INTERNATIONAL STANDARD

ISO/IEC 23094-3

First edition
2022-09

Information technology — General video coding —

Part 3: Conformance and reference software for low complexity enhancement video coding

Technologies de l'information — Codage vidéo général —

Partie 3: Conformité et logiciel de référence pour le codage vidéo d'amélioration de faible complexité

Reference number
ISO/IEC 23094-3:2022(E)

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Contents

Page

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of document should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives or www.iec.ch/members_experts/refdocs).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents) or the IEC list of patent declarations received (see https://patents.iec.ch).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT) see www.iso.org/iso/foreword.html. In the IEC, see www.iec.ch/understanding-standards.

This document was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 29, *Coding of audio, picture, multimedia and hypermedia information*.

A list of all parts in the ISO/IEC 23094 series can be found on the ISO and IEC websites.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html and www.iec.ch/national-committees.

Information technology — General video coding —

Part 3: Conformance and reference software for low complexity enhancement video coding

1 Scope

This document specifies a set of tests and procedures designed to verify whether bitstreams and decoders meet normative requirements specified in ISO/IEC 23094-2.

An encoder can claim conformance to ISO/IEC 23094-2 if the bitstreams that it generates are conforming bitstreams. Characteristics of coded bitstreams and decoders are defined in ISO/IEC 23094-2. Decoder characteristics define the properties and capabilities of the applied decoding process. The capabilities of a decoder specify which bitstreams the decoder can decode and reconstruct. A bitstream can be decoded by a decoder if the characteristics of the bitstream are within the specified decoder capabilities.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 23094-2:2021, *Information technology – General video coding — Part 2: Low complexity enhancement video coding*

ISO/IEC 14882, *Programming languages — C++*

3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 23094-2 and the following apply.

ISO and IEC maintain terminology databases for use in standardization at the following addresses:

— ISO Online browsing platform: available at https://www.iso.org/obp

— IEC Electropedia: available at https://www.electropedia.org/

3.1
bitstream
sequence of bits in the form of a NAL unit stream or a raw bitstream, that forms the representation of coded pictures and associated data forming one or more coded video sequences

3.2
decoder
embodiment of a process that operates on a *bitstream* (3.1) and may conform to the decoding process requirements specified for conformance

Note 1 to entry: The decoder does not include the display process, which is outside the scope of this document.

3.3
encoder
embodiment of a process that produces a *bitstream* (3.1)

Note 1 to entry: The process is not specified in this document (except in regard to identification of the reference software encoder).

3.4
reference software decoder
software which may decode a *bitstream* (3.1) encoded according to the syntax structure and which is accompanying this document

3.5
reference software encoder
encoding software which is accompanying this document

3.6
test base bitstream
bitstream (3.1) that is conformant to Rec. ITU-T H.264 | ISO/IEC 14496-10 or Rec. ITU-T H.265 | ISO/IEC 23008-2

3.7
test bitstream
bitstream (3.1) that is the combination of a *test base bitstream* (3.6) and a *test enhancement bitstream* (3.8)

3.8
test enhancement bitstream
bitstream (3.1) that is conformant to ISO/IEC 23094-2

4 Abbreviated terms

For the purposes of this document, the abbreviated terms given in ISO/IEC 23094-2 and the following apply.

API	Application programming interface
AVC	Advanced video coding
EVC	Essential video coding
HEVC	High efficiency video coding
LTM	LCEVC test model
VVC	Versatile video coding

5 Conventions

For the purposes of this document, relevant conventions are specified in of ISO/IEC 23094-2:2021, Clause 5.

6 Conformance testing for ISO/IEC 23094-2

6.1 General

The following subclauses specify normative tests for verifying conformance of video bitstreams as well as decoders. Those normative tests make use of test data (bitstream test suites) provided as an electronic attachment to this document and the reference software decoder specified in ISO/IEC 23094-2.

For the purpose of testing according to 6.5, test bitstreams are provided in the bitstream test suite. In the following subclauses, the normative tests use the test bitstreams for verifying the conformance of the decoding process of the test enhancement bitstreams.

A test base bitstream is conformant in accordance with the respective International Standard it implements. The conformance testing of the test base bitstream is not in the scope of this document.

This document includes an electronic attachment containing the conformance bitstreams identified within the text and the reference software. The electronic attachment is available at the following address: https://standards.iso.org/iso-iec/23094/-3/ed-1/en/.

6.2 Bitstream conformance

Bitstream conformance for ISO/IEC 23094-2 is specified by ISO/IEC 23094-2:2021, C.4.

6.3 Decoder conformance

Decoder conformance for ISO/IEC 23094-2 is specified by ISO/IEC 23094-2:2021, C.5.

6.4 Procedure to test bitstreams

A bitstream that claims conformance with ISO/IEC 23094-2 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 ISO/IEC 23094-2.

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) specified in ISO/IEC 23094-2 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 ISO/IEC 23094-2 including the hypothetical reference decoder (HRD) conformance (based on Annexes C, D, and E). These complementary tests may be performed using other video bitstream verifiers that perform more complete tests than those implemented by the reference software decoder.

ISO/IEC 23094-2 contains several informative recommendations that are not an integral part of that 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 ISO/IEC 23094-2.

A verifier may not necessarily perform all stages of the decoding process specified in ISO/IEC 23094-2 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.

Annex A of this document provides information on the accuracy tolerance of decoding implementations if a different precision than the fixed-point precision specified in ISO/IEC 23094-2 is used.

6.5 Procedure to test decoder conformance

6.5.1 Conformance bitstreams

A bitstream has values of profile_idc and level_idc 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 of ISO/IEC 23094-2:2021 to properly perform the decoding process.

6.5.2 Contents of the bitstream file

The conformance bitstreams are included in this document as an electronic attachment. The following information might be included in a single zipped file for each such bitstream.

— *.bit – bitstream

— *.txt – description

— *.yuv.md5 – MD5 check sum of the complete decoded yuv file

— *.md5 – MD5 check sum of the bitstream file

— *.opl – output picture log

— *.cfg – config file used to generate bitstream with LTM reference software encoder

— *_userdata.bin – Byte file with the content of the user data present within the bitstream

In case a single zipped file contains multiple bitstreams, each bitstream shall be tested separately. The above listed files are available for each such bitstream within the zipped file.

6.5.3 Operation modes for decoder conformance testing

For decoder conformance testing, there are two operation modes:

1) *Operation mode A:* The reference software decoder may be used to separate the test enhancement bitstream from the test bitstream and to generate the decoded base pictures, as specified in Annex B. A decoder shall use as inputs the decoded base pictures and the test enhancement bitstream. In this operation mode, the decoder shall be able to decode the test enhancement bitstream in accordance with ISO/IEC 23094-2:2021, C.1.

2) *Operation mode B:* A decoder shall use as an input the test bitstream as provided. In this operation mode, the decoder shall be able to decode the test bitstream in accordance with ISO/IEC 23094-2:2021, C.1.

A decoder under test may choose either of the two operation modes to verify decoder conformance.

6.5.4 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 ISO/IEC 23094-2:2021, Clause 8 and Annex C.

For output order conformance, it is a requirement that all of the decoded pictures specified for output in ISO/IEC 23094-2:2021, Annex C 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 the values specified in ISO/IEC 23094-2:2021, Clause 8.

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 ISO/IEC 23094-2:2021, Annex C.

The display process, which ordinarily follows the output of the decoding process, is outside the scope of this document.

6.5.5 Recommendations

This clause does not form an integral part of this document.

In addition to the requirements, it is desirable that conforming decoders implement various informative recommendations specified in ISO/IEC 23094-2 that are not an integral part of that International Standard. This clause discusses some of these recommendations.

It is recommended that a conforming decoder be able to resume the decoding process as soon as possible after the loss or corruption of part of a bitstream. In most cases it is possible to resume decoding at the next IDR NALu. It is recommended that a conforming decoder be able to perform concealment for the video packets for which all the coded data has not been received.

6.5.6 Static tests for output order conformance

Static tests of a video decoder require testing of the decoded samples. This clause will explain how this test can be accomplished when the decoded samples at the output of the decoding process are available. It may not be possible to perform this type of test with a production decoder (due to the lack of an appropriate accessible interface in the design at which to perform the test). In that case this test should be performed by the manufacturer during the design and development phase. Static tests are used for testing the decoding process. The test will check that the values of the samples decoded by the decoder under test shall be identical to the values of the samples decoded by the reference decoder. When a hash of the values of the samples of the decoded pictures is attached to the bitstream file, a corresponding hash operation performed on the values of the samples of the decoded pictures produced by the decoder under test shall produce the same results.

6.5.7 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 ISO/IEC 23094-2:2021, Clause 8 and Annex C and to verify that the HRD models (as specified by the CPB and DPB specification in ISO/IEC 23094-2:2021, Annex C) are not violated when the bits of the bitstream are delivered at the proper rate.

The dynamic test is often easier to perform on a complete decoding 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 decoded pictures are correct at the output of the display process. However, since the display process is not within the normative scope of ISO/IEC 23094-2, 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 output order and timing of the decoded pictures shall be correct.

If buffering period and picture timing SEI messages are included in the base 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 base test bitstream.

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

— fixed_pic_rate_flag shall be inferred to be equal to 1.

— low_delay_hrd_flag shall be inferred to be equal to 0.

— cbr_flag shall be inferred to be equal to 0.

— The frame rate of the bitstream shall be inferred to be equal to the frame rate value specified in Table 1, where the bitstream is listed. If this is missing, then a frame rate of either 25 or 30 000 ÷ 1 001 can be inferred.

— time_scale shall be set equal to 90 000 and the value of num_units_in_tick shall be computed based on frame rate.

— The bit rate of the base test bitstream shall be inferred to be equal to the maximum value for the level specified in the specification that the base test bitstream implements.

— CPB and DPB sizes shall be inferred to be equal to the maximum value for the level as stated in the specification of the base.

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 90 000 (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 ÷ 90 000 and subsequent access units are removed at intervals based on the frame distance, i.e., 2 * (90 000 ÷ num_units_in_tick) or the field distance, i.e., (90 000 / 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.

6.5.8 Decoder conformance test of a particular profile and level

In order for a decoder of a particular profile and level to claim output order conformance to ISO/IEC 23094-2 as specified by this document, the decoder shall successfully pass the static test specified in 6.5.6 with all the bitstreams of the normative test suite specified for testing decoders of this particular profile and level combination.

In order for a decoder of a particular profile and level to claim output timing conformance to ISO/IEC 23094-2 as specified by this document, the decoder shall successfully pass both the static test specified in 6.5.6 and the dynamic test specified in 6.5.7 with all the bitstreams of the normative test suite specified for testing decoders of this particular profile and level. Table 1 specifies the normative test suites for each profile and level combination. The test suite for a particular profile and level combination is the list of bitstreams that are marked with an 'X' in the column corresponding to that profile and level combination.

'X' indicates that the bitstream is designed to test both the dynamic and static conformance of the decoder.

The bitstream column specifies the bitstream used for each test.

A decoder that conforms to the profiles at a specific level shall be capable of decoding the specified bitstreams in Table 1.

6.6 Specification of the test bitstreams

6.6.1 General

Some characteristics of each bitstream listed in Table 1 are specified in this clause.

6.6.2 Test bitstreams

6.6.2.1 Test bitstream MP_SET-1

Specification: Streams with sets of coding tools in Main profile.

Functional stage: Test the decoding process of Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the full set of coding tools in Main profile is enabled.

6.6.2.2 Test bitstream TT-1

Specification: Streams with 2x2 directional decomposition transform used in Main profile.

Functional stage: Test the decoding process of the 2x2 directional decomposition transform in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the 2x2 directional decomposition transform is used in Main profile.

6.6.2.3 Test bitstream TT-2

Specification: Streams with 4x4 directional decomposition transform changed to 2x2 directional decomposition transform changed to 4x4 directional decomposition transform used in Main profile.

Functional stage: Test the decoding process with changes of the kernel size of the directional decomposition transform within the bitstream in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the kernel size of the directional decomposition transform is changed within the bitstream in Main profile.

6.6.2.4 Test bitstream UPS-1

Specification: Streams with the nearest sample upsampler kernel used in Main profile.

Functional stage: Test the decoding process of the nearest sample upsampler kernel in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the nearest sample upsampler kernel is used in Main profile.

6.6.2.5 Test bitstream UPS-2

Specification: Streams with the bilinear upsampler kernel used in Main profile.

Functional stage: Test the decoding process of the bilinear upsampler kernel in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the bilinear upsampler kernel is used in Main profile.

6.6.2.6 Test bitstream UPS-3

Specification: Streams with the cubic upsampler kernel used in Main profile.

Functional stage: Test the decoding process of the cubic upsampler kernel in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the cubic upsampler kernel is used in Main profile.

6.6.2.7 Test bitstream UPS-4

Specification: Streams with the modified cubic upsampler kernel used in Main profile.

Functional stage: Test the decoding process of the modified cubic upsampler kernel in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the modified cubic upsampler kernel is used in Main profile.

6.6.2.8 Test bitstream UPS-5

Specification: Streams with the adaptive cubic upsampler kernel used in Main profile.

Functional stage: Test the decoding process of the adaptive cubic upsampler kernel in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the adaptive cubic upsampler kernel is used in Main profile.

6.6.2.9 Test bitstream UPS-6

Specification: Streams with the nearest upsampler kernel changed to the cubic upsampler kernel changed to the modified cubic upsampler kernel used in Main profile.

Functional stage: Test the decoding process with changes of the upsampler kernel within the bitstream in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the upsampler kernel is changed within the bitstream in Main profile.

6.6.2.10 Test bitstream QUANT-1-1

Specification: Streams with the default method of the dequantization offset mode used and the value of the dequantization offset parameter set to 0 in Main profile.

Functional stage: Test the decoding process of the default method of the dequantization offset mode in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the default method of the dequantization offset mode is used and the value of the dequantization offset parameter is set to 0 in Main profile.

6.6.2.11 Test bitstream QUANT-1-2

Specification: Streams with the default method of the dequantization offset mode used and the value of the dequantization offset parameter set to 30 in Main profile.

Functional stage: Test the decoding process of the default method of the dequantization offset mode in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the default method of the dequantization offset mode is used and the value of the dequantization offset parameter is set to 30 in Main profile.

6.6.2.12 Test bitstream QUANT-1-3

Specification: Streams with the default method of the dequantization offset mode used and the value of the dequantization offset parameter set to 64 in Main profile.

Functional stage: Test the decoding process of the default method of the dequantization offset mode in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the default method of the dequantization offset mode is used and the value of the dequantization offset parameter is set to 64 in Main profile.

6.6.2.13 Test bitstream QUANT-1-4

Specification: Streams with the const-offset method of the dequantization offset mode used and the value of the dequantization offset parameter set to 10 in Main profile.

Functional stage: Test the decoding process of the const-offset method of the dequantization offset mode in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the const-offset method of the dequantization offset mode is used and the value of the dequantization offset parameter is set to 10 in Main profile.

6.6.2.14 Test bitstream QUANT-1-5

Specification: Streams with the const-offset method of the dequantization offset mode used and the value of the dequantization offset parameter set to 40 in Main profile.

Functional stage: Test the decoding process of the const-offset method of the dequantization offset mode in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the const-offset method of the dequantization offset mode is used and the value of the dequantization offset parameter is set to 40 in Main profile.

6.6.2.15 Test bitstream QUANT-1-6

Specification: Streams with the const-offset method of the dequantization offset mode used and the value of the dequantization offset parameter set to 72 in Main profile.

Functional stage: Test the decoding process of the const-offset method of the dequantization offset mode in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the const-offset method of the dequantization offset mode is used and the value of the dequantization offset parameter is set to 72 in Main profile.

6.6.2.16 Test bitstream QUANT-2-1

Specification: Streams with the value of the temporal step width modifier set to 10 in Main profile.

Functional stage: Test the decoding process of the temporal step width modifier in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the value of the temporal step width modifier is set to 10 in Main profile.

6.6.2.17 Test bitstream QUANT-2-2

Specification: Streams with the value of the temporal step width modifier set to 20 in Main profile.

Functional stage: Test the decoding process of the temporal step width modifier in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the value of the temporal step width modifier is set to 20 in Main profile.

6.6.2.18 Test bitstream QUANT-2-3

Specification: Streams with the value of the temporal step width modifier set to 30 in Main profile.

Functional stage: Test the decoding process of the temporal step width modifier in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the value of the temporal step width modifier is set to 30 in Main profile.

6.6.2.19 Test bitstream QUANT-3-1

Specification: Streams with the value of the chroma step width multiplier set to 15 in Main profile.

Functional stage: Test the decoding process of the chroma step width multiplier in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the value of the chroma step width multiplier is set to 15 in Main profile.

6.6.2.20 Test bitstream QUANT-3-2

Specification: Streams with the value of the chroma step width multiplier set to 35 in Main profile.

Functional stage: Test the decoding process of the chroma step width multiplier in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the value of the chroma step width multiplier is set to 35 in Main profile.

6.6.2.21 Test bitstream QUANT-3-3

Specification: Streams with the value of the chroma step width multiplier set to 45 in Main profile.

Functional stage: Test the decoding process of the chroma step width multiplier in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the value of the chroma step width multiplier is set to 45 in Main profile.

6.6.2.22 Test bitstream QUANT-4-1

Specification: Streams with the default quantization matrix in Main profile.

Functional stage: Test the decoding process of the default quantization matrix in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the default quantization matrix is used in Main profile.

6.6.2.23 Test bitstream QUANT-4-2

Specification: Streams with the same custom quantization matrix for both sub-layers in Main profile.

Functional stage: Test the decoding process of the same custom quantization matrix for both sub-layers in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the same custom quantization matrix for both sub-layers is used in Main profile.

6.6.2.24 Test bitstream QUANT-4-3

Specification: Streams with a custom quantization matrix for sub-layer 2 in Main profile.

Functional stage: Test the decoding process of a custom quantization matrix for sub-layer 2 in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which a custom quantization matrix for sub-layer 2 is used in Main profile.

6.6.2.25 Test bitstream QUANT-4-4

Specification: Streams with a custom quantization matrix for sub-layer 1 in Main profile.

Functional stage: Test the decoding process of a custom quantization matrix for sub-layer 1 in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which a custom quantization matrix for sub-layer 1 is used in Main profile.

6.6.2.26 Test bitstream QUANT-4-5

Specification: Streams with different custom quantization matrices for both sub-layers in Main profile.

Functional stage: Test the decoding process of different custom quantization matrices for both sub-layers in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which different custom quantization matrices for both sub-layers are used in Main profile.

6.6.2.27 Test bitstream QUANT-5-1

Specification: Streams with the value of the step width of enhancement sub-layer 2 set to 16 in Main profile.

Functional stage: Test the decoding process of the step width of enhancement sub-layer 2 in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the step width of enhancement sub-layer 2 is set to 16 in Main profile.

6.6.2.28 Test bitstream QUANT-5-2

Specification: Streams with the value of the step width of enhancement sub-layer 2 set to 800 in Main profile.

Functional stage: Test the decoding process of the step width of enhancement sub-layer 2 in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the step width of enhancement sub-layer 2 is set to 800 in Main profile.

6.6.2.29 Test bitstream QUANT-5-3

Specification: Streams with the value of the step width of enhancement sub-layer 2 set to 1 000 in Main profile.

Functional stage: Test the decoding process of the step width of enhancement sub-layer 2 in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the step width of enhancement sub-layer 2 is set to 1 000 in Main profile.

6.6.2.30 Test bitstream QUANT-5-4

Specification: Streams with the value of the step width of enhancement sub-layer 2 set to 2 000 in Main profile.

Functional stage: Test the decoding process of the step width of enhancement sub-layer 2 in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the step width of enhancement sub-layer 2 is set to 2 000 in Main profile.

6.6.2.31 Test bitstream QUANT-5-5

Specification: Streams with the value of the step width of enhancement sub-layer 2 set to 3 000 in Main profile.

Functional stage: Test the decoding process of the step width of enhancement sub-layer 2 in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the step width of enhancement sub-layer 2 is set to 3 000 in Main profile.

6.6.2.32 Test bitstream QUANT-6-1

Specification: Streams with, initially, the default method of the dequantization offset mode used and the value of the dequantization offset parameter set to 0, changed to the const-offset method of the dequantization offset mode and the value of the dequantization offset parameter changed to 10 used in Main profile.

Functional stage: Test the decoding process with a change of the dequantization offset mode and dequantization offset parameter in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the dequantization offset mode and the value of the dequantization offset parameter are changed within the bitstream in Main profile.

6.6.2.33 Test bitstream QUANT-6-2

Specification: Streams with, initially, the const-offset method of the dequantization offset mode used and the value of the dequantization offset parameter set to 70, changed to the default method of the dequantization offset mode and the value of the dequantization offset parameter changed to 30 used in Main profile.

Functional stage: Test the decoding process with a change of the dequantization offset mode and dequantization offset parameter in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the dequantization offset mode and the value of the dequantization offset parameter are changed within the bitstream in Main profile.

6.6.2.34 Test bitstream QUANT-6-3

Specification: Streams with the default quantization matrix changed to different custom quantization matrices for both sub-layers changed to a custom quantization matrix for sub-layer 1 only used in Main profile.

Functional stage: Test the decoding process with changes of the quantization matrix mode in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the quantization matrix mode is changed within the bitstream in Main profile.

6.6.2.35 Test bitstream QUANT-6-4

Specification: Streams with the value of the step width of enhancement sub-layer 2 changed from 1 500, to 500, and to 2 500 and the value of the step width of enhancement sub-layer 1 changed from 32 767, to 600, and to 1 600 in Main profile.

Functional stage: Test the decoding process with changes of the values of the step widths of enhancement sub-layers 1 and 2 in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the values of the step widths of enhancement sub-layers 1 and 2 are changed within the bitstream in Main profile.

6.6.2.36 Test bitstream ZMTP-1

Specification: Streams with temporal prediction disabled used in Main profile.

Functional stage: Test the decoding process of the temporal prediction disabled in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the temporal prediction is disabled in Main profile.

6.6.2.37 Test bitstream ZMTP-2

Specification: Streams with temporal prediction enabled used in Main profile.

Functional stage: Test the decoding process of the temporal prediction in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the temporal prediction is enabled in Main profile.

6.6.2.38 Test bitstream ZMTP-3

Specification: Streams with temporal prediction initially disabled, then enabled and finally disabled used in Main profile.

Functional stage: Test the decoding process of the temporal prediction being switched on and off in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the temporal prediction is being switched on and off within the bitstream in Main profile.

6.6.2.39 Test bitstream RS-1

Specification: Streams with temporal tile prediction disabled used in Main profile.

Functional stage: Test the decoding process of the temporal tile prediction disabled in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the temporal tile prediction is disabled in Main profile.

6.6.2.40 Test bitstream RS-2

Specification: Streams with temporal tile prediction enabled used in Main profile.

Functional stage: Test the decoding process of the temporal tile prediction in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the temporal tile prediction is enabled in Main profile.

6.6.2.41 Test bitstream PR-1

Specification: Streams with predicted residuals process disabled used in Main profile.

Functional stage: Test the decoding process of the predicted residuals process disabled in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the predicted residuals process is disabled in Main profile.

6.6.2.42 Test bitstream PR-2

Specification: Streams with predicted residuals process enabled used in Main profile.

Functional stage: Test the decoding process of the predicted residuals process in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the predicted residuals process is enabled in Main profile.

6.6.2.43 Test bitstream L1D-1

Specification: Streams with L-1 filter disabled used in Main profile.

Functional stage: Test the decoding process of the L-1 filter disabled in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the L-1 filter is disabled in Main profile.

6.6.2.44 Test bitstream L1D-2

Specification: Streams with L-1 filter enabled used in Main profile.

Functional stage: Test the decoding process of the L-1 filter in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the L-1 filter is enabled in Main profile.

6.6.2.45 Test bitstream TD-1

Specification: Streams with no tiling used in Main profile.

Functional stage: Test the decoding process without tiling in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which no tiling is used in Main profile.

6.6.2.46 Test bitstream TD-2

Specification: Streams with a tiling size of 512x256 used in Main profile.

Functional stage: Test the decoding process of tiling with default sizes in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which a tiling size of 512x256 is used in Main profile.

6.6.2.47 Test bitstream TD-3

Specification: Streams with a tiling size of 1 024x512 used in Main profile.

Functional stage: Test the decoding process of tiling with default sizes in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which a tiling size of 1 024x512 is used in Main profile.

6.6.2.48 Test bitstream TD-4

Specification: Streams with a tiling size of 256x256 used in Main profile.

Functional stage: Test the decoding process of tiling with custom sizes in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which a tiling size of 256x256 is used in Main profile.

6.6.2.49 Test bitstream TD-5

Specification: Streams with a tiling size of 1 024x1 024 used in Main profile.

Functional stage: Test the decoding process of tiling with custom sizes in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which a tiling size of 1 024x1 024 is used in Main profile.

6.6.2.50 Test bitstream SD-1

Specification: Streams with no scaling for both levels 1 and 2 used in Main profile.

Functional stage: Test the decoding process the scaling modes for levels 1 and 2 in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which no scaling is used for both levels in Main profile.

6.6.2.51 Test bitstream SD-2

Specification: Streams with no scaling used for level 1 and two-dimensional (2D) 2:1 scaling across both dimensions used for level 2 in Main profile.

Functional stage: Test the decoding process the scaling modes for levels 1 and 2 in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which no scaling is used for level 1 and two-dimensional (2D) 2:1 scaling across both dimensions is used for level 2 in Main profile.

6.6.2.52 Test bitstream SD-3

Specification: Streams with one-dimensional (1D) 2:1 scaling only across the horizontal dimension used for both levels 1 and 2 in Main profile.

Functional stage: Test the decoding process the scaling modes for levels 1 and 2 in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which one-dimensional (1D) 2:1 scaling only across the horizontal dimension is used for both levels 1 and 2 in Main profile.

6.6.2.53 Test bitstream SD-4

Specification: Streams with one-dimensional (1D) 2:1 scaling only across the horizontal dimension used for level 1 and two-dimensional (2D) 2:1 scaling across both dimensions used for level 2 in Main profile.

Functional stage: Test the decoding process the scaling modes for levels 1 and 2 in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which one-dimensional (1D) 2:1 scaling only across the horizontal dimension is used for level 1 and two-dimensional (2D) 2:1 scaling across both dimensions is used for level 2 in Main profile.

6.6.2.54 Test bitstream SD-5

Specification: Streams with two-dimensional (1D) 2:1 scaling across both dimensions used for both levels 1 and 2 in Main profile.

Functional stage: Test the decoding process the scaling modes for levels 1 and 2 in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which two-dimensional (2D) 2:1 scaling across both dimensions is used for both levels 1 and 2 in Main profile.

6.6.2.55 Test bitstream RBD-1

Specification: Streams with a base bit depth of 8 bits and an enhancement bit depth of at least 10 bits and level1_depth_flag disabled in Main profile.

Functional stage: Test the decoding process with different base and enhancement bit depths in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which a base bit depth of 8 bits and an enhancement bit depth of at least 10 bits are used and level1_depth_flag is disabled in Main profile.

6.6.2.56 Test bitstream RBD-2

Specification: Streams with a base bit depth of 8 bits and an enhancement bit depth of at least 10 bits and level1_depth_flag enabled in Main profile.

Functional stage: Test the decoding process with different base and enhancement bit depths in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which a base bit depth of 8 bits and an enhancement bit depth of at least 10 bits are used and level1_depth_flag is enabled in Main profile.

6.6.2.57 Test bitstream RBD-3

Specification: Streams with a base bit depth of 10 bits and an enhancement bit depth of at least 12 bits and level1_depth_flag disabled in Main profile.

Functional stage: Test the decoding process with different base and enhancement bit depths in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which a base bit depth of 10 bits and an enhancement bit depth of at least 12 bits are used and level1_depth_flag is disabled in Main profile.

6.6.2.58 Test bitstream RBD-4

Specification: Streams with a base bit depth of 10 bits and an enhancement bit depth of at least 12 bits and level1_depth_flag enabled in Main profile.

Functional stage: Test the decoding process with different base and enhancement bit depths in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which a base bit depth of 10 bits and an enhancement bit depth of at least 12 bits are used and level1_depth_flag is enabled in Main profile.

6.6.2.59 Test bitstream ECS-1

Specification: Streams with processing of all planes (luma and chroma) in Main profile.

Functional stage: Test the decoding process with processing of all planes (luma and chroma) in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which all planes (luma and chroma) are processed in Main profile.

6.6.2.60 Test bitstream ECS-2

Specification: Streams with processing without data in enhancement layer in Main profile.

Functional stage: Test the decoding process without data in the enhancement layer in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which no data is signaled in the enhancement layer in Main profile.

6.6.2.61 Test bitstream HDUD-1

Specification: Streams with 2-bits of user data enabled and residuals sub-layer 1 disabled in Main profile.

Functional stage: Test the decoding and extraction process of user data in Main profile.

Purpose: Check that the decoder can properly decode and extract the user data of bitstreams in which 2-bits of user data are used and residuals sub-layer 1 is disabled in Main profile.

6.6.2.62 Test bitstream HDUD-2

Specification: Streams with 2-bits of user data enabled and residuals sub-layer 1 enabled in Main profile.

Functional stage: Test the decoding and extraction process of user data in Main profile.

Purpose: Check that the decoder can properly decode and extract the user data of bitstreams in which 2-bits of user data are used and residuals sub-layer 1 is enabled in Main profile.

6.6.2.63 Test bitstream HDUD-3

Specification: Streams with 6-bits of user data enabled and residuals sub-layer 1 disabled in Main profile.

Functional stage: Test the decoding and extraction process of user data in Main profile.

Purpose: Check that the decoder can properly decode and extract the user data of bitstreams in which 6-bits of user data are used and residuals sub-layer 1 is disabled in Main profile.

6.6.2.64 Test bitstream HDUD-4

Specification: Streams with 6-bits of user data enabled and residuals sub-layer 1 enabled in Main profile.

Functional stage: Test the decoding and extraction process of user data in Main profile.

Purpose: Check that the decoder can properly decode and extract the user data of bitstreams in which 6-bits of user data are used and residuals sub-layer 1 is enabled in Main profile.

6.6.2.65 Test bitstream CONF-1

Specification: Streams with conformance window cropping on the left in Main profile.

Functional stage: Test the decoding process of the conformance cropping window of the width in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the conformance window cropping on the left is used in Main profile.

6.6.2.66 Test bitstream CONF-2

Specification: Streams with conformance window cropping on the right in Main profile.

Functional stage: Test the decoding process of the conformance cropping window of the width in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the conformance window cropping on the right is used in Main profile.

6.6.2.67 Test bitstream CONF-3

Specification: Streams with conformance window cropping on the top in Main profile.

Functional stage: Test the decoding process of the conformance cropping window of the height in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the conformance window cropping on the top is used in Main profile.

6.6.2.68 Test bitstream CONF-4

Specification: Streams with conformance window cropping on the bottom in Main profile.

Functional stage: Test the decoding process of the conformance cropping window of the height in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the conformance window cropping on the bottom is used in Main profile.

6.6.2.69 Test bitstream CST-1

Specification: Streams with monochrome colour format in Main profile.

Functional stage: Test the decoding process of the colour format in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the monochrome colour format is used in Main profile.

6.6.2.70 Test bitstream CST-2

Specification: Streams with 4:2:0 colour format in Main profile.

Functional stage: Test the decoding process of the colour format in Main profile.

Purpose: Check that the decoder can properly decode bitstreams in which the 4:2:0 colour format is used in Main profile.

6.7 Normative test suites for ISO/IEC 23094-2

Legend:

X – Bitstream is for static and dynamic test

Table 1 — Bitstreams for Main profile

Categories	Sub category (if applicable)	Description	Bitstream	File name	Main Profile	Level	Frame rate (Frame/sec)
Tool set		Tool set of Main profile	MP_SET-1	MP_SET-1_v-nova_v03	X	2.1 & 3.1	50
Transform type	Transform type	2x2 directional decomposition transform	TT-1	TT-1_v-nova_v03	X	2.1 & 3.1	50
	Transform type change	Transform kernel size changed from 4x4, to 2x2, and to 4x4	TT-2	TT-2_v-nova_v02	X	2.1 & 3.1	50
Upsampling	Upsampling kernel	Nearest	UPS-1	UPS-1_v-nova_v03	X	2.1 & 3.1	50
		Bilinear	UPS-2	UPS-2_v-nova_v03	X	2.1 & 3.1	50
		Cubic	UPS-3	UPS-3_v-nova_v03	X	2.1 & 3.1	50
		Modified cubic	UPS-4	UPS-4_v-nova_v03	X	2.1 & 3.1	50
		Adaptive cubic (coefficients: 3 048, 5 144, 4 144, 1 048)	UPS-5	UPS-5_v-nova_v03	X	2.1 & 3.1	50
	Upsampling kernel change	Nearest changed to cubic changed to modified cubic	UPS-6	UPS-6_v-nova_v02	X	2.1 & 3.1	50
Quantization	Dequantization offset mode	Default method, value: 0	QUANT-1-1	QUANT-1-1_v-nova_v03	X	2.1 & 3.1	50
		Default method, value: 30	QUANT-1-2	QUANT-1-2_v-nova_v03	X	2.1 & 3.1	50
		Default method, value: 64	QUANT-1-3	QUANT-1-3_v-nova_v03	X	2.1 & 3.1	50
		Const-offset method, value:10	QUANT-1-4	QUANT-1-4_v-nova_v03	X	2.1 & 3.1	50
		Const-offset method, value: 40	QUANT-1-5	QUANT-1-5_v-nova_v03	X	2.1 & 3.1	50
		Const-offset method, value: 72	QUANT-1-6	QUANT-1-6_v-nova_v03	X	2.1 & 3.1	50
	Temporal step width modifier	Temporal step width modifier, value: 10	QUANT-2-1	QUANT-2-1_v-nova_v03	X	2.1 & 3.1	50
		Temporal step width modifier, value: 20	QUANT-2-2	QUANT-2-2_v-nova_v03	X	2.1 & 3.1	50
		Temporal step width modifier, value: 30	QUANT-2-3	QUANT-2-3_v-nova_v03	X	2.1 & 3.1	50
	Chroma step width multiplier	Chroma step width multiplier, value: 15	QUANT-3-1	QUANT-3-1_v-nova_v05	X	2.1 & 3.1	50
		Chroma step width multiplier, value: 35	QUANT-3-2	QUANT-3-2_v-nova_v05	X	2.1 & 3.1	50
		Chroma step width multiplier, value: 45	QUANT-3-3	QUANT-3-3_v-nova_v05	X	2.1 & 3.1	50
	Quantization matrix	Mode 1: both default	QUANT-4-1	QUANT-4-1_v-nova_v03	X	2.1 & 3.1	50
		Mode 2: same and custom	QUANT-4-2	QUANT-4-2_v-nova_v03	X	2.1 & 3.1	50
		Mode 3: sub-layer 2 custom	QUANT-4-3	QUANT-4-3_v-nova_v03	X	2.1 & 3.1	50
		Mode 4: sub-layer 1 custom	QUANT-4-4	QUANT-4-4_v-nova_v03	X	2.1 & 3.1	50
		Mode 5: different and custom	QUANT-4-5	QUANT-4-5_v-nova_v03	X	2.1 & 3.1	50

Table 1 *(continued)*

Categories	Sub category (if applicable)	Description	Bitstream	File name	Main Profile	Level	Frame rate (Frame/sec)
	Step width sub-layer 2	Step width sub-layer 2, value: 16	QUANT-5-1	QUANT-5-1_v-nova_v03	X	2.1 & 3.1	50
		Step width sub-layer 2, value: 800	QUANT-5-2	QUANT-5-2_v-nova_v03	X	2.1 & 3.1	50
		Step width sub-layer 2, value: 1 000	QUANT-5-3	QUANT-5-3_v-nova_v03	X	2.1 & 3.1	50
		Step width sub-layer 2, value: 2 000	QUANT-5-4	QUANT-5-4_v-nova_v03	X	2.1 & 3.1	50
		Step width sub-layer 2, value: 3 000	QUANT-5-5	QUANT-5-5_v-nova_v03	X	2.1 & 3.1	50
	Quantization configuration change	Change of dequantization offset mode & value	QUANT-6-1	QUANT-6-1_v-nova_v02	X	2.1 & 3.1	50
		Change of dequantization offset mode & value	QUANT-6-2	QUANT-6-2_v-nova_v02	X	2.1 & 3.1	50
		Change of quantization matrix	QUANT-6-3	QUANT-6-3_v-nova_v02	X	2.1 & 3.1	50
		Change of sub-layers 1 and 2 step widths	QUANT-6-4	QUANT-6-4_v-nova_v02	X	2.1 & 3.1	50
Temporal prediction	Temporal prediction	Temporal prediction Off	ZMTP-1	ZMTP-1_v-nova_v03	X	2.1 & 3.1	50
		Temporal prediction On	ZMTP-2	ZMTP-2_v-nova_v03	X	2.1 & 3.1	50
	Temporal prediction change	Temporal prediction Off, switched On, switched Off	ZMTP-3	ZMTP-3_v-nova_v02	X	2.1 & 3.1	50
Temporal tile prediction		Temporal tile prediction Off	RS-1	RS-1_v-nova_v03	X	2.1 & 3.1	50
		Temporal tile prediction On	RS-2	RS-2_v-nova_v03	X	2.1 & 3.1	50
Predicted residuals		Predicted residuals Off	PR-1	PR-1_v-nova_v03	X	2.1 & 3.1	50
		Predicted residuals On	PR-2	PR-2_v-nova_v03	X	2.1 & 3.1	50
L-1 filter		L-1 filter Off	L1D-1	L1D-1_v-nova_v03	X	2.1 & 3.1	50
		L-1 filter On	L1D-2	L1D-2_v-nova_v03	X	2.1 & 3.1	50
Tiling	Off	Tiling Off	TD-1	TD-1_v-nova_v03	X	2.1 & 3.1	50
	Default size	Tiling size 512x256	TD-2	TD-2_v-nova_v03	X	2.1 & 3.1	50
		Tiling size 1 024x512	TD-3	TD-3_v-nova_v03	X	2.1 & 3.1	50
	Custom size	Tiling size 256x256	TD-4	TD-4_v-nova_v03	X	2.1 & 3.1	50
		Tiling size 1 024x1 024	TD-5	TD-5_v-nova_v03	X	2.1 & 3.1	50

Table 1 *(continued)*

Categories	Sub category (if applicable)	Description	Bitstream	File name	Main Profile	Level	Frame rate (Frame/sec)
Scaling mode		Level 1: 0 (none) Level 2: 0 (none)	SD-1	SD-1_v-nova_v03	X	2.1 & 3.1	50
		Level 1: 0 (none) Level 2: 2 (2D)	SD-2	SD-2_v-nova_v03	X	2.1 & 3.1	50
		Level 1: 1 (1D) Level 2: 1 (1D)	SD-3	SD-3_v-nova_v03	X	2.1 & 3.1	50
		Level 1: 1 (1D) Level 2: 2 (2D)	SD-4	SD-4_v-nova_v03	X	2.1 & 3.1	50
		Level 1: 2 (2D) Level 2: 2 (2D)	SD-5	SD-5_v-nova_v03	X	2.1 & 3.1	50
Relative bit depth	Base 8 bit, enhancement 10 bit or higher	level1_depth_flag Off	RBD-1	RBD-1_v-nova_v03	X	2.1 & 3.1	50
		level1_depth_flag On	RBD-2	RBD-2_v-nova_v03	X	2.1 & 3.1	50
	Base 10 bit, enhancement 12 bit or higher	level1_depth_flag Off	RBD-3	RBD-3_v-nova_v03	X	2.1 & 3.1	50
		level1_depth_flag On	RBD-4	RBD-4_v-nova_v03	X	2.1 & 3.1	50
Enhancement		Process luma & chroma planes (planes_type = 1)	ECS-1	ECS-1_v-nova_v05	X	2.1 & 3.1	50
		No enhancement data	ECS-2	ECS-2_v-nova_v02	X	2.1 & 3.1	50
User data	2-bits	User data 2-bits, residuals sub-layer 1 Off	HDUD-1	HDUD-1_v-nova_v03	X	2.1 & 3.1	50
		User data 2-bits, residuals sub-layer 1 On	HDUD-2	HDUD-2_v-nova_v03	X	2.1 & 3.1	50
	6-bits	User data 6-bits, residuals sub-layer 1 Off	HDUD-3	HDUD-3_v-nova_v03	X	2.1 & 3.1	50
		User data 2-bits, residuals sub-layer 1 On	HDUD-4	HDUD-4_v-nova_v03	X	2.1 & 3.1	50
Conformance window	Cropping of width	conf_win_left_offset	CONF-1	CONF-1_v-nova_v03	X	2.1 & 3.1	50
		conf_win_right_offset	CONF-2	CONF-2_v-nova_v03	X	2.1 & 3.1	50
	Cropping of height	conf_win_top_offset	CONF-3	CONF-3_v-nova_v03	X	2.1 & 3.1	50
		conf_win_bottom_offset	CONF-4	CONF-4_v-nova_v03	X	2.1 & 3.1	50
Chroma sampling type		Color format: monochrome	CST-1	CST-1_v-nova_v03	X	2.1 & 3.1	50
		Color format: 4:2:0	CST-2	CST-2_v-nova_v03	X	2.1 & 3.1	50

7 Reference software description

7.1 General

The LCEVC Test Model (LTM) is implemented in C++ programming language in accordance with ISO/IEC 14882.