|  |  |
| --- | --- |
| **Joint Collaborative Team on Video Coding (JCT-VC)**  **of ITU-T SG16 WP3 and ISO/IEC JTC1/SC29/WG11**  7th Meeting: Geneva, CH, 21-30 November, 2011 | Document: JCTVC-G331 |

|  |  |  |  |
| --- | --- | --- | --- |
| *Title:* | **On NAL unit types and slice types** | | |
| *Status:* | Input Document to JCT-VC | | |
| *Purpose:* | Proposal | | |
| *Author(s) or Contact(s):* | Ye-Kui Wang Ying Chen  5775 Morehouse Dr San Diego, CA 92121 USA | Tel: Email: | 1-858-651-8345 [yekuiw@qualcomm.com](mailto:yekuiw@qualcomm.com)  1-858-845-6589 [cheny@qualcomm.com](mailto:cheny@qualcomm.com) |
| *Source:* | Qualcomm Incorporated | | |

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

# Abstract

At the previous JCT-VC meeting, it was agreed to change nal\_ref\_idc (2 bits) to nal\_ref\_flag (1 bit), and change nal\_unit\_type from 5 bits to 6 bits. Consequently, the total number of NAL unit types doubled from 32 to 64. This document proposes an allocation of the 64 NAL unit types to different categories of NAL unit types, and raises some NAL unit type related questions for discussion. Furthermore, it is proposed to add slice types 3 to 5, with similar semantics as slice types 5 to 7 in AVC.

# Allocation of NAL unit types

## Summary of the current allocation of NAL unit types

In HEVC WD4d4, the 32 NAL unit types are categorized as follows:

* Unspecified: 0, 24..31
  + shall not precede the first VCL NAL unit of the coded picture
* Reserved: 2, 3, 10, 11, 13..23
  + 14..18: may start a new access unit
  + 20..23: shall not precede the first VCL NAL unit of the coded picture
  + 2, 3, 10, 11, 13: no restriction specified
* Other (specified and not reserved) NAL unit types: 1, 4..9, 12
  + 1, 4..9: may start a new access unit
    - 6: shall precede the first VCL NAL unit of the coded picture
    - 9: shall be the first NAL unit in an access unit
  + 12: shall not precede the first VCL NAL unit of the coded picture

## Proposed allocation of NAL unit types

### Summary

In the proposal, the 64 NAL unit types are categorized as follows:

* Unspecified: 0, 48..63
  + shall not precede the first VCL NAL unit of the coded picture
* Reserved: 10..47
  + 10..25: may start a new access unit
  + 26..47: shall not precede the first VCL NAL unit of the coded picture
* Other (specified and not reserved) NAL unit types: 1..9
  + 1..8: may start a new access unit
    - 4: shall precede the first VCL NAL unit of the coded picture
    - 8: shall be the first NAL unit in an access unit
  + 9: shall not precede the first VCL NAL unit of the coded picture

The text changes are provided in the following subsections, wherein changes are marked. In lieu with the proposed changes, some purely editorial changes throughout the HEVC WD would be needed. The proponents will provide those changes to the editors if the proposal is agreed.

### Semantics of nal\_unit\_type

**nal\_unit\_type** specifies the type of RBSP data structure contained in the NAL unit as specified in Table 7‑1.

NAL units that use nal\_unit\_type equal to 0 or in the range of 48..63, inclusive, shall not affect the decoding process specified in this Recommendation | International Standard.

NOTE 2 – NAL unit types 0 and 48..63, inclusive, may be used as determined by the application. No decoding process for these values of nal\_unit\_type is specified in this Recommendation | International Standard. Since different applications might use NAL unit types 0 and 48..63, inclusive for different purposes, particular care must be exercised in the design of encoders that generate NAL units with nal\_unit\_type equal to 0 or in the range of 48..63, inclusive, and in the design of decoders that interpret the content of NAL units with nal\_unit\_type equal to 0 or in the range of 48..63, inclusive.

Decoders shall ignore (remove from the bitstream and discard) the contents of all NAL units that use reserved values of nal\_unit\_type.

NOTE 3 – This requirement allows future definition of compatible extensions to this Recommendation | International Standard.

Table 7‑1 – NAL unit type codes and NAL unit type classes

|  |  |  |
| --- | --- | --- |
| **nal\_unit\_type** | **Content of NAL unit and RBSP syntax structure** | **NAL unit type class** |
| 0 | Unspecified | non-VCL |
| 1 | Coded slice of a non-IDR and non-CRA picture slice\_layer\_rbsp( ) | VCL |
| 2 | Coded slice of a CRA picture  slice\_layer\_rbsp( ) | VCL |
| 3 | Coded slice of an IDR picture slice\_layer\_rbsp( ) | VCL |
| 4 | Supplemental enhancement information (SEI) sei\_rbsp( ) | non-VCL |
| 5 | Sequence parameter set seq\_parameter\_set\_rbsp( ) | non-VCL |
| 6 | Picture parameter set pic\_parameter\_set\_rbsp( ) | non-VCL |
| 7 | Adaptation parameter set aps\_rbsp( ) | non-VCL |
| 8 | Access unit delimiter access\_unit\_delimiter\_rbsp( ) | non-VCL |
| 9 | Filler data filler\_data\_rbsp( ) | non-VCL |
| 10..47 | Reserved | n/a |
| 48..63 | Unspecified | non-VCL |

Coded slice NAL unit collectively refers to a coded slice with nal\_unit\_type in the range of 1 to 3, inclusive. The variable IdrPicFlag is specified as

IdrPicFlag = ( ( nal\_unit\_type = = 3 ) ? 1 : 0 ) (7‑1)

When the value of nal\_unit\_type is equal to 3 for a NAL unit containing a slice of a particular picture, the picture shall not contain NAL units with nal\_unit\_type equal to 1 or 2. For coded video sequences conforming to one or more of the profiles specified in Annex A that are decoded using the decoding process specified in clauses 2-9, such a picture is referred to as an IDR picture.

### Order of NAL units and coded pictures and association to access units

This subclause specifies the order of NAL units and coded pictures and association to access unit for coded video sequences that conform to one or more of the profiles specified in Annex A that are decoded using the decoding process specified in clauses 2-9.

An access unit consists of one coded picture and zero or more non-VCL NAL units. The association of VCL NAL units to coded pictures is described in subclause 7.4.1.2.5.

The first access unit in the bitstream starts with the first NAL unit of the bitstream.

The first of any of the following NAL units after the last VCL NAL unit of a coded picture specifies the start of a new access unit:

– access unit delimiter NAL unit (when present),

– sequence parameter set NAL unit (when present),

– picture parameter set NAL unit (when present),

– adaptation parameter set NAL unit (when present),

– SEI NAL unit (when present),

– NAL units with nal\_unit\_type in the range of 10 to 25, inclusive (when present),

– first VCL NAL unit of a coded picture (always present).

The constraints for the detection of the first VCL NAL unit of a coded picture are specified in subclause 7.4.1.2.4.

The following constraints shall be obeyed by the order of the coded pictures and non-VCL NAL units within an access unit:

– When an access unit delimiter NAL unit is present, it shall be the first NAL unit. There shall be at most one access unit delimiter NAL unit in any access unit.

– When any SEI NAL units are present, they shall precede the first VCL NAL unit of the coded picture.

– When an SEI NAL unit containing a buffering period SEI message is present, the buffering period SEI message shall be the first SEI message payload of the first SEI NAL unit in the access unit.

– NAL units having nal\_unit\_type equal to 0, 9, or in the range of 26 to 47, inclusive, shall not precede the first VCL NAL unit of the coded picture.

NOTE 1 – Sequence parameter set NAL units or picture parameter set NAL units may be present in an access unit, but cannot follow the last VCL NAL unit of the coded picture within the access unit, as this condition would specify the start of a new access unit.

## Discussions

1. Should “may start a new access unit” be changed to “may precede the first VCL NAL unit of the coded picture”, such that the opposite is exactly “shall not precede the first VCL NAL unit of the coded picture”?
2. Should a coded slice NAL unit for which cross-slice-boundary in-picture prediction is allowed use a different NAL unit type than a coded slice NAL unit for which cross-slice-boundary in-picture prediction is disallowed?
3. Should a different NAL unit type be used for SEI NAL units containing slice-level SEI messages, if any, as such an SEI NAL unit may succeed the first VCL NAL unit of the coded picture in the same access unit?

# Additional slice types

The semantics of the slice\_type syntax element in AVC is as follows.

slice\_type specifies the coding type of the slice according to .

Table ‑ – Name association to slice\_type

|  |  |
| --- | --- |
| slice\_type | Name of slice\_type |
| 0 | P (P slice) |
| 1 | B (B slice) |
| 2 | I (I slice) |
| 3 | SP (SP slice) |
| 4 | SI (SI slice) |
| 5 | P (P slice) |
| 6 | B (B slice) |
| 7 | I (I slice) |
| 8 | SP (SP slice) |
| 9 | SI (SI slice) |

When slice\_type has a value in the range 5..9, it is a requirement of bitstream conformance that all other slices of the current coded picture shall have a value of slice\_type equal to the current value of slice\_type or equal to the current value of slice\_type minus 5.

NOTE 1 – Values of slice\_type in the range 5..9 can be used by an encoder to indicate that all slices of a picture have the same value of (slice\_type % 5). Values of slice\_type in the range 5..9 are otherwise equivalent to corresponding values in the range 0..4.

When nal\_unit\_type is equal to 5 (IDR picture), slice\_type shall be equal to 2, 4, 7, or 9.

When max\_num\_ref\_frames is equal to 0, slice\_type shall be equal to 2, 4, 7, or 9.

We propose to enable the similar functionality as in AVC, with the following changes to the semantics of the slice\_type syntax element (changes are marked):

**slice\_type** specifies the coding type of the slice according to Table 7‑3.

Table 7‑3 – Name association to slice\_type

|  |  |
| --- | --- |
| slice\_type | Name of slice\_type |
| 0 | P (P slice) |
| 1 | B (B slice) |
| 2 | I (I slice) |
| 3 | P (P slice) |
| 4 | B (B slice) |
| 5 | I (I slice) |

When slice\_type has a value in the range 3..5, it is a requirement of bitstream conformance that all other slices of the current coded picture shall have a value of slice\_type equal to the current value of slice\_type or equal to the current value of slice\_type minus 3.

NOTE 1 – Values of slice\_type in the range 3..5 can be used by an encoder to indicate that all slices of a picture have the same value of (slice\_type % 3). Values of slice\_type in the range 3..5 are otherwise equivalent to corresponding values in the range 0..2.

When nal\_unit\_type is equal to 5 (IDR picture), slice\_type shall be equal to 2 or 5.

When max\_num\_ref\_frames is equal to 0, slice\_type shall be equal to 2 or 5.

Accordingly, the syntax tables ref\_pic\_list\_modification( ) and ref\_pic\_list\_combination( ) are changed as follows (changes are marked).

|  |  |
| --- | --- |
| ref\_pic\_list\_modification( ) { | Descriptor |
| if( slice\_type % 3 != 2 ) { |  |
| **ref\_pic\_list\_modification\_flag\_l0** | u(1) |
| if( ref\_pic\_list\_modification\_flag\_l0 ) |  |
| do { |  |
| **modification\_of\_pic\_nums\_idc** | ue(v) |
| if( modification\_of\_pic\_nums\_idc = = 0 | |  modification\_of\_pic\_nums\_idc = = 1 ) |  |
| **abs\_diff\_pic\_num\_minus1** | ue(v) |
| else if( modification\_of\_pic\_nums\_idc = = 2 ) |  |
| **long\_term\_pic\_num** | ue(v) |
| } while( modification\_of\_pic\_nums\_idc != 3 ) |  |
| } |  |
| if( slice\_type % 3 = = 1 ) { |  |
| **ref\_pic\_list\_modification\_flag\_l1** | u(1) |
| if( ref\_pic\_list\_modification\_flag\_l1 ) |  |
| do { |  |
| **modification\_of\_pic\_nums\_idc** | ue(v) |
| if( modification\_of\_pic\_nums\_idc = = 0 | |  modification\_of\_pic\_nums\_idc = = 1 ) |  |
| **abs\_diff\_pic\_num\_minus1** | ue(v) |
| else if( modification\_of\_pic\_nums\_idc = = 2 ) |  |
| **long\_term\_pic\_num** | ue(v) |
| } while( modification\_of\_pic\_nums\_idc != 3 ) |  |
| } |  |
| } |  |

|  |  |
| --- | --- |
| ref\_pic\_list\_combination( ) { | Descriptor |
| if( slice\_type % 3 = = 1 ) { |  |
| **ref\_pic\_list\_combination\_flag** | u(1) |
| if( ref\_pic\_list\_combination\_flag ) { |  |
| **num\_ref\_idx lc\_active\_minus1** | ue(v) |
| **ref\_pic\_list\_modification\_flag\_lc** | u(1) |
| if( ref\_pic\_list\_modification\_flag\_lc) |  |
| for ( i =0; i <= num\_ref\_idx\_lc\_active\_minus1; i++ ) { |  |
| **pic\_from\_list\_0\_flag** | u(1) |
| **ref\_idx\_list\_curr** | ue(v) |
| } |  |
| } |  |
| } |  |
| } |  |

# 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).**