|  |  |
| --- | --- |
| **Joint Collaborative Team on 3D Video Coding Extensions**  **of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11**  5th Meeting: Vienna, AT, 27 July -2 Aug. 2013 | Document: JCT3V- E1009 |

|  |  |  |  |
| --- | --- | --- | --- |
| *Title:* | **Draft 2 of conformance testing of the 3D-AVC** | | |
| *Status:* | Output document | | |
| *Purpose:* | Draft | | |
| *Author(s) or Contact(s):* | Dmytro Rusanovskyy  Jyranmaankatu 33,  37140 Nokia,  Finland  Teruhiko Suzuki  2-10-1, Osaki, Shinagawa-ku,  Tokyo, 141-8610  JAPAN  Dong Tian  201 Broadway, 8th floor  Cambridge, MA 02139  U.S.A. | Tel: Email: | [Dmytro.Rusanovskyy@ieee.org](mailto:Dmytro.Rusanovskyy@ieee.org)  +81-50-3750-2740 [teruhikos@jp.sony.com](mailto:teruhikos@jp.sony.com)  +1-617-621-7556  [tian@merl.com](mailto:tian@merl.com) |
| *Source:* | Editors | | |

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

# Abstract

This document is a draft 2 of new amendment to ITU-T Rec. H.264.1 | ISO/IEC 14496-4 adding conformance test of 3D-AVC.

FINAL DRAFT AMENDMENT© ISO/IEC 2009 – All rights reservedISO/IEC 14496-4:2004/PDAM 42 63Part 4: Conformance testing, AMENDMENT 41: Conformance testing of 3D-AVCInformation technology — Coding of audio-visual objectsÉlément introductif — Élément central — Partie 4: Titre de la partieInformation technology — Coding of audio-visual objects — Part 4: Conformance testing, AMENDMENT 42: Conformance testing of 3D-AVCE2013-04-26(50) ApprovalISO/IECISO/IEC J200442AmendmentInternational Standard 2013ISO/IEC 14496‑ISO/IEC 14496‑4ISO/IEC 14496-4/PDAM 42  Coding of audio, picture, multimedia and hypermedia informationInformation technology11291 2Heading 2,H2,H21,Œ©o‚µ 2,h2,?co??E 2,뙥2,?c1,?co?ƒÊ 2,?2,Œ1,Œ2,Œ©2,DO NOT USE\_h2,título 2,Œ©1,Œ©\_o‚µ 2,2,Header 2,2nd levelHeading 1    02 STD Version 2.1c250   4K:\29IS\14496\14496-4\2nd E\Amd38\w10895.doc

# Foreword

This Amendment establishes conformance test requirements for conformance to ITU‑T Rec. H.264 | ISO/IEC 14496‑10.

In this Amendment, additional text to ITU‑T Rec. H.264 | ISO/IEC 14496‑10 is specified for testing the conformance of ITU‑T Rec. H.264 | ISO/IEC 14496‑10 video decoders including in particular the 3D-AVC Profiles.

The following subclauses specify the normative tests for verifying conformance of ITU‑T Rec. H.264 | ISO/IEC 14496‑10 video bitstreams and decoders. These normative tests make use of test data (bitstream test suites) provided as an electronic annex to this document, and of the reference software decoder specified in ITU‑T Rec. H.264.2 | ISO/IEC 14496‑5 with source code available in electronic format.

*Replace 10.6.4 with the followings:*



### Procedure to test bitstreams

A bitstream that claims conformance with ITU‑T H.264 | ISO/IEC 14496‑10 shall pass the following normative test:

The bitstream shall be decoded by processing it with the reference software decoder. When processed by the reference software decoder, the bitstream shall not cause any error or non‑conformance messages to be reported by the reference software decoder. This test should not be applied to bitstreams that are known to contain errors introduced by transmission, as such errors are highly likely to result in bitstreams that lack conformance to ITU-T H.264 | ISO/IEC 14496‑10.

Successfully passing the reference software decoder test provides only a strong presumption that the bitstream under test is conforming to the video layer, i.e., that it does indeed meet all the requirements for the video layer (except Annexes C, D and E and clauses G.12, H.12, I.12 and J.12) specified in ITU-T H.264 | ISO/IEC 14496-10 that are tested by the reference software decoder.

Additional tests may be necessary to more thoroughly check that the bitstream properly meets all the requirements specified in ITU-T H.264 | ISO/IEC 14496-10 including the hypothetical reference decoder (HRD) conformance (based on Annexes C, D and E and clauses G.12, H.12, I.12 and J.12). These complementary tests may be performed using other video bitstream verifiers that perform more complete tests than those implemented by the reference software decoder.

ITU-T H.264 | ISO/IEC 14496-10 contains several informative recommendations that are not an integral part of that Recommendation | International Standard. When testing a bitstream for conformance, it may also be useful to test whether or not the bitstream follows those recommendations.

To check correctness of a bitstream, it is necessary to parse the entire bitstream and to extract all the syntax elements and other values derived from those syntactic elements and used by the decoding process specified in ITU‑T H.264 | ISO/IEC 14496-10.

A verifier may not necessarily perform all stages of the decoding process specified in ITU‑T H.264 | ISO/IEC 14496-10 in order to verify bitstream correctness. Many tests can be performed on syntax elements in a state prior to their use in some processing stages.

*Replace 10.6.5.1 with the following:*



#### Conformance bitstreams

A bitstream has values of profile\_idc, level\_idc, and constraint\_setX\_flag (where X is a number in the range of 0 to 6, inclusive) corresponding to a set of specified constraints on a bitstream for which a decoder conforming to a specified profile and level is required in Annex A, clause G.10, H.10, I.10 or J.10 of ITU-T H.264 | ISO/IEC 14496-10 to properly

*Replace 10.6.5.3 with the following:*



#### Requirements on output of the decoding process and timing

Two classes of decoder conformance are specified:

– output order conformance; and

– output timing conformance.

The output of the decoding process is specified in clauses 8, G.8, G.12, H.8, H.12, I.8, I.12, J.8, J.12 and Annex C of ITU‑T H.264 | ISO/IEC 14496-10.

For output order conformance, it is a requirement that all of the decoded pictures specified for output in Annex C, clause G.12, H.12, I.12 or J.12 of ITU‑T H.264 | ISO/IEC 14496-10 shall be output by a conforming decoder in the specified order and that the values of the decoded samples in all of the pictures that are output shall be (exactly equal to) the values specified in clause 8, clause G.8, H.8, I.8 or J.8 of ITU-T H.264 | ISO/IEC 14496-10.

For output timing conformance, it is a requirement that a conforming decoder shall also output the decoded samples at the rates and times specified in Annex C, clause G.12, H.12, I.12 or J.12 of ITU-T H.264 | ISO/IEC 14496-10.

The display process, which ordinarily follows the output of the decoding process, is outside the scope of this Recommendation | International Standard.

*Replace 10.6.5.6 with the following:*



#### Dynamic tests for output timing conformance

Dynamic tests are applied to check that all the decoded samples are output and that the timing of the output of the decoder's decoded samples conforms to the specification of clauses 8, G.8, G.12, H.8, H.12, I.8, I.12 andJ,12; and Annex C of ITU‑T H.264 | ISO/IEC 14496-10, and to verify that the HRD models (as specified by the CPB and DPB specification in Annex C, clause G.12, H.12, I.12 or J.12 of ITU-T H.264 | ISO/IEC 14496-10) are not violated when the bits are delivered at the proper rate.

The dynamic test is often easier to perform on a complete decoder system, which may include a systems decoder, a video decoder and a display process. It may be possible to record the output of the display process and to check that display order and timing of fields or frames are correct at the output of the display process. However, since the display process is not within the normative scope of ITU-T H.264 | ISO/IEC 14496-10, there may be cases where the output of the display process differs in timing or value even though the video decoder is conforming. In this case, the output of the video decoder itself (before the display process) would need to be captured in order to perform the dynamic tests on the video decoder. In particular the field or frame order and timing shall be correct.

If buffering period SEI and picture timing SEI are included in the test bitstream, HRD conformance shall be verified using the values of initial\_cpb\_removal\_delay, initial\_cpb\_removal\_delay\_offset, cpb\_removal\_delay and dpb\_removal\_delay that are included in the bitstream.

If buffering period SEI and picture timing SEI are not included in the bitstream, the following inferences shall be made to generate the missing parameters:

– fixed\_frame\_rate\_flag shall be inferred to be 1.

– low\_delay\_hrd\_flag shall be inferred to be 0.

– cbr\_flag shall be inferred to be 0.

– The frame rate of the bitstream shall be inferred to be the frame rate value specified in the corresponding table of clause 6.7, where the bitstream is listed. If this is missing, then a frame rate of either 25 or 30000 ÷ 1001 can be inferred.

– time\_scale shall be set to 90,000 and the value of num\_units\_in\_tick shall be computed based on field rate (twice the frame rate).

– The bit rate of the bitstream shall be inferred to be the maximum value for the level specified in Table A‑1 in ITU-T H.264 | ISO/IEC 14496-10.

– CPB and DPB sizes shall be inferred to be the maximum value for the level specified in Table A-1 in ITU‑T H.264 | ISO/IEC 14496-10.

With the above inferences, the HRD shall be operated as follows.

– The CPB is filled starting at time t = 0, until it is full, before removal of the first access unit. This means that the initial\_cpb\_removal\_delay shall be inferred to be equal to the total CPB buffer size divided by the bit rate divided by 90000 (rounded downwards) and initial\_cpb\_removal\_delay\_offset shall be inferred to be equal to zero.

– The first access unit is removed at time t = initial\_cpb\_removal\_delay ÷ 90000 and subsequent access units are removed at intervals based on the frame distance, i.e., 2 \* (90000 ÷ num\_units\_in\_tick) or the field distance, i.e., (90000 / num\_units\_in\_tick), depending on whether the access unit is coded as a frame picture or field picture.

– Using these inferences, the CPB will not overflow or underflow and the DPB will not overflow.

*In 10.6.5.7, add the following at the end of the sub clause 10.6.5.7:*

A decoder that conforms to the 3D-AVC profile at a specific level shall be capable of decoding the specified bitstreams in Table 6. A decoder that conforms to the 3D-AVC profile shall also be capable of decoding all bitstreams that are required to be decoded by a Main or High profile decoder of the same level. In addition to the specified bitstreams in Table 6, a decoder that conforms to the 3D-AVC profile shall be capable of decoding the bitstreams in Tables 1, 2 and 4 that correspond to these requirements.

*Add the following after the sub clause 10.6.6.34.2:*



#### Test Bitstreams – 3D-AVC Profile

##### Test bitstream #MVDDR-1. #MVDDR-2

**Specification**: All slices are coded as I, P or B slices. Only the first picture is coded as an IDR access unit. Each view component contains only one slice. num\_views\_minus1 is equal to 1. NumDepthViews is equal to 2. The width and the height of depth view components are half of the texture view components. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T H.264 | ISO/IEC 14496-10. All tools specified in XX are enabled.

**Functional stage**: Decoding of texture view component and depth view component.

**Purpose**: Check that the decoder can properly decode lower resolution depth view component.

##### Test bitstream #MVDDR-3. #MVDDR-4

**Specification**: All slices are coded as I, P or B slices. Only the first picture is coded as an IDR access unit. Each view component contains only one slice. num\_views\_minus1 is equal to 1. NumDepthViews is equal to 2. The width and the height of depth view components are equal to the texture view components. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T H.264 | ISO/IEC 14496-10. All tools specified in JCT3V-D1009 are enabled.

**Functional stage**: Decoding of texture view component and depth view component.

**Purpose**: Check that the decoder can properly decode the same resolution depth view component.

##### Test bitstream #MVDCT-1

**Specification**: All slices are coded as I, P or B slices. Only the first picture is coded as an IDR access unit. Each view component contains only one slice. num\_views\_minus1 is equal to 2. NumDepthViews is equal to 3. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T H.264 | ISO/IEC 14496-10. View synthesis prediction (VSP) for dependent texture views is enabled.

**Functional stage**: Decoding of texture view component and depth view component.

**Purpose**: Check that the decoder can properly decode bitstream if VSP is utilized.

##### Test bitstream #MVDCT-2

**Specification**: All slices are coded as I, P or B slices. Only the first picture is coded as an IDR access unit. Each view component contains only one slice. num\_views\_minus1 is equal to 2. NumDepthViews is equal to 3. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T H.264 | ISO/IEC 14496-10. Depth-based motion vector prediction (DMVP) is enabled.

**Functional stage**: Decoding of texture view component and depth view component.

**Purpose**: Check that the decoder can properly decode bitstream if DMVP is utilized.

##### Test bitstream #MVDCT-3

**Specification**: All slices are coded as I, P or B slices. Only the first picture is coded as an IDR access unit. Each view component contains only one slice. num\_views\_minus1 is equal to 2. NumDepthViews is equal to 3. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T H.264 | ISO/IEC 14496-10. Adaptive luminance compensation (ALC) is enabled for coded texture views.

**Functional stage**: Decoding of texture view component and depth view component.

**Purpose**: Check that the decoder can properly decode bitstream if ALC is utilized.

##### Test bitstream #MVDCT-4

**Specification**: All slices are coded as I, P or B slices. Only the first picture is coded as an IDR access unit. Each view component contains only one slice. num\_views\_minus1 is equal to 2. NumDepthViews is equal to 3. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T H.264 | ISO/IEC 14496-10. Non-linear depth representation (NLDR) is enabled for coded depth views.

**Functional stage**: Decoding of texture view component and depth view component.

**Purpose**: Check that the decoder can properly decode bitstream if NLDR is utilized.

##### Test bitstream #MVDCT-5

**Specification**: All slices are coded as I, P or B slices. Only the first picture is coded as an IDR access unit. Each view component contains only one slice. num\_views\_minus1 is equal to 2. NumDepthViews is equal to 3. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T H.264 | ISO/IEC 14496-10. Slice header prediction (SHP) is enabled for texture component of dependent views and depth component of the base texture view.

**Functional stage**: Decoding of texture view component and depth view component.

**Purpose**: Check that the decoder can properly decode bitstream if SHP is utilized.

##### Test bitstream #MVDCT-6

**Specification**: All slices are coded as I, P or B slices. Only the first picture is coded as an IDR access unit. Each view component contains only one slice. num\_views\_minus1 is equal to 1. NumDepthViews is equal to 2. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T H.264 | ISO/IEC 14496-10. Multiview video plus depth (MVD) data is coded in order when texture component preceeds depth component of the same view. NBDV disparity derivation tools is utilized for improving coding efficiency.

**Functional stage**: Decoding of texture view component and depth view component.

**Purpose**: Check that the decoder can properly decode bitstream if texture preceeds the depth component and NBDV is used for disparity deriction.

*Add the following after Table AMD38.1:*

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Table AMD42.1 – Bitstreams for the 3D-AVC profiles | | | | | | | |
| Categories | Bitstream | Donated by | File Name | Multi-view Depth High |  | Level | Frame Rate (Frame/Sec) |
| Depth Resolution | MVDDR-1 | Nokia |  | X |  | 3 and higher | 30 |
| MVDDR-2 | Nokia |  | X |  | 4 and higher | 25 |
| Coding tools | MVDCT-1 | MERL |  | X |  | 3 and higher | 30 |
| MVDCT-2 | Mediatek (TBC) |  | X |  | 4 and higher | 25 |
| MVDCT-3 | Samsung  (TBC) |  | X |  | 3 and higher | 30 |
| MVDCT-4 | Poznan |  | X |  | 4 and higher | 25 |
| MVDCT-5 | Qualcomm  (TBC) |  | X |  | 4 and higher | 25 |
| MVDCT-6 | Qualcom (TBC) |  | X |  | 3 and higher | 30 |