set. The value of num\_long\_term\_ref\_pics\_sps shall be in the range of 0 to 32, inclusive.

**lt\_ref\_pic\_poc\_lsb\_sps**[ i ] specifies the least significant bits of the picture order count of the i-th long-term reference picture specified in the sequence parameter set. The number of bits used to represent lt\_ref\_pic\_poc\_lsb\_sps[ i ] shall be equal to log2\_max\_pic\_order\_cnt\_lsb\_minus4 + 4.

### Slice header

|  |  |
| --- | --- |
| slice\_header( ) { | Descriptor |
| ... |  |
| if( IdrPicFlag ) { |  |
| **idr\_pic\_id** | ue(v) |
| **no\_output\_of\_prior\_pics\_flag** | u(1) |
| } else { |  |
| **pic\_order\_cnt\_lsb** | u(v) |
| **short\_term\_ref\_pic\_set\_sps\_flag** | u(1) |
| if( !short\_term\_ref\_pic\_set\_sps\_flag ) |  |
| short\_term\_ref\_pic\_set( num\_short\_term\_ref\_pic\_sets ) |  |
| else |  |
| **short\_term\_ref\_pic\_set\_idx** | u(v) |
| if( long\_term\_ref\_pics\_present\_flag ) { |  |
| **num\_long\_term\_pics** | ue(v) |
| if( num\_long\_term\_ref\_pics\_sps ) |  |
| **num\_long\_term\_sps** | ue(v) |
| for( i = 0; i < num\_long\_term\_sps + num\_long\_term\_pics; i++ ) { |  |
| if ( i < num\_long\_term\_sps ) |  |
| **lt\_idx\_sps**[ i ] | u(v) |
| else |  |
| **poc\_lsb\_lt**[ i ] | u(v) |
| **delta\_poc\_msb\_present\_flag**[ i ] | u(1) |
| if( delta\_poc\_msb\_present\_flag[ i ] ) |  |
| **delta\_poc\_msb\_cycle\_lt**[ i ] | ue(v) |
| **used\_by\_curr\_pic\_lt\_flag**[ i ] | u(1) |
| } |  |
| } |  |
| } |  |
| ... |  |
| } |  |

**num\_long\_term\_pics** specifies the number of the long-term reference pictures that are to be included in the long-term reference picture set of the current picture and that are directly signalled in the slice header. The value of num\_long\_term\_pics shall be in the range of 0 to sps\_max\_dec\_pic\_buffering[ sps\_max\_temporal\_layers\_minus1 ] – NumNegativePics[ StRpsIdx ] – NumPositivePics[ StRpsIdx ] – num\_long\_term\_sps, inclusive. When not present, the value of num\_long\_term\_pics is inferred to be equal to 0.

**num\_long\_term\_sps** specifies the number of long-term reference pictures that are specified in the active sequence parameter set and that are to be included in the long-term reference picture set of the current picture. If num\_long\_term\_sps is not present, the value is inferred to be equal to 0. The value of num\_long\_term\_sps shall be in the range of 0 to Min( num\_long\_term\_ref\_pics\_sps, max\_dec\_pic\_buffering[ max\_temporal\_layers\_minus1 ] – NumNegativePics[ StRpsIdx ] – NumPositivePics[ StRpsIdx ] – num\_long\_term\_pics ), inclusive.

**lt\_idx\_sps**[ i ] specifies the index, to the list of long-term reference pictures specified in the active sequence parameter set, of the i-th long-term reference picture inherited from the referred sequence parameter set to the long-term reference picture set of the current picture. The value of lt\_idx\_sps[ i ] shall be in the range of 0 to num\_long\_term\_ref\_pics\_sps - 1, inclusive. The number of bits used to represent lt\_idx\_sps[ i ] shall be equal to Ceil( Log2( num\_long\_term\_ref\_pics\_sps ) ). For any values of j and k in the range of 0 to num\_long\_term\_sps – 1, inclusive, if j is less than k, lt\_ref\_pic\_poc\_lsb\_sps [ lt\_idx\_sps[ j ] ] shall not be less than lt\_ref\_pic\_poc\_lsb\_sps [ lt\_idx\_sps[ k ] ].

The variable PocLsbLt[ i ] is derived as follows.

if(i < num\_long\_term\_sps)  
 PocLsbLt[ i ] = lt\_ref\_pic\_poc\_lsb\_sps [ lt\_idx\_sps[ i ] ]  
 else  
 PocLsbLt [ i ] = poc\_lsb\_lt[ i ]

The value of PocLsbLt[ i ] shall be in the range of 0 to MaxPicOrderCntLsb − 1, inclusive.

**...**

**delta\_poc\_msb\_cycle\_lt**[ i ]is used to determine the value of the most significant bits of the picture order count value of the i-th long-term reference picture that is included in the long-term reference picture set of the current picture.

The variable DeltaPocMSBCycleLt[ i ] is derived as follows.

if( i = = 0 | | i = = num\_long\_term\_sps | | PocLsbLt[ i − 1 ] != PocLsbLt[ i ] )   
 DeltaPocMSBCycleLt[ i ] = delta\_poc\_msb\_cycle\_lt[ i ]  
 else (7‑42)  
 DeltaPocMSBCycleLt[ i ] = delta\_poc\_msb\_cycle\_lt[ i ] + DeltaPocMSBCycleLt[ i − 1 ]

The value of DeltaPocMSBCycleLt[ i ] \* MaxPicOrderCntLsb + pic\_order\_cnt\_lsb – PocLsbLt[ i ] shall be in the range of 1 to 224 – 1, inclusive.

### Change on the decoding process for reference picture set

This process is invoked once per picture, after decoding of a slice header but prior to the decoding of any coding unit and prior to the decoding process for reference picture list construction of the slice as specified in subclause 8.3.3. The process may result in marking one or more reference pictures as "unused for reference".

….

* Otherwise, the following applies for derivation of the five lists of picture order count values and the numbers of entries.

for( i = 0, j = 0, k = 0; i < NumNegativePics[ StRpsIdx ] ; i++ )  
 if( UsedByCurrPicS0[ StRpsIdx ][ i ] )  
 PocStCurrBefore[ j++ ] = PicOrderCntVal + DeltaPocS0[ StRpsIdx ][ i ]  
 else  
 PocStFoll[ k++ ] = PicOrderCntVal + DeltaPocS0[ StRpsIdx ][ i ]  
NumPocStCurrBefore = j  
  
for( i = 0, j = 0; i < NumPositivePics[ StRpsIdx ]; i++ )  
 if( UsedByCurrPicS1[ StRpsIdx ][ i ] )  
 PocStCurrAfter[ j++ ] = PicOrderCntVal + DeltaPocS1[ StRpsIdx ][ i ]  
 else  
 PocStFoll[ k++ ] = PicOrderCntVal + DeltaPocS1[ StRpsIdx ][ i ]  
NumPocStCurrAfter = j  
NumPocStFoll = k (8‑5)  
  
for( i = 0, j = 0, k = 0; i < num\_long\_term\_sps + num\_long\_term\_pics; i++ )  
 if( delta\_poc\_msb\_present\_flag[ i ] )  
 if( used\_by\_curr\_pic\_lt\_flag[ i ] )  
 PocLtCurr[ j++ ] = PicOrderCntVal − DeltaPocMSBCycleLt[ i ] \* MaxPicOrderCntLsb −  
 pic\_order\_cnt\_lsb + PocLsbLt[ i ]  
 else  
 PocLtFoll[ k++ ] = PicOrderCntVal − DeltaPocMSBCycleLt[ i ] \* MaxPicOrderCntLsb −  
 pic\_order\_cnt\_lsb + PocLsbLt[ i ]  
 else  
 if( used\_by\_curr\_pic\_lt\_flag[ i ] )  
 PocLtCurr[ j++ ] = PocLsbLt[ i ]  
 else  
 PocLtFoll[ k++ ] = PocLsbLt[ i ]  
  
NumPocLtCurr = j  
NumPocLtFoll = k

## Option 2

The difference in option2 compared to option 1 is as follows:

* used\_by\_curr\_pic\_lt flag for long term pictures signaled in SPS is also signaled in SPS.
* lt\_idx\_sps[ i ] index sent in slice header is ue(v) coded (instead of u(v) coding in option1).

The syntax for option2 for SPS is shown in section 2.4 and for slice header is shown in section 2.5. For the semantics only differences compared to section 2.1 and 2.2 respectively are described.

### SPS changes

|  |  |
| --- | --- |
| seq\_parameter\_set\_rbsp( ) { | Descriptor |
| **...** |  |
| **num\_short\_term\_ref\_pic\_sets** | ue(v) |
| for( i = 0; i < num\_short\_term\_ref\_pic\_sets; i++) |  |
| short\_term\_ref\_pic\_set( i ) |  |
| **long\_term\_ref\_pics\_present\_flag** | u(1) |
| if( long\_term\_ref\_pics\_present\_flag ) { |  |
| **num\_long\_term\_ref\_pics\_sps** | ue(v) |
| for( i = 0; i < num\_long\_term\_ref\_pics\_sps; i++ ) { |  |
| **lt\_ref\_pic\_poc\_lsb\_sps**[ i ] | u(v) |
| **used\_by\_curr\_pic\_lt\_sps\_flag**[ i ] | u(1) |
| } |  |
| } |  |
| ... |  |
| } |  |

**used\_by\_curr\_pic\_lt\_sps\_flag**[ i ] equal to 0 specifies that the i-th long-term reference picture specified in the sequence parameter set is not used for reference by the current picture.

## Slice header

|  |  |
| --- | --- |
| slice\_header( ) { | Descriptor |
| ... |  |
| if( IdrPicFlag ) { |  |
| **idr\_pic\_id** | ue(v) |
| **no\_output\_of\_prior\_pics\_flag** | u(1) |
| } else { |  |
| **pic\_order\_cnt\_lsb** | u(v) |
| **short\_term\_ref\_pic\_set\_sps\_flag** | u(1) |
| if( !short\_term\_ref\_pic\_set\_sps\_flag ) |  |
| short\_term\_ref\_pic\_set( num\_short\_term\_ref\_pic\_sets ) |  |
| else |  |
| **short\_term\_ref\_pic\_set\_idx** | u(v) |
| if( long\_term\_ref\_pics\_present\_flag ) { |  |
| **num\_long\_term\_pics** | ue(v) |
| if( num\_long\_term\_ref\_pics\_sps ) |  |
| **num\_long\_term\_sps** | ue(v) |
| for( i = 0; i < num\_long\_term\_sps + num\_long\_term\_pics; i++ ) { |  |
| if ( i < num\_long\_term\_sps ) |  |
| **lt\_idx\_sps**[ i ] | ue(v) |
| else { |  |
| **poc\_lsb\_lt**[ i ] | u(v) |
| **used\_by\_curr\_pic\_lt\_flag**[ i ] | u(1) |
| } |  |
| **delta\_poc\_msb\_present\_flag**[ i ] | u(1) |
| if( delta\_poc\_msb\_present\_flag[ i ] ) |  |
| **delta\_poc\_msb\_cycle\_lt**[ i ] | ue(v) |
| **~~used\_by\_curr\_pic\_lt\_flag~~**~~[ i ]~~ | ~~u(1)~~ |
| } |  |
| } |  |
| } |  |
| ... |  |
| } |  |

**lt\_idx\_sps**[ i ] specifies the index, to the list of long-term reference pictures specified in the active sequence parameter set, of the i-th long-term reference picture inherited from the referred sequence parameter set to the long-term reference picture set of the current picture. The value of lt\_idx\_sps[ i ] shall be in the range of 0 to num\_long\_term\_ref\_pics\_sps - 1, inclusive. ~~The number of bits used to represent lt\_idx\_sps[ i ] shall be equal to Ceil( Log2( num\_long\_term\_ref\_pics\_sps ) ).~~ For any values of j and k in the range of 0 to num\_long\_term\_sps – 1, inclusive, if j is less than k, lt\_ref\_pic\_poc\_lsb\_sps [ lt\_idx\_sps[ j ] ] shall not be less than lt\_ref\_pic\_poc\_lsb\_sps [ lt\_idx\_sps[ k ] ].

The variable PocLsbLt[ i ] is derived as follows.

if(i < num\_long\_term\_sps)  
 PocLsbLt[ i ] = lt\_ref\_pic\_poc\_lsb\_sps [ lt\_idx\_sps[ i ] ]  
 else  
 PocLsbLt [ i ] = poc\_lsb\_lt[ i ]

The value of PocLsbLt[ i ] shall be in the range of 0 to MaxPicOrderCntLsb − 1, inclusive.

## Change on the decoding process for reference picture set

This process is invoked once per picture, after decoding of a slice header but prior to the decoding of any coding unit and prior to the decoding process for reference picture list construction of the slice as specified in subclause 8.3.3. The process may result in marking one or more reference pictures as "unused for reference".

….

* Otherwise, the following applies for derivation of the five lists of picture order count values and the numbers of entries.

for( i = 0, j = 0, k = 0; i < NumNegativePics[ StRpsIdx ] ; i++ )  
 if( UsedByCurrPicS0[ StRpsIdx ][ i ] )  
 PocStCurrBefore[ j++ ] = PicOrderCntVal + DeltaPocS0[ StRpsIdx ][ i ]  
 else  
 PocStFoll[ k++ ] = PicOrderCntVal + DeltaPocS0[ StRpsIdx ][ i ]  
NumPocStCurrBefore = j  
  
for( i = 0, j = 0; i < NumPositivePics[ StRpsIdx ]; i++ )  
 if( UsedByCurrPicS1[ StRpsIdx ][ i ] )  
 PocStCurrAfter[ j++ ] = PicOrderCntVal + DeltaPocS1[ StRpsIdx ][ i ]  
 else  
 PocStFoll[ k++ ] = PicOrderCntVal + DeltaPocS1[ StRpsIdx ][ i ]  
NumPocStCurrAfter = j  
NumPocStFoll = k (8‑5)  
  
for( i = 0, j = 0, k = 0; i < num\_long\_term\_sps + num\_long\_term\_pics; i++ )  
 if( delta\_poc\_msb\_present\_flag[ i ] )  
 if( used\_by\_curr\_pic\_lt\_flag[ i ] )  
 PocLtCurr[ j++ ] = PicOrderCntVal − DeltaPocMSBCycleLt[ i ] \* MaxPicOrderCntLsb −  
 pic\_order\_cnt\_lsb + PocLsbLt[ i ]  
 else  
 PocLtFoll[ k++ ] = PicOrderCntVal − DeltaPocMSBCycleLt[ i ] \* MaxPicOrderCntLsb −  
 pic\_order\_cnt\_lsb + PocLsbLt[ i ]  
 else  
 if(used\_by\_curr\_pic\_lt\_sps\_flag[ lt\_idx\_sps[ i ] ])  
 PocLtCurr[ j++ ] = PocLsbLt[ i ]  
 else  
 PocLtFoll[ k++ ] = PocLsbLt[ i ]  
  
NumPocLtCurr = j  
NumPocLtFoll = k

# Description of experiments

The test case 2.6 in the common conditions for reference picture marking and list construction, described in JCTVC-H0725[1], is used to determine the signalling performance of the proposed method. The test condition describes two scenes alternating at fixed time intervals, and scene changes occur at pre-determined time points. The first case considers scene intervals of 10s – 30s – 10s – 30s and the second case considers scene intervals of 20s – 60s – 20s – 60s. A comparison of the bit-counts of the syntax elements related to long-term reference picture signalling in the proposed option1 method, proposed option2 method and the latest HEVC working draft (WD7) is presented below. Results shown below include the bit-counts for two full scene-cycles ( 80s in Test 1 and 160 s in Test 2). The proposed method provides, on an average, 36% reduction (option 1) and 42% reduction (option 2) in the bit-count of the related syntax elements. The syntax elements used to compute the bit-counts are as follows.

1. For WD7: num\_long\_term\_pics, poc\_lsb\_lt, delta\_poc\_msb\_present\_flag, delta\_poc\_msb\_cycle\_lt, and used\_by\_curr\_pic\_lt\_flag.
2. For proposed method:

Option 1: all the elements used for the WD7 bit-count, and num\_long\_term\_ref\_pic\_sps, lt\_ref\_pic\_poc\_lsb\_sps, num\_long\_term\_sps, and lt\_idx\_sps.

Option 2: all the elements used for the WD7 bit-count, and num\_long\_term\_ref\_pic\_sps, lt\_ref\_pic\_poc\_lsb\_sps, used\_by\_curr\_pic\_lt\_sps\_flag , num\_long\_term\_sps, and lt\_idx\_sps.

|  |  |  |  |
| --- | --- | --- | --- |
|  | WD7 | Proposed option1 method | Proposed option 2 method |
| **Test 1: 10s – 30s – 10s – 30s** |  |  |  |
| LTRP signalling-related bit-count in SPS | 1 | 30 | 60 |
| LTRP signalling-related bit-count in slice header | 56896 | 36507 | 32939 |
| Total LTRP signalling-related bit-count | 56897 | 36537 | 32999 |
| % reduction |  | 36% | 42% |
| **Test 2: 20s – 60s – 20s – 60s** |  |  |  |
| LTRP signalling-related bit-count in SPS | 1 | 30 | 60 |
| LTRP signalling-related bit-count in slice header | 114695 | 73906 | 66754 |
| Total LTRP signalling-related bit-count | 114696 | 73936 | 66814 |
| % reduction |  | 36% | 42% |

As the option 2 method saves more bits and has similar complexity as option 1, we prefer option 2 method.

# References

[1] Y-K. Wang, M. M. Hannuksela, T. K. Tan, R. Sjöberg, and Yan Ye, “Common conditions for reference picture marking and list construction proposals,” JCT-VC documents JCTVC-H0725, San Jose, February 2012.

# 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).**

**Panasonic 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).**

**Sharp 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).**

**LG Electronics 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).**