International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

H.265.1

(10/2016)

SERIES H: AUDIOVISUAL AND MULTIMEDIA SYSTEMS

Infrastructure of audiovisual services – Coding of moving video

---

Conformance specification for ITU-T H.265 high efficiency video coding

Recommendation ITU-T H.265.1

ITU-T

ITU-T H-SERIES RECOMMENDATIONS

AUDIOVISUAL AND MULTIMEDIA SYSTEMS

CHARACTERISTICS OF VISUAL TELEPHONE SYSTEMS	H.100–H.199
INFRASTRUCTURE OF AUDIOVISUAL SERVICES	
General	H.200–H.219
Transmission multiplexing and synchronization	H.220–H.229
Systems aspects	H.230–H.239
Communication procedures	H.240–H.259
Coding of moving video	**H.260–H.279**
Related systems aspects	H.280–H.299
Systems and terminal equipment for audiovisual services	H.300–H.349
Directory services architecture for audiovisual and multimedia services	H.350–H.359
Quality of service architecture for audiovisual and multimedia services	H.360–H.369
Telepresence	H.420–H.429
Supplementary services for multimedia	H.450–H.499
MOBILITY AND COLLABORATION PROCEDURES	
Overview of Mobility and Collaboration, definitions, protocols and procedures	H.500–H.509
Mobility for H-Series multimedia systems and services	H.510–H.519
Mobile multimedia collaboration applications and services	H.520–H.529
Security for mobile multimedia systems and services	H.530–H.539
Security for mobile multimedia collaboration applications and services	H.540–H.549
Mobility interworking procedures	H.550–H.559
Mobile multimedia collaboration inter-working procedures	H.560–H.569
BROADBAND, TRIPLE-PLAY AND ADVANCED MULTIMEDIA SERVICES	
Broadband multimedia services over VDSL	H.610–H.619
Advanced multimedia services and applications	H.620–H.629
Ubiquitous sensor network applications and Internet of Things	H.640–H.649
IPTV MULTIMEDIA SERVICES AND APPLICATIONS FOR IPTV	
General aspects	H.700–H.719
IPTV terminal devices	H.720–H.729
IPTV middleware	H.730–H.739
IPTV application event handling	H.740–H.749
IPTV metadata	H.750–H.759
IPTV multimedia application frameworks	H.760–H.769
IPTV service discovery up to consumption	H.770–H.779
Digital Signage	H.780–H.789
E-HEALTH MULTIMEDIA SERVICES AND APPLICATIONS	
Personal health systems	H.810–H.819
Interoperability compliance testing of personal health systems (HRN, PAN, LAN, TAN and WAN)	H.820–H.859
Multimedia e-health data exchange services	H.860–H.869

For further details, please refer to the list of ITU-T Recommendations.

Recommendation ITU-T H.265.1

Conformance specification for ITU-T H.265 high efficiency video coding

Summary

Recommendation ITU-T H.265.1 "Conformance specification for ITU-T H.265 high efficiency video coding" specifies tests for (non-exhaustive) testing to verify whether bitstreams and decoders meet the normative requirements specified in Rec. ITU-T H.265 | ISO/IEC 23008-2. The bitstreams provided in this Recommendation correspond to the 04-2013 version of Rec. ITU-T H.265 | ISO/IEC 23008-2.

The second version of this specification added tests for the format range and high throughput extensions (RExt), scalable extensions (SHVC), multi-view extensions (MV-HEVC), and 3D extensions (3D-HEVC), and includes some additional tests and minor corrections to the tests specified for the prior profiles.

This Recommendation was developed jointly with ISO/IEC JTC 1/SC 29/WG 11 (MPEG) and corresponds in a technically aligned manner to ISO/IEC 23008-8.

The conformance bitstreams needed for this Recommendation are available at the following link: http://handle.itu.int/11.1002/2000/12297.

History

Edition	Recommendation	Approval	Study Group	Unique ID*
1.0	ITU-T H.265.1	2014-10-14	16	11.1002/1000/12297
2.0	ITU-T H.265.1	2016-10-29	16	11.1002/1000/12906

Keywords

Conformance bitstream, conformance specification, high efficiency video coding.

* To access the Recommendation, type the URL http://handle.itu.int/ in the address field of your web browser, followed by the Recommendation's unique ID. For example, http://handle.itu.int/11.1002/1000/11830-en.

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure, e.g., interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had not received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database at http://www.itu.int/ITU-T/ipr/.

CONTENTS

Recommendation ITU-T H.265.1

Conformance specification for ITU-T H.265 high efficiency video coding

1 Scope

This Recommendation | International Standard[1] specifies a set of tests and procedures designed to indicate whether encoders or decoders meet the normative requirements specified in Rec. ITU-T H.265 | ISO/IEC 23008-2.

2 Normative references

2.1 General

The following Recommendations and International Standards contain provisions which, through reference in this text, constitute provisions of this Recommendation | International Standard. At the time of publication, the editions indicated were valid. All Recommendations and Standards are subject to revision, and parties to agreements based on this Recommendation | International Standard are encouraged to investigate the possibility of applying the most recent edition of the Recommendations and Standards listed below. Members of IEC and ISO maintain registers of currently valid International Standards. The Telecommunication Standardization Bureau of the ITU maintains a list of currently valid ITU-T Recommendations.

2.2 Identical Recommendations | International Standards

– None.

2.3 Paired Recommendations | International Standards equivalent in technical content

– Recommendation ITU-T H.265 (in force), *High efficiency video coding*.

– ISO/IEC 23008-2: in force, *Information technology – High efficiency video coding and media delivery in heterogeneous environment – Part 2: High Efficiency Video Coding*.

– Recommendation ITU-T H.265.2 (in force), *High efficiency coding reference software*.

– ISO/IEC 23008-5: in force, *Information technology – High efficiency video coding and media delivery in heterogeneous environment – Part 2: High Efficiency Video Coding Reference Software*.

2.4 Additional references

– None.

3 Definitions

For the purposes of this Recommendation | International Standard, the terms, definitions, abbreviations and symbols specified in Rec. ITU-T H.265 | ISO/IEC 23008-2 (particularly in clause 3) apply. The following terms are further clarified for purposes herein as follows.

3.1 **bitstream**: A Rec. ITU-T H.265 | ISO/IEC 23008-2 video bitstream.

3.2 **decoder**: A Rec. ITU-T H.265 | ISO/IEC 23008-2 video decoder, i.e., an embodiment of the decoding process specified by Rec. ITU-T H.265 | ISO/IEC 23008-2. The decoder does not include the display process, which is outside the scope of this Recommendation | International Standard.

3.3 **encoder**: An embodiment of a process, not specified in this Recommendation | International Standard (except in regard to identification of the reference software encoder), that produces a bitstream.

3.4 **reference software decoder**: The software decoder provided in Rec. ITU-T H.265.2 | ISO/IEC 23008-5.

3.5 **reference software encoder**: The software encoder provided in Rec. ITU-T H.265.2 | ISO/IEC 23008-5.

---

[1] This Recommendation | International Standard includes an electronic attachment containing the conformance bitstreams identified within the text. The conformance bitstreams needed for this Recommendation are available at the following link:http://handle.itu.int/11.1002/2000/12297. The bitstreams can also be downloaded from the ITU-T Test Signal Database.

4 Abbreviations and acronyms

For the purposes of this Recommendation | International Standard, relevant abbreviations and acronyms are specified in clause 4 of Rec. ITU-T H.265 | ISO/IEC 23008-2.

5 Conventions

For the purposes of this Recommendation | International Standard, relevant conventions are specified in clause 5 of Rec. ITU-T H.265 | ISO/IEC 23008-2.

6 Conformance testing for ITU-T H.265 | ISO/IEC 23008-2

6.1 Introduction

The following clauses 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 annex to this Recommendation | International Standard and the reference software decoder specified in Rec. ITU-T H.265.2 | ISO/IEC 23008-5.

6.2 Bitstream conformance

Bitstream conformance for Rec. ITU-T H.265 | ISO/IEC 23008-2 is specified by clause C.4 of Rec. ITU-T H.265 | ISO/IEC 23008-2.

6.3 Decoder conformance

Decoder conformance for Rec. ITU-T H.265 | ISO/IEC 23008-2 is specified by clause C.5 of Rec. ITU-T H.265 | ISO/IEC 23008-2.

6.4 Procedure to test bitstreams

A bitstream that claims conformance with Rec. ITU-T H.265 | ISO/IEC 23008-2 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 Rec. ITU-T H.265 | ISO/IEC 23008-2.

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 Rec. ITU-T H.265 | ISO/IEC 23008-2 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 Rec. ITU-T H.265 | ISO/IEC 23008-2 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.

Rec. ITU-T H.265 | ISO/IEC 23008-2 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 Rec. ITU-T H.265 | ISO/IEC 23008-2.

A verifier may not necessarily perform all stages of the decoding process specified in Rec. ITU-T H.265 | ISO/IEC 23008-2 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.

6.5 Procedure to test decoder conformance

6.5.1 Conformance bitstreams

A bitstream has values of general_profile_idc, general_tier_flag, and general_level_idc corresponding to a set of specified constraints on a bitstream for which a decoder conforming to a specified profile, tier, and level is required in Annex A of Rec. ITU-T H.265 | ISO/IEC 23008-2 to properly perform the decoding process.

6.5.2 Contents of the bitstream file

The conformance bitstreams are included in this Recommendation | International Standard as an electronic attachment. The following information is included in a single zipped file for each such bitstream.

- bitstream;
- decoded pictures or hashes of decoded pictures (may not be present);
- short description of the bitstream;
- trace file (results while decoding the bitstream, in ASCII format).

In cases where the decoded pictures or hashes of decoded pictures are not available, the reference software decoder shall be used to generate the necessary reference decoded pictures from the bitstream.

6.5.3 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 clause 8 and Annex C of Rec. ITU-T H.265 | ISO/IEC 23008-2.

For output order conformance, it is a requirement that all of the decoded pictures specified for output in Annex C of Rec. ITU-T H.265 | ISO/IEC 23008-2 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 of Rec. ITU-T H.265 | ISO/IEC 23008-2.

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 of Rec. ITU-T H.265 | ISO/IEC 23008-2.

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

6.5.4 Recommendations (informative)

This clause does not form an integral part of this Recommendation | International Standard.

In addition to the requirements, it is desirable that conforming decoders implement various informative recommendations specified in Rec. ITU-T H.265 | ISO/IEC 23008-2 that are not an integral part of that Recommendation | 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 start code or slice header. It is recommended that a conforming decoder be able to perform concealment for the coding tree blocks or video packets for which all the coded data has not been received.

6.5.5 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.

6.5.6 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 clause 8 and Annex C of Rec. ITU-T H.265 | ISO/IEC 23008-2, and to verify that the HRD models (as specified by the CPB and DPB specification in Annex C of Rec. ITU-T H.265 | ISO/IEC 23008-2) 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 Rec. ITU-T H.265 | ISO/IEC 23008-2, 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 test bitstream, HRD conformance shall be verified using the values of nal_initial_cpb_removal_delay, nal_initial_cpb_removal_offset, au_cpb_removal_delay_minus1 and pic_dpb_output_delay that are included in the bitstream.

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

– fixed_pic_rate_within_cvs_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 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.
– vui_time_scale shall be set equal to 90 000 and the value of vui_num_units_in_tick shall be computed based on frame rate.
– The bit rate of the bitstream shall be inferred to be equal to the maximum value for the level specified in Table A.1 in Rec. ITU-T H.265 | ISO/IEC 23008-2.
– CPB and DPB sizes shall be inferred to be equal to the maximum value for the level specified in Table A.1 in Rec. ITU-T H.265 | ISO/IEC 23008-2.

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 nal_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 nal_initial_cpb_removal_offset shall be inferred to be equal to zero.
– The first access unit is removed at time t = nal_initial_cpb_removal_delay ÷ 90000 and subsequent access units are removed at intervals based on the frame distance, i.e., (90000 ÷ vui_num_units_in_tick) or the field distance, i.e., (90000 ÷ num_units_in_tick).
– Using these inferences, the CPB will not overflow or underflow and the DPB will not overflow.

6.5.7 Decoder conformance test of a particular profile, tier, and level

In order for a decoder of a particular profile, tier, and level to claim output order conformance to Rec. ITU-T H.265 | ISO/IEC 23008-2 as specified by this Recommendation | International Standard, the decoder shall successfully pass the static test specified in clause 6.5.5 with all the bitstreams of the normative test suite specified for testing decoders of this particular profile, tier, and level combination.

In order for a decoder of a particular profile, tier, and level to claim output timing conformance to Rec. ITU-T H.265 | ISO/IEC 23008-2 as specified by this Recommendation | International Standard, the decoder shall successfully pass both the static test specified in clause 6.5.5 and the dynamic test specified in clause 6.5.6 with all the bitstreams of the normative test suite specified for testing decoders of this particular profile, tier, and level. Tables 1 through 6 specify the normative test suites for each profile, tier, and level combination. The test suite for a particular profile, tier, and level combination is the list of bitstreams that are marked with an 'X' in the column corresponding to that profile, tier, and level combination. In the column 'Main tier', 'X' indicates that the bitstream is for the Main tier. A decoder conforming to the Main tier shall be capable of decoding the specified bitstreams, among the testing profile-level combination, indicated by 'X' in the 'Main tier' column in the associated table. A decoder conforming to the High tier shall be capable of decoding all the specified bitstreams, among the testing profile-level combinations, in the associated table.

'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 Main profile, Main Still Picture profile, or Main 10 profile at a specific level shall be capable of decoding the specified bitstreams in Table 1. A decoder that conforms to the Multiview Main profile at specific level shall be capable of decoding the specified bitstreams in Table 2. In addition to the bitstreams defined in Table 3, a decoder that conforms to the Multiview Main profile shall be capable of decoding the Main profile bitstreams specified in Table 1.

A decoder that conforms to the 3D Main profile (as specified in subclause I.11 of Rec. ITU-T H.265 | ISO/IEC 23008-2) at specific level shall be capable of decoding the specified bitstreams in Table 3. In addition to the bitstreams defined in Table 3, a decoder that conforms to the 3D Main profile shall be capable of decoding the Multiview Main profile bitstreams specified in Table 2.

A decoder that conforms to the Monochrome, Monochrome 12, Monochrome 16, Main 12, Main 4:2:2 10, Main 4:2:2 12, Main 4:4:4, Main 4:4:4 10, Main 4:4:4 12, Main Intra, Main 10 Intra, Main 12 Intra, Main 4:2:2 10 Intra, Main 4:2:2 12 Intra, Main 4:4:4 Intra, Main 4:4:4 10 Intra, Main 4:4:4 12 Intra, Main 4:4:4 16 Intra, Main 4:4:4 Still Picture, or Main 4:4:4 16 Still Picture profile (as specified in subclause A.3.5 of Rec. ITU-T H.265 | ISO/IEC 23008-2), which are collectively referred to as the format range extensions profiles, shall be capable of decoding the specified bitstreams in Table 4. A decoder that conforms to some format range extensions profiles is also required to be capable of decoding bitstreams that conform to other particular profiles. Thus, in addition to the specified bitstreams in Table 4, a decoder that conforms to a format range extensions profile shall also be capable of decoding the bitstreams specified in Table 1 that conform to the decoding requirements specified for the format range extensions profile in subclause A.3.5 of Rec. ITU-T H.265 | ISO/IEC 23008-2.

A decoder that conforms to the High Throughput 4:4:4 16 Intra profile (as specified in subclause A.3.6 of Rec. ITU-T H.265 | ISO/IEC 23008-2) at specific level shall be capable of decoding the specified bitstreams in Table 4.

A decoder that conforms to a list of profile, tier, level, INBLD capability quadruplets such that one of the quadruplets corresponds to a profile indication of Scalable Main or Scalable Main 10 profile (as specified in sub-clause H.11.1.1 of Rec. ITU-T H.265 | ISO/IEC 23008-2) at specific level, shall be capable of decoding the specified bitstreams in Table 5. A decoder that conforms to a list of profile, tier, level, INBLD capability quadruplets such that one of the quadruplets corresponds to a profile indication of Scalable Main or Scalable Main 10 profile at specific level shall also be capable of decoding the specified bitstreams in Table 1.

A decoder that conforms to a list of profile, tier, level, INBLD capability quadruplets such that one of the quadruplets corresponds to a profile indication of Scalable Monochrome, Scalable Monochrome 12, Scalable Monochrome 16 or Scalable Main 4:4:4 (as specified in sub-clause H.11.1.2 of Rec. ITU-T H.265 | ISO/IEC 23008-2, collectively referred to as scalable format range extension profiles) at specific level, shall be capable of decoding the specified bitstreams in Table 6. A decoder that conforms to a list of profile, tier, level, INBLD capability quadruplets such that one of the quadruplets corresponds to a scalable format range extension profile is also required to be capable of decoding bitstreams that conform to other particular profiles. Thus, in addition to the specified bitstreams in Table 6, a decoder that conforms to a list of profile, tier, level, INBLD capability quadruplets such that one of the quadruplets corresponds to a scalable format range extension profile shall also be capable of decoding the bitstreams specified in Table 1 and Table 5 that conform to the decoding requirements specified for the scalable format range extensions profile in subclause H.11.1.2 of Rec. ITU-T H.265 | ISO/IEC 23008-2.

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. In Table 1, the value "29.97" shall be interpreted as an approximation of an exact value of 30000 ÷ 1001 and the value "59.94" shall be interpreted as an approximation of an exact value of 60000 ÷ 1001.

6.6.2 Test bitstreams – Block structure

6.6.2.1 Test bitstream STRUCT_A

Specification: All slices are coded as I, P or B slices. Each picture contains one slice. Various CTU and maximum CU sizes are used.

Functional stage: Test the reconstruction process of slices.

Purpose: Check that the decoder can properly decode I, P and B slices with various CTU and maximum CU sizes.

6.6.2.2 Test bitstream STRUCT_B

Specification: All slices are coded as I, P or B slices. Each picture contains one slice. Various CTU and minimum CU sizes are used.

Functional stage: Test the reconstruction process of slices.

Purpose: Check that the decoder can properly decode I, P and B slices with various CTU and minimum CU sizes.

6.6.3 Test bitstreams – Intra coding

6.6.3.1 Test bitstreams IPRED_A, IPRED_B, and IPRED_C

Specification: All slices are coded as I slices. Each picture contains one slice. All intra prediction modes (35 modes for each of luma 32x32, luma 16x16, luma 8x8, luma 4x4, chroma 16x16, chroma 8x8 and chroma 4x4, for a total 245 modes) are used. The IPRED_B bitstream contains only one picture, and conforms to the Main Still Picture profile.

Functional stage: Test the reconstruction process of I slices.

Purpose: Check that the decoder can properly decode I slices with all intra prediction modes.

6.6.3.2 Test bitstream CIP_A

Specification: The bitstream contains one I slice and one B slice, using one slice per picture. Both SAO and the deblocking filter are disabled.

Functional stage: Test the reference sample substitution process for intra sample prediction.

Purpose: Check that the decoder can properly decode slices of coded pictures containing intra TUs with unavailable samples for intra prediction.

6.6.3.3 Test bitstream CIP_B

Specification: The bitstream contains an I-picture and 4 P-pictures. Each picture contains only one slice. constrained_intra_pred_flag is equal to 1.

Functional stage: Test the reference sample substitution process for intra sample prediction.

Purpose: Check that the decoder can properly decode slices of coded pictures containing intra TUs with unavailable samples for intra prediction.

6.6.3.4 Test bitstream CIP_C

Specification: The bitstream contains one I slice and one B slice, using more than one slice per picture. Both SAO and the deblocking filter are disabled.

Functional stage: Test the reference sample substitution process for intra sample prediction.

Purpose: Check that the decoder can properly decode slices of coded pictures containing intra TUs with unavailable samples for intra prediction.

6.6.4 Test bitstreams – Inter frame coding

6.6.4.1 Test bitstream MERGE_A

Specification: All slices are coded as I or B slices. Each picture contains only one slice. five_minus_max_num_merge_cand is set equal to 4.

Functional stage: Test the reconstruction process of motion vector prediction.

Purpose: Check that the decoder can properly decode with the maximum number of merging candidates equal to any value permitted by the standard (i.e., 1, 2, 3, 4, 5).

6.6.4.2 Test bitstream MERGE_B

Specification: All slices are coded as I or B slices. Each picture contains only one slice. five_minus_max_num_merge_cand is set equal to 3.

Functional stage: Test the reconstruction process of motion vector prediction.

Purpose: Check that the decoder can properly decode with the maximum number of merging candidates equal to any value permitted by the standard (i.e., 1, 2, 3, 4, 5).

6.6.4.3 Test bitstream MERGE_C

Specification: All slices are coded as I or B slices. Each picture contains only one slice. five_minus_max_num_merge_cand is set equal to 2.

Functional stage: Test the reconstruction process of motion vector prediction.

Purpose: Check that the decoder can properly decode with the maximum number of merging candidates equal to any value permitted by the standard (i.e., 1, 2, 3, 4, 5).

6.6.4.4 Test bitstream MERGE_D

Specification: All slices are coded as I or B slices. Each picture contains only one slice. five_minus_max_num_merge_cand is set equal to 1.

Functional stage: Test the reconstruction process of motion vector prediction.

Purpose: Check that the decoder can properly decode with the maximum number of merging candidates equal to any value permitted by the standard (i.e., 1, 2, 3, 4, 5).

6.6.4.5 Test bitstream MERGE_E

Specification: All slices are coded as I or B slices. Each picture contains only one slice. five_minus_max_num_merge_cand is set equal to 0.

Functional stage: Test the reconstruction process of motion vector prediction.

Purpose: Check that the decoder can properly decode with the maximum number of merging candidates equal to any value permitted by the standard (i.e., 1, 2, 3, 4, 5).

6.6.4.6 Test bitstream MERGE_F

Specification: All slices are coded as I or B slices. Each picture contains only one slice. sps_temporal_mvp_enabled_flag is equal to 0 and five_minus_max_num_merge_cand is equal to 0.

Functional stage: Test the reconstruction process of motion vector prediction.

Purpose: Check that the decoder can properly decode when the temporal merging candidate is not included in the merge candidate set.

6.6.4.7 Test bitstream MERGE_G

Specification: All slices are coded as I or B slices. Each picture contains only one slice. five_minus_max_num_merge_cand is set equal to 0.

Functional stage: Test the reconstruction process of motion vector prediction.

Purpose: Check that the decoder can properly decode with merge index ranging from 0 to 4.

6.6.4.8 Test bitstream PMERGE_A

Specification: All slices are coded as I or B slices. Each picture contains only one slice. log2_parallel_merge_level_minus2 is set equal to 0.

Functional stage: Test the reconstruction process of motion vector prediction.

Purpose: Check that the decoder can properly decode parallel merge level values permitted by the standard (i.e., 2, 3, 4, 5, 6 for luma CTB size 64x64).

6.6.4.9 Test bitstream PMERGE_B

Specification: All slices are coded as I or B slices. Each picture contains only one slice. log2_parallel_merge_level_minus2 is set equal to 1.

Functional stage: Test the reconstruction process of motion vector prediction.

Purpose: Check that the decoder can properly decode parallel merge level values permitted by the standard (i.e., 2, 3, 4, 5, 6 for luma CTB size 64x64).

6.6.4.10 Test bitstream PMERGE_C

Specification: All slices are coded as I or B slices. Each picture contains only one slice. log2_parallel_merge_level_minus2 is set equal to 2.

Functional stage: Test the reconstruction process of motion vector prediction.

Purpose: Check that the decoder can properly decode the parallel merge level values permitted by the standard (i.e., 2, 3, 4, 5, 6 for luma CTB size 64x64).

6.6.4.11 Test bitstream PMERGE_D

Specification: All slices are coded as I or B slices. Each picture contains only one slice. log2_parallel_merge_level_minus2 is set equal to 3.

Functional stage: Test the reconstruction process of motion vector prediction.

Purpose: Check that the decoder can properly decode the parallel merge level values permitted by the standard (i.e., 2, 3, 4, 5, 6 for luma CTB size 64x64).

6.6.4.12 Test bitstream PMERGE_E

Specification: All slices are coded as I or B slices. Each picture contains only one slice. log2_parallel_merge_level_minus2 is set equal to 4.

Functional stage: Test the reconstruction process of motion vector prediction.

Purpose: Check that the decoder can properly decode the parallel merge level values permitted by the standard (i.e., 2, 3, 4, 5, 6 for luma CTB size 64x64).

6.6.4.13 Test bitstream AMVP_A

Specification: All slices are coded as I or P slices. Each picture contains only one slice. num_ref_idx_l0_default_active_minus1 is equal to 0, num_ref_idx_l1_default_active_minus1 is equal to 0 and num_ref_idx_active_override_flag is equal to 0.

Functional stage: Test the reconstruction process of motion vector prediction.

Purpose: Check that the decoder can properly decode when motion vector scaling is not needed for spatial motion vector prediction candidate generation (all inter-coded PUs within the same slice have the same inter_pred_idc and ref_idx_l0).

6.6.4.14 Test bitstream AMVP_B

Specification: All slices are coded as I or B slices. Each picture contains only one slice. Multiple reference pictures are used. For some slices, num_ref_idx_l0_default_active_minus1 is equal to 3 and num_ref_idx_active_override_flag is equal to 0. For other B slices, num_ref_idx_l0_default_active_minus1 is equal to 1, num_ref_idx_l1_default_active_minus1 is equal to 1 and num_ref_idx_active_override_flag is equal to 0.

Functional stage: Test the reconstruction process of motion vector prediction.

Purpose: Check that the decoder can properly decode when motion vector scaling is not needed for spatial motion vector prediction candidate generation.

6.6.4.15 Test bitstream AMVP_C

Specification: All slices are coded as I or P slices. Each picture contains only one slice.

Functional stage: Test the reconstruction process of motion vector prediction, specifically, motion vector prediction during the low delay condition.

Purpose: Check that the decoder can properly decode when motion vector scaling is not needed for spatial motion vector prediction candidate generation.

6.6.4.16 Test bitstream TMVP_A

Specification: Each picture contains only one slice. slice_temporal_mvp_enabled_flag is set equal to 0 for pictures 0 to 8 and 1 for pictures 9 to 16.

Functional stage: Test the reconstruction process of motion vector prediction.

Purpose: Check that the decoder can properly decode for different slice_temporal_mvp_enabled_flag values.

6.6.4.17 Test bitstream MVDL1ZERO_A

Specification: The bitstream contains multiple B slices per picture. Randomized on and off switching of the mvd_l1_zero_flag is included for multiple B slices.

Functional stage: Test the reconstruction process of motion vector prediction.

Purpose: Check that the decoder can properly decode when the parsing of list 1 motion vector difference for bi-prediction varies according to values of mvd_l1_zero_flag.

6.6.4.18 Test bitstream MVCLIP_A

Specification: Each picture contains only one slice. Motion vector prediction and merge candidate motion vectors are clipped to 16-bit values. Clipped motion vector prediction and merge candidates are selected.

Functional stage: Test the reconstruction process of motion vector prediction.

Purpose: Check that the decoder can properly decode when clipping of motion vector prediction and merge candidate motion vectors to 16-bit values occurs.

6.6.4.19 Test bitstream MVEDGE_A

Specification: Each picture contains only one slice. The bitstream includes motion vectors pointing to the padded edge regions in a picture.

Functional stage: Test the reconstruction process of motion vector prediction.

Purpose: Check that the decoder can properly decode motion vectors pointing to the padded edge regions of a picture.

6.6.4.20 Test bitstream WP_A

Specification: All slices are coded as I or P slices. Each picture contains only one slice. weighted_pred_flag is equal to 1. Plural reference indices are assigned to each reference picture.

Functional stage: Weighted sample prediction process for P slices with plural reference indices.

Purpose: Check that the decoder can properly decode weighted sample prediction for P slices with plural reference indices.

6.6.4.21 Test bitstream WP_B

Specification: All slices are coded as I, P or B slices. Each picture contains only one slice. weighted_pred_flag is equal to 1 and weighted_bipred_flag is equal to 1. Plural reference indices are assigned to each reference picture.

Functional stage: Weighted sample prediction process for P and B slices with plural reference indices.

Purpose: Check that the decoder can properly decode weighted sample prediction for P and B slices with plural reference indices.

6.6.5 Test bitstreams – Transform and quantization

6.6.5.1 Test bitstream RQT_A

Specification: All slices are coded as I or B slices. Each picture contains only one slice. max_transform_hierarchy_depth_inter and max_transform_hierarchy_depth_intra are both set equal to 0.

Functional stage: Test the reconstruction process of slices with residual quadtree.

Purpose: Check that the decoder can properly decode slices with residual quadtree with intra and inter depth equal to 0.

6.6.5.2 Test bitstream RQT_B

Specification: All slices are coded as I or B slices. Each picture contains only one slice. max_transform_hierarchy_depth_inter and max_transform_hierarchy_depth_intra are both set equal to 1.

Functional stage: Test the reconstruction process of slices with residual quadtree.

Purpose: Check that the decoder properly decodes slices with residual quadtree with intra and inter depth equal to 1.

6.6.5.3 Test bitstream RQT_C

Specification: All slices are coded as I or B slices. Each picture contains only one slice. max_transform_hierarchy_depth_inter and max_transform_hierarchy_depth_intra are both set equal to 2.

Functional stage: Test the reconstruction process of slices with residual quadtree.

Purpose: Check that the decoder properly decodes slices with residual quadtree with intra and inter depth equal to 2.

6.6.5.4 Test bitstream RQT_D

Specification: All slices are coded as I or B slices. Each picture contains only one slice. max_transform_hierarchy_depth_inter and max_transform_hierarchy_depth_intra are both set equal to 3.

Functional stage: Test the reconstruction process of slices with residual quadtree.

Purpose: Check that the decoder properly decodes slices with residual quadtree with intra and inter depth equal to 3.

6.6.5.5 Test bitstream RQT_E

Specification: All slices are coded as I or B slices. Each picture contains only one slice. max_transform_hierarchy_depth_inter and max_transform_hierarchy_depth_intra are both set equal to 4.

Functional stage: Test the reconstruction process of slices with residual quadtree.

Purpose: Check that the decoder properly decodes slices with residual quadtree with intra and inter depth equal to 4.

6.6.5.6 Test bitstream RQT_F

Specification: All slices are coded as I or B slices. Each picture contains only one slice. max_transform_hierarchy_depth_inter is set equal to 2 and max_transform_hierarchy_depth_intra is set equal to 0.

Functional stage: Test the reconstruction process of slices with residual quadtree.

Purpose: Check that the decoder properly decodes slices with residual quadtree with different intra and inter depths.

6.6.5.7 Test bitstream RQT_G

Specification: All slices are coded as I or B slices. Each picture contains only one slice. max_transform_hierarchy_depth_inter is set equal to 0 and max_transform_hierarchy_depth_intra is set equal to 2.

Functional stage: Test the reconstruction process of slices with residual quadtree.

Purpose: Check that the decoder properly decodes slices with residual quadtree with different intra and inter depths.

6.6.5.8 Test bitstream TUSIZE_A

Specification: All slices are coded as I or P slices. Each picture contains only one slice. log2_min_luma_transform_block_size_minus2 is set equal to 2. The maximum luma CB size is 64x64, the minimum luma CB size is 32x32, the minimum transform size for luma is 16x16 and for chroma is 8x8.

Functional stage: Test the reconstruction process of slices with limited minimum transform size.

Purpose: Check that the decoder properly decodes slices with residual quadtree with minimum transform size that are not the default 4x4.

6.6.5.9 Test bitstream DELTAQP_A

Specification: All slices are coded as I or B slices. Each picture contains only one slice. The maximum luma CB size is equal to 64x64 and the minimum luma CB size is equal to 8x8. diff_cu_qp_delta_depth is set randomly to values in the range of 0 to 3. CuQpDeltaVal is set randomly from −26 to 25.

Functional stage: Test the reconstruction process of slices with nonzero values of CuQpDeltaVal.

Purpose: Check that the decoder properly decodes slices with different values of CuQpDeltaVal.

6.6.5.10 Test bitstream DELTAQP_B

Specification: All slices are coded as I, P or B slices. Each picture contains more than one slice. The maximum luma CB size is equal to 64x64 and the minimum luma CB size is equal to 8x8. CuQpDeltaVal is set randomly from −26 to 25. slice_cb_qp_offset and slice_cr_qp_offset are set randomly from −4 to 4.

Functional stage: Test the reconstruction process of slices with nonzero values of CuQpDeltaVal.

Purpose: Check that the decoder properly handles various combinations of chroma QP offset.

6.6.5.11 Test bitstream DELTAQP_C

Specification: All slices are coded as I or B slices. Each picture contains only one slice. The maximum luma CB size is equal to 64x64 and the minimum luma CB size is equal to 8x8. diff_cu_qp_delta_depth is set randomly to values in the range of 0 to 3. CuQpDeltaVal is set randomly from −26 to 25. In some TUs, the cbfLuma or cbfChroma is equal to 0.

Functional stage: Test the reconstruction process of slices with nonzero values of CuQpDeltaVal.

Purpose: Check that the decoder properly decodes slices with different values of CuQpDeltaVal.

6.6.5.12 Test bitstream INITQP_A

Specification: All slices are coded as I or B slices. The value of init_qp_minus26 is set from −26 to 25.

Functional stage: Test QP initialization based on init_qp_minus26.

Purpose: Check that the decoder properly decodes different init_qp_minus26 values.

6.6.5.13 Test bitstream SLIST_A

Specification: All slices are coded as I or B slices. Each picture contains one slice. One SPS and more than one PPS are included. The SPS includes scaling list data. One of the PPSs does not include scaling list data. In other PPSs, different scaling lists data are included. In each picture, the PPS is overridden. scaling_list_enabled_flag is set equal to 1.

Functional stage: Test the scaling list reconstruction process. Tests switching of scaling list data in SPS and PPS.

Purpose: Check that the decoder properly decodes slices of coded frames with the scaling list, with different coding modes of the scaling list, when no scaling list is included in the PPS and when scaling list data are included in the PPS.

6.6.5.14 Test bitstream SLIST_B

Specification: All slices are coded as I or B slices. Each picture contains one slice. More than one SPS and more than one PPS are included. One of the SPSs does not include scaling list data. One of the PPSs does not include scaling list data. In other SPSs and PPSs, different scaling lists data are included. In each picture, the PPS is overridden. scaling_list_enabled_flag is set equal to 0 or 1.

Functional stage: Test the scaling list reconstruction process. Tests switching of scaling list off, default scaling list and scaling list in parameter sets.

Purpose: Check that the decoder can properly decode slices of coded frames with the scaling list, different coding modes of the scaling list and when there are multiple SPSs and PPSs.

6.6.5.15 Test bitstream SLIST_C

Specification: All slices are coded as I or B slices. Each picture contains one slice. One SPS and more than one PPS are included. The SPS does not include scaling list data. One of the PPSs does not include scaling list data. In other PPSs, different scaling lists data are included. In each picture, the PPS is overridden. scaling_list_enabled_flag is set equal to 1.

Functional stage: Test the scaling list reconstruction process. Tests switching of default scaling list and scaling list in PPS.

Purpose: Check that the decoder can properly decode slices of coded frames with the scaling list, different coding modes of the scaling list, when no scaling list data are present and switching of default scaling list and scaling list data in PPS occurs.

6.6.5.16 Test bitstream SLIST_D

Specification: All slices are coded as I or B slices. Each picture contains more than one slice. More than one SPS and more than one PPS are included. One of the SPSs does not include scaling list data. One of the PPSs does not include scaling list data. In other SPSs and PPSs, different scaling lists data are included. In each picture, the PPS is overridden. scaling_list_enabled_flag is set equal to 0 or 1.

Functional stage: Test the scaling list reconstruction process. Tests switching of scaling list off, default scaling list and scaling list in parameter sets.

Purpose: Check that the decoder can properly decode slices of coded frames with the scaling list, different coding modes of the scaling list and when there are multiple SPSs and PPSs.

6.6.6 Test bitstreams – Deblocking filter

6.6.6.1 Test bitstream DBLK_A

Specification: All slices are coded as I, P or B slices. Each picture contains more than one slice. More than one PPS is used. QP is set randomly to values in the range of 22 to 51. pps_beta_offset_div2 is randomly set in each picture from −6 to 6. slice_beta_offset_div2 and slice_tc_offset_div2 are randomly set in each slice from −6 to 6.

Functional stage: Test the deblocking filter process.

Purpose: Check that the decoder can properly decode slices with various combinations of deblocking filter control parameters.

6.6.6.2 Test bitstream DBLK_B

Specification: All slices are coded as I, P or B slices. Each picture contains more than one slice. More than one PPS is used. pps_cb_qp_offset and pps_cr_qp_offset are randomly set to values in the range from −12 to 12. slice_cb_qp_offset and slice_cr_qp_offset are randomly set from −4 to 4.

Functional stage: Test the deblocking filter process.

Purpose: Check that the decoder can properly decode when the deblocking filter varies according to various combinations of QP.

6.6.6.3 Test bitstream DBLK_C

Specification: All slices are coded as I, P or B slices. Each picture contains more than one slice. pps_deblocking_filter_disabled_flag is set equal to 0. slice_deblocking_filter_disabled_flag is randomly set equal to 0 or 1.

Functional stage: Test the deblocking filter process.

Purpose: Check that the decoder can properly decode with the deblocking filter being enabled and disabled across slices.

6.6.6.4 Test bitstream DBLK_D

Specification: All slices are coded as I or B slices. Each picture contains more than one slice and tile. slice_loop_filter_across_slices_enabled_flag is set equal to 0 and loop_filter_across_tiles_enabled_flag is set equal to 1.

Functional stage: Test the deblocking filter process.

Purpose: Check that the decoder can properly decode with the deblocking filter being enabled and disabled at slice and tile boundaries.

6.6.6.5 Test bitstream DBLK_E

Specification: All slices are coded as I or B slices. Each picture contains more than one slice and tile. slice_loop_filter_across_slices_enabled_flag is set equal to 1 and loop_filter_across_tiles_enabled_flag is set equal to 0.

Functional stage: Test the deblocking filter process.

Purpose: Check that the decoder can properly decode with the deblocking filter being enabled and disabled at slice and tile boundaries.

6.6.6.6 Test bitstream DBLK_F

Specification: All slices are coded as I or B slices. Each picture contains more than one slice and tile. slice_loop_filter_across_slices_enabled_flag is set equal to 0 and loop_filter_across_tiles_enabled_flag is set equal to 1.

Functional stage: Test the deblocking filter process.

Purpose: Check that the decoder can properly decode with the deblocking filter being enabled and disabled at slice and tile boundaries.

6.6.6.7 Test bitstream DBLK_G

Specification: All slices are coded as I or B slices. Each picture contains more than one slice and tile. slice_loop_filter_across_slices_enabled_flag is set equal to 1 and loop_filter_across_tiles_enabled_flag is set equal to 0.

Functional stage: Test the deblocking filter process.

Purpose: Check that the decoder can properly decode with the deblocking filter being enabled and disabled at slice and tile boundaries.

6.6.7 Test bitstreams – Sample adaptive offset

6.6.7.1 Test bitstream SAO_A

Specification: All slices are coded as I or B slices. Each picture contains only one slice. sao_merge_left_flag and sao_merge_up_flag are randomly set equal to 0 or 1.

Functional stage: Test the reconstruction process of sample adaptive offset.

Purpose: Check that the decoder can properly decode with random SAO merge left/up flag values.

6.6.7.2 Test bitstream SAO_B

Specification: All slices are coded as I or B slices. Each picture contains only one slice and contains more than one tile. slice_sao_luma_flag and slice_sao_chroma_flag are randomly set equal to 0 or 1.

Functional stage: Test the reconstruction process of sample adaptive offset.

Purpose: Check that the decoder can properly decode with tiles and randomly enabled SAO for luma and/or SAO for chroma per slice.

6.6.7.3 Test bitstream SAO_C

Specification: All slices are coded as I or P slices. Each picture contains only one slice. All SAO offset values in this bitstream have maximum allowed magnitude 7 and random sign.

Functional stage: Test the reconstruction process of sample adaptive offset.

Purpose: Check that the decoder can properly decode with maximum SAO offset values.

6.6.7.4 Test bitstream SAO_D

Specification: All slices are coded as I or P slices. Each picture contains only one slice. SAO offset values in this bitstream have random values in the range −7..7.

Functional stage: Test the reconstruction process of sample adaptive offset.

Purpose: Check that the decoder can properly decode with random SAO offset values.

6.6.7.5 Test bitstream SAO_E

Specification: All slices are coded as I or B slices. Each picture contains only one slice. A set of SAO parameters is associated with each CTB for all frames, therefore no SAO merge flags (up or left) are used. Only the band offset SAO type is used and the four SAO offset values are randomly set to −7 or 7. The luma CTB size is set equal to 16x16.

Functional stage: Tests loading of maximum SAO information at CTB level and frame.

Purpose: Check that the decoder can properly decode with the maximum possible SAO information.

6.6.7.6 Test bitstream SAO_F

Specification: All slices are coded as I or B slices. Each picture contains only one slice. A set of SAO parameters is associated to each CTB for all frames, therefore, no SAO merge flags (up or left) are used. Only the band offset SAO type is used and the four SAO offset values are randomly set to −7 or 7. The luma CTB size is set equal to 32x32.

Functional stage: Tests loading of maximum SAO information at CTB level and frame.

Purpose: Check that the decoder can properly decode with the maximum possible SAO information.

6.6.7.7 Test bitstream SAO_G

Specification: All slices are coded as I or B slices. Each picture contains only one slice. A set of SAO parameters is associated to each CTB for all frames, therefore, no SAO merge flags (up or left) are used. Only the band offset SAO type is used and the four SAO offset values are randomly set to −7 or 7. The luma CTB size is set equal to 64x64.

Functional stage: Tests loading of maximum SAO information at CTB level and frame.

Purpose: Check that the decoder can properly decode with the maximum possible SAO information.

6.6.7.8 Test bitstream SAO_H

Specification: All slices are coded as I slices. Each picture contains multiple slices. SAO edge modes 2 and 3 are used (diagonal). slice_loop_filter_across_slices_enabled_flag is set to 0.

Functional stage: Test the SAO decoding process with different diagonal neighbour availability.

Purpose: Check that, when slice_loop_filter_across_slices_enabled_flag is set to zero, the decoder can properly apply the SAO filter in CTU corners regardless of whether diagonally neighbouring CTUs are available or not.

6.6.7.9 Test bitstream SAODBLK_A

Specification: All slices are coded as I or P slices. Each picture contains only one slice. The bitstream includes 1) non-rectangular shaped slices and 2) the minimum size of a CTU is used, which for chroma planes takes the value of 8x8 and exactly coincides with a block that is used for deblocking.

Functional stage: Test the SAO decoding process with various coding parameter settings.

Purpose: Check that that the decoder can properly decode loop filtering on slice boundaries.

6.6.7.10 Test bitstream SAODBLK_B

Specification: All slices are coded as I or P slices. Each picture contains both tiles and slices. The bitstream includes 1) non-rectangular shaped slices and 2) the minimum size of a CTU is used, which for chroma planes takes the value of 8x8 and exactly coincides with a block that is used for deblocking.

Functional stage: Test the SAO decoding process with various coding parameter settings.

Purpose: Check that the decoder can properly decode loop filtering on slice boundaries.

6.6.8 Test bitstreams – Entropy coding

6.6.8.1 Test bitstream MAXBINS_A

Specification: All slices are coded as I slices. Each picture contains only one slice. The number of bins per CTU is constructed to be within 95% of the maximum number which is 4096 bits per CTU with luma CTB size 16x16. pcm_enabled_flag is set equal to 1.

Functional stage: Test the parsing process.

Purpose: Check that the decoder can properly decode slices with the maximum number of bins per CTU.

6.6.8.2 Test bitstream MAXBINS_B

Specification: All slices are coded as I or B slices. Each picture contains only one slice. The number of bins per CTU is constructed to be within 95% of the maximum number which is 4096 bits per CTU with luma CTB size 16x16. pcm_enabled_flag is set equal to 1.

Functional stage: Test the parsing process.

Purpose: Check that the decoder can properly decode slices with the maximum number of bins per CTU.

6.6.8.3 Test bitstream MAXBINS_C

Specification: All slices are coded as I or B slices. Each picture contains only one slice. The number of bins per CTU is constructed to be within 95% of the maximum number which is 4096 bits per CTU with luma CTB size 16x16. pcm_enabled_flag is set equal to 1.

Functional stage: Test the parsing process.

Purpose: Check that the decoder can properly decode slices with the maximum number of bins per CTU.

6.6.8.4 Test bitstream CAINIT_A

Specification: All slices are coded as I or B slices. Each picture contains only one slice. There is one PPS. cabac_init_present_flag is equal to 0 in PPS.

Functional stage: Test the parsing process.

Purpose: Check that the decoder properly decodes when cabac_init_flag is not signalled in the slice header of P or B slices.

6.6.8.5 Test bitstream CAINIT_B

Specification: All slices are coded as I or B slices. Each picture contains only one slice. There is one PPS. cabac_init_present_flag is equal to 1 in PPS. cabac_init_flag is signalled for B slices in the slice header referring the PPS. cabac_init_flag can be set to 0 or 1.

Functional stage: Test the parsing process.

Purpose: Check that the decoder properly decodes with different cabac_init_flag values in B slices.

6.6.8.6 Test bitstream CAINIT_C

Specification: All slices are coded as I or P slices. Each picture contains only one slice. There is one PPS. cabac_init_present_flag is equal to 1 in PPS. cabac_init_flag is signalled for P slices in the slice header referring the PPS. cabac_init_flag can be set to 0 or 1.

Functional stage: Test the parsing process.

Purpose: Check that the decoder properly decodes with different cabac_init_flag values in P slices.

6.6.8.7 Test bitstream CAINIT_D

Specification: All slices are coded as I or B slices which are uni-directionally predicted. Each picture contains only one slice. There is one PPS. cabac_init_present_flag is equal to 1 in PPS. cabac_init_flag is signalled for B slices in the slice header referring the PPS. cabac_init_flag can be set to 0 or 1.

Functional stage: Test the parsing process.

Purpose: Check that the decoder properly decodes with different cabac_init_flag values in P slices.

6.6.8.8 Test bitstream CAINIT_E

Specification: All slices are coded as I or P slices. Each picture contains only one slice. Each slice contains four tiles (two columns of tiles and two rows of tiles with uniform spacing). There is one PPS. cabac_init_present_flag is equal to 1 in PPS. cabac_init_flag is signalled for P slices in the slice header referring the PPS. cabac_init_flag can be set to 0 or 1.

Functional stage: Test the parsing process.

Purpose: Check that the decoder properly decodes when cabac_init_flag is switched in P slices with the use of tiles.

6.6.8.9 Test bitstream CAINIT_F

Specification: All slices are coded as I or uni-directionally predicted B slices. Each picture contains only one slice. Each slice contains four tiles (two columns of tiles and two rows of tiles with uniform spacing). There is one PPS. cabac_init_present_flag is equal to 1 in PPS. cabac_init_flag is signalled for uni-directionally predicted B slices in the slice header referring the PPS. cabac_init_flag can be set to 0 or 1.

Functional stage: Test the parsing process.

Purpose: Check that the decoder properly decodes when cabac_init_flag is switched in B slices with the use of tiles.

6.6.8.10 Test bitstream CAINIT_G

Specification: All slices are coded as I or P slices. Each picture contains only one slice. Each slice contains multiple dependent slice segments. Each dependent slice contains three CTUs or less. There is one PPS. cabac_init_present_flag is equal to 1 in PPS. cabac_init_flag is signalled for P slices in the slice header referring the PPS. cabac_init_flag can be set to 0 or 1.

Functional stage: Test the parsing process.

Purpose: Check that the decoder properly decodes when cabac_init_flag is switched in P slices with dependent slice segments.

6.6.8.11 Test bitstream CAINIT_H

Specification: All slices are coded as I or uni-directionally predicted B slices. Each picture contains only one slice. Each slice contains multiple dependent slice segments. Each dependent slice contains three CTUs or less. There is one PPS. cabac_init_present_flag is equal to 1 in PPS. cabac_init_flag is signalled for uni-directionally predicted B slices in the slice header referring the PPS. cabac_init_flag can be set to 0 or 1.

Functional stage: Test the parsing process.

Purpose: Check that the decoder properly decodes when cabac_init_flag is switched in B slices with dependent slice segments.

6.6.8.12 Test bitstream SDH_A

Specification: All slices are coded as I or B slices. Each picture contains only one slice. sign_data_hiding_enabled_flag is set equal to 1. The bitstream includes various configurations of sign data hiding.

Functional stage: Test the parsing process.

Purpose: Check that the decoder properly decodes with sign data hiding.

6.6.9 Test bitstreams – Temporal scalability

6.6.9.1 Test bitstream TSCL_A

Specification: All slices are coded as I or B slices. Each picture contains only one slice. The bitstream includes four temporal layers.

Functional stage: Test temporal scalability.

Purpose: Check that the decoder properly decodes temporal layers.

6.6.9.2 Test bitstream TSCL_B

Specification: All slices are coded as I or P slices. Each picture contains only one slice. The bitstream includes four temporal layers.

Functional stage: Test temporal scalability.

Purpose: Check that the decoder properly decodes temporal layers.

6.6.10 Test bitstreams – Parallel processing tools

6.6.10.1 Test bitstream TILES_A

Specification: All slices are coded as I or P slices. Each picture contains only one slice. num_tile_columns_minus1 and num_tile_rows_minus1 are set equal to 4, which is the maximum value for level 4.1. uniform_spacing_flag is set equal to 0. The values of column_width_minus1[i] and row_height_minus1[i] are set randomly for each picture. loop_filter_across_tiles_enabled_flag is set randomly for each picture.

Functional stage: Test dependency breaks at tile boundaries.

Purpose: Check that the decoder properly decodes when there is random non-uniform tile spacing with a maximum number of tiles and the deblocking filter is enabled and disabled at tile boundaries.

6.6.10.2 Test bitstream TILES_B

Specification: All slices are coded as I or P slices. Each picture contains a random number of slices. All slice boundaries are aligned with tile boundaries. num_tile_columns_minus1 and num_tile_rows_minus1 are set equal to 4, which is the maximum value for level 4.1. uniform_spacing_flag is set equal to 0. The values of column_width_minus1[i] and row_height_minus1[i] are set randomly for each picture. loop_filter_across_tiles_enabled_flag is set randomly for each picture. pps_loop_filter_across_slices_enabled_flag is set randomly for each frame. slice_loop_filter_across_slices_enabled_flag is set randomly for each slice.

Functional stage: Test dependency breaks at tile boundaries and enabling/disabling the deblocking filter at tile/slice boundaries.

Purpose: Check that the decoder properly decodes when there is random non-uniform tile spacing with a maximum number of tiles and the deblocking filter is enabled and disabled at tile and slice boundaries.

6.6.10.3 Test bitstream WPP_A

Specification: entropy_coding_sync_enabled_flag is set equal to 1. A luma CTB size of 64x64 is used. The bitstream contains six repeated patterns of pictures with a particular ordering and referencing relationship, which are each eight pictures long. The first three of these groups of eight pictures have pictures with the following characteristics:

– One slice in the frame, QP is constant.
– One slice in the frame, the QP of each CU is set equal to a value such that $\mathrm{Abs}(QP - SliceQp_Y) > 2$.
– Maximum number of independent slice segments in the frame, at least one slice segment is one CTU long, at least one slice segment is two CTUs long, QP is constant.
– Maximum number of independent slice segments in the frame, at least one slice segment is one CTU long, at least one slice segment is two CTUs long, the QP of each CU is set equal to a value such that $\mathrm{Abs}(QP - SliceQp_Y) > 2$.
– Maximum number of dependent slice segments in the frame, at least one dependent slice segment is one CTU long, at least one dependent slice segment is two CTUs long, QP is constant.
– Maximum number of dependent slice segments in the frame, at least one dependent slice segment is one CTU long, at least one dependent slice segment is two CTUs long, the QP of each CU is set equal to a value such that $\mathrm{Abs}(QP - SliceQp_Y) > 2$.
– Random combination of independent/dependent slice segments, QP is constant.
– Random combination of independent/dependent slice segments, the QP of each CU is set equal to a value such that $\mathrm{Abs}(QP - SliceQp_Y) > 2$.

The first of these groups of eight pictures is coded using all intra CUs, and the second is coded with CU skipping disabled. The final three of these groups of eight pictures feature a mixture of single slice pictures, and pictures coded using multiple slices and multiple slice segments. Random amounts of slice segment header extension bytes are encoded in each slice header.

Functional stage: Tests that entropy coding is correctly synchronized with different slice types. Tests that the QP predictor is reset to $SliceQp_Y$ at the beginning of every CTU row. May be used to test handling of entry points by a parallel decoder.

Purpose: Check that the decoder properly decodes when entropy coding synchronization is enabled and handle entry points when slice segment header extension data bytes are present. This bitstream does not conform to the Main profile or Main 10 profile since slice_segment_header_extension_present_flag is equal to 1. However, Main profile and Main 10 profile decoders shall be able to decode this bitstream and ignore the slice segment header extension data bytes.

6.6.10.4 Test bitstream WPP_B

Specification: entropy_coding_sync_enabled_flag is set equal to 1. A luma CTB size of 32x32 is used. The bitstream contains six repeated patterns of pictures with a particular ordering and referencing relationship, which are each eight pictures long. The first three of these groups of eight pictures have pictures with the following characteristics:

- One slice in the frame, QP is constant.
- One slice in the frame, the QP of each CU is set equal to a value such that $Abs(QP - SliceQp_Y) > 2$.
- Maximum number of independent slice segments in the frame, at least one slice segment is one CTU long, at least one slice segment is two CTUs long, QP is constant.
- Maximum number of independent slice segments in the frame, at least one slice segment is one CTU long, at least one slice segment is two CTUs long, the QP of each CU is set equal to a value such that $Abs(QP - SliceQp_Y) > 2$.
- Maximum number of dependent slice segments in the frame, at least one dependent slice segment is one CTU long, at least one dependent slice segment is two CTUs long, QP is constant.
- Maximum number of dependent slice segments in the frame, at least one dependent slice segment is one CTU long, at least one dependent slice segment is two CTUs long, the QP of each CU is set equal to a value such that $Abs(QP - SliceQp_Y) > 2$.
- Random combination of independent/dependent slice segments, QP is constant.
- Random combination of independent/dependent slice segments, the QP of each CU is set equal to a value such that $Abs(QP - SliceQp_Y) > 2$.

The first of these groups of eight pictures is coded using all intra CUs, and the second is coded with CU skipping disabled. The final three of these groups of eight pictures feature a mixture of single slice pictures, and pictures coded using multiple slices and multiple slice segments. Random amounts of slice segment header extension bytes are encoded in each slice header.

Functional stage: Tests that entropy coding is correctly synchronized with different slice types. Tests that the QP predictor is reset to $SliceQp_Y$ at the beginning of every CTU row. May be used to test handling of entry points by a parallel decoder.

Purpose: Check that the decoder properly decodes when entropy coding synchronization is enabled and and handle entry points when slice segment header extension data bytes are present. This bitstream does not conform to the Main profile or Main 10 profile since slice_segment_header_extension_present_flag is equal to 1. However, Main profile and Main 10 profile decoders shall be able to decode this bitstream and ignore the slice segment header extension data bytes.

6.6.10.5 Test bitstream WPP_C

Specification: entropy_coding_sync_enabled_flag is set equal to 1. A luma CTB size of 16x16 is used. The bitstream contains six repeated patterns of pictures with a particular ordering and referencing relationship, which are each eight pictures long. The first three of these groups of eight pictures have pictures with the following characteristics:

- One slice in the frame, QP is constant.
- One slice in the frame, the QP of each CU is set equal to a value such that $Abs(QP - SliceQp_Y) > 2$.
- Maximum number of independent slice segments in the frame, at least one slice segment is one CTU long, at least one slice segment is two CTUs long, QP is constant.
- Maximum number of independent slice segments in the frame, at least one slice segment is one CTU long, at least one slice segment is two CTUs long, the QP of each CU is set equal to a value such that $Abs(QP - SliceQp_Y) > 2$.
- Maximum number of dependent slice segments in the frame, at least one dependent slice segment is one CTU long, at least one dependent slice segment is two CTUs long, QP is constant.

– Maximum number of dependent slice segments in the frame, at least one dependent slice segment is one CTU long, at least one dependent slice segment is two CTUs long, the QP of each CU is set equal to a value such that Abs(QP − $SliceQp_Y$) > 2.

– Random combination of independent/dependent slice segments, QP is constant.

– Random combination of independent/dependent slice segments, the QP of each CU is set equal to a value such that Abs(QP − $SliceQp_Y$) > 2.

The first of these groups of eight pictures is coded using all intra CUs, and the second is coded with CU skipping disabled. The final three of these groups of eight pictures feature a mixture of single slice pictures, and pictures coded using multiple slices and multiple slice segments. Random amounts of slice segment header extension bytes are encoded in each slice header.

Functional stage: Tests that entropy coding is correctly synchronized with different slice types. Tests that the QP predictor is reset to $SliceQp_Y$ at the beginning of every CTU row. May be used to test handling of entry points by a parallel decoder.

Purpose: Check that the decoder properly decodes when entropy coding synchronization is enabled and and handle entry points when slice segment header extension data bytes are present. This bitstream does not conform to the Main profile or Main 10 profile since slice_segment_header_extension_present_flag is equal to 1. However, Main profile and Main 10 profile decoders shall be able to decode this bitstream and ignore the slice segment header extension data bytes.

6.6.10.6 Test bitstream WPP_D

Specification: entropy_coding_sync_enabled_flag is set equal to 1. A luma CTB size of 64x64 is used. The picture is one CTU wide. The bitstream contains six repeated patterns of pictures with a particular ordering and referencing relationship, which are each eight pictures long. These are encoded with varying numbers of slices and slice segments. Even frames have fixed QP, while the QP established at the CU level in odd frames is set such that Abs(QP − $SliceQp_Y$) > 2. The first of these groups of eight pictures is coded using all intra CUs, and the second is coded with CU skipping disabled. Random amounts of slice segment header extension bytes are encoded in each slice header.

Functional stage: Tests that entropy coding is correctly synchronized when a picture is one CTU wide. Tests that the QP predictor is reset to $SliceQp_Y$ at the beginning of every CTU row. May be used to test handling of entry points by a parallel decoder.

Purpose: Check that the decoder properly decodes when a picture is one CTU wide and handle entry points when slice segment header extension data bytes are present. This bitstream does not conform to the Main profile or Main 10 profile since slice_segment_header_extension_present_flag is equal to 1. However, Main profile and Main 10 profile decoders shall be able to decode this bitstream and ignore the slice segment header extension data bytes.

6.6.10.7 Test bitstream WPP_E

Specification: entropy_coding_sync_enabled_flag is set equal to 1. A luma CTB size of 64x64 is used. The picture is two CTUs wide. The bitstream contains six repeated patterns of pictures with a particular ordering and referencing relationship, which are each eight pictures long. These are encoded with varying numbers of slices and slice segments. Even frames have fixed QP, while the QP established at the CU level in odd frames is set such that Abs(QP − $SliceQp_Y$) > 2. The first of these groups of eight pictures is coded using all intra CUs, and the second is coded with CU skipping disabled. Random amounts of slice segment header extension bytes are encoded in each slice header.

Functional stage: Tests that entropy coding is correctly synchronized when a picture is two CTUs wide. Tests that the QP predictor is reset to $SliceQp_Y$ at the beginning of every CTU row. May be used to test handling of entry points by a parallel decoder.

Purpose: Check that the decoder properly decodes when a picture is two CTUs wide and handle entry points when slice segment header extension data bytes are present. This bitstream does not conform to the Main profile or Main 10 profile since slice_segment_header_extension_present_flag is equal to 1. However, Main profile and Main 10 profile decoders shall be able to decode this bitstream and ignore the slice segment header extension data bytes.

6.6.10.8 Test bitstream WPP_F

Specification: entropy_coding_sync_enabled_flag is set equal to 1. A luma CTB size of 64x64 is used. The picture is three CTUs wide. The bitstream contains six repeated patterns of pictures with a particular ordering and referencing relationship, which are each eight pictures long. These are encoded with varying numbers of slices and slice segments. Even frames have fixed QP, while the QP established at the CU level in odd frames is set such that Abs(QP − $SliceQp_Y$) > 2. The first of these groups of eight pictures is coded using all intra CUs, and the second is coded with CU skipping disabled. Random amounts of slice segment header extension bytes are encoded in each slice header.

Functional stage: Tests that entropy coding is correctly synchronized when a picture is three CTUs wide. Tests that the QP predictor is reset to $SliceQp_Y$ at the beginning of every CTU row. May be used to test handling of entry points by a parallel decoder.

Purpose: Check that the decoder properly decodes when a picture is three CTUs wide and handle entry points when slice segment header extension data bytes are present. This bitstream does not conform to the Main profile or Main 10 profile since slice_segment_header_extension_present_flag is equal to 1. However, Main profile and Main 10 profile decoders shall be able to decode this bitstream and ignore the slice segment header extension data bytes.

6.6.10.9 Test bitstream ENTP_A

Specification: All slices are coded as I slices. Each picture contains only one slice. Four tiles are included in a picture. num_tile_columns_minus1 and num_tile_rows_minus1 are set equal to 1. uniform_spacing_flag is set equal to 1. There are some tiles in the picture with PicOrderCntVal equal to 4 that contains emulation prevention bytes.

Functional stage: Test entry point signalling.

Purpose: Check that the decoder properly decodes when entry point signalling in a slice header is used with tiles and emulation prevention bytes occur in the substream(s).

6.6.10.10 Test bitstream ENTP_B

Specification: All slices are coded as I or B slices. Each picture contains only one slice. Six tiles are included in a picture. num_tile_columns_minus1 is set equal to 1 and num_tile_rows_minus1 are set equal to 2. uniform_spacing_flag is set equal to 1. There are some pictures (e.g., with PicOrderCntVal equal to 4, 6, 10, 12, 18, and 20) that contain a tile in which emulation prevention bytes occur.

Functional stage: Test entry point signalling.

Purpose: Check that the decoder properly decodes when entry point signalling in a slice header is used with tiles and emulation prevention bytes occur in the substream(s).

6.6.10.11 Test bitstream ENTP_C

Specification: All slices are coded as I or B slices. Each picture contains only one slice. entropy_coding_sync_enabled_flag is set equal to 1.

Functional stage: Test entry point signalling.

Purpose: Check that the decoder properly decodes when entropy coding synchronization is used.

6.6.11 Test bitstreams – Other coding tools

6.6.11.1 Test bitstream IPCM_A

Specification: All slices are coded as I slices. Each picture contains only one slice. pcm_enabled_flag is equal to 1. Both pcm_sample_bit_depth_luma_minus1 and pcm_sample_bit_depth_chroma_minus1 are equal to 7. log2_min_pcm_luma_coding_block_size_minus3, log2_diff_max_min_pcm_luma_coding_block_size, and pcm_loop_filter_disabled_flag are equal to 0, 2 and 0, respectively.

Functional stage: Test parsing of pcm_flag in the coding unit syntax.

Purpose: Check that the decoder properly decodes pcm_flag.

6.6.11.2 Test bitstream IPCM_B

Specification: All slices are coded as I slices. Each picture contains only one slice. pcm_enabled_flag is equal to 1. Both pcm_sample_bit_depth_luma_minus1 and pcm_sample_bit_depth_chroma_minus1 are equal to 5. log2_min_pcm_luma_coding_block_size_minus3, log2_diff_max_min_pcm_luma_coding_block_size, and pcm_loop_filter_disabled_flag are equal to 0, 1 and 0, respectively.

Functional stage: Test parsing of pcm_flag in the coding unit syntax. Test parsing of pcm_sample_luma and pcm_sample_chroma data.

Purpose: Check that the decoder properly decodes pcm_flag, and pcm_sample_luma and pcm_sample_chroma data.

6.6.11.3 Test bitstream IPCM_C

Specification: All slices are coded as I slices. Each picture contains only one slice. pcm_enabled_flag is equal to 1. Both pcm_sample_bit_depth_luma_minus1 and pcm_sample_bit_depth_chroma_minus1 are equal to 7.

log2_min_pcm_luma_coding_block_size_minus3, log2_diff_max_min_pcm_luma_coding_block_size, and pcm_loop_filter_disabled_flag are equal to 0, 1 and 1, respectively.

Functional stage: Test parsing of pcm_flag in the coding unit syntax. Test parsing of pcm_sample_luma and pcm_sample_chroma data. Test skipping of loop filtering on samples associated with pcm_flag equal to 1.

Purpose: Check that the decoder properly decodes pcm_flag, pcm_sample_luma and pcm_sample_chroma data and skips loop filtering on samples associated with pcm_flag equal to 1.

6.6.11.4 Test bitstream IPCM_D

Specification: All slices are coded as I slices. Each picture contains only one slice. pcm_enabled_flag is equal to 1. Both pcm_sample_bit_depth_luma_minus1 and pcm_sample_bit_depth_chroma_minus1 are equal to 7. log2_min_pcm_luma_coding_block_size_minus3, log2_diff_max_min_pcm_luma_coding_block_size, and pcm_loop_filter_disabled_flag are equal to 0, 1 and 0, respectively. transquant_bypass_enabled_flag is equal to 1.

Functional stage: Test parsing of pcm_flag in the coding unit syntax. Test parsing of pcm_sample_luma and pcm_sample_chroma data in PCM sample syntax. Test skipping of loop filtering on samples associated with both cu_transquant_bypass_flag and pcm_flag equal to 1.

Purpose: Check that the decoder properly decodes pcm_flag, pcm_sample_luma and pcm_sample_chroma data and skips loop filtering on samples associated with both cu_transquant_bypass_flag and pcm_flag equal to 1.

6.6.11.5 Test bitstream IPCM_E

Specification: Contains a single coded picture. The coded picture contains only one intra slice. pcm_enabled_flag is equal to 1. pcm_sample_bit_depth_luma_minus1 and pcm_sample_bit_depth_chroma_minus1 are equal to 5 and 7, respectively. log2_min_pcm_luma_coding_block_size_minus3, log2_diff_max_min_pcm_luma_coding_block_size, and pcm_loop_filter_disabled_flag are equal to 1, 0 and 0, respectively.

Functional stage: Test parsing of pcm_flag in the coding unit syntax. Test parsing of pcm_sample_luma and pcm_sample_chroma data with different bit depths.

Purpose: Check that the decoder can properly decode pcm_flag, and pcm_sample_luma and pcm_sample_chroma data with different pcm_sample_bit_depth_luma_minus1 and pcm_sample_bit_depth_chroma_minus1 values.

6.6.11.6 Test bitstream TS_A

Specification: Each picture contains only one slice. transform_skip_enabled_flag is set equal to 1 for pictures 0 to 8 and equal to 0 for pictures 9 to 16.

Functional stage: Test reconstruction process of slices with transform_skip_enabled_flag is equal to 1.

Purpose: Check that the decoder can properly decode transform_skip_enabled_flag.

6.6.11.7 Test bitstreams AMP_A, AMP_D, AMP_E and AMP_F

Specification: All slices are coded as I, P or B slices. Each picture contains only one slice. All asymmetric motion partition modes (2NxnU, 2NxND, nLx2N, nRx2N) are included.

Functional stage: Test reconstruction process of slices with amp_enabled_flag equal to 1.

Purpose: Check that the decoder can properly decode slices with all asymmetric motion partition modes.

6.6.11.8 Test bitstream AMP_B

Specification: All slices are coded as I, P or B slices. Each picture contains only one slice. Asymmetric motion partition is only utilized for PUs of which size is larger than minimum CU.

Functional stage: Test reconstruction process of slices with amp_enabled_flag equal to 1.

Purpose: Check that the decoder can properly decode slices with a constraint on asymmetric motion partition modes.

6.6.11.9 Test bitstream LS_A

Specification: All slices are coded as I or B slices. Each picture contains only one slice. All the CUs are transform/quantization/filtering bypass CUs. SAO and the deblocking filter are enabled in the SPS and PPS.

Functional stage: Test reconstruction process of transform/quantization/filtering bypass coding.

Purpose: Check that the decoder can properly decode transform/quantization/filtering bypass coding.

6.6.11.10 Test bitstream LS_B

Specification: All slices are coded as I or B slices. Each picture contains only one slice. At least 50% of the CUs do not use transform/quantization/filtering bypass, and where there at are least 100 CUs in each of the following categories:

– The luma CB is 64x64, it has cu_transquant_bypass_flag on, at least one of the neighbouring CUs uses SAO, at least one of the neighbouring CUs uses deblocking filtering.

– The luma CB is 32x32, it has cu_transquant_bypass_flag on, at least one of the neighbouring CUs uses SAO, at least one of the neighbouring CUs uses deblocking filtering.

– The luma CB is 16x16, it has cu_transquant_bypass_flag on, at least one of the neighbouring CUs uses SAO, at least one of the neighbouring CUs uses deblocking filtering.

– The luma CB is 8x8, it has cu_transquant_bypass_flag on, at least one of the neighbouring CUs uses SAO, at least one of the neighbouring CUs uses deblocking filtering.

Functional stage: Test reconstruction process of transform/quantization/filtering bypass coding.

Purpose: Check that the decoder can properly decode transform/quantization/filtering bypass coding.

6.6.12 Test bitstreams – High level syntax

6.6.12.1 Test bitstream NUT_A

Specification: All slices are coded as I or P slices. Each picture contains only one slice. Three temporal layers are used. The bitstream exercises various VCL NAL unit types.

Functional stage: Test decoding of various VCL NAL unit types.

Purpose: Check that the decoder can properly decode the VCL NAL unit types: TRAIL_N, TRAIL_R, TSA_N, TSA_R, STSA_N, STSA_R, RADL_R, RASL_R, RADL_N, RASL_N, BLA_W_LP, BLA_W_RADL, BLA_N_LP, IDR_W_RADL, IDR_N_LP, and CRA_NUT.

6.6.12.2 Test bitstream FILLER_A

Specification: All slices are coded as I or B slices. Each picture contains only one slice. This bitstream contains some NAL units with nal_unit_type equal to 38 (filler data) at the end of every access unit.

Functional stage: Test decoding with filler data NAL units present.

Purpose: Check that the decoder can properly decode a bitstream containing NAL units with the NAL unit type FD_NUT.

6.6.12.3 Test bitstream VPSID_A

Specification: All slices are coded as I or B slices. Each picture contains only one slice. This bitstream contains two VPSs with the correct one having the vps_video_parameter_set_id value 4. The bitstream has 3 temporal layers and the correct VPS has the vps_temporal_id_nesting_flag turned off.

Functional stage: Test decoding of vps_video_parameter_set_id.

Purpose: Check that the decoder properly decodes vps_video_parameter_set_id.

6.6.12.4 Test bitstream PS_B

Specification: All slices are coded as I or P slices. Each picture contains only one slice. sps_extension_flag is set equal to 1. Data is included after the sps_extension_flag.

Functional stage: Test decoding of VPS, SPS and PPS.

Purpose: Check that the decoder properly handles the extension_flag in SPS. This bitstream does not conform to the Main profile or Main 10 profile of the first edition of the high efficiency video coding specification since sps_extension_flag is equal to 1. However, Main profile and Main 10 profile decoders shall be able to decode this bitstream and ignore the SPS extension.

6.6.12.5 Test bitstream VPSSPSPPS_A

Specification: All slices are coded as I slices. The resolution of each picture is changed. All parameter sets are encoded at the beginning of bitstream prior to any pictures. Some of the parameter sets are duplicated and the order of parameter sets is arbitrary.

Functional stage: Test decoding of VPS, SPS and PPS.

Purpose: Check that the decoder properly handles VPS, SPS and PPS.

6.6.12.6 Test bitstream PPS_A

Specification: All slices are coded as I or B slices. Each picture contains only one slice. Bitstream includes multiple PPS signalling with random PPS parameters (constrained intra, transform skip, tile configurations, WP, loop filter, etc.) that get randomly selected by coded pictures.

Functional stage: Test decoding of PPS.

Purpose: Check that the decoder properly handles multiple PPSs being signalled.

6.6.12.7 Test bitstream SLPPLP_A

Specification: All slices are coded as I or B slices. Each picture contains only one slice. The bitstream contains three temporal layers (two sublayers).

Functional stage: Test decoding of sub_layer_profile_present_flag and sub_layer_level_present_flag.

Purpose: Check that the decoder properly handles sub_layer_profile_present_flag and sub_layer_level_present_flag.

6.6.12.8 Test bitstream OPFLAG_A

Specification: All slices are coded as I or B slices. Each picture contains only one slice. The value of output_flag_present_flag is 1, indicating that the syntax element pic_output_flag is signalled in the slice header.

Functional stage: Test parsing of output_flag_present_flag in the PPS. Test parsing of pic_output_flag in slice header syntax.

Purpose: Check that the decoder properly decodes slice headers containing pic_output_flag.

6.6.12.9 Test bitstream OPFLAG_B

Specification: All slices are coded as I slices. Each picture contains only one slice. The value of output_flag_present_flag is 1, indicating that the syntax element pic_output_flag is signalled in the slice header. Pictures with PicOrderCntVal equal to 39 and 73 are set to be not for output.

Functional stage: Test picture output.

Purpose: Check that the decoder properly decodes slice headers containing pic_output_flag.

6.6.12.10 Test bitstream OPFLAG_C

Specification: All slices are coded as I or B slices. Each picture contains only one slice. The value of output_flag_present_flag is 1, indicating that the syntax element pic_output_flag is signalled in the slice header. Picture with PicOrderCntVal equal to 20, 31, 56, and 72 are set to be not for output.

Functional stage: Test picture output.

Purpose: Check that the decoder properly decodes slice headers containing pic_output_flag.

6.6.12.11 Test bitstream NoOutPrior_A

Specification: All slices are coded as I or B slices. Each picture contains only one slice. The value of NoRaslOutputFlag and no_output_of_prior_pics_flag is equal to 1 when a CRA picture follows an end of sequence NAL unit. One CRA picture is included in the middle of the bitstream. An end of sequence NAL unit is present in the bitstream in the decoding order that is right before the CRA picture.

Functional stage: Test picture output.

Purpose: Check that the decoder properly decodes a CRA picture that occurs immediately after an end of sequence NAL unit which should result in all stored decoded pictures in the DPB to be removed without outputting them.

6.6.12.12 Test bitstream NoOutPrior_B

Specification: All slices are coded as I or B slices. Each picture contains only one slice. Two IDR pictures are included in the bitstream. The first IDR picture is the first picture in the bitstream. The second IDR picture is in the middle of the bitstream. The value of no_output_of_prior_pics_flag for the second IDR picture is set equal to 1.

Functional stage: Test picture output.

Purpose: Check that the decoder properly decodes an IDR picture with no_output_of_prior_pics_flag equal to 1 which should result in all stored decoded pictures in the DPB to be removed without outputting them.

6.6.12.13 Test bitstream PICSIZE_A

Specification: All slices are coded as I or B slices. Each picture contains only one slice. Each picture includes four tile columns.

Functional stage: Test picture size capability.

Purpose: Check that the decoder properly decodes pictures whose size is 1056x8440.

6.6.12.14 Test bitstream PICSIZE_B

Specification: All slices are coded as I or B slices. The bitstream is designed to test maximum height for level 5.1. Picture size is 8440x1056. Picture width is not a multiple of 16.

Functional stage: Test picture size capability.

Purpose: Check that the decoder properly decodes pictures whose size is 8440x1056 (picture width not a multiple of 16).

6.6.12.15 Test bitstream PICSIZE_C

Specification: All slices are coded as I or B slices. Each picture includes two tile columns.

Functional stage: Test picture size capability.

Purpose: Check that the decoder properly decodes pictures whose size is 528x4216.

6.6.12.16 Test bitstream PICSIZE_D

Specification: All slices are coded as I or B slices. The bitstream is designed to test maximum height for level 4.1. Picture size is 4216x528.

Functional stage: Test picture size capability.

Purpose: Check that the decoder properly decodes pictures whose size is 4216x528.

6.6.12.17 Test bitstream POC_A

Specification: All slices are coded as I or B slices. The bitstream is designed to test some rules related to PicOrderCntVal derivation.

Functional stage: Test PicOrderCntVal derivation process.

Purpose: Check that the decoder properly decodes different PicOrderCntVal values.

6.6.12.18 Test bitstream RAP_A

Specification: All slices are coded as I or B slices. Each picture contains only one slice. The first picture in the bitstream is a CRA picture and is followed by seven RASL pictures that are not decodable. There are two subsequent CRA pictures with RASL pictures, following the first CRA picture in this bitstream. These subsequent RASL pictures should be decodable since the associated CRA picture is not the first CRA picture in the bitstream.

Functional stage: Test reconstruction process starting with a CRA picture followed by RASL pictures that cannot be decoded of slices with inter RPS prediction.

Purpose: Check that the decoder properly decodes when the CRA picture is the first picture in the bitstream and is followed by RASL pictures that are not decodable.

6.6.12.19 Test bitstream RAP_B

Specification: All slices are coded as I or B slices. A CRA picture with leading pictures following end of sequence NAL unit is included. VPS, SPS and PPS are present in the bitstream repeatedly. The conformance window for cropping is signalled in the bitstream.

Functional stage: Test reconstruction process of a CRA picture.

Purpose: Check that the decoder properly decodes a CRA picture with leading pictures following an end of sequence NAL unit.

6.6.12.20 Test bitstream RPS_A

Specification: All slices are coded as I or B slices. Each picture contains only one slice. The slice header includes the inter RPS prediction method for sending the RPS for short-term pictures. The last three frames of this 44-frame sequence contain slice header RPS using the inter RPS in addition to the RPS sent in the PPS.

Functional stage: Test reconstruction process of slices with inter RPS prediction.

Purpose: Check that the decoder properly decodes slices using the inter RPS prediction method.

6.6.12.21 Test bitstream RPS_B

Specification: All slices are coded as I or B slices. Each picture contains only one slice. The bitstream includes random RPS signalling in slice headers along with random picture coding order within a series of pictures.

Functional stage: Test reconstruction process of slices without inter RPS prediction.

Purpose: Check that the decoder properly decodes slices using the inter RPS prediction method.

6.6.12.22 Test bitstream RPS_C

Specification: All slices are coded as I or P slices. Each picture contains only one slice. Two temporal layers are used. 15 reference pictures are used. The bitstream exercises short-term reference pictures in the RPS.

Functional stage: Test short-term RPS handling.

Purpose: Check that the decoder properly decodes when short-term picture handling is used in the RPS.

6.6.12.23 Test bitstream RPS_D

Specification: All slices are coded as I or P slices. Each picture contains only one slice. Two temporal layers are used. The bitstream exercises short-term and long-term reference pictures in the RPS.

Functional stage: Test long-term and short-term RPS handling.

Purpose: Check that the decoder properly decodes when long-term and short-term picture handling is used in the RPS.

6.6.12.24 Test bitstream RPS_E

Specification: All slices are coded as I or B slices. Each picture contains only one slice. The bitstream includes random RPS signalling with LTRPs in slice headers along with random picture coding order within a series of pictures.

Functional stage: Test reconstruction process of slices without inter RPS prediction.

Purpose: Check that the decoder properly decodes slices with long-term reference pictures without inter RPS prediction.

6.6.12.25 Test bitstream RPS_F

Specification: The inter RPS prediction signals some RPS entries that are not used by the current picture. (used_by_curr_pic_flag[j] equal to 0 and use_delta_flag[j] equal to 1).

Functional stage: Test reconstruction process of slices without inter RPS prediction.

Purpose: Check that the decoder properly decodes slices with the inter RPS prediction method including RPS entries that are not used by the current picture.

6.6.12.26 Test bitstream LTRPSPS_A

Specification: The bitstream is coded under typical random access conditions with the following modifications. Eight long-term reference picture candidates (four different slice_pic_order_cnt_lsb values and two values of used_by_curr_pic_lt_flag[i], giving a total of eight) are signalled in the SPS. The slice headers refer to long-term reference pictures that are either referred to from the SPS or may be explicitly signalled in the slice header. Reference picture list modification is applied on some pictures.

Functional stage: Test parsing of long_term_ref_pics_present_flag, num_long_term_ref_pics_sps, lt_ref_pic_poc_lsb_sps, and used_by_curr_pic_lt_sps_flag in SPS. Test parsing of num_long_term_sps and lt_idx_sps in slice header syntax.

Purpose: Check that the decoder can properly decode slice headers when long-term reference pictures from the list of candidates in the SPS are specified.

6.6.12.27 Test bitstream RPLM_A

Specification: All slices are coded as I or B slices. Each picture contains only one slice. The bitstream includes random reference picture list modification with varying list sizes.

Functional stage: Test reconstruction process of slices with reference list modification.

Purpose: Check that the decoder properly decodes slices with reference list modification.

6.6.12.28 Test bitstream RPLM_B

Specification: All slices are coded as I or B slices. Each picture contains more than one slice. The bitstream includes random reference picture list modification with varying list sizes.

Functional stage: Test reconstruction process of slices with reference list modification.

Purpose: Check that the decoder properly decodes slices with reference list modification.

6.6.12.29 Test bitstream SLICES_A

Specification: Each picture contains more than one slice with different slice type.

Functional stage: Test reconstruction process for pictures comprised of slices with different slice_type values.

Purpose: Check that the decoder properly decodes pictures comprised of slices with different slice_type values.

6.6.12.30 Test bitstream DSLICE_A

Specification: All slices are coded as I or B slices. Each picture contains more than one slice. dependent_slice_segments_enabled_flag is set equal to 1.

Functional stage: Test reconstruction process for independent and dependent slice segments.

Purpose: Check that the decoder properly decodes independent and dependent slice segments.

6.6.12.31 Test bitstream DSLICE_B

Specification: All slices are coded as I or B slices. Each picture contains more than one slice. dependent_slice_segments_enabled_flag is set equal to 1. entropy_coding_sync_enabled_flag is set equal to 1.

Functional stage: Test reconstruction process for dependent slice segments.

Purpose: Check that the decoder properly decodes dependent slice segments in combination with entropy coding synchronization.

6.6.12.32 Test bitstream DSLICE_C

Specification: All slices are coded as I or B slices. Each picture contains more than one slice. dependent_slice_segments_enabled_flag is set equal to 1. tiles_enabled_flag is set equal to 1.

Functional stage: Test reconstruction process for dependent slice segments.

Purpose: Check that the decoder properly decodes dependent slice segments in combination with tiles.

6.6.12.33 Test bitstream BUMPING_A

Specification: All slices are coded as I or B slices. Each picture contains more than one slice. All pictures with PicOrderCntVal values in the range of 0 to 65 are to be output except the pictures with PicOrderCntVal values equal to 4, 5, 6, 7, 15, 21, 22, 23, 30, 31, 36, 37, 38, 39, 54, 55, and 56. Those pictures are not output since they have not been output yet when IRAP pictures with no_output_of_prior_pics_flag equal to 1 are encountered in the bitstream. Four temporal layers are used and IRAP pictures with no_output_of_prior_pics_flag equal to 1 are present in the bitstream.

Functional stage: Test the bumping process.

Purpose: Check that the decoder properly handles tests output order conformance, in particular when applying the bumping process.

6.6.12.34 Test bitstream CONFWIN_A

Specification: All slices are coded as I or B slices. Each picture contains more than one slice. The value of conf_win_bottom_offset, conf_win_top_offset, conf_win_left_offset and conf_win_right_offset are set equal to 1.

Functional stage: Test conformance window usage.

Purpose: Check that the decoder properly handles conf_win_bottom_offset, conf_win_top_offset, conf_win_left_offset, and conf_win_right_offset.

6.6.12.35 Test bitstream HRD_A

Specification: All slices are coded as I or B slices. Each access unit contains four decoding units. sub_pic_hrd_params_present_flag is set equal to 1 and du_common_cpb_removal_delay_flag is set equal to 0.

Functional stage: Test decoding-unit-based HRD.

Purpose: Check that the decoder can properly decode with decoding-unit-based CPB removal time signalling.

6.6.12.36 Test bitstream EXT_A

Specification: A three-picture bitstream containing extension data in the VPS, SPS, PPS, and slice headers. Note that this bitstream is not a conforming bitstream, but conforming decoders are required to be able to parse it and decode the pictures in the bitstream.

Functional stage: Test decoding of bitstreams that contain extension data.

Purpose: Check that the decoder can properly handle extension data. This bitstream does not conform to the Main profile or Main 10 profile since extension data are present. However, Main profile and Main 10 profile decoders shall be able to decode this bitstream and ignore the extension data.

6.6.13 Test bitstreams – 10 bit

6.6.13.1 Test bitstream WP_A_MAIN10

Specification: All slices are coded as I or P slices. Each picture contains only one slice. bit_depth_luma_minus8 is set equal to 2 and bit_depth_chroma_minus8 is set equal to 2. weighted_pred_flag is set equal to 1. Plural reference indices are assigned to each reference picture.

Functional stage: Weighted sample prediction process for P slices with plural reference indices.

Purpose: Check that the decoder can properly decode weighted sample prediction for P slices with plural reference indices.

6.6.13.2 Test bitstream WP_B_MAIN10

Specification: All slices are coded as I, P or B slices. Each picture contains only one slice. bit_depth_luma_minus8 is set equal to 2 and bit_depth_chroma_minus8 is set equal to 2. weighted_pred_flag is set equal to 1 and weighted_bipred_flag is equal to 1. Plural reference indices are assigned to each reference picture.

Functional stage: Weighted sample prediction process for P and B slices with plural reference indices.

Purpose: Check that the decoder can properly decode weighted sample prediction for P and B slices with plural reference indices.

6.6.13.3 Test bitstream TSUNEQBD_A_MAIN10

Specification: Each picture contains only one slice. bit_depth_luma_minus8 is set equal to 2 and bit_depth_chroma_minus8 is set equal to 1. In the PPS, transform_skip_enabled_flag set equal to 1. In residual_coding(), transform_skip_flag set equal to 1 for Y, Cb, Cr for both intra and inter prediction modes.

Functional stage: Test the parsing and reconstruction process of slices with transform skip mode for luma and chroma with different bit depths.

Purpose: Check that the decoder can properly decode intra and inter prediction slices with transform skip and unequal bit depth (luma: 10-bit, chroma: 9-bit).

6.6.13.4 Test bitstream DBLK_A_MAIN10

Specification: All slices are coded as I or B slices. Each picture contains only one slice. bit_depth_luma_minus8 is set equal to 2 and bit_depth_chroma_minus8 is set equal to 2. Some QP values are set to negative values.

Functional stage: Test deblocking filter process for 10 bit video.

Purpose: Check that the decoder can properly decode negative values of QP that affect the deblocking filter process.

6.6.13.5 Test bitstream INITQP_B_MAIN10

Specification: All slices are coded as I or B slices. bit_depth_luma_minus8 is set equal to 2 and bit_depth_chroma_minus8 is set equal to 2. The value of init_qp_minus26 is set from −38 to 25.

Functional stage: Test the initial QP.

Purpose: Check that the decoder can properly decode different init_qp_minus26 values.

6.6.13.6 Test bitstream WPP_A_MAIN10

Specification: bit_depth_luma_minus8 is set equal to 2 and bit_depth_chroma_minus8 is set equal to 2. entropy_coding_sync_enabled_flag is set equal to 1. A luma CTB size of 64x64 is used. The bitstream contains six repeated patterns of pictures with a particular ordering and referencing relationship, which are each eight pictures long. The first three of these groups of eight pictures have pictures with the following characteristics:

- One slice in the frame, QP is constant.
- One slice in the frame, the QP of each CU is set equal to a value such that $\text{Abs}(QP - SliceQp_Y) > 2$.
- Maximum number of independent slice segments in the frame, at least one slice segment is one CTU long, at least one slice segment is two CTUs long, QP is constant.
- Maximum number of independent slice segments in the frame, at least one slice segment is one CTU long, at least one slice segment is two CTUs long, the QP of each CU is set equal to a value such that $\text{Abs}(QP - SliceQp_Y) > 2$.
- Maximum number of dependent slice segments in the frame, at least one dependent slice segment is one CTU long, at least one dependent slice segment is two CTUs long, QP is constant.
- Maximum number of dependent slice segments in the frame, at least one dependent slice segment is one CTU long, at least one dependent slice segment is two CTUs long, the QP of each CU is set equal to a value such that $\text{Abs}(QP - SliceQp_Y) > 2$.
- Random combination of independent/dependent slice segments, QP is constant.
- Random combination of independent/dependent slice segments, the QP of each CU is set equal to a value such that $\text{Abs}(QP - SliceQp_Y) > 2$.

The first of these groups of eight pictures is coded using all intra CUs, and the second is coded with CU skipping disabled. The final three of these groups of eight pictures feature a mixture of single slice pictures, and pictures coded using multiple slices and multiple slice segments. Random amounts of slice segment header extension bytes are encoded in each slice header.

Functional stage: Tests that entropy coding is correctly synchronized with different slice types. Tests that the QP predictor is reset to $SliceQp_Y$ at the beginning of every CTU row. May be used to test handling of entry points by a parallel decoder.

Purpose: Check that the decoder properly decodes when entropy coding synchronization is enabled and can handle entry points when slice segment header extension data bytes are present. This bitstream does not conform to the Main 10 profile since slice_segment_header_extension_present_flag is equal to 1. However, Main 10 profile decoders shall be able to decode this bitstream and ignore the slice segment header extension data bytes.

6.6.13.7 Test bitstream WPP_B_MAIN10

Specification: bit_depth_luma_minus8 is set equal to 2 and bit_depth_chroma_minus8 is set equal to 2. entropy_coding_sync_enabled_flag is set equal to 1. A luma CTB size of 32x32 is used. The bitstream contains six repeated patterns of pictures with a particular ordering and referencing relationship, which are each eight pictures long. The first three of these groups of eight pictures have pictures with the following characteristics:

- One slice in the frame, QP is constant.
- One slice in the frame, the QP of each CU is set equal to a value such that $\text{Abs}(QP - SliceQp_Y) > 2$.
- Maximum number of independent slice segments in the frame, at least one slice segment is one CTU long, at least one slice segment is two CTUs long, QP is constant.
- Maximum number of independent slice segments in the frame, at least one slice segment is one CTU long, at least one slice segment is two CTUs long, the QP of each CU is set equal to a value such that $\text{Abs}(QP - SliceQp_Y) > 2$.
- Maximum number of dependent slice segments in the frame, at least one dependent slice segment is one CTU long, at least one dependent slice segment is two CTUs long, QP is constant.
- Maximum number of dependent slice segments in the frame, at least one dependent slice segment is one CTU long, at least one dependent slice segment is two CTUs long, the QP of each CU is set equal to a value such that $\text{Abs}(QP - SliceQp_Y) > 2$.
- Random combination of independent/dependent slice segments, QP is constant.
- Random combination of independent/dependent slice segments, the QP of each CU is set equal to a value such that $\text{Abs}(QP - SliceQp_Y) > 2$.

The first of these groups of eight pictures is coded using all intra CUs, and the second is coded with CU skipping disabled. The final three of these groups of eight pictures feature a mixture of single slice pictures, and pictures coded using multiple slices and multiple slice segments. Random amounts of slice segment header extension bytes are encoded in each slice header.

Functional stage: Tests that entropy coding is correctly synchronized with different slice types. Tests that the QP predictor is reset to $SliceQp_Y$ at the beginning of every CTU row. May be used to test handling of entry points by a parallel decoder.

Purpose: Check that the decoder properly decodes when entropy coding synchronization is enabled and handle entry points when slice segment header extension data bytes are present. This bitstream does not conform to the Main 10 profile since slice_segment_header_extension_present_flag is equal to 1. However, Main 10 profile decoders shall be able to decode this bitstream and ignore the slice segment header extension data bytes.

6.6.13.8 Test bitstream WPP_C_MAIN10

Specification: bit_depth_luma_minus8 is set equal to 2 and bit_depth_chroma_minus8 is set equal to 2. entropy_coding_sync_enabled_flag is set equal to 1. A luma CTB size of 16x16 is used. The bitstream contains six repeated patterns of pictures with a particular ordering and referencing relationship, which are each eight pictures long. The first three of these groups of eight pictures have pictures with the following characteristics:

- One slice in the frame, QP is constant.
- One slice in the frame, the QP of each CU is set equal to a value such that $Abs(QP - SliceQp_Y) > 2$.
- Maximum number of independent slice segments in the frame, at least one slice segment is one CTU long, at least one slice segment is two CTUs long, QP is constant.
- Maximum number of independent slice segments in the frame, at least one slice segment is one CTU long, at least one slice segment is two CTUs long, the QP of each CU is set equal to a value such that $Abs(QP - SliceQp_Y) > 2$.
- Maximum number of dependent slice segments in the frame, at least one dependent slice segment is one CTU long, at least one dependent slice segment is two CTUs long, QP is constant.
- Maximum number of dependent slice segments in the frame, at least one dependent slice segment is one CTU long, at least one dependent slice segment is two CTUs long, the QP of each CU is set equal to a value such that $Abs(QP - SliceQp_Y) > 2$.
- Random combination of independent/dependent slice segments, QP is constant.
- Random combination of independent/dependent slice segments, the QP of each CU is set equal to a value such that $Abs(QP - SliceQp_Y) > 2$.

The first of these groups of eight pictures is coded using all intra CUs, and the second is coded with CU skipping disabled. The final three of these groups of eight pictures feature a mixture of single slice pictures, and pictures coded using multiple slices and multiple slice segments. Random amounts of slice segment header extension bytes are encoded in each slice header.

Functional stage: Tests that entropy coding is correctly synchronized with different slice types. Tests that the QP predictor is reset to $SliceQp_Y$ at the beginning of every CTU row. May be used to test handling of entry points by a parallel decoder.

Purpose: Check that the decoder properly decodes when entropy coding synchronization is enabled and handle entry points when slice segment header extension data bytes are present. This bitstream does not conform to the Main 10 profile since slice_segment_header_extension_present_flag is equal to 1. However, Main 10 profile decoders shall be able to decode this bitstream and ignore the slice segment header extension data bytes.

6.6.13.9 Test bitstream WPP_D_MAIN10

Specification: bit_depth_luma_minus8 is set equal to 2 and bit_depth_chroma_minus8 is set equal to 2. entropy_coding_sync_enabled_flag is set equal to 1. A luma CTB size of 64x64 is used. The picture is one CTU wide. The bitstream contains six repeated patterns of pictures with a particular ordering and referencing relationship, which are each eight pictures long. Each of these groups of eight pictures are encoded with varying numbers of slices and slice segments. Even frames have fixed QP, while the QP established at the CU level in odd frames is set such that $Abs(QP - SliceQp_Y) > 2$. The first of these groups of eight pictures is coded using all intra CUs, and the second is coded with CU skipping disabled. Random amounts of slice segment header extension bytes are encoded in each slice header.

Functional stage: Tests that entropy coding is correctly synchronized when a picture is one CTU wide. Tests that the QP predictor is reset to $SliceQp_Y$ at the beginning of every CTU row. May be used to test handling of entry points by a parallel decoder.

Purpose: Check that the decoder properly decodes when a picture is one CTU wide and handle entry points when slice segment header extension data bytes are present. This bitstream does not conform to the Main 10 profile since slice_segment_header_extension_present_flag is equal to 1. However, Main 10 profile decoders shall be able to decode this bitstream and ignore the slice segment header extension data bytes.

6.6.13.10 Test bitstream WPP_E_MAIN10

Specification: bit_depth_luma_minus8 is set equal to 2 and bit_depth_chroma_minus8 is set equal to 2. entropy_coding_sync_enabled_flag is set equal to 1. A luma CTB size of 64x64 is used. The picture is two CTUs wide. The bitstream contains six repeated patterns of pictures with a particular ordering and referencing relationship, which are each eight pictures long. These are encoded with varying numbers of slices and slice segments. Even frames have fixed QP, while the QP established at the CU level in odd frames is set such that $\text{Abs}(\text{QP} - \text{SliceQp}_Y) > 2$. The first of these groups of eight pictures is coded using all intra CUs, and the second is coded with CU skipping disabled. Random amounts of slice segment header extension bytes are encoded in each slice header.

Functional stage: Tests that entropy coding is correctly synchronized when a picture is two CTUs wide. Tests that the QP predictor is reset to SliceQp_Y at the beginning of every CTU row. May be used to test handling of entry points by a parallel decoder.

Purpose: Check that the decoder properly decodes when a picture is two CTUs wide and handle entry points when slice segment header extension data bytes are present. This bitstream does not conform to the Main 10 profile since slice_segment_header_extension_present_flag is equal to 1. However, Main 10 profile decoders shall be able to decode this bitstream and ignore the slice segment header extension data bytes.

6.6.13.11 Test bitstream WPP_F_MAIN10

Specification: bit_depth_luma_minus8 is set equal to 2 and bit_depth_chroma_minus8 is set equal to 2. entropy_coding_sync_enabled_flag is set equal to 1. A luma CTB size of 64x64 is used. The picture is three CTUs wide. The bitstream contains six repeated patterns of pictures with a particular ordering and referencing relationship, which are each eight pictures long. These are encoded with varying numbers of slices and slice segments. Even frames have fixed QP, while the QP established at the CU level in odd frames is set such that $\text{Abs}(\text{QP} - \text{SliceQp}_Y) > 2$. The first of these groups of eight pictures is coded using all intra CUs, and the second is coded with CU skipping disabled. Random amounts of slice segment header extension bytes are encoded in each slice header.

Functional stage: Tests that entropy coding is correctly synchronized when a picture is three CTUs wide. Tests that the QP predictor is reset to SliceQp_Y at the beginning of every CTU row. May be used to test handling of entry points by a parallel decoder.

Purpose: Check that the decoder properly decodes when a picture is three CTUs wide and handle entry points when slice segment header extension data bytes are present. This bitstream does not conform to the Main 10 profile since slice_segment_header_extension_present_flag is equal to 1. However, Main 10 profile decoders shall be able to decode this bitstream and ignore the slice segment header extension data bytes.

6.6.14 Test bitstreams – MV-HEVC

6.6.14.1 Test bitstream MVHEVCS-A

Specification: All slices are coded as I, P or B slices. Only the first picture of each view is coded as an Instantaneous Decoding Refresh (IDR) picture. Each picture contains only one slice. NumViews is equal to 2. NumDirectRefLayers of the non-base view is equal to 1. For each picture in the non-base view, inter-view prediction is enabled. The two views are with the same spatial resolution. All NAL units are encapsulated into the byte stream format specified in Rec. ITU-T H.265 | ISO/IEC 23008-2.

The number of layers is 2, the resolution is 1024x768 and the number of frames is 48.

The output layer sets are specified as follows:

– OLS_0 – layer: 0, PTL idx: 1 (Main 4), output_layer_flag: 1

– OLS_1 – layer: 0, PTL idx: 1 (Main 4), output_layer_flag: 1
layer: 1, PTL idx: 2 (Multiview Main 4), output_layer_flag: 1

Functional stage: Decoding of two views with inter-view prediction and inter-prediction.

Purpose: To test the most normal case for MV-HEVC.

6.6.14.2 Test bitstream MVHEVCS-B

Specification: All slices of the base view are coded as I slices and all slices of the non-base view are coded with intra prediction or inter-view prediction, thus are P slices. Only the first picture of each view is coded as an IDR picture. Each picture contains only one slice. NumViews is equal to 2. NumDirectRefLayers of the non-base view is equal to 1. The two views are with the same spatial resolution. All NAL units are encapsulated into the byte stream format specified in Rec. ITU-T H.265 | ISO/IEC 23008-2.

The number of layers is 2, the resolution is 1024x768 and the number of frames is 64.

The output layer sets are specified as follows:

– OLS_0 – layer: 0, PTL idx: 1 (Main 4), output_layer_flag: 1
– OLS_1 – layer: 0, PTL idx: 1 (Main 4), output_layer_flag: 1
layer: 1, PTL idx: 2 (Multiview Main 4), output_layer_flag: 1

Functional stage: Decoding of two views with only inter-view prediction and intra-prediction.

Purpose: To test the all intra base view case.

6.6.14.3 Test bitstream MVHEVCS-C

Specification: All slices are coded as I, P or B slices. Only the first picture of each view is coded as an IDR picture. Each picture contains only one slice. NumViews is equal to 2. NumDirectRefLayers is always equal to 0, meaning inter-view prediction is disabled. The two views are with the same spatial resolution. All NAL units are encapsulated into the byte stream format specified in Rec. ITU-T H.265 | ISO/IEC 23008-2.

The number of layers is 2, the resolution is 1024x768 and the number of frames is 30.

The output layer sets are specified as follows:

– OLS_0 – layer: 0, PTL idx: 1 (Main 4), output_layer_flag: 1
– OLS_1 – layer: 0, PTL idx: 1 (Main 4), output_layer_flag: 1
layer: 1, PTL idx: 2 (Multiview Main 4), output_layer_flag: 1

Functional stage: Decoding of two views with only inter-prediction and intra-prediction.

Purpose: To test the simulcast case.

6.6.14.4 Test bitstream MVHEVCS-D

Specification: All slices are coded as I, P or B slices. Only the first picture of each view is coded as an IDR picture. Each picture contains only one slice. NumViews is equal to 2. NumDirectRefLayers is always equal to 0, meaning inter-view prediction is disabled. The two views are with different spatial resolutions. All NAL units are encapsulated into the byte stream format specified in Rec. ITU-T H.265 | ISO/IEC 23008-2.

The number of layers is 2, the resolution is 1024x768 for layer 0 and 512x384 for layer 1, and the number of frames is 25.

The output layer sets are specified as follows:

– OLS_0 – layer: 0, PTL idx: 1 (Main 5.1), output_layer_flag: 1
– OLS_1 – layer: 0, PTL idx: 1 (Main 5.1), output_layer_flag: 1
layer: 1, PTL idx: 2 (Multiview Main 5.1), output_layer_flag: 1

Functional stage: Decoding of two views with only inter-prediction and intra-prediction.

Purpose: To test the case when different spatial resolutions for two views are used.

6.6.14.5 Test bitstream MVHEVCS-E

Specification: All slices are coded as I, P or B slices. Only the first picture of each view is coded as an IDR picture. In addition, every two group of pictures start with an access unit which contains pictures that are all clean random access (CRA) pictures. Each picture contains only one slice. NumViews is equal to 2. NumDirectRefLayers of the non-base view is equal to 1. For each picture in the non-base view, inter-view prediction is enabled. The two views are with the same spatial resolution. All NAL units are encapsulated into the byte stream format specified in Rec. ITU-T H.265 | ISO/IEC 23008-2.

The number of layers is 2, the resolution is 1024x768 and the number of frames is 48.

The output layer sets are specified as follows:

– OLS_0 – layer: 0, PTL idx: 1 (Main 4), output_layer_flag: 1
– OLS_1 – layer: 0, PTL idx: 1 (Main 4), output_layer_flag: 1
layer: 1, PTL idx: 2 (Multiview Main 4), output_layer_flag: 1

Functional stage: Decoding of two views with inter-view prediction and inter-prediction.

Purpose: To test the random access hierarchical B case.

6.6.14.6 Test bitstream MVHEVCS-F

Specification: All slices are coded as I, P or B slices. Only the first picture of each view is coded as an IDR picture. Thus the first access unit is an IRAP access unit. In addition, for each of every two following group of pictures, an access unit which contains pictures that are all CRA pictures are requested. Other pictures are non-IRAP pictures. Each picture contains only one slice. NumViews is equal to 2. NumActiveRefLayerPics is equal to 1 for each picture in an IRAP access unit of the non-base view, and 0 otherwise, i.e., only for each picture in the non-base view in the IRAP access unit, inter-view prediction is enabled. The two views are with the same spatial resolution. All NAL units are encapsulated into the byte stream format specified in Rec. ITU-T H.265 | ISO/IEC 23008-2.

The number of layers is 2, the resolution is 1024x768 and the number of frames is 48.

The output layer sets are specified as follows:

- OLS_0 – layer: 0, PTL idx: 1 (Main 4), output_layer_flag: 1
- OLS_1 – layer: 0, PTL idx: 1 (Main 4), output_layer_flag: 1
  layer: 1, PTL idx: 2 (Multiview Main 4), output_layer_flag: 1

Functional stage: Decoding of two views with inter-view prediction only for random access points.

Purpose: To test the flexibility of inter-view prediction applicability scope (inter-view prediction only for IRAP pictures).

6.6.14.7 Test bitstream MVHEVCS-G

Specification: All slices are coded as I, P or B slices. Only the first picture of each view is coded as an IDR picture. Thus the first access unit is an IRAP access unit. In addition, every two group of pictures start with an access unit which contains pictures that are all CRA pictures. Each picture contains only one slice. NumViews is equal to 3. NumDirectRefLayers of the non-base view is equal to 1. For each picture in the non-base view, inter-view prediction is enabled. The three views are with the same spatial resolution. All NAL units are encapsulated into the byte stream format specified in Rec. ITU-T H.265 | ISO/IEC 23008-2.

The number of layers is 3, the resolution is 1024x768 and the number of frames is 49.

The output layer sets are specified as follows:

- OLS_0 – layer: 0, PTL idx: 1 (Main 5.1), output_layer_flag: 1
- OLS_1 – layer: 0, PTL idx: 1 (Main 5.1), output_layer_flag: 1
  layer: 1, PTL idx: 2 (Multiview Main 5.1), output_layer_flag: 1
- OLS_2 – layer: 0, PTL idx: 1 (Main 5.1), output_layer_flag: 1
  layer: 1, PTL idx: 2 (Multiview Main 5.1), output_layer_flag: 1
- OLS_3 – layer: 0, PTL idx: 1 (Main 5.1), output_layer_flag: 1
  layer: 1, PTL idx: 2 (Multiview Main 5.1), output_layer_flag: 1
  layer: 2, PTL idx: 2 (Multiview Main 5.1), output_layer_flag: 1

Functional stage: Decoding of three views with inter-view prediction and inter-prediction.

Purpose: To test the random access hierarchical B case (random access 3-view case, PIP configuration: wherein the middle view is the base view and each of the left and right views depends only on the base view).

6.6.14.8 Test bitstream MVHEVCS-H

Specification: All slices are coded as I, P or B slices. Only the first picture of each view is coded as an IDR picture. Thus the first access unit is an IRAP access unit. In addition, every two group of pictures start with an access unit which contains pictures that are all CRA pictures. Each picture contains only one slice. NumViews is equal to 3. NumDirectRefLayers of the first non-base view is equal to 1 and NumDirectRefLayers of the second non-base view is equal to 2. For each picture in the non-base view, inter-view prediction is enabled. The three views are with the same spatial resolution. All NAL units are encapsulated into the byte stream format specified in Rec. ITU-T H.265 | ISO/IEC 23008-2.

The number of layers is 3, the resolution is 1920x1088 and the number of frames is 50.

The output layer sets are specified as follows:

- OLS_0 – layer: 0, PTL idx: 1 (Main 5.1), output_layer_flag: 1
- OLS_1 – layer: 0, PTL idx: 1 (Main 5.1), output_layer_flag: 1
  layer: 1, PTL idx: 2 (Multiview Main 5.1), output_layer_flag: 1

– OLS_2 – layer: 0, PTL idx: 1 (Main 5.1), output_layer_flag: 1
layer: 1, PTL idx: 2 (Multiview Main 5.1), output_layer_flag: 1
layer: 2, PTL idx: 2 (Multiview Main 5.1), output_layer_flag: 1

Functional stage: Decoding of three views with inter-view prediction and inter-prediction.

Purpose: To test the random access hierarchical B case (random access 3-view case, IBP configuration: wherein the left view is the base view, right view depends only on the base view and the middle view depends on both left view and right view).

6.6.14.9 Test bitstream MVHEVCS-I

Specification: All slices are coded as I, P or B slices. Only the first picture of each view is coded as an IDR picture. Thus the first access unit is an IRAP access unit. In addition, every two group of pictures start with an access unit which contains pictures that are all CRA pictures. Each picture contains only one slice. NumViews is equal to 3. NumDirectRefLayers of the first non-base view is equal to 1 and NumDirectRefLayers of the second non-base view is equal to 2. The P view is coded with viewOrderIdx equal to 1. For each picture in the non-base view, inter-view prediction is enabled. Each view contains both texture and depth and the views are with the same spatial resolution. All NAL units are encapsulated into the byte stream format specified in Rec. ITU-T H.265 | ISO/IEC 23008-2. The vertical constraint of the inter-view motion vector applies only to the P view (for both texture layer and depth layer of this view).

The number of layers is 6, the resolution is 1024x768 and the number of frames is 9.

The output layer sets are specified as follows:

– OLS_0 – layer: 0, PTL idx: 1 (Main 5.1), output_layer_flag: 1, inbld_flag: 0

– OLS_1 – layer: 0, PTL idx: 1 (Main 5.1), output_layer_flag: 1, inbld_flag: 0
layer: 1, PTL idx: 2 (Multiview Main 5.1), output_layer_flag: 1, inbld_flag: 0
layer: 2, PTL idx: 2 (Multiview Main 5.1), output_layer_flag: 1, inbld_flag: 0

– OLS_2 – layer: 0, PTL idx: 1 (Main 5.1), output_layer_flag: 1, inbld_flag: 0
layer: 1, PTL idx: 2 (Multiview Main 5.1), output_layer_flag: 1, inbld_flag: 0
layer: 2, PTL idx: 2 (Multiview Main 5.1), output_layer_flag: 1, inbld_flag: 0
layer: 3(auxiliary depth), PTL idx: 3 (Main 5.1), output_layer_flag: 1, inbld_flag: 1
layer: 4(auxiliary depth), PTL idx: 2 (Multiview Main 5.1), output_layer_flag: 1, inbld_flag: 0
layer: 5(auxiliary depth), PTL idx: 2 (Multiview Main 5.1), output_layer_flag: 1, inbld_flag: 0

Functional stage: Decoding of three views with inter-view prediction and inter-prediction.

Purpose: To test the auxiliary picture case.

6.6.15 Test bitstreams – 3D-HEVC

all intra base view configuration: A bitstream in this configuration contains two or three views, each containing one texture view and one depth view. All slices of the base view are coded as I slices and all slices of the non-base view may be coded with only intra prediction or inter-view prediction, thus I or P slices. Only the first picture of each view is coded as an IDR picture. Each picture contains only one slice. NumDirectRefLayers of the non-base views is equal to 1. All texture and depth views are with the same spatial resolution. All NAL units are encapsulated into the byte stream format specified in Rec. ITU-T H.265 | ISO/IEC 23008-2.

random access base view configuration: A bitstream in this configuration contains two or three views, each containing one texture view and one depth view. All slices are coded as I, P or B slices. Only the first picture of each view is coded as an IDR picture. In addition, every two group of pictures start with an access unit which contains pictures that are all CRA pictures. Each picture contains only one slice. NumDirectRefLayers of the non-base view is equal to 1. For each picture in a non-base view, inter-view prediction may be enabled. All texture and depth views are with the same spatial resolution. All NAL units are encapsulated into the byte stream format specified in Rec. ITU-T H.265 | ISO/IEC 23008-2.

3-view configuration: A bitstream in this configuration contains three views, each containing one texture view and one depth view. NumViews is equal to 3. It can be either all intra base view configuration or random access base view configuration.

2-view configuration: A bitstream in this configuration contains two views, each containing one texture view and one depth view. NumViews is equal to 2. It can be either all intra base view configuration or random access base view configuration.

6.6.15.1 Test bitstreams – Texture tools

In this category, test bitstreams are tested with the following default tools enabled. The default tools include inter-view sample prediction, Neighboring Block based Disparity Vector (NBDV, without accessing depth views for disparity vector derivation), and the merge candidate list construction by including additional merge candidates for texture views. The intra prediction tools for depth views are disabled. The inter-component prediction for both the depth dependent texture tools and the texture dependent depth tool are disabled. Specifically, depth refined disparity vector derivation (DoNBDV, as specified by depth_ref_enabled_flag), View Synthesis Prediction (VSP, as specified by vsp_mc_enabled_flag) and Depth Predicted Sub-block Partitioning (DBBP, as specified by dbbp_enabled_flag) are disabled.

In the test bistreams of this category, unless specified explicitly, the depth views are coded with MV-HEVC mechanisms (i.e., only inter-view sample prediction and inter-layer motion prediction). Inter-view sample prediction is enabled by allowing an inter-view reference picture to be put into a reference picture list of the current picture.

6.6.15.1.1 Test bitstream 3DHC_T_A

Specification: The bitstream is coded with random access base view configuration as well as 3-view configuration. In addition to inter-view motion prediction, advanced residual prediction (ARP) is enabled (with iv_res_pred_enabled_flag set equal to 1) in the non-base texture views.

The number of layers is 6, the resolution is 1024x768 and the number of frames is 48.

Functional stage: Decoding of three views with inter-view motion prediction and advanced residual prediction.

Purpose: To test the ARP.

6.6.15.1.2 Test bitstream 3DHC_T_B

Specification: The bitstream is coded with random access base view configuration as well as 3-view configuration. In addition to inter-view motion prediction, sub-PU inter-view motion prediction is enabled (with log2_ivmc_sub_pb_size_minus3 less than 3) in the non-base texture views.

The number of layers is 6, the resolution is 1024x768 and the number of frames is 50.

Functional stage: Decoding of three views with sub-PU inter-view motion prediction.

Purpose: To test the sub-PU inter-view motion prediction.

6.6.15.1.3 Test bitstream 3DHC_T_C

Specification: The bitstream is coded with random access base view configuration as well as 3-view configuration. In addition to inter-view motion prediction, illumination compensation is enabled (with slice_ic_enabled_flag set equal to 1) in the non-base texture views.

The number of layers is 6, the resolution is 1024x768 and the number of frames is 64.

Functional stage: Decoding of three views with illumination compensation.

Purpose: To test the illumination compensation.

6.6.15.1.4 Test bitstream 3DHC_T_D

Specification: The bitstream is coded with random access base view configuration as well as 3-view configuration. In addition to inter-view motion prediction, ARP, sub-PU inter-view motion prediction and illumination compensation are enabled in the non-base texture views.

The number of layers is 6, the resolution is 1024x768 and the number of frames is 64.

Functional stage: Decoding of three views with all texture coding tools enabled.

Purpose: To test the combined texture coding tools.

6.6.15.1.5 Test bitstream 3DHC_T_E

Specification: The bitstream contains three views, each containing only one texture view but no depth view. All slices are coded as I, P or B slices. Only the first picture of each view is coded as an IDR picture. In addition, every two group of pictures start with an access unit which contains pictures that are all CRA pictures. Each picture contains only one slice. NumDirectRefLayers of the non-base view is equal to 1. For each picture in a non-base view, inter-view prediction are enabled, with inter-view motion prediction, ARP, sub-PU inter-view motion prediction and illumination compensation. All NAL units are encapsulated into the byte stream format specified in Rec. ITU-T H.265 | ISO/IEC 23008-2.

The number of layers is 6 and the resolution is 1920x1088 and the number of frames is 60.

Functional stage: Decoding of three texture views only with all texture coding tools enabled.

Purpose: To test the texture coding tools for texture only bitstream.

6.6.15.2 Test bitstreams – Depth tools

In this category, test bitstreams are tested with the default tools enabled for texture coding. The default tools include inter-view sample prediction, NBDV, and the merge candidate list construction. The intra prediction tools for depth views are disabled by default. The inter-component prediction for both the depth dependent texture tools (DoNBDV, VSP and DBBP) and the texture dependent depth tools are disabled. Depth views are coded by default with MV-HEVC mechanisms (i.e., only inter-view sample prediction).

Depth intra coding tools are tested in both random access base view configuration and all intra base view configuration.

Depth inter coding tools are tested in random access base view configuration.

6.6.15.2.1 Test bitstream 3DHC_D1_A

Specification: The bitstream is coded with random access base view configuration as well as 3-view configuration. In addition to inter-view sample prediction for depth, depth intra wedge prediction is enabled (with intra_dc_only_wedge_enabled_flag set equal to 1,no_dim_flag set equal to 0 and depth_intra_mode_idx_flag set equal to 0) for depth views.

The number of layers is 6, the resolution is 1920x1088 and the number of frames is 30.

Functional stage: Decoding of three views with depth intra wedge prediction enabled for depth.

Purpose: To test the depth intra wedge prediction in random access base view configuration.

6.6.15.2.2 Test bitstream 3DHC_D1_B

Specification: The bitstream is coded with all intra base view configuration as well as 3-view configuration. depth intra wedge prediction is enabled (with intra_dc_only_wedge_enabled_flag set equal to 1, no_dim_flag set equal to 0 and depth_intra_mode_idx_flag set equal to 0) for depth views.

The number of layers is 6, the resolution is 1920x1088 and the number of frames is 3.

Functional stage: Decoding of three views with depth intra wedge prediction enabled for depth.

Purpose: To test the depth intra wedge prediction in all intra base view configuration.

6.6.15.2.3 Test bitstream 3DHC_D1_C

Specification: The bitstream is coded with random access base view configuration as well as 3-view configuration. In addition to inter-view sample prediction for depth, intra Segmental DC (SDC) mode is enabled (with intra_dc_only_wedge_enabled_flag set equal to 1 and no_dim_flag set equal to 1) for depth views.

The number of layers is 6, the resolution is 1024x768 and the number of frames is 50.

Functional stage: Decoding of three views with intra SDC enabled for depth.

Purpose: To test the intra SDC in random access base view configuration.

6.6.15.2.4 Test bitstream 3DHC_D1_D

Specification: The bitstream is coded with all intra base view configuration as well as 3-view configuration. In addition, intra Segmental DC (SDC) mode is enabled for depth views.

The number of layers is 6, the resolution is 1024x768 and the number of frames is 50.

Functional stage: Decoding of three views with intra SDC enabled for depth.

Purpose: To conform the intra SDC in all intra base view configuration.

6.6.15.2.5 Test bitstream 3DHC_D1_E

Specification: The bitstream is coded with random access base view configuration as well as 3-view configuration. In addition to inter-view sample prediction for depth, intra skip mode (with skip_intra_enabled_flag set equal to 1) is enabled for depth views.

The number of layers is 6, the resolution is 1920x1088 and the number of frames is 30.

Functional stage: Decoding of three views with depth intra skip mode enabled for depth.

Purpose: To test the depth intra skip mode in random access base view configuration.

6.6.15.2.6 Test bitstream 3DHC_D1_F

Specification: The bitstream is coded with all intra base view configuration as well as 3-view configuration. In addition, intra skip mode is enabled for depth views.

The number of layers is 6, the resolution is 1920x1088 and the number of frames is 30.

Functional stage: Decoding of three views with depth intra skip mode enabled for depth.

Purpose: To test the depth intra skip mode in all intra base view configuration.

6.6.15.2.7 Test bitstream 3DHC_D1_G

Specification: The bitstream is coded with random access base view configuration as well as 3-view configuration. In addition to inter-view sample prediction for depth, the all depth intra tools are enabled together with depth lookup table (DLT, with dlt_flag set equal to 1).

The number of layers is 6, the resolution is 1024x768 and the number of frames is 64.

Functional stage: Decoding of three views with depth intra coding tools enabled.

Purpose: To test the depth Intra coding tools in random access base view configuration.

6.6.15.2.8 Test bitstream 3DHC_D1_H

Specification: The bitstream is coded with all intra base view configuration as well as 3-view configuration. In addition, the all depth Intra tools are enabled together with DLT.

The number of layers is 6, the resolution is 1024x768 and the number of frames is 64.

Functional stage: Decoding of three views with depth intra coding tools enabled.

Purpose: To test the depth intra coding tools in all intra base view configuration.

6.6.15.2.9 Test bitstream 3DHC_D2_A

Specification: The bitstream is coded with random access base view configuration as well as 3-view configuration. In addition to inter-view sample prediction for depth, the inter-view motion prediction is enabled for depth views.

The number of layers is 6, the resolution is 1920x1088 and the number of frames is 60.

Functional stage: Decoding of three views with inter-view motion prediction for depth views.

Purpose: To test the inter-view motion prediction for depth.

6.6.15.2.10 Test bitstream 3DHC_D2_B

Specification: The bitstream is coded with random access base view configuration as well as 3-view configuration. In addition to inter-view sample prediction for depth, the inter Segmental SDC (inter SDC) is enabled (with inter_dc_only_enabled_flag set equal to 1) for depth views.

The number of layers is 6, the resolution is 1920x1088 and the number of frames is 50.

Functional stage: Decoding of three views with inter SDC enabled for depth views.

Purpose: To test the inter SDC for depth.

6.6.15.3 Test bitstreams – Depth dependent texture tools

6.6.15.3.1 Test bitstream 3DHC_DT_A

Specification: The bitstream is coded with random access base view configuration as well as 3-view configuration. In addition to ARP, sub-PU inter-view motion prediction and illumination compensation, depth refined disparity vector derivation (DoNBDV) is enabled (with depth_ref_enabled_flag set equal to 1) for non-base texture views.

The number of layers is 6, the resolution is 1024x768 and the number of frames is 60.

Functional stage: Decoding of three views with DoNBDV enabled for texture views.

Purpose: To test the DoNBDV for texture.

6.6.15.3.2 Test bitstream 3DHC_DT_B

Specification: The bitstream is coded with random access base view configuration as well as 3-view configuration. In addition to ARP, sub-PU inter-view motion prediction and illumination compensation, VSP is enabled (with vsp_mc_enabled_flag set equal to 1) for non-base texture views.

The number of layers is 6, the resolution is 1024x768 and the number of frames is 25.

Functional stage: Decoding of three views with VSP enabled for texture views.

Purpose: To test the VSP for texture.

6.6.15.3.3 Test bitstream 3DHC_DT_C

Specification: The bitstream is coded with random access base view configuration as well as 3-view configuration. In addition to ARP, sub-PU inter-view motion prediction and illumination compensation, DBBP is enabled (with dbbp_enabled_flag set equal to 1) for non-base texture views.

The number of layers is 6, the resolution is 1024x768 and the number of frames is 64.

Functional stage: Decoding of three views with DBBP enabled for texture views.

Purpose: To test the inter DBBP for texture.

6.6.15.3.4 Test bitstream 3DHC_DT_D

Specification: The bitstream is coded with random access base view configuration as well as 3-view configuration. In addition to ARP, sub-PU inter-view motion prediction and illumination compensation, all the depth dependent texture tools (DoNBDV, VSP and DBBP) are enabled for non-base texture views.

The number of layers is 6, the resolution is 1024x768 and the number of frames is 25.

Functional stage: Decoding of three views with depth dependent texture tools.

Purpose: To test the combined depth dependent texture tools.

6.6.15.4 Test bitstreams – Texture dependent depth tools

In this category, test bitstreams are tested with the all texture coding tools and depth dependent texture tools enabled for non-base texture views. Depth views are coded with depth coding tools enabled by default and with the texture dependent depth tools disabled unless otherwise specified.

6.6.15.4.1 Test bitstream 3DHC_TD_A

Specification: The bitstream is coded with random access base view configuration as well as 3-view configuration. In addition to all the texture coding tools, all the depth coding tools and depth dependent texture coding tools, the texture merge candidate with sub-block partitioning (sub-PU MPI) is enabled (with tex_mc_enabled_flag set equal to 1 and log2_texmc_sub_pb_size_minus3 set less than 3).

The number of layers is 6, the resolution is 1024x768 and the number of frames is 48.

Functional stage: Decoding of three views with sub-PU MPI enabled for depth views.

Purpose: To test the sub-PU MPI for depth.

6.6.15.4.2 Test bitstream 3DHC_TD_B

Specification: The bitstream is coded with random access base view configuration as well as 3-view configuration. In addition to all the texture coding tools, all the depth coding tools and depth dependent texture coding tools, the texture merge candidate (MPI) is enabled (with tex_mc_enabled_flag set equal to 1 and log2_texmc_sub_pb_size_minus3 set equal to 3).

The number of layers is 6, the resolution is 1920x1088 and the number of frames is 30.

Functional stage: Decoding of three views with MPI enabled for depth views.

Purpose: To test the MPI for depth.

6.6.15.4.3 Test bitstream 3DHC_TD_C

Specification: The bitstream is coded with random access base view configuration as well as 3-view configuration. In addition to all the texture coding tools, all the depth coding tools and depth dependent texture coding tools, quad-tree limitation (QTL) is enabled (with, cqt_cu_part_pred_enabled_flag set equal to 1).

The number of layers is 6, the resolution is 1024x768 and the number of frames is 60.

Functional stage: Decoding of three views with QTL enabled for depth views.

Purpose: To test the QTL for depth.

6.6.15.4.4 Test bitstream 3DHC_TD_D

Specification: The bitstream is coded with random access base view configuration as well as 3-view configuration. In addition to all the texture coding tools, all the depth coding tools and depth dependent texture coding tools, depth intra contour prediction mode is enabled (with intra_contour_enabled_flag set equal to 1, no_dim_flag set equal to 0 and depth_intra_mode_idx_flag set equal to 1).

The number of layers is 6, the resolution is 1920x1088 and the number of frames is 30.

Functional stage: Decoding of three views with depth intra contour prediction mode enabled for depth views.

Purpose: To test the depth intra contour prediction for depth in random access base view configuration.

6.6.15.4.5 Test bitstream 3DHC_TD_E

Specification: The bitstream is coded with all intra base view configuration as well as 3-view configuration. In addition to all the texture coding tools, all the depth coding tools and depth dependent texture coding tools, depth intra contour prediction mode is enabled.

The number of layers is 6, the resolution is 1920x1088 and the number of frames is 3.

Functional stage: Decoding of three views with depth intra contour prediction mode enabled for depth views.

Purpose: To test the depth intra contour prediction for depth in all intra base view configuration.

6.6.15.5 Test bitstreams – Other combined cases

6.6.15.5.1 Test bitstream 3DHC_C_A

Specification: The bitstream is coded with random access base view configuration as well as 3-view configuration. All the texture coding tools, all the depth coding tools, all the depth dependent texture coding tools and all the texture dependent depth coding tools are enabled.

The number of layers is 6, the resolution is 1920x1088 and the number of frames is 60.

Functional stage: Decoding of three views with coding tools enabled.

Purpose: To test the common test condition case.

6.6.15.5.2 Test bitstream 3DHC_C_B

Specification: The bitstream is coded with all intra base view configuration as well as 3-view configuration. All the texture coding tools, all the depth coding tools, all the depth dependent texture coding tools and all the texture dependent depth coding tools are enabled.

The number of layers is 6, the resolution is 1920x1088 and the number of frames is 60.

Functional stage: Decoding of three views with coding tools enabled for all intra base view configuration.

Purpose: To test the all intra base view configuration.

6.6.15.5.3 Test bitstream 3DHC_C_C

Specification: The bitstream is coded with random access base view configuration as well as 2-view configuration. All the texture coding tools, all the depth coding tools, all the depth dependent texture coding tools and all the texture dependent depth coding tools are enabled.

The number of layers is 4, the resolution is 1024x768 and the number of frames is 64.

Functional stage: Decoding of two views with coding tools enabled.

Purpose: To test the 2-view case.

6.6.16 Test bitstreams – Format range and high throughput extensions

6.6.16.1 Test bitstream ADJUST_IPRED_ANGLE_A_RExt

Specification: All slices are coded as I slices. bit_depth_luma_minus8 is set equal to 2 and bit_depth_chroma_minus8 is set equal to 2. chroma_format_idc is set equal to 2. All intra prediction modes for all PU sizes of chroma (35 modes for each chroma PU size of 16x32, 8x16, and 4x8).

Functional stage: Intra prediction.

Purpose: Check that the decoder properly decodes with all the intra prediction modes.

6.6.16.2 Test bitstream CCP_8bit_RExt

Specification: All slices are coded as I or B slices. bit_depth_luma_minus8 is set equal to 0 and bit_depth_chroma_minus8 is set equal to 0. chroma_format_idc is set equal to 3. cross_component_prediction_enabled_flag is set equal to 1.

Functional stage: Test reconstruction process of cross component prediction.

Purpose: Check that the decoder properly decodes with cross-component prediction.

6.6.16.3 Test bitstream CCP_10bit_RExt

Specification: All slices are coded as I or B slices. bit_depth_luma_minus8 is set equal to 2 and bit_depth_chroma_minus8 is set equal to 2. chroma_format_idc is set equal to 3. cross_component_prediction_enabled_flag is set equal to 1.

Functional stage: Test the reconstruction process of cross component prediction.

Purpose: Check that the decoder properly decodes with cross-component prediction.

6.6.16.4 Test bitstream CCP_12bit_RExt

Specification: All slices are coded as I or B slices. bit_depth_luma_minus8 is set equal to 4 and bit_depth_chroma_minus8 is set equal to 4. chroma_format_idc is set equal to 3. cross_component_prediction_enabled_flag is set equal to 1.

Functional stage: Test the reconstruction process of cross component prediction.

Purpose: Check that the decoder properly decodes with cross-component prediction.

6.6.16.5 Test bitstream Bitdepth_A_RExt

Specification: All slices are coded as I or B slices. The value of bit_depth_luma_minus8 is higher than that of bit_depth_chroma_minus8. chroma_format_idc is set equal to 3.

Functional stage: Test the reconstruction process with different bit depth for luma and chroma.

Purpose: Check that the decoder properly decodes when the bit depth of luma and chroma are not the same.

6.6.16.6 Test bitstream Bitdepth_B_RExt

Specification: All slices are coded as I or B slices. The value of bit_depth_chroma_minus8 is higher than that of bit_depth_luma_minus8. chroma_format_idc is set equal to 3.

Functional stage: Test the reconstruction process of different bit depth for luma and chroma.

Purpose: Check that the decoder properly decodes when the bit depth of luma and chroma are not the same.

6.6.16.7 Test bitstream QMATRIX_A_RExt

Specification: All slices are coded as I or B slices. The value of general_max_422chroma_constraint_flag, general_max_420chroma_constraint_flag and general_max_monochrome_constraint_flag is equal to 0. Various scaling list data are included with and without transform skip for all transform block sizes over different QP values.

Functional stage: Test the reconstruction process with scaling lists.

Purpose: Check that the decoder properly decodes with various scaling list data, with and without transform skip for all transform block sizes over different QP values.

6.6.16.8 Test bitstream SAO_A_RExt

Specification: All slices are coded as I or B slices. bit_depth_luma_minus8 is set equal to 4 and bit_depth_chroma_minus8 is set equal to 4. chroma_format_idc is set equal to 3. log2_sao_offset_scale_luma is set equal to 2. log2_sao_offset_scale_chroma is set equal to 2.

Functional stage: Test the reconstruction process of SAO.

Purpose: Check that the decoder properly decodes the specified PPS bit shift parameters for scaling up the SAO offset values.

6.6.16.9 Test bitstream PERSIST_PARAM_A_RExt

Specification: All slices are coded as I slices. bit_depth_luma_minus8 is set equal to 4 and bit_depth_chroma_minus8 is set equal to 4. chroma_format_idc is set equal to 3. persistent_rice_adaptation_enabled_flag is set equal to 0 or 1.

Functional stage: Test the binarization process with persistent Golomb Rice parameters.

Purpose: Check that the decoder properly decodes with the incrementing and decrementing of all four of the persistent Golomb Rice parameters, and the enabling/disabling of the tool.

6.6.16.10 Test bitstream HIGH_TP_8BIT_RExt

Specification: All slices are coded as I slices. bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 0. chroma_format_idc is set equal to 3. The bitstream consists of a concatenation of two coded video sequences of one frame each:

– the first with extended_precision_processing_flag equal to 0

– the second with extended_precision_processing_flag equal to 1

Functional stage: Test the extended precision processing function.

Purpose: Check that the decoder properly decodes a bitstream for the High Throughput 4:4:4 16 Intra profile with general_lower_bit_rate_constraint_flag equal to 1.

6.6.16.11 Test bitstream HIGH_TP_10BIT_RExt

Specification: All slices are coded as I slices. bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 2. chroma_format_idc is set equal to 3. The bitstream consists of a concatenation of two coded video sequences of one frame each:

– the first with extended_precision_processing_flag equal to 0

– the second with extended_precision_processing_flag equal to 1

Functional stage: Test the extended precision processing function.

Purpose: Check that the decoder properly decodes a bitstream for the High Throughput 4:4:4 16 Intra profile with general_lower_bit_rate_constraint_flag equal to 1.

6.6.16.12 Test bitstream HIGH_TP_12BIT_RExt

Specification: All slices are coded as I slices. bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 4. chroma_format_idc is set equal to 3. The bitstream consists of a concatenation of two coded video sequences of one frame each:

– the first with extended_precision_processing_flag equal to 0

– the second with extended_precision_processing_flag equal to 1

Functional stage: Test the extended precision processing function.

Purpose: Check that the decoder properly decodes a bitstream for the High Throughput 4:4:4 16 Intra profile with general_lower_bit_rate_constraint_flag equal to 1.

6.6.16.13 Test bitstream HIGH_TP_16BIT_RExt

Specification: All slices are coded as I slices. bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 8. chroma_format_idc is set equal to 3. The bitstream consists of a concatenation of two coded video sequences of one frame each:

– the first with extended_precision_processing_flag equal to 0

– the second with extended_precision_processing_flag equal to 1

Functional stage: Test the extended precision processing function.

Purpose: Check that the decoder properly decodes a bitstream for the High Throughput 4:4:4 16 Intra profile with general_lower_bit_rate_constraint_flag equal to 1.

6.6.16.14 Test bitstream HIGH_TP_8BIT_RExt

Specification: All slices are coded as I slices. bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 0. chroma_format_idc is set equal to 3. The bitstream consists of a concatenation of two coded video sequences of one frame each:

– the first with extended_precision_processing_flag equal to 0

– the second with extended_precision_processing_flag equal to 1

Functional stage: Test the extended precision processing function.

Purpose: Check that the decoder properly decodes a bitstream for the Main 4:4:4 16 Intra profile with general_lower_bit_rate_constraint_flag equal to 1.

6.6.16.15 Test bitstream HIGH_TP_10BIT_RExt

Specification: All slices are coded as I slices. bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 2. chroma_format_idc is set equal to 3. The bitstream consists of a concatenation of two coded video sequences of one frame each:

– the first with extended_precision_processing_flag equal to 0

– the second with extended_precision_processing_flag equal to 1

Functional stage: Test the extended precision processing function.

Purpose: Check that the decoder properly decodes a bitstream for the Main 4:4:4 16 Intra profile with general_lower_bit_rate_constraint_flag equal to 1.

6.6.16.16 Test bitstream HIGH_TP_12BIT_RExt

Specification: All slices are coded as I slices. bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 4. chroma_format_idc is set equal to 3. The bitstream consists of a concatenation of two coded video sequences of one frame each:

– the first with extended_precision_processing_flag equal to 0

– the second with extended_precision_processing_flag equal to 1

Functional stage: Test the extended precision processing function.

Purpose: Check that the decoder properly decodes a bitstream for the Main 4:4:4 16 Intra profile with general_lower_bit_rate_constraint_flag equal to 1.

6.6.16.17 Test bitstream HIGH_TP_16BIT_RExt

Specification: All slices are coded as I slices. bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 8. chroma_format_idc is set equal to 3. The bitstream consists of a concatenation of two coded video sequences of one frame each:

– the first with extended_precision_processing_flag equal to 0

– the second with extended_precision_processing_flag equal to 1

Functional stage: Test the extended precision processing function.

Purpose: Check that the decoder properly decodes a bitstream for the Main 4:4:4 16 Intra profile with general_lower_bit_rate_constraint_flag equal to 1.

6.6.16.18 Test bitstream IPCM_A_RExt

Specification: All slices are coded as I slices. chroma_format_idc is set equal to 2. pcm_enabled_flag is equal to 1. Both pcm_sample_bit_depth_luma_minus1 and pcm_sample_bit_depth_chroma_minus1 are equal to 7. log2_min_pcm_luma_coding_block_size_minus3, log2_diff_max_min_pcm_luma_coding_block_size, and pcm_loop_filter_disabled_flag are equal to 0, 1 and 0, respectively.

Functional stage: Test the parsing of pcm_flags in coding unit syntax. Test parsing of 4:2:2 format.

Purpose: Check that decoder properly decodes slices of coded frames containing pcm_flag.

6.6.16.19 Test bitstream IPCM_B_RExt

Specification: All slices are coded as I slices. chroma_format_idc is set equal to 3. pcm_enabled_flag is equal to 1. Both pcm_sample_bit_depth_luma_minus1 and pcm_sample_bit_depth_chroma_minus1 are equal to 7. log2_min_pcm_luma_coding_block_size_minus3, log2_diff_max_min_pcm_luma_coding_block_size, and pcm_loop_filter_disabled_flag are equal to 0, 1 and 0, respectively.

Functional stage: Test the parsing of pcm_flags in coding unit syntax. Test parsing of 4:4:4 format, pcm_sample_luma and pcm_sample_chroma data.

Purpose: Check that decoder properly decodes slices of coded frames containing pcm_flag.

6.6.16.20 Test bitstream TSCTX_8bit_I_RExt

Specification: All slices are coded as I slices. bit_depth_luma_minus8 is set equal to 0 and bit_depth_chroma_minus8 is set equal to 0. chroma_format_idc is set equal to 3. transform_skip_context_enabled_flag is set equal to 1. log2_max_transform_skip_block_size_minus2 is equal to 0.

Functional stage: Test the reconstruction process with transform skip.

Purpose: Check that decoder properly decodes with transform skip context.

6.6.16.21 Test bitstream TSCTX_8bit_RExt

Specification: All slices are coded as I or B slices. bit_depth_luma_minus8 is set equal to 0 and bit_depth_chroma_minus8 is set equal to 0. chroma_format_idc is set equal to 3. transform_skip_context_enabled_flag is set equal to 1. log2_max_transform_skip_block_size_minus2 is equal to 0.

Functional stage: Test the reconstruction process with transform skip.

Purpose: Check that decoder properly decodes with transform skip context.

6.6.16.22 Test bitstream TSCTX_10bit_I_RExt

Specification: All slices are coded as I slices. bit_depth_luma_minus8 is set equal to 2 and bit_depth_chroma_minus8 is set equal to 2. chroma_format_idc is set equal to 3. transform_skip_context_enabled_flag is set equal to 1. log2_max_transform_skip_block_size_minus2 is equal to 0.

Functional stage: Test the reconstruction process with transform skip.

Purpose: Check that decoder properly decodes with transform skip context.

6.6.16.23 Test bitstream TSCTX_10bit_RExt

Specification: All slices are coded as I or B slices. bit_depth_luma_minus8 is set equal to 2 and bit_depth_chroma_minus8 is set equal to 2. chroma_format_idc is set equal to 3. transform_skip_context_enabled_flag is set equal to 1. log2_max_transform_skip_block_size_minus2 is equal to 0.

Functional stage: Test the reconstruction process with transform skip.

Purpose: Check that decoder properly decodes with transform skip context.

6.6.16.24 Test bitstream TSCTX_12bit_I_RExt

Specification: All slices are coded as I slices. bit_depth_luma_minus8 is set equal to 4 and bit_depth_chroma_minus8 is set equal to 4. chroma_format_idc is set equal to 3. transform_skip_context_enabled_flag is set equal to 1. log2_max_transform_skip_block_size_minus2 is equal to 0.

Functional stage: Test the reconstruction process with transform skip.

Purpose: Check that decoder properly decodes with transform skip context.

6.6.16.25 Test bitstream TSCTX_12bit_RExt

Specification: All slices are coded as I or B slices. bit_depth_luma_minus8 is set equal to 4 and bit_depth_chroma_minus8 is set equal to 4. chroma_format_idc is set equal to 3. transform_skip_context_enabled_flag is set equal to 1. log2_max_transform_skip_block_size_minus2 is equal to 0.

Functional stage: Test the reconstruction process with transform skip.

Purpose: Check that decoder properly decodes with transform skip context.

6.6.16.26 Test bitstream ExplictRdpcm_A_RExt

Specification: All slices are coded as I or B slices. bit_depth_luma_minus8 is set equal to 4 and bit_depth_chroma_minus8 is set equal to 4. chroma_format_idc is set equal to 3. explicit_rdpcm_enabled_flag is set equal to 1.

Functional stage: Test the reconstruction process with the residual modification process for blocks using a transform bypass.

Purpose: Check that decoder properly decodes with the explicit residual modification process for blocks using a transform bypass where on even numbered TUs the residual modification process for blocks using a transform bypass is disabled, while on odd numbered TUs it is decided via prediction error minimization.

6.6.16.27 Test bitstream ExplictRdpcm_B_RExt

Specification: All slices are coded as I or B slices. bit_depth_luma_minus8 is set equal to 4 and bit_depth_chroma_minus8 is set equal to 4. chroma_format_idc is set equal to 3. explicit_rdpcm_enabled_flag is set equal to 1.

Functional stage: Test the reconstruction process with the residual modification process for blocks using a transform bypass.

Purpose: Check that decoder properly decodes with explicit the residual modification process for blocks using a transform bypass where on even numbered TUs the residual modification process for blocks using a transform bypass is disabled while on odd numbered TUs it is decided via prediction error minimization.

6.6.16.28 Test bitstream Main_422_10_A_RExt

Specification: All slices are coded as I or B slices. bit_depth_luma_minus8 is set equal to 2 and bit_depth_chroma_minus8 is set equal to 2. chroma_format_idc is set equal to 2.

Functional stage: Test the reconstruction process with various combinations of tools.

Purpose: Check that decoder properly decodes with various combinations of coding tools.

6.6.16.29 Test bitstream Main_422_10_B_RExt

Specification: All slices are coded as I or B slices. bit_depth_luma_minus8 is set equal to 2 and bit_depth_chroma_minus8 is set equal to 2. chroma_format_idc is set equal to 2.

Functional stage: Test reconstruction process with various combinations of tools.

Purpose: Check that decoder properly decodes bitstream with various combinations of coding tools.

6.6.16.30 Test bitstream GENERAL_8b_400_RExt

Specification: All slices are coded as I slices. bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 0. chroma_format_idc is set equal to 0.

Functional stage: Test the extended precision processing function.

Purpose: Check that the decoder properly decodes a bitstream for the Monochrome profile with general_lower_bit_rate_constraint_flag equal to 1.

6.6.16.31 Test bitstream GENERAL_8b_420_RExt

Specification: All slices are coded as I slices. bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 0. chroma_format_idc is set equal to 1.

Functional stage: Test the extended precision processing function.

Purpose: Check that the decoder properly decodes a bitstream for the Main Intra profile with general_lower_bit_rate_constraint_flag equal to 1.

6.6.16.32 Test bitstream GENERAL_8b_444_RExt

Specification: All slices are coded as I slices. bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 0. chroma_format_idc is set equal to 3. The bitstream consists of a concatenation of sequences, each just two pictures:

- the first of each pair has cross_component_prediction_prediction_enabled_flag equal to 0,
- the second has cross_component_prediction_prediction_enabled_flag equal to 1.

The concatenated sequences consist of:

- Sequence 0: transform_skip_rotation_enabled_flag equal to 0, transform_skip_context_enabled_flag equal to 0, implicit_rdpcm_enabled_flag equal to 0, intra_smoothing_disabled_flag equal to 0, persistent_rice_adaptation_enabled_flag equal to 0, chroma_qp_offset_list_enabled_flag equal to 0
- Sequence 1: as in sequence 0, but with transform_skip_rotation equal to 1
- Sequence 2: as in sequence 0, but with transform_skip_context equal to 1
- Sequence 3: as in sequence 0, but with implicit_rdpcm equal to 1

– Sequence 4: as in sequence 0, but with intra_smoothing_disabled equal to 1
– Sequence 5: as in sequence 0, but with persistent_rice_adaptation_enabled_flag equal to 1
– Sequence 6: as in sequence 0, but with chroma_qp_offset_list_enabled_flag equal to 1

Functional stage: Test the extended precision processing function.

Purpose: Check that the decoder properly decodes a bitstream for the Main 4:4:4 Intra profile with general_lower_bit_rate_constraint_flag equal to 1.

6.6.16.33 Test bitstream GENERAL_10b_420_RExt

Specification: All slices are coded as I slices. bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 2. chroma_format_idc is set equal to 1.

Functional stage: Test the extended precision processing function.

Purpose: Check that the decoder properly decodes a bitstream for the Main 10 Intra profile with general_lower_bit_rate_constraint_flag equal to 1.

6.6.16.34 Test bitstream GENERAL_10b_422_RExt

Specification: All slices are coded as I slices. bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 2. chroma_format_idc is set equal to 2. The bitstream consists of a concatenation of sequences, each containing just one picture:

– Sequence 0: chroma_qp_offset_list_enabled_flag equal to 0
– Sequence 1: chroma_qp_offset_list_enabled_flag equal to 1

Functional stage: Test the extended precision processing function.

Purpose: Check that the decoder properly decodes a bitstream for the Main 4:2:2 10 Intra profile with general_lower_bit_rate_constraint_flag equal to 1.

6.6.16.35 Test bitstream GENERAL_10b_444_RExt

Specification: All slices are coded as I slices. bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 2. chroma_format_idc is set equal to 3. The bitstream consists of a concatenation of sequences, each just two pictures:

– the first of each pair has cross_component_prediction_prediction_enabled_flag equal to 0,
– the second has cross_component_prediction_prediction_enabled_flag equal to 1.

The concatenated sequences consist of:

– Sequence 0: transform_skip_rotation_enabled_flag equal to 0, transform_skip_context_enabled_flag equal to 0, implicit_rdpcm_enabled_flag equal to 0, intra_smoothing_disabled_flag equal to 0, persistent_rice_adaptation_enabled_flag equal to 0
– Sequence 1: as in sequence 0, but with transform_skip_rotation equal to 1
– Sequence 2: as in sequence 0, but with transform_skip_context equal to 1
– Sequence 3: as in sequence 0, but with implicit_rdpcm equal to 1
– Sequence 4: as in sequence 0, but with intra_smoothing_disabled equal to 1
– Sequence 5: as in sequence 0, but with persistent_rice_adaptation_enabled_flag equal to 1

Functional stage: Test the extended precision processing function.

Purpose: Check that the decoder properly decodes a bitstream for the Main 4:4:4 10 Intra profile with general_lower_bit_rate_constraint_flag equal to 1.

6.6.16.36 Test bitstream GENERAL_12b_400_RExt

Specification: All slices are coded as I slices. bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 4. chroma_format_idc is set equal to 0.

Functional stage: Test the extended precision processing function.

Purpose: Check that the decoder properly decodes a bitstream for the Monochrome 12 profile with general_lower_bit_rate_constraint_flag equal to 1.

6.6.16.37 Test bitstream GENERAL_12b_420_RExt

Specification: All slices are coded as I slices. bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 4. chroma_format_idc is set equal to 1.

Functional stage: Test the extended precision processing function.

Purpose: Check that the decoder properly decodes a bitstream for the Main 12 Intra profile with general_lower_bit_rate_constraint_flag equal to 1.

6.6.16.38 Test bitstream GENERAL_12b_422_RExt

Specification: All slices are coded as I slices. bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 4. chroma_format_idc is set equal to 2. The bitstream consists of a concatenation of sequences, each containing just one picture:

- Sequence 0: chroma_qp_offset_list_enabled_flag equal to 0
- Sequence 1: chroma_qp_offset_list_enabled_flag equal to 1

Functional stage: Test the extended precision processing function.

Purpose: Check that the decoder properly decodes a bitstream for the Main 4:2:2 12 Intra profile with general_lower_bit_rate_constraint_flag equal to 1.

6.6.16.39 Test bitstream GENERAL_12b_444_RExt

Specification: All slices are coded as I slices. bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 4. chroma_format_idc is set equal to 3. The bitstream consists of a concatenation of sequences, each just two pictures:

- the first of each pair has cross_component_prediction_prediction_enabled_flag equal to 0,
- the second has cross_component_prediction_prediction_enabled_flag equal to 1.

The concatenated sequences consist of:

- Sequence 0: transform_skip_rotation_enabled_flag equal to 0, transform_skip_context_enabled_flag equal to 0, implicit_rdpcm_enabled_flag equal to 0, intra_smoothing_disabled_flag equal to 0, persistent_rice_adaptation_enabled_flag equal to 0, chroma_qp_offset_list_enabled_flag equal to 0
- Sequence 1: as in sequence 0, but with transform_skip_rotation equal to 1
- Sequence 2: as in sequence 0, but with transform_skip_context equal to 1
- Sequence 3: as in sequence 0, but with implicit_rdpcm equal to 1
- Sequence 4: as in sequence 0, but with intra_smoothing_disabled equal to 1
- Sequence 5: as in sequence 0, but with persistent_rice_adaptation_enabled_flag equal to 1
- Sequence 6: as in sequence 0, but with chroma_qp_offset_list_enabled_flag equal to 1

Functional stage: Test the extended precision processing function.

Purpose: Check that the decoder properly decodes a bitstream for the Main 4:4:4 12 Intra profile with general_lower_bit_rate_constraint_flag equal to 1.

6.6.16.40 Test bitstream GENERAL_16b_400_RExt

Specification: All slices are coded as I slices. bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 8. chroma_format_idc is set equal to 0. The bitstream consists of a concatenation of sequences, each containing just one picture:

- Sequence 0: transform_skip_rotation_enabled_flag equal to 0, transform_skip_context_enabled_flag equal to 0, implicit_rdpcm_enabled_flag equal to 0, intra_smoothing_disabled_flag equal to 0, persistent_rice_adaptation_enabled_flag equal to 0, extended_precision_processing_flag equal to 0
- Sequence 1: as in sequence 0, but with transform_skip_rotation equal to 1
- Sequence 2: as in sequence 0, but with transform_skip_context equal to 1
- Sequence 3: as in sequence 0, but with implicit_rdpcm equal to 1
- Sequence 4: as in sequence 0, but with intra_smoothing_disabled equal to 1
- Sequence 5: as in sequence 0, but with persistent_rice_adaptation_enabled_flag equal to 1
- Sequence 6: as in sequence 0, but with extended_precision_processing_flag equal to 1

Functional stage: Test the extended precision processing function.

Purpose: Check that the decoder properly decodes a bitstream for the Monochrome 16 profile with general_lower_bit_rate_constraint_flag equal to 1.

6.6.16.41 Test bitstream GENERAL_16b_444_RExt

Specification: All slices are coded as I slices. bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 8. chroma_format_idc is set equal to 3. The bitstream consists of a concatenation of sequences, each just two pictures:

- the first of each pair has cross_component_prediction_prediction_enabled_flag equal to 0,
- the second has cross_component_prediction_prediction_enabled_flag equal to 1.

The concatenated sequences consist of:

- Sequence 0: transform_skip_rotation_enabled_flag equal to 0, transform_skip_context_enabled_flag equal to 0, implicit_rdpcm_enabled_flag equal to 0, intra_smoothing_disabled_flag equal to 0, persistent_rice_adaptation_enabled_flag equal to 0, extended_precision_processing_flag equal to 0, chroma_qp_offset_list_enabled_flag equal to 0
- Sequence 1: as in sequence 0, but with transform_skip_rotation equal to 1
- Sequence 2: as in sequence 0, but with transform_skip_context equal to 1
- Sequence 3: as in sequence 0, but with implicit_rdpcm equal to 1
- Sequence 4: as in sequence 0, but with intra_smoothing_disabled equal to 1
- Sequence 5: as in sequence 0, but with persistent_rice_adaptation_enabled_flag equal to 1
- Sequence 6: as in sequence 0, but with extended_precision_processing_flag equal to 1
- Sequence 7: as in sequence 0, but with chroma_qp_offset_list_enabled_flag equal to 1

Functional stage: Test the extended precision processing function.

Purpose: Check that the decoder properly decodes a bitstream for the Main 4:4:4 16 Intra profile with general_lower_bit_rate_constraint_flag equal to 1.

6.6.16.42 Test bitstream GENERAL_16b_444_HighThroughput_RExt

Specification: All slices are coded as I slices. bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 8. chroma_format_idc is set equal to 3. The bitstream consists of a concatenation of sequences, each just two pictures:

- the first of each pair has cross_component_prediction_prediction_enabled_flag equal to 0,
- the second has cross_component_prediction_prediction_enabled_flag equal to 1.

The concatenated sequences consist of:

- Sequence 0: transform_skip_rotation_enabled_flag equal to 0, transform_skip_context_enabled_flag equal to 0, implicit_rdpcm_enabled_flag equal to 0, intra_smoothing_disabled_flag equal to 0, persistent_rice_adaptation_enabled_flag equal to 0, extended_precision_processing_flag equal to 0, chroma_qp_offset_list_enabled_flag equal to 0
- Sequence 1: as in sequence 0, but with transform_skip_rotation equal to 1
- Sequence 2: as in sequence 0, but with transform_skip_context equal to 1
- Sequence 3: as in sequence 0, but with implicit_rdpcm equal to 1
- Sequence 4: as in sequence 0, but with intra_smoothing_disabled equal to 1
- Sequence 5: as in sequence 0, but with persistent_rice_adaptation_enabled_flag equal to 1
- Sequence 6: as in sequence 0, but with extended_precision_processing_flag equal to 1
- Sequence 7: as in sequence 0, but with chroma_qp_offset_list_enabled_flag equal to 1

Functional stage: Test the extended precision processing function.

Purpose: Check that the decoder properly decodes a bitstream for the High Throughput 4:4:4 16 Intra profile with general_lower_bit_rate_constraint_flag equal to 1.

6.6.16.43 Test bitstream WAVETILES_RExt

Specification: All slices are coded as I slices. bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 8. chroma_format_idc is set equal to 3. The purpose of the bitstream is to exercise the combination of simultaneously using wavefronts and tiles.

The bitstream consists of four sequences of one picture:

– Sequence 1: 1 slice per picture
– Sequence 2: 1 slice per tile
– Sequence 3: 1 slice for at most 6 CTUs
– Sequence 4: 1 slice segment for at most 6 CTUs
– Sequences 3 and 4 split the picture into multiple slice / slice segments per tile.

Functional stage: Test the extended precision processing function.

Purpose: Check that the decoder properly decodes a bitstream for the High Throughput 4:4:4 16 Intra profile with general_lower_bit_rate_constraint_flag equal to 1.

6.6.17 Test bitstreams – Scalable extensions

6.6.17.1 Test bitstreams – Layer ID

6.6.17.1.1 Test bitstream LAYERID_A

Specification: All slices are coded as I, P and B slices. The LAYERID_A bitstream has two layers –with layer ID values 0 and 2. The layer with layer ID equal to 2 is coded as an auxiliary layer. The bitstream contains only nine access units.

Functional stage: Test gaps in the layer ID values.

Purpose: Check that the decoder can properly decode bitstreams that have layers with gaps in the layer ID values.

Coding structure: Hierarchical B-frames with GOP size of 8.

Number of access units: 9

Profile, tier and level (PTL) information: Three PTL structures are specified:
– PTL Idx 0: Main profile, Main tier, Level 4.1 (whole bitstream)
– PTL Idx 1: Main profile, Main tier, Level 3.1
– PTL Idx 2: Main profile, Main tier, Level 4

Output layer sets: The bitstream contains three output layer sets:
– OLS 0 includes the following layer:
  – Layer with layer ID value 0, PTL Idx 1
– OLS 1 includes the following layers:
  – Layer with layer ID value 0, PTL Idx 1
  – Layer with layer ID value 2, PTL Idx 2
– OLS 2 includes the following layer:
  – Layer with layer ID value 2, PTL Idx 2

Resolution of each layer: 1280x720

Frame rate: 24 fps

6.6.17.1.1 Test bitstream MVD_A_IDCC

Specification: All slices are coded as I, P and B slices. The MVD_A_IDCC bitstream has two layers. The bitstream contains only eight access units. The value of DefaultTargetOutputLayerIdc is set equal to 1. Layer 1 uses inter-layer sample prediction from Layer 0.

Functional stage: Test disabling inter-layer motion vector prediction.

Purpose: Check that the decoder can properly decode when the enhancement layer performs inter-layer sample prediction from the base layer and motion vector prediction is disabled.

Coding structure: Hierarchical B-frames with four temporal sub-layers.

Number of access units: 8

Profile, tier and level (PTL) information: Three PTL structures are specified:
– PTL Idx 0: Main profile, Main tier, Level 4 (whole bitstream)
– PTL Idx 1: Main profile, Main tier, Level 3.1

– PTL Idx 2: Scalable Main profile, Main tier, Level 4

Output layer sets: The bitstream contains two output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2

Resolution of layers:

- Layer 0: 960x540 (output), 960x544 (coded)
- Layer 1: 1920x1080 (output), 1920x1280 (coded)

Frame rate: 24 fps

6.6.17.1.2 Test bitstream MVD_A_NOKIA

Specification: All slices are coded as I and P slices. The MVD_A_NOKIA bitstream has three layers. The bitstream contains only four access units. Layer 1 uses inter-layer sample prediction and inter-layer motion vector prediction from Layer 0. Layer 2 uses inter-layer sample prediction from Layer 1 and inter-layer motion vector prediction from Layer 0.

Functional stage: Test inter-layer sample prediction and inter-layer motion vector prediction from different layers.

Purpose: Check that the decoder can properly decode when the enhancement layer uses inter-layer sample prediction and inter-layer motion vector prediction from different layers.

Coding structure: Low delay P.

Number of access units: 4

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 4.1 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 3.1
- PTL Idx 2: Scalable Main profile, Main tier, Level 4

Output layer sets: The bitstream contains three output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
- OLS 2 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
  - Layer 2, PTL Idx 2

Resolution of each layer: 1280x720

Frame rate: 24 fps

6.6.17.1.3 Test bitstream MAXTID_A

Specification: All slices are coded as I, P and B slices. The MAXTID_A bitstream has three layers. The bitstream contains only nine access units. The values of max_tid_il_ref_pics_plus1[0][1] and max_tid_il_ref_pics_plus1[1][2] are set equal to 3 and 2, respectively.

Functional stage: Test parsing of the syntax elements max_tid_il_ref_pics_plus1[ ][ ] and inter-layer prediction restrictions based on temporal sub-layer.

Purpose: Check that the decoder can properly parse and decode the syntax elements max_tid_ref_present_flag and max_tid_il_ref_pics_plus1[ ][ ].

Coding structure: Hierarchical B-frames with four temporal sub-layers.

Number of access units: 9

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 4 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 3.1
- PTL Idx 2: Scalable Main profile, Main tier, Level 4

Output layer sets: The bitstream contains three output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
- OLS 2 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
  - Layer 2, PTL Idx 2

Resolution of each layer:

- Layer 0: 960x540 (output), 960x544 (coded)
- Layer 1: 1920x1080
- Layer 2: 1920x1080

Frame rate: 24 fps

6.6.17.1.4 Test bitstream MAXTID_B

Specification: All slices are coded as I, P and B slices. The MAXTID_B bitstream has three layers. The bitstream contains only nine access units. The value of the syntax element max_tid_ref_present_flag is set equal to 0.

Functional stage: Test parsing of the syntax element max_tid_ref_present_flag.

Purpose: Check that the decoder can properly parse the syntax element max_tid_ref_present_flag and properly infer values of max_tid_il_ref_pics_plus1[i][j] when max_tid_ref_present_flag equal to 0.

Coding structure: Hierarchical B-frames with four temporal sub-layers.

Number of access units: 9

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 4 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 3.1
- PTL Idx 2: Scalable Main profile, Main tier, Level 4

Output layer sets: The bitstream contains three output layer sets:

- OLS 0 includes the following layers:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
- OLS 2 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
  - Layer 2, PTL Idx 2

Resolution of each layer:

- Layer 0: 960x540 (output), 960x544 (coded)

– Layer 1: 1920x1080
– Layer 2: 1920x1080

Frame rate: 24 fps

6.6.17.1.5 Test bitstream MAXTID_C

Specification: All slices are coded as I, P and B slices. The MAXTID_C bitstream has three layers. The bitstream contains only nine access units. The values of max_tid_il_ref_pics_plus1[0][1] and max_tid_il_ref_pics_plus1[1][2] are both set equal to 0.

Functional stage: Test parsing of the syntax element max_tid_il_ref_pics_plus1[ ][ ] and inter-layer prediction restrictions based on temporal sub-layer.

Purpose: Check that the decoder can properly parse and decode the syntax elements max_tid_ref_present_flag and max_tid_il_ref_pics_plus1.

Coding structure: Hierarchical B-frames with four temporal sub-layers.

Number of access units: 9

Profile, tier and level (PTL) information: Three PTL structures are specified:

– PTL Idx 0: Main profile, Main tier, Level 4 (whole bitstream)
– PTL Idx 1: Main profile, Main tier, Level 3.1
– PTL Idx 2: Scalable Main profile, Main tier, Level 4

Output layer sets: The bitstream contains three output layer sets:

– OLS 0 includes the following layers:
  – Layer 0, PTL Idx 1
– OLS 1 includes the following layers:
  – Layer 0, PTL Idx 1
  – Layer 1, PTL Idx 2
– OLS 2 includes the following layers:
  – Layer 0, PTL Idx 1
  – Layer 1, PTL Idx 2
  – Layer 2, PTL Idx 2

Resolution of each layer:

– Layer 0: 960x540 (output), 960x544 (coded)
– Layer 1: 1920x1080
– Layer 2: 1920x1080

Frame rate: 24 fps

6.6.17.1.6 Test bitstream INACTIVE_A

Specification: All slices are coded as I, P and B slices. The INACTIVE_A bitstream has three layers. The bitstream contains 100 access units. Layer 0 is a direct reference layer of Layer 1, and Layer 0 and Layer 1 are direct reference layers of Layer 2. The value of default_ref_layers_active_flag is set equal to 0 in the VPS. The layers are of the same resolution.

Functional stage: Test parsing of the syntax elements inter_layer_pred_enabled_flag, num_inter_layer_ref_pics_minus1, and inter_layer_pred_layer_idc[ ].

Purpose: Check that the decoder can properly parse and decode the bitstream when the active reference layers are explicitly signalled.

Coding structure: Random access

Number of access units: 100

Profile, tier and level (PTL) information: Three PTL structures are specified:

– PTL Idx 0: Main profile, Main tier, Level 3 (whole bitstream)

- PTL Idx 1: Main profile, Main tier, Level 2.1
- PTL Idx 2: Scalable Main profile, Main tier, Level 2.1

Output layer sets: The bitstream contains three output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
- OLS 2 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
  - Layer 2, PTL Idx 2

Resolution of each layer: 416x240

Frame rate: 50 fps

6.6.17.1.7 Test bitstream REFLAYER_A

Specification: All slices are coded as I, P and B slices. The bitstream has three layers. The REFLAYER_A bitstream contains 9 access units. Layer 0 is a direct reference layer of Layer 1, and Layer 0 and Layer 1 are direct reference layers of Layer 2. The value of default_ref_layers_active_flag is set equal to 0 in the VPS. The reference layer are used for both inter-layer sample prediction and inter-layer motion vector prediction. The value of direct_dependency_all_layers_flag is equal to 1 and the value of direct_dependency_all_layers_type is equal to 2.

Functional stage: Test multiple active reference layers with different inter-layer prediction layers at different access units.

Purpose: Check that the decoder can properly parse and decode the bitstream when different active reference layers are chosen for prediction at different access units.

Coding structure: Low Delay B

Number of access units: 9

Profile, tier and level (PTL) information: Four PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 3.1 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 3
- PTL Idx 2: Scalable Main profile, Main tier, Level 3
- PTL Idx 3: Scalable Main profile, Main tier, Level 4

Output layer sets: The bitstream contains three output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
- OLS 2 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
  - Layer 2, PTL Idx 3

Resolution of each layer:

- Layer 0: 960x540 (output), 960x544 (coded)
- Layer 1: 960x540 (output), 960x544 (coded)
- Layer 2: 1920x1080 (output), 1920x1080 (coded)

Frame rate: 24 fps

6.6.17.1.8 Test bitstream REFLAYER_B

Specification: All slices are coded as I and B slices. The bitstream has three layers. The REFLAYER_B bitstream contains 9 access units. Layer 0 is a direct reference layer of Layer 1, and Layer 0 and Layer 1 are direct reference layers of Layer 2. For Layer 1, both inter-layer sample prediction and inter-layer motion vector prediction are performed from Layer 0. For Layer 2, only inter-layer sample prediction is performed from Layer 0, and both inter-layer sample prediction and inter-layer motion vector prediction are performed from Layer 1.

Functional stage: Test multiple active reference layers with different inter-layer prediction layers at different access units, and different layers used for inter-layer sample prediction and inter-layer motion vector prediction.

Purpose: Check that the decoder can properly parse and decode the bitstream when different active reference layers are chosen for prediction at different access units, and different types of inter-layer prediction are performed from different layers.

Coding structure: Low Delay B

Number of access units: 9

Profile, tier and level (PTL) information: Four PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 3.1 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 3
- PTL Idx 2: Scalable Main profile, Main tier, Level 3
- PTL Idx 3: Scalable Main profile, Main tier, Level 4

Output layer sets: The bitstream contains three output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
- OLS 2 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
  - Layer 2, PTL Idx 3

Resolution of each layer:

- Layer 0: 960x540 (output), 960x544 (coded)
- Layer 1: 960x540 (output), 960x544 (coded)
- Layer 2: 1920x1080 (output), 1920x1080 (coded)

Frame rate: 24 fps

6.6.17.1.9 Test bitstream REFLAYER_C

Specification: All slices are coded as I and B slices. The bitstream has three layers. The REFLAYER_C bitstream contains 9 access units. Layer 0 is a direct reference layer of Layer 1, and Layer 0 and Layer 1 are direct reference layers of Layer 2.

Functional stage: Test multiple active reference layers with different inter-layer prediction layers at different access units.

Purpose: Check that the decoder can properly parse and decode the bitstream when different active reference layers are chosen for prediction at different access units.

Coding structure: Low Delay B

Number of access units: 9

Profile, tier and level (PTL) information: Four PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 3.1 (whole bitstream)

- PTL Idx 1: Main profile, Main tier, Level 3
- PTL Idx 2: Scalable Main profile, Main tier, Level 3.1
- PTL Idx 3: Scalable Main profile, Main tier, Level 4

Output layer sets: The bitstream contains three output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
- OLS 2 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
  - Layer 2, PTL Idx 3

Resolution of each layer:

- Layer 0: 960x540 (output), 960x544 (coded)
- Layer 1: 1280x720 (output), 1280x720 (coded)
- Layer 2: 1920x1080 (output), 1920x1080 (coded)

Frame rate: 24 fps

6.6.17.1.10 Test bitstream REFLAYER_D

Specification: All slices are coded as I and B slices. The bitstream has three layers. The REFLAYER_D bitstream contains 9 access units. Layer 0 is a direct reference layer of Layer 1, and Layer 0 and Layer 1 are direct reference layers of Layer 2. The reference layer are used for both inter-layer sample prediction and inter-layer motion vector prediction with the direct_dependency_all_layers_flag is equal to 1 and direct_dependency_all_layers_type is equal to 2. Inter-layer prediction restrictions based on the temporal ID are indicated using values of the syntax element max_tid_il_ref_pics_plus1[ ][ ].

Functional stage: Test multiple active reference layers with different inter-layer prediction layers at different access units, where inter-layer prediction may be restricted based on TemporalID value.

Purpose: Check that the decoder can properly parse and decode the bitstream when different active reference layers are chosen for prediction at different access units and inter-layer prediction may be disabled at some access units based on TemporalID value.

Coding structure: Random Access B with GOP size 8 and four temporal sub-layers.

Number of access units: 9

Profile, tier and level (PTL) information: Four PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 3.1 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 3
- PTL Idx 2: Scalable Main profile, Main tier, Level 3.1
- PTL Idx 3: Scalable Main profile, Main tier, Level 4

Output layer sets: The bitstream contains four output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
- OLS 2 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2

– Layer 2, PTL Idx 3

Resolution of each layer:

- Layer 0: 960x540 (output), 960x544 (coded)
- Layer 1: 1280x720 (output), 1280x720 (coded)
- Layer 2: 1920x1080 (output), 1920x1080 (coded)

Frame rate: 24 fps

6.6.17.2 Test bitstreams – VPS syntax

6.6.17.2.1 Test bitstream SPLITFLAG_A

Specification: All slices are coded as I, P and B slices. The SPLITFLAG_A bitstream has two layers and the value of splitting_flag is set equal to 1.

Functional stage: Test inference of layer dimension ID values from the layer ID values.

Purpose: Check that the decoder can properly parse the syntax element splitting_flag and infer the values of dimension ID values from the layer ID values.

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0
- PTL Idx 1: Main profile, Main tier, Level 3.1
- PTL Idx 2: Scalable Main profile, Main tier, Level 4

Output layer sets: The bitstream contains two output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2

Resolution of each layer:

- Layer 0: 960x540
- Layer 1: 1920x1080

Frame rate: 24 fps

6.6.17.2.2 Test bitstream VUI_A

Specification: All slices are coded as I, P and B slices. The VUI_A bitstream has two layers. The bitstream contains five access units. The value of vps_vui_present_flag is set equal to 1 and VPS VUI syntax structure is signalled.

Functional stage: Test VPS VUI.

Purpose: Check that the decoder can properly parse when VPS VUI is present in the bitstream. The decoder is expected to be able to parse the VPS VUI.

Coding structure: Random access

Number of access units: 5

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 4.1 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 3.1
- PTL Idx 2: Scalable Main profile, Main tier, Level 4.1

Output layer sets: The bitstream contains two output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1

  – Layer 1, PTL Idx 2

Resolution of each layer:

- Layer 0: 960x540 (output), 960x640 (coded)
- Layer 1: 1920x1080 (output), 1920x1280 (coded)

Frame rate: 50 fps

6.6.17.2.3 Test bitstream VUI_B

Specification: All slices are coded as I, P and B slices. The VUI_B bitstream has two layers. The bitstream contains five access units. The value of vps_vui_present_flag is set equal to 1 and VPS VUI syntax structure is signalled. Information about tiles is included in the VUI.

Functional stage: Test VPS VUI.

Purpose: Check that the decoder can properly parse when VPS VUI is present in the bitstream. The decoder is expected to be able to parse the VPS VUI.

Coding structure: Random access

Number of access units: 5

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 4.1 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 3.1
- PTL Idx 2: Scalable Main profile, Main tier, Level 4.1

Output layer sets: The bitstream contains two output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2

Resolution of each layer:

- Layer 0: 960x540 (output), 960x640 (coded)
- Layer 1: 1920x1080 (output), 1920x1280 (coded)

Frame rate: 50 fps

6.6.17.2.4 Test bitstream VUI_C

Specification: All slices are coded as I, P and B slices. The VUI_C bitstream has two layers. The bitstream contains five access units. The value of vps_vui_present_flag is set equal to 1 and VPS VUI syntax structure is signalled. Information about wavefront for each layer is included in the VUI.

Functional stage: Test VPS VUI.

Purpose: Check that the decoder can properly parse when VPS VUI is present in the bitstream. The decoder is expected to be able to parse the VPS VUI.

Coding structure: Random access

Number of access units: 5

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 4.1 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 3.1
- PTL Idx 2: Scalable Main profile, Main tier, Level 4.1

Output layer sets: The bitstream contains two output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1

- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2

Resolution of each layer:

- Layer 0: 960x540 (output), 960x640 (coded)
- Layer 1: 1920x1080 (output), 1920x1280 (coded)

Frame rate: 50 fps

6.6.17.2.5 Test bitstream NONVUI_A

Specification: All slices are coded as I, P and B slices. The NONVUI_A bitstream has two layers. The bitstream contains five access units. The value of vps_non_vui_extension_length is set equal to 100 in the VPS extension indicating the presence of VPS non-VUI extension bytes. The bits forming the non-VUI extension are set at random to 0 or 1.

Functional stage: Test VPS non-VUI extension data.

Purpose: Check that the decoder can properly parse the syntax element vps_non_vui_extension_length and the VPS non-VUI extension bytes when present in the bitstream.

Coding structure: Random access

Number of access units: 5

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 4.1 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 3.1
- PTL Idx 2: Scalable Main profile, Main tier, Level 4.1

Output layer sets: The bitstream contains two output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2

Resolution of each layer:

- Layer 0: 960x540 (output), 960x640 (coded)
- Layer 1: 1920x1080 (output), 1920x1280 (coded)

Frame rate: 50 fps

6.6.17.2.6 Test bitstream NONVUI_B

Specification: All slices are coded as I, P and B slices. The NONVUI_B bitstream has two layers. The bitstream contains five access units. The value of vps_non_vui_extension_length is set equal to 10 in the VPS extension indicating the presence of VPS non-VUI extension bytes. The bits forming the non-VUI extension are set equal to 1.

Functional stage: Test VPS non-VUI extension data.

Purpose: Check that the decoder can properly parse the syntax element vps_non_vui_extension_length and the VPS non-VUI extension bytes when present in the bitstream.

Coding structure: Random access

Number of access units: 5

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 4.1 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 3.1
- PTL Idx 2: Scalable Main profile, Main tier, Level 4.1

Output layer sets: The bitstream contains two output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2

Resolution of each layer:

- Layer 0: 960x540 (output), 960x640 (coded)
- Layer 1: 1920x1080 (output), 1920x1280 (coded)

Frame rate: 50 fps

6.6.17.2.7 Test bitstream NONVUI_C

Specification: All slices are coded as I, P and B slices. The NONVUI_C bitstream has two layers. The bitstream contains five access units. The value of vps_non_vui_extension_length is set equal to 10 in the VPS extension indicating the presence of VPS non-VUI extension bytes. The bits forming the non-VUI extension are set equal to 0.

Functional stage: Test VPS non-VUI extension data.

Purpose: Check that the decoder can properly parse the syntax element vps_non_vui_extension_length and the VPS non-VUI extension bytes when present in the bitstream. Emulation prevention bytes would be inserted due to presence of 0's in the VPS non-VUI extension bytes and the decoder is expected to properly remove the emulation prevention bytes and parse the non-VUI extension.

Coding structure: Random access

Number of access units: 5

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 4.1 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 3.1
- PTL Idx 2: Scalable Main profile, Main tier, Level 4.1

Output layer sets: The bitstream contains two output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2

Resolution of each layer:

- Layer 0: 960x540 (output), 960x640 (coded)
- Layer 1: 1920x1080 (output), 1920x1280 (coded)

Frame rate: 50 fps

6.6.17.3 Test bitstreams – DPB

6.6.17.3.1 Test bitstream DPB_A

Specification: All slices are coded as I, P and B slices. The DPB_A bitstream has two layers and within each layer, there are four temporal sub-layers. The bitstream contains nine access units. The sub-DPB information for the output layer sets are signalled, and the maximum sub-layer DPB size is signalled as four for all sub-layers.

Functional stage: Test sub-DPB behaviour with sizes specified for various sub-DPBs and different maximum sub-layers.

Purpose: Check that the decoder can properly parse the syntax elements in the dpb_size( ) syntax structure and properly decode the bitstream when the sub-layer DPB information is present.

Coding structure: Random access

Number of access units: 9

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 3.1 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 3
- PTL Idx 2: Scalable Main profile, Main tier, Level 3.1

Output layer sets: The bitstream contains two output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2

Resolution of each layer:

- Layer 0: 960x540 (output), 960x544 (coded)
- Layer 1: 1280x720

Frame rate: 30 fps

6.6.17.3.2 Test bitstream DPB_B

Specification: All slices are coded as I and B slices. The DPB_B bitstream has two layers and within each layer there are three temporal sub-layers. The bitstream contains 9 access units. The sub-DPB information for the output layer sets are signalled, and the value of maximum sub-layer DPB size is different for different maximum sub-layers.

Functional stage: Test sub-DPB behaviour with different sizes specified for various sub-DPBs and different maximum sub-layers.

Purpose: Check that the decoder can properly parse the syntax elements in the dpb_size( ) syntax structure and properly decode the bitstream when the sub-layer DPB information is present.

Coding structure: Low Delay B

Number of access units: 9

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 3.1 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 3
- PTL Idx 2: Scalable Main profile, Main tier, Level 3.1

Output layer sets: The bitstream contains two output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2

Resolution of each layer:

- Layer 0: 960x540 (output), 960x544 (coded)
- Layer 1: 1280x720

Frame rate: 30 fps

6.6.17.4 Test bitstreams – Picture resolution

6.6.17.4.1 Test bitstream SRATIOS_A

Specification: All slices are coded as I, P and B slices. The SRATIOS_A bitstream contains 60 access units. Non-trivial values of ScaledReferenceOffset and ReferenceLayerOffset are used for the resampling process: horizontal ratio of approximately 1.492 (936/627) and vertical ratio of approximately 1.487 (516/347).

Functional stage: Test non-trivial scaling ratios for spatial scalability.

Purpose: Check that the decoder can properly parse and decode the bitstream when the scaling ratio is non-trivial. This bitstream also tests the parsing of the syntax elements related to scaled reference layer offsets and reference layer offsets.

Coding structure: Random access

Number of access units: 60

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 4.1 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 3.1
- PTL Idx 2: Scalable Main profile, Main tier, Level 4.1

Output layer sets: The bitstream contains two output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2

Resolution of each layer:

- Layer 0: 1280x720 (output), 1280x720 (coded)
- Layer 1: 1920x1080 (output), 1920x1280 (coded)

Frame rate: 60 fps

6.6.17.4.2 Test bitstream SRATIOS_B

Specification: All slices are coded as I, P and B slices. The SRATIOS_B bitstream contains 50 access units. Default values of ScaledReferenceOffset and ReferenceLayerOffset are used and unusual picture sizes are used for the resampling process: horizontal ratio of approximately 1.168 (1280/1096) and vertical ratio of approximately 1.169 (720/616).

Functional stage: Test non-trivial scaling ratios for spatial scalability.

Purpose: Check that the decoder can properly parse and decode the bitstream when the scaling ratio is non-trivial.

Coding structure: Random access

Number of access units: 50

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 4.1 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 3.1
- PTL Idx 2: Scalable Main profile, Main tier, Level 3.1

Output layer sets: The bitstream contains two output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2

Resolution of each layer:

- Layer 0: 1096x616
- Layer 1: 1280x720

Frame rate: 50 fps

6.6.17.4.3 Test bitstream SNR_A

Specification: All slices are coded as I, P and B slices. The SNR_A bitstream has two layers and they are coded at the same resolution but at different qualities. The bitstream contains four access units. Layer 1 is predicted from Layer 0. The value of DefaultTargetOutputLayerIdc is equal to 1.

Functional stage: Test SNR scalability.

Purpose: Check that the decoder can properly parse and decode the bitstream containing two layers at the same resolution but different qualities.

Coding structure: Low delay B

Number of access units: 4

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 4 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 4
- PTL Idx 2: Scalable Main profile, Main tier, Level 4

Output layer sets: The bitstream contains two output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2

Resolution of each layer:

- Layer 0: 1920x1080 (output), 1920x1080 (coded)
- Layer 1: 1920x1080 (output), 1920x1280 (coded)

Frame rate: 24 fps

6.6.17.4.4 Test bitstream SNR_B

Specification: All slices are coded as I, P and B slices. The SNR_B bitstream has three layers and they are coded at the same resolution but at different qualities. The bitstream contains four access units. Layer 1 is predicted from Layer 0 and Layer 2 is predicted from Layer 1. The value of DefaultTargetOutputLayerIdc is equal to 1.

Functional stage: Test SNR scalability.

Purpose: Check that the decoder can properly parse and decode the bitstream containing two layers at the same resolution but different qualities.

Coding structure: Low delay B

Number of access units: 4

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 4 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 4
- PTL Idx 2: Scalable Main profile, Main tier, Level 4

Output layer sets: The bitstream contains three output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
- OLS 2 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
  - Layer 2, PTL Idx 2

Resolution of each layer:

- Layer 0: 1920x1080 (output), 1920x1080 (coded)

– Layer 1: 1920x1080 (output), 1920x1280 (coded)
– Layer 2: 1920x1080 (output), 1920x1280 (coded)

Frame rate: 24 fps

6.6.17.4.5 Test bitstream SNR_C

Specification: All slices are coded as I, P and B slices. The SNR_C bitstream has three layers and they are coded at the same resolution but at different qualities. The bitstream contains four access units. The three layers are at the same resolution and coded at different qualities. Layer 1 is predicted from Layer 0 and Layer 2 is predicted from Layer 1. The value of DefaultTargetOutputLayerIdc is equal to 1.

Functional stage: Test SNR scalability.

Purpose: Check that the decoder can properly parse and decode the bitstream containing two layers at the same resolution but different qualities.

Coding structure: Low delay B

Number of access units: 4

Profile, tier and level (PTL) information: Three PTL structures are specified:

– PTL Idx 0: Main profile, Main tier, Level 4 (whole bitstream)
– PTL Idx 1: Main profile, Main tier, Level 4
– PTL Idx 2: Scalable Main profile, Main tier, Level 4

Output layer sets: The bitstream contains three output layer sets:

– OLS 0 includes the following layer:
  – Layer 0, PTL Idx 1
– OLS 1 includes the following layers:
  – Layer 0, PTL Idx 1
  – Layer 1, PTL Idx 2
– OLS 1 includes the following layers:
  – Layer 0, PTL Idx 1
  – Layer 1, PTL Idx 2
  – Layer 2, PTL Idx 2

Resolution of each layer:

– Layer 0: 1920x1080 (output), 1920x1080 (coded)
– Layer 1: 1920x1080 (output), 1920x1280 (coded)
– Layer 2: 1920x1080 (output), 1920x1280 (coded)

Frame rate: 24 fps

6.6.17.4.6 Test bitstream REPFMT_A

Specification: All slices are coded as I and P slices. The REPFMT_A bitstream has two layers; one representation format syntax structure in the VPS extension is signalled and both layers are associated with the same representation format syntax structure. The bitstream contains four access units. The value of the syntax element conformance_window_vps_flag is equal to 1 in the rep_format( ) structure.

Functional stage: Test parsing of representation format syntax structure in the VPS extension.

Purpose: Check that the decoder can properly parse and decode the representation format syntax structure(s) specified in the VPS extension.

Coding structure: Low delay P

Number of access units: 4

Profile, tier and level (PTL) information: Three PTL structures are specified:

– PTL Idx 0: Main profile, Main tier, Level 3.1 (whole bitstream)
– PTL Idx 1: Main profile, Main tier, Level 3

– PTL Idx 2: Scalable Main profile, Main tier, Level 3.1

Output layer sets: The bitstream contains two output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2

Resolution of each layer:

- Layer 0: 960x540 (output), 960x544 (coded)
- Layer 1: 960x540 (output), 960x544 (coded)

Frame rate: 30 fps

6.6.17.4.7 Test bitstream REPFMT_B

Specification: All slices are coded as I and P slices. The REPFMT_B bitstream has three layers; two representation format syntax structures are signalled in the VPS extension. The bitstream contains four access units. The value of the syntax element conformance_window_vps_flag is equal to 1 in one of the rep_format( ) structures.

Functional stage: Test parsing of representation format syntax structure in the VPS extension.

Purpose: Check that the decoder can properly parse and decode representation format syntax structure defined in the VPS extension.

Coding structure: Low delay P

Number of access units: 4

Profile, tier and level (PTL) information: Four PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 3.1 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 3
- PTL Idx 2: Scalable Main profile, Main tier, Level 3.1
- PTL Idx 3: Scalable Main profile, Main tier, Level 4

Output layer sets: The bitstream contains three output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
- OLS 2 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
  - Layer 2, PTL Idx 3

Resolution of each layer:

- Layer 0: 960x540 (output), 960x544 (coded)
- Layer 1: 960x540 (output), 960x544 (coded)
- Layer 2: 1920x1080 (coded)

Frame rate: 30 fps

6.6.17.4.8 Test bitstream REPFMT_C

Specification: All slices are coded as I and P slices. The REPFMT_C bitstream has three layers; two representation format syntax structures are signalled in the VPS extension. The bitstream contains four access units. The value of the syntax element conformance_window_vps_flag is equal to 1 in one of the rep_format( ) structure.

Functional stage: Test parsing of representation format syntax structure in the VPS extension.

Purpose: Check that the decoder can properly parse and decode representation format syntax structure specified in the VPS extension.

Coding structure: Low delay P

Number of access units: 4

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 3.1 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 3
- PTL Idx 2: Scalable Main profile, Main tier, Level 4

Output layer sets: The bitstream contains three output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
- OLS 2 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
  - Layer 2, PTL Idx 2

Resolution of each layer:

- Layer 0: 960x540 (output), 960x544 (coded)
- Layer 1: 1920x1080 (coded)
- Layer 2: 1920x1080 (coded)

Frame rate: 30 fps

6.6.17.4.9 Test bitstream RESCHANGE_A

Specification: All slices are coded as I and P slices. The RESCHANGE_A bitstream has three layers; two representation format syntax structures are signalled in the VPS extension. The bitstream contains four access units. The active SPS for Layer 0 signals a resolution that is different from the resolution specified in the rep_format( ) structure associated with Layer 0.

Functional stage: Test parsing of representation format syntax structure in the VPS extension and update of picture resolution in the SPS.

Purpose: Check that the decoder can properly parse and decode the bitstream when picture resolution specified in the SPS is different from picture resolution specified in the VPS.

Coding structure: Low delay P

Number of access units: 4

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 3.1 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 3.1
- PTL Idx 2: Scalable Main profile, Main tier, Level 4

Output layer sets: The bitstream contains two output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2

Resolution of each layer:

– Layer 0: 960x536
– Layer 1: 1920x1080

Frame rate: 30 fps

6.6.17.4.10 Test bitstream ADAPTRES_A

Specification: All slices are coded as I and P slices. The ADAPTRES_A bitstream contains two layers. At the switch point, the picture at the higher layer is coded as an IRAP picture with P_SLICEs using lower layer picture as inter-layer reference for this use case.

Functional stage: Test adaptive resolution change.

Purpose: Check that the decoder can properly parse and decode the bitstream when skip pictures are used to code IRAP pictures such that multiple layers are used to effect adaptive resolution change.

Coding structure: Low delay P

Number of access units: 30

Profile, tier and level (PTL) information: Three PTL structures are specified:

– PTL Idx 0: Main profile, Main tier, Level 3.1 (whole bitstream)
– PTL Idx 1: Main profile, Main tier, Level 3.1
– PTL Idx 2: Scalable Main profile, Main tier, Level 4

Output layer sets: The bitstream contains three output layer sets:

– OLS 0 includes the following layer:
  – Layer 0, PTL Idx 1
– OLS 1 includes the following layers:
  – Layer 0, PTL Idx 1
  – Layer 1, PTL Idx 2
– OLS 2 includes the following layers:
  – Layer 0, PTL Idx 1
  – Layer 1, PTL Idx 2
  – Layer 2, PTL Idx 2

Resolution of each layer:

– Layer 0: 1920x720
– Layer 1: 1920x1080
– Layer 2: 1920x1080

Frame rate: 24 fps

6.6.17.4.11 Test bitstream SPSREPFMT_A

Specification: All slices are coded as I, P and B slices. The SPSREPFMT_A bitstream has two layers; four representation format syntax structures are signalled in the VPS extension. The default inference rule of the representation format is applied to the layers. In the active SPS of Layer 1, the representation format is updated.

Functional stage: Test representation format update in SPS.

Purpose: Check that the decoder can properly parse and decode the bitstream when picture resolution specified in the SPS is different from picture resolution specified in the VPS.

Profile, tier and level (PTL) information: Three PTL structures are specified:

– PTL Idx 0: Main profile, Main tier, Level 5.1 (whole bitstream)
– PTL Idx 1: Main profile, Main tier, Level 5.1
– PTL Idx 2: Scalable Main profile, Main tier, Level 5.1

Output layer sets: The bitstream contains two output layer sets:

– OLS 0 includes the following layer:

– Layer 0, PTL Idx 1

– OLS 1 includes the following layers:

  – Layer 0, PTL Idx 1

  – Layer 1, PTL Idx 2

Resolution of each layer: 960x540

Frame rate: 24 fps

6.6.17.4.12 Test bitstream CONFCROP_A

Specification: All slices are coded as I and P slices. The CONFCROP_A bitstream has two layers and conformance cropping windows are specified for both the layers in the VPS extension. The bitstream contains eight access units.

Functional stage: Test conformance cropping window in VPS extension.

Purpose: Check that the decoder can properly parse and decode the bitstream when conformance cropping windows are specified in the VPS extension, and the output picture is cropped with the right parameters.

Coding structure: Low delay P

Number of access units: 8

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 3.1 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 3.1
- PTL Idx 2: Scalable Main profile, Main tier, Level 5

Output layer sets: The bitstream contains two output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2

Resolution of each layer:

- Layer 0: 960x540 (output), 960x640 (coded)
- Layer 1: 1920x1080 (output), 1920x1280 (coded)

Frame rate: 30 fps

6.6.17.4.13 Test bitstream CONFCROP_B

Specification: All slices are coded as I and P slices. The CONFCROP_B bitstream has two layers and conformance cropping windows are specified for both the layers in the VPS extension. The bitstream contains eight access units.

Functional stage: Test conformance cropping window in VPS extension.

Purpose: Check that the decoder can properly parse and decode the bitstream when conformance cropping windows are specified in the VPS extension, and the output picture is cropped with the right parameters.

Coding structure: Low delay P

Number of access units: 8

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 3.1 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 3
- PTL Idx 2: Scalable Main profile, Main tier, Level 4

Output layer sets: The bitstream contains two output layer sets:

- OLS 0 includes the following layers:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:

- Layer 0, PTL Idx 1
- Layer 1, PTL Idx 2

Resolution of each layer:

- Layer 0: 720x480 (output), 960x536 (coded)
- Layer 1: 1600x900 (output), 1920x1072 (coded)

Frame rate: 30 fps

6.6.17.4.14 Test bitstream CONFCROP_C

Specification: All slices are coded as I and P slices. The CONFCROP_C bitstream has three layers and conformance cropping windows are specified for all the layers in the VPS extension. The bitstream contains four access units.

Functional stage: Test conformance cropping window in VPS extension.

Purpose: Check that the decoder can properly parse and decode the bitstream when conformance cropping windows are specified in the VPS extension, and the output picture is cropped with the right parameters.

Coding structure: Low delay P

Number of access units: 4

Profile, tier and level (PTL) information: Four PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 3.1 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 3
- PTL Idx 2: Scalable Main profile, Main tier, Level 3.1
- PTL Idx 3: Scalable Main profile, Main tier, Level 4

Output layer sets: The bitstream contains three output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
- OLS 2 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
  - Layer 2, PTL Idx 3

Resolution of each layer:

- Layer 0: 720x480 (output), 960x536 (coded)
- Layer 1: 960x540 (output), 1280x720 (coded)
- Layer 2: 1760x900 (output), 1920x1080 (coded)

Frame rate: 30 fps

6.6.17.5 Test bitstreams – Offsets and phase adjustments

6.6.17.5.1 Test bitstream SCREFOFF_A

Specification: All slices are coded as I, P and B slices. The SCREFOFF_A bitstream has two layers. The bitstream contains 20 access units. Layer 0 is a direct reference layer of Layer 1. The scaled reference layer offsets are signalled such that layers have different resolutions and scaling ratio equal to 1.

Functional stage: Test scaled reference layer offsets.

Purpose: Check that the decoder can properly parse the syntax elements related to scaled reference layer offsets in the bitstream and properly decode the bitstream.

Coding structure: Random access

Number of access units: 20

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 3.1 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 3
- PTL Idx 2: Scalable Main profile, Main tier, Level 3.1

Output layer sets: The bitstream contains two output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2

Resolution of each layer:

- Layer 0: 400x400
- Layer 1: 940x540

Frame rate: 50 fps

6.6.17.5.2 Test bitstream REFREGOFF_A

Specification: All slices are coded as I, P and B slices. The REFREGOFF_A bitstream has two layers. The bitstream contains nine access units. Layer 0 is a direct reference layer of Layer 1.The reference region offsets are signalled such that the scaling ratio is equal to 2.

Functional stage: Test reference region offsets in the PPS.

Purpose: Check that the decoder can properly parse the syntax elements related to reference region offsets in the bitstream and properly decode the bitstream.

Coding structure: Random access

Number of access units: 9

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 4.1 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 4.1
- PTL Idx 2: Scalable Main profile, Main tier, Level 4.1

Output layer sets: The bitstream contains two output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2

Resolution of each layer: 1280x800

Frame rate: 30 fps

6.6.17.5.3 Test bitstream RESPHASE_A

Specification: All slices are coded as I, P and B slices. The RESPHASE_A bitstream has two layers. The bitstream contains 60 access units. Layer 0 is a direct reference layer of Layer 1. Non-trivial resampling phase values are signalled.

Functional stage: Test resampling phase.

Purpose: Check that the decoder can properly parse the syntax elements related to resampling phase in the bitstream and properly decode the bitstream.

Coding structure: Random access

Number of access units: 60

Profile, tier and level (PTL) information: Three PTL structures are specified:

– PTL Idx 0: Main profile, Main tier, Level 4.1 (whole bitstream)
– PTL Idx 1: Main profile, Main tier, Level 3.1
– PTL Idx 2: Scalable Main profile, Main tier, Level 4.1

Output layer sets: The bitstream contains two output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2

Resolution of each layer:

- Layer 0: 1280x720 (output), 1280x720 (coded)
- Layer 1: 1920x1080 (output), 1920x1280 (coded)

Frame rate: 60 fps

6.6.17.5.4 Test bitstream OLS_A

Specification: All slices are coded as I and P slices. The OLS_A bitstream has three layers. The bitstream contains four access units. An additional output layer set is signalled in addition to the three output layers sets specified.

Functional stage: Test additional output layer set.

Purpose: Check that the decoder can properly parse the syntax elements associated with additional output layer sets and properly decode the bitstream.

Coding structure: Low delay P

Number of access units: 4

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 4.1 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 3.1
- PTL Idx 2: Scalable Main profile, Main tier, Level 4

Output layer sets: The bitstream contains four output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
- OLS 2 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
  - Layer 2, PTL Idx 2
- OLS 3 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
  - Layer 2, PTL Idx 2

Resolution of each layer: 1280x720

Frame rate: 24 fps

6.6.17.5.5 Test bitstream OLS_B

Specification: All slices are coded as I and P slices. The OLS_B bitstream has three layers. The bitstream contains four access units. The value of default_target_output_layer_idc is set equal to 2.

Functional stage: Test non-default output layer sets.

Purpose: Check that the decoder can properly parse the syntax elements associated with explicitly signalled output layer sets and properly decode the bitstream.

Coding structure: Low delay P

Number of access units: 4

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 4.1 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 3.1
- PTL Idx 2: Scalable Main profile, Main tier, Level 4

Output layer sets: The bitstream contains three output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
- OLS 2 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
  - Layer 2, PTL Idx 2

Resolution of each layer: 1280x720

Frame rate: 24 fps

6.6.17.5.6 Test bitstream OLS_C

Specification: All slices are coded as I and P slices. The OLS_C bitstream has four layers. The bitstream contains four access units. Layers with layer IDs 1 and 2 are not used for inter-layer prediction in some of the output layer sets, thus making them unnecessary layers for prediction purposes for those output layer sets.

Functional stage: Test layers not necessary for prediction.

Purpose: Check that the decoder can properly decode the bitstream when layers not necessary for prediction are part of output layer sets.

Coding structure: Low delay P

Number of access units: 4

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 4.1 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 3.1
- PTL Idx 2: Scalable Main profile, Main tier, Level 4

Output layer sets: The bitstream contains four output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
- OLS 2 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1 (No PTL)
  - Layer 2, PTL Idx 2

– OLS 3 includes the following layers:
  – Layer 0, PTL Idx 1
  – Layer 1 (No PTL)
  – Layer 2 (No PTL)
  – Layer 3, PTL Idx 2

Resolution of each layer: 1280x720

Frame rate: 24 fps

6.6.17.5.7 Test bitstream DISFLAG_A

Specification: All slices are coded as I, P and B slices. The DISFLAG_A bitstream has two layers. The bitstream contains eight access units. Some of the pictures are marked as discardable pictures (discardable_flag set equal to 1).

Functional stage: Test discardable pictures.

Purpose: Check that the decoder can properly parse the syntax element discardable_flag and properly decode the bitstream when discardable pictures are present.

Coding structure: Random access

Number of access units: 8

Profile, tier and level (PTL) information: Three PTL structures are specified:

– PTL Idx 0: Main profile, Main tier, Level 4.1 (whole bitstream)
– PTL Idx 1: Main profile, Main tier, Level 3.1
– PTL Idx 2: Scalable Main profile, Main tier, Level 4.1

Output layer sets: The bitstream contains two output layer sets:

– OLS 0 includes the following layer:
  – Layer 0, PTL Idx 1
– OLS 1 includes the following layers:
  – Layer 0, PTL Idx 1
  – Layer 1, PTL Idx 2

Resolution of each layer:

– Layer 0: 960x540 (output)
– Layer 1: 1920x1080 (output)

Frame rate: 50 fps

6.6.17.6 Test bitstreams – Scaling lists

6.6.17.6.1 Test bitstream PPSSLIST_A

Specification: All slices are coded as I, P, and B slices. The PPSSLIST_A bitstream has two layers. Each picture contains one slice. The value of vps_max_layers_minus1 is set equal to 1 and the value of pps_infer_scaling_list_flag for the enhancement layer is equal to 1.

Functional stage: Test inference of PPS scaling list.

Purpose: Check that the decoder can properly parse the syntax elements associated with inference of scaling list in the PPS, properly infer the scaling list parameter of one PPS from that of another, and properly decode the bitstream.

Profile, tier and level (PTL) information: Three PTL structures are specified:

– PTL Idx 0: Main profile, Main tier, Level 4.1 (whole bitstream)
– PTL Idx 1: Main profile, Main tier, Level 4.1
– PTL Idx 2: Scalable Main profile, Main tier, Level 4.1

Output layer sets: The bitstream contains two output layer sets:

– OLS 0 includes the following layer:
  – Layer 0, PTL Idx 1

- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2

Resolution of each layer:

- Layer 0: 960x540
- Layer 1: 1280x720

6.6.17.6.2 Test bitstream SPSSLIST_A

Specification: All slices are coded as I, P, and B slices. The SPSSLIST_A bitstream has two layers. Each picture contains one slice. The value of vps_max_layers_minus1 is set equal to 1 and the value of sps_infer_scaling_list_flag for the enhancement layer is equal to 1.

Functional stage: Test inference of SPS scaling list.

Purpose: Check that the decoder can properly parse the syntax elements associated with inference of scaling list in the SPS, properly infer the scaling list parameter of one SPS from that of another, and properly decode the bitstream.

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 4.1 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 4.1
- PTL Idx 2: Scalable Main profile, Main tier, Level 4.1

Output layer sets: The bitstream contains two output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2

Resolution of each layer:

- Layer 0: 960x540
- Layer 1: 1280x720

Frame rate: 24 fps

6.6.17.7 Test bitstreams – Colour gamut scalability

6.6.17.7.1 Test bitstream CGS_A

Specification: All slices are coded as I and P slices. The CGS_A bitstream has two layers. The base layer has bit depth equal to 8 and is coded in BT.709 container; the enhancement layer has bit depth equal to 10 and is coded in BT.2020 container. The bitstream contains one access unit. The value of cm_octant_depth is set equal to 1 and the value of split_octant_flag is set equal to 1.

Functional stage: Test colour gamut scalability with 8-bit base layer and 10-bit enhancement layer.

Purpose: Check that the decoder can properly decode the bitstream when the colour gamut scalability is enabled and the base and the enhancement layers have different bit depths.

Number of access units: 1

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 4.1 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 4.1
- PTL Idx 2: Scalable Main 10 profile, Main tier, Level 4.1

Output layer sets: The bitstream contains two output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:

- Layer 0, PTL Idx 1
- Layer 1, PTL Idx 2

Resolution of each layer: 1920x1080

Frame rate: 50 fps

6.6.17.7.2 Test bitstream CGS_B

Specification: All slices are coded as I and P slices. The CGS_B bitstream has two layers. The base layer has bit depth equal to 8 and is coded in BT.709 container; the enhancement layer has bit depth equal to 10 and is coded in BT.2020 container. The bitstream contains one access unit. The value of cm_octant_depth is set equal to 0.

Functional stage: Test colour gamut scalability with 8-bit base layer and 10-bit enhancement layer.

Purpose: Check that the decoder can properly decode the bitstream when the colour gamut scalability is enabled and the base and the enhancement layers have different bit depths.

Number of access units: 1

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 4.1 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 4.1
- PTL Idx 2: Scalable Main 10 profile, Main tier, Level 4.1

Output layer sets: The bitstream contains two output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2

Resolution of each layer: 1920x1080

Frame rate: 50 fps

6.6.17.7.3 Test bitstream CGS_C

Specification: All slices are coded as I and P slices. The CGS_C bitstream has two layers. The base layer has bit depth equal to 10 and is coded in BT.709 container; the enhancement layer has bit depth equal to 10 and is coded in BT.2020 container. The bitstream contains one access unit. The value of cm_octant_depth is set equal to 1 and the value of split_octant_flag is set equal to 1.

Functional stage: Test colour gamut scalability with 10-bit base layer and 10-bit enhancement layer.

Purpose: Check that the decoder can properly decode the bitstream when the colour gamut scalability is enabled and the base and the enhancement layers have the same bit depths.

Number of access units: 1

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main 10 profile, Main tier, Level 4.1 (whole bitstream)
- PTL Idx 1: Main 10 profile, Main tier, Level 4.1
- PTL Idx 2: Scalable Main 10 profile, Main tier, Level 4.1

Output layer sets: The bitstream contains two output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2

Resolution of each layer: 1920x1080

Frame rate: 50 fps

6.6.17.7.4 Test bitstream CGS_D

Specification: All slices are coded as I and P slices. The CGS_D bitstream has two layers. The base layer has bit depth equal to 10 and is coded in BT.709 container; the enhancement layer has bit depth equal to 10 and is coded in BT.2020 container. The bitstream contains one access unit. The value of cm_octant_depth is set equal to 0.

Functional stage: Test colour gamut scalability with 10-bit base layer and 10-bit enhancement layer.

Purpose: Check that the decoder can properly decode the bitstream when the colour gamut scalability is enabled and the base and the enhancement layers have the same bit depths.

Number of access units: 1

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main 10 profile, Main tier, Level 4.1 (whole bitstream)
- PTL Idx 1: Main 10 profile, Main tier, Level 4.1
- PTL Idx 2: Scalable Main 10 profile, Main tier, Level 4.1

Output layer sets: The bitstream contains two output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2

Resolution of each layer: 1920x1080

Frame rate: 50 fps

6.6.17.7.5 Test bitstream CGS_E

Specification: All slices are coded as I and P slices. The CGS_E bitstream has two layers. The base layer has bit depth equal to 8 and is coded in BT.709 container; the enhancement layer has bit depth equal to 10 and is coded in BT.2020 container. The bitstream contains one access unit. The value of cm_octant_depth is set equal to 1 and the value of split_octant_flag is set equal to 1.

Functional stage: Test colour gamut scalability with 8-bit base layer and 10-bit enhancement layer together with spatial scalability.

Purpose: Check that the decoder can properly decode the bitstream when the colour gamut scalability is enabled, the base and the enhancement layers have different bit depths and scaling ratio is not equal to 1.

Number of access units: 1

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 4.1 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 4.1
- PTL Idx 2: Scalable Main 10 profile, Main tier, Level 4.1

Output layer sets: The bitstream contains two output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2

Resolution of each layer:

- Layer 0: 960x540
- Layer 1: 1920x1080

Frame rate: 60 fps

6.6.17.7.6 Test bitstream CGS_F

Specification: All slices are coded as I and P slices. The CGS_F bitstream has two layers. The base layer has bit depth equal to 8 and is coded with BT.709 container; the enhancement layer has bit depth equal to 10 and is coded in BT.2020 container. The bitstream contains one access unit. The value of cm_octant_depth is set equal to 0.

Functional stage: Test colour gamut scalability with 8-bit base layer and 10-bit enhancement layer together with spatial scalability.

Purpose: Check that the decoder can properly decode the bitstream when the colour gamut scalability is enabled, the base and the enhancement layers have different bit depths and scaling ratio is not equal to 1.

Number of access units: 1

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 4.1 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 4.1
- PTL Idx 2: Scalable Main 10 profile, Main tier, Level 4.1

Output layer sets: The bitstream contains two output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2

Resolution of each layer:

- Layer 0: 960x540
- Layer 1: 1920x1080

Frame rate: 60 fps

6.6.17.7.7 Test bitstream CGS_G

Specification: All slices are coded as I, P and B slices. The CGS_G bitstream has two layers. The base layer has bit depth equal to 8 and is coded in BT.709 container; the enhancement layer has bit depth equal to 10 and is coded in BT.2020 container. The bitstream contains one access unit. The value of cm_octant_depth is set equal to 1 and the value of split_octant_flag is set equal to 0.

Functional stage: Test colour gamut scalability with 8-bit base layer and 10-bit in enhancement layer together.

Purpose: Check that the decoder can properly decode the bitstream when the colour gamut scalability is enabled, the base and the enhancement layers have different bit depths and more than one access unit is coded.

Number of access units: 2

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 4.1 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 4.1
- PTL Idx 2: Scalable Main 10 profile, Main tier, Level 4.1

Output layer sets: The bitstream contains two output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2

Resolution of each layer: 1920x1080

Frame rate: 50 fps

6.6.17.7.8 Test bitstream CGS_H

Specification: All slices are coded as I, P and B slices. The CGS_H bitstream has two layers. The base layer has bit depth equal to 10 and is coded with BT.709 container; the enhancement layer has bit depth equal to 10 and is coded in BT.2020 container. The bitstream contains one access unit. The value of cm_octant_depth is set equal to 1 and the value of split_octant_flag is set equal to 0.

Functional stage: Test colour gamut scalability with 10-bit base layer and 10-bit enhancement layer.

Purpose: Check that the decoder can properly decode the bitstream when the colour gamut scalability is enabled, and the base and the enhancement layers have the same bit depths and more than one access unit is coded.

Number of access units: 2

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main 10 profile, Main tier, Level 4.1 (whole bitstream)
- PTL Idx 1: Main 10 profile, Main tier, Level 4.1
- PTL Idx 2: Scalable Main 10 profile, Main tier, Level 4.1

Output layer sets: The bitstream contains two output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2

Resolution of each layer: 1920x1080

Frame rate: 50 fps

6.6.17.7.9 Test bitstream CGS_I

Specification: All slices are coded as I, P and B slices. The CGS_I bitstream has two layers. The base layer has bit depth equal to 8 and is coded in BT.709 container; the enhancement layer has bit depth equal to 10 and is coded in BT.2020 container. The bitstream contains one access unit. The value of cm_octant_depth is set equal to 1 and the value of split_octant_flag is set equal to 0. The two layers have different picture resolutions.

Functional stage: Test colour gamut scalability with 8-bit base layer and 10-bit enhancement layer together with spatial scalability.

Purpose: Check that the decoder can properly decode the bitstream when the colour gamut scalability is enabled, the base and the enhancement layers have different bit depths, scaling ratio is not equal to 1, and more than one access unit is coded.

Number of access units: 2

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 4.1 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 4.1
- PTL Idx 2: Scalable Main 10 profile, Main tier, Level 4.1

Output layer sets: The bitstream contains two output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2

Resolution of each layer:

- Layer 0: 960x540
- Layer 1: 1920x1080

Frame rate: 60 fps

6.6.17.8 Test bitstreams – Additional extensibility

6.6.17.8.1 Test bitstream PSEXT_A

Specification: All slices are coded as I, P and B slices. The PSEXT_A bitstream has two layers. The bitstream contains four access units. The value of pps_extension_6bits is set equal to 0x5 for both layers, and the value of sps_extension_6bits is set equal to 0 and 0x63 for the base and enhancement layers, respectively.

Functional stage: Test additional extensions of SPS and PPS.

Purpose: Check that the decoder can properly parse the PPS and SPS when the value of sps_extension_6bits and pps_extension_6bits are not equal to 0.

Coding structure: Low delay P

Number of access units: 4

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 3.1 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 3.1
- PTL Idx 2: Scalable Main profile, Main tier, Level 3.1

Output layer sets: The bitstream contains two output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2

Resolution of each layer: 1280x720

Frame rate: 30 fps

6.6.17.8.2 Test bitstream LAYERID63_A

Specification: All slices are coded as I, P and B slices. The LAYERID63_A bitstream has two layers. The bitstream contains ten access units. Random bits are added to fill the payload of NAL units that have layer ID equal to 63.

Functional stage: Test nuh_layer_id value of 63.

Purpose: Check that the decoder can properly parse NAL units when the value of nuh_layer_id is equal to 63.

Number of access units: 10

6.6.17.9 Test bitstreams – Picture order count

6.6.17.9.1 Test bitstream POC_A

Specification: All slices are coded as I and P slices. The POC_A bitstream has two layers. The bitstream contains nine access units. The value of poc_reset_idc is set equal to 2 and the value of vps_poc_lsb_aligned_flag is set equal to 1, thus resetting the POC of the pictures.

Functional stage: Test unaligned POC and POC reset.

Purpose: Check that the decoder can properly decode when the POC is reset.

Coding structure: Low delay P

Number of access units: 9

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 3.1 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 3.1
- PTL Idx 2: Scalable Main profile, Main tier, Level 4

Output layer sets: The bitstream contains two output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2

Resolution of each layer: 1280x720

- Layer 0: 1280x720
- Layer 1: 1920x1080

Frame rate: 24 fps

6.6.17.9.2 Test bitstream POC_B

Specification: All slices are coded as I and P slices. The POC_B bitstream has two layers. The bitstream contains ten access units. The value of poc_reset_idc is set equal 1 and the value of vps_poc_lsb_aligned_flag is set equal to 0, thus resetting the MSB of the POC of the pictures.

Functional stage: Test reset of POC MSBs.

Purpose: Check that the decoder can properly decode when the POC MSBs are reset.

Coding structure: Low delay P

Number of access units: 10

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 3.1 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 3.1
- PTL Idx 2: Scalable Main profile, Main tier, Level 4

Output layer sets: The bitstream contains two output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2

Resolution of each layer: 1280x720

- Layer 0: 1280x720
- Layer 1: 1920x1080

Frame rate: 24 fps

6.6.17.10 Test bitstreams – Base layer type

6.6.17.10.1 Test bitstream HYBRID_A

Specification: All slices are coded as P and B slices. The HYBRID_A bitstream has two layers, but only the enhancement layer is present in the coded bitstream and the decoded base layer pictures are expected to be provided externally. The bitstream contains 15 access units.

Functional stage: Test hybrid scalability.

Purpose: Check that the decoder can properly decode when the base layer is externally provided and the enhancement layer predicts from the base layer (hybrid scalability).

Coding structure: Random access

Number of access units: 15

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 4.1 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 3.1

– PTL Idx 2: Scalable Main profile, Main tier, Level 4.1

Output layer sets: The bitstream contains two output layer sets:

- OLS 0 (externally provided)
- OLS 1 includes the following layer:
  - Layer 1, PTL Idx 2

Resolution of each layer:

- Layer 0: 1280x720 (output), 1280x720 (coded)
- Layer 1: 1920x1080 (output), 1920x1280 (coded)

Frame rate: 30 fps

6.6.17.10.2 Test bitstream INBLD_A

Specification: All slices are coded as I or P slices. The INBLD_A bitstream has two layers with layer ID values 2 and 3. Layer with Layer ID value 2 is an independent non-base layer. The layers with layer ID value 0 and 1 are not present in the bitstream.

Functional stage: Test independent non-base layer decoding.

Purpose: Check that the decoder can properly decode the bitstream that does not have a layer with layer ID equal to 0 and is indicated to be an independent non-base layer.

Coding structure: Low delay P

Number of access units: 4

Profile, tier and level (PTL) information: Four PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 4.1 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 3.1
- PTL Idx 2: Scalable Main profile, Main tier, Level 4
- PTL Idx 3: Main profile, Main tier, Level 4 (independent non-base layer PTL)

Output layer sets: The bitstream contains five output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
- OLS 2 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
  - Layer 2, PTL Idx 3
- OLS 3 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
  - Layer 2, PTL Idx 3
  - Layer 3, PTL Idx 2
- OLS 4 includes the following layers:
  - Layer 2, PTL Idx 3 (additional layer set)
  - Layer 3, PTL Idx 2

Resolution of each layer: 1280x720

Frame rate: 24 fps

6.6.17.10.3 Test bitstream SIM_A

Specification: All slices are coded as I or B slices. The SIM_A bitstream has two layers. The bitstream contains eight access units. The enhancement layer is independently coded (without prediction from the base layer). The value of DefaultTargetOutputLayerIdc is set equal to 0.

Functional stage: Test simulcast functionality.

Purpose: Check that the decoder can properly decode the bitstream when the enhancement layer is coded independent of the lower layer(s).

Coding structure: Random access

Number of access units: 8

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 4 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 3.1
- PTL Idx 2: Scalable Main profile, Main tier, Level 4

Output layer sets: The bitstream contains two output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2

Resolution of each layer: 1280x720

- Layer 0: 960x540 (output), 960x544 (coded)
- Layer 1: 1920x1080 (output), 1920x1280(coded)

Frame rate: 24 fps

6.6.17.10.4 Test bitstream SIM_B

Specification: All slices are coded as I and B slices. The SIM_B bitstream has two layers with four temporal sub-layers. The bitstream contains eight access units. The enhancement layer is independently coded (without prediction from the base layer). The value of DefaultTargetOutputLayerIdc is set equal to 2. One additional layer is specified that only contains the enhancement layer.

Functional stage: Test simulcast functionality.

Purpose: Check that the decoder can properly decode the bitstream when the enhancement layer is independent of the lower layers.

Coding structure: Random access

Number of access units: 8

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 4 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 3.1
- PTL Idx 2: Scalable Main profile, Main tier, Level 4

Output layer sets: The bitstream contains three output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
- OLS 2 includes the following layers:
  - Layer 1, PTL Idx 1

Resolution of each layer: 1280x720

– Layer 0: 960x540 (output), 960x544 (coded)
– Layer 1: 1920x1080 (output), 1920x1280(coded)

Frame rate: 24 fps

6.6.17.11 Test bitstreams – Signalling of level information

6.6.17.11.1 Test bitstream SLLEV_A

Specification: All slices are coded as I, P and B slices. The SLLEV_A bitstream has two layers with three temporal sub-layers. The bitstream contains eight access units. The value of the syntax element sub_layer_level_present_flag[ ] is equal to 1 for certain profile, tier, level triplets and the level of the sub-layer representations is explicitly present in the bitstream for those profile, tier, level triplets.

Functional stage: Test parsing level information for temporal sub-layers representations.

Purpose: Check that the decoder can properly parse the syntax elements corresponding to the level information for temporal sub-layer representations, including sub_layer_level_present_flag[ ] and sub_layer_level_idc[ ].

Coding structure: Random access

Number of access units: 8

Profile, tier and level (PTL) information: Three PTL structures are specified:

– PTL Idx 0: Main profile, Main tier, Level 3.1 (whole bitstream)
– PTL Idx 1: Main profile, Main tier, Level 3.1
  – Sub-layer 0: Level 3
  – Sub-layer 1: Level 3
– PTL Idx 2: Scalable Main profile, Main tier, Level 4.1
  – Sub-layer 0: Level 4
  – Sub-layer 1: Level 4

Output layer sets: The bitstream contains two output layer sets:

– OLS 0 includes the following layer:
  – Layer 0, PTL Idx 1
– OLS 1 includes the following layers:
  – Layer 0, PTL Idx 1
  – Layer 1, PTL Idx 2

Resolution of each layer:

– Layer 0: 640x360
– Layer 1: 1280x720

Frame rate: 60 fps

6.6.17.12 Test bitstreams – Auxiliary pictures

6.6.17.12.1 Test bitstream ALPHA_A_BBC

Specification: The ALPHA_A_BBC bitstream has two layers where the enhancement layer is the alpha plane coded as an 8-bit monochrome auxiliary picture. The texture signal is the VenueVu5 sequence described in document JCTVC-K0296.

Functional stage: Test alpha channels coded as auxiliary picture layer.

Purpose: Check that the decoder can properly parse and decode a bitstream containing an auxiliary picture layer.

Profile, tier and level (PTL) information: Three PTL structures are specified:

– PTL Idx 0: Main profile, Main tier, Level 4.1 (whole bitstream)
– PTL Idx 1: Main profile, Main tier, Level 4.1
– PTL Idx 2: Scalable Main profile, Main tier, Level 4.1

Output layer sets: The bitstream contains two output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2

Resolution of each layer:

- Layer 0: 1920x1080
- Layer 1: 1920x1080

Frame rate: 30 fps

6.6.17.12.2 Test bitstream DEPTH_A

Specification: All slices are coded as I, P and B slices. The DEPTH_A bitstream has four layers. The bitstream contains four access units. The layers with layer IDs 0 and 1 are texture layers and the layers with layer IDs 2 and 3 are associated auxiliary depth layers.

Functional stage: Test depth layers coded as auxiliary picture layer

Purpose: Check that the decoder can properly parse and decode a bitstream that contains depth information coded as auxiliary picture layer.

Coding structure: Low delay P

Number of access units: 4

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 4.1 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 3.1
- PTL Idx 2: Scalable Main profile, Main tier, Level 4

Output layer sets: The bitstream contains four output layer sets:

- OLS 0 includes the following layers:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
  - Layer 2, PTL Idx 2
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
  - Layer 2, PTL Idx 2
  - Layer 3, PTL Idx 2

Resolution of each layer: 1024x768

Frame rate: 30 fps

6.6.17.13 Test bitstreams – Layers

6.6.17.13.1 Test bitstream 8LAYERS

Specification: All slices are coded as I, P and B slices. The 8LAYERS bitstream has eight layers. The bitstream contains nine access units.

Functional stage: Test 8 layers in a bitstream

Purpose: Check that the decoder can properly parse and decode a bitstream that contains 8 layers

Coding structure: Random access

Number of access units: 9

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main profile, Main tier, Level 4.1 (whole bitstream)
- PTL Idx 1: Main profile, Main tier, Level 3.1
- PTL Idx 2: Scalable Main profile, Main tier, Level 4.1

Output layer sets: The bitstream contains eight output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
- OLS 2 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
  - Layer 2, PTL Idx 2
- OLS 3 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
  - Layer 2, PTL Idx 2
  - Layer 3, PTL Idx 2
- OLS 4 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
  - Layer 2, PTL Idx 2
  - Layer 3, PTL Idx 2
  - Layer 4, PTL Idx 2
- OLS 5 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
  - Layer 2, PTL Idx 2
  - Layer 3, PTL Idx 2
  - Layer 4, PTL Idx 2
  - Layer 5, PTL Idx 2
- OLS 6 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2
  - Layer 2, PTL Idx 2
  - Layer 3, PTL Idx 2
  - Layer 4, PTL Idx 2
  - Layer 5, PTL Idx 2
  - Layer 6, PTL Idx 2
- OLS 7 includes the following layers:

– Layer 0, PTL Idx 1
– Layer 1, PTL Idx 2
– Layer 2, PTL Idx 2
– Layer 3, PTL Idx 2
– Layer 4, PTL Idx 2
– Layer 5, PTL Idx 2
– Layer 6, PTL Idx 2
– Layer 7, PTL Idx 2

Resolution of each layer:

– Layer 0: 176x144
– Layers 1 to 7: 176x144 or 352x288

6.6.17.14 Test bitstreams – Scalable range extensions

6.6.17.14.1 Test bitstream SREXT_A

Specification: All slices are coded as I, P and B slices. The SREXT_A bitstream has two layers. The bitstream contains four access units. The two layers conform to the Monochrome profile. The value of DefaultTargetOutputLayerIdc is set equal to 1.

Functional stage: Test SNR scalability with 8-bit monochrome layers.

Purpose: Check that the decoder can properly parse and decode a bitstream that contains two 8-bit monochrome layers with inter-layer prediction enabled and spatial scaling ratio equal to 1.

Coding structure: Low delay P

Number of access units: 4

Profile, tier and level (PTL) information: Three PTL structures are specified:

– PTL Idx 0: Monochrome profile, Main tier, Level 4 (whole bitstream)
– PTL Idx 1: Monochrome profile, Main tier, Level 4
– PTL Idx 2: Scalable Monochrome profile, Main tier, Level 4

Output layer sets: The bitstream contains two output layer sets:

– OLS 0 includes the following layer:
  – Layer 0, PTL Idx 1
– OLS 1 includes the following layers:
  – Layer 0, PTL Idx 1
  – Layer 1, PTL Idx 2

Resolution of each layer: 512x512 8-bit monochrome (output), 512x512 8-bit (coded)

Frame rate: 30 fps

6.6.17.14.2 Test bitstream SREXT_B

Specification: All slices are coded as I, P and B slices. The SREXT_B bitstream has two layers. The bitstream contains four access units. The two layers conform to the Monochrome 12 profile. The value of DefaultTargetOutputLayerIdc is set equal to 1.

Functional stage: Test SNR scalability with 12-bit monochrome layers.

Purpose: Check that the decoder can properly parse and decode a bitstream that contains two 12-bit monochrome layers with inter-layer prediction enabled and spatial scaling ratio equal to 1.

Coding structure: Low delay P

Number of access units: 4

Profile, tier and level (PTL) information: Three PTL structures are specified:

– PTL Idx 0: Monochrome 12 profile, Main tier, Level 4 (whole bitstream)

– PTL Idx 1: Monochrome 12 profile, Main tier, Level 4
– PTL Idx 2: Scalable Monochrome 12 profile, Main tier, Level 4

Output layer sets: The bitstream contains two output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2

Resolution of each layer: 512x512 12-bit monochrome (output), 512x512 12-bit (coded)

Frame rate: 30 fps

6.6.17.14.3 Test bitstream SREXT_C

Specification: All slices are coded as I, P and B slices. The SREXT_C bitstream has two layers. The bitstream contains four access units. The two layers conform to the Monochrome 16 profile. The value of DefaultTargetOutputLayerIdc is set equal to 1.

Functional stage: Test SNR scalability with 16-bit monochrome layers.

Purpose: Check that the decoder can properly parse and decode a bitstream that contains two 16-bit monochrome layers with inter-layer prediction enabled and spatial scaling ratio equal to 1.

Coding structure: Low delay P

Number of access units: 4

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Monochrome 16 profile, Main tier, Level 4 (whole bitstream)
- PTL Idx 1: Monochrome 16 profile, Main tier, Level 4
- PTL Idx 2: Scalable Monochrome 16 profile, Main tier, Level 4

Output layer sets: The bitstream contains two output layer sets:

- OLS 0 includes the following layers:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2

Resolution of each layer: 1240x960 12-bit monochrome (output), 1240x960 12-bit (coded)

Frame rate: 30 fps

6.6.17.14.4 Test bitstream SREXT_D

Specification: All slices are coded as I, P and B slices. The SREXT_D bitstream has two layers. The bitstream contains 4 access units. The base layer is monochrome and has bit-depth equal to 8, and the enhancement layer is monochrome and has bit-depth equal to 12. The value of DefaultTargetOutputLayerIdc is set equal to 1.

Functional stage: Test SNR and bit depth scalability with monochrome layers.

Purpose: Check that the decoder can properly parse and decode a bitstream that contains two monochrome layers of different bitdepths, with inter-layer prediction enabled and spatial scaling ratio equal to 1.

Coding structure: Low delay B

Number of access units: 4

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Monochrome profile, Main tier, Level 4 (whole bitstream)
- PTL Idx 1: Monochrome 12 profile, Main tier, Level 4
- PTL Idx 2: Scalable Monochrome 12 profile, Main tier, Level 4

Output layer sets: The bitstream contains two output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2

Resolution of each layer:

- Layer 0: 512x512 8-bit monochrome (output), 512x512 8-bit (output)
- Layer 1: 512x512 10-bit monochrome (output), 512x512 10-bit (output)

Frame rate: 30 fps

6.6.17.14.5 Test bitstream SREXT_E

Specification: All slices are coded as I, P and B slices. The SREXT_E bitstream has two layers. The bitstream contains four access units. Both layers are 4:4:4 layers and have bit-depth equal to 8. The value of DefaultTargetOutputLayerIdc is set equal to 1.

Functional stage: Test SNR scalability with 4:4:4 layers.

Purpose: Check that the decoder can properly parse and decode a bitstream that contains two 4:4:4 layers of different bitdepths, with inter-layer prediction enabled and spatial scaling ratio equal to 1.

Coding structure: Low delay B

Number of access units: 4

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main 4:4:4 profile, Main tier, Level 4 (whole bitstream)
- PTL Idx 1: Main 4:4:4 profile, Main tier, Level 4
- PTL Idx 2: Scalable Main 4:4:4 profile, Main tier, Level 4

Output layer sets: The bitstream contains two output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2

Resolution of each layer: 1280x720, 8 bits, 4:4:4

Frame rate: 30 fps

6.6.17.14.6 Test bitstream SREXT_F

Specification: All slices are coded as I, P and B slices. The SREXT_F bitstream has two layers. The bitstream contains four access units. Both layers are 4:4:4 layers and have bit-depth equal to 8 and different picture resolutions. The value of DefaultTargetOutputLayerIdc is set equal to 1.

Functional stage: Test SNR and spatial scalability with 4:4:4 layers.

Purpose: Check that the decoder can properly parse and decode a bitstream that contains two 4:4:4 layers of different picture resolutions and inter-layer prediction enabled.

Coding structure: Low delay B

Number of access units: 4

Profile, tier and level (PTL) information: Three PTL structures are specified:

- PTL Idx 0: Main 4:4:4 profile, Main tier, Level 4 (whole bitstream)
- PTL Idx 1: Main 4:4:4 profile, Main tier, Level 4
- PTL Idx 2: Scalable Main 4:4:4 profile, Main tier, Level 4

Output layer sets: The bitstream contains two output layer sets:

- OLS 0 includes the following layer:
  - Layer 0, PTL Idx 1
- OLS 1 includes the following layers:
  - Layer 0, PTL Idx 1
  - Layer 1, PTL Idx 2

Resolution of each layer:

- Layer 0: 600x510, 8 bits, 4:4:4
- Layer 1: 1280x720, 8 bits, 4:4:4

Frame rate: 30 fps

6.7 Normative conformance test suites for Rec. ITU-T H.265 | ISO/IEC 23008-2

6.7.1 Bitstreams for Main, Main Still Picture, and Main 10 profiles

Legend:

X – Bitstream is for static and dynamic test

Table 1 – Bitstreams for Main, Main Still Picture, and Main 10 profiles

Categories	Subcategory	Bitstream	File name	Main	Main 10	Main Still Picture	Main tier	Level	Frame rate (Frames/s)
Block structure	Block structure and partitioning	STRUCT_A	STRUCT_A_Samsung_7	X	X		X	5.1 and higher	50
		STRUCT_B	STRUCT_B_Samsung_6	X	X		X	5.1 and higher	50
Intra coding	Intra prediction	IPRED_A	IPRED_A_docomo_2	X	X		X	5.1 and higher	30
		IPRED_B	IPRED_B_Nokia_3			X	X	4.0 and higher	N/A
		IPRED_C	IPRED_C_Mitsubishi_3	X	X		X	3.0 and higher	30
	Constrained intra prediction	CIP_A	CIP_A_Panasonic_3	X	X			4.0 and higher	30
		CIP_B	CIP_B_NEC_3	X	X			2.0 and higher	30
		CIP_C	CIP_C_Panasonic_2	X	X			4.0 and higher	30
Inter coding	Merge	MERGE_A	MERGE_A_TI_3	X	X			2.0 and higher	30
		MERGE_B	MERGE_B_TI_3	X	X			2.0 and higher	30
		MERGE_C	MERGE_C_TI_3	X	X			2.0 and higher	30
		MERGE_D	MERGE_D_TI_3	X	X			2.0 and higher	30
		MERGE_E	MERGE_E_TI_3	X	X			2.0 and higher	30
		MERGE_F	MERGE_F_MTK_4	X	X			4.0 and higher	30
		MERGE_G	MERGE_G_HHI_4	X	X			3.1 and higher	60

Table 1 – Bitstreams for Main, Main Still Picture, and Main 10 profiles

Categories	Subcategory	Bitstream	File name	Main	Main 10	Main Still Picture	Main tier	Level	Frame rate (Frames/s)
	Parallel merge	PMERGE_A	PMERGE_A_TI_3	X	X			2.0 and higher	30
		PMERGE_B	PMERGE_B_TI_3	X	X			2.0 and higher	30
		PMERGE_C	PMERGE_C_TI_3	X	X			2.0 and higher	30
		PMERGE_D	PMERGE_D_TI_3	X	X			2.0 and higher	30
		PMERGE_E	PMERGE_E_TI_3	X	X			2.0 and higher	30
	Motion vector prediction	AMVP_A	AMVP_A_MTK_4	X	X			4.0 and higher	50
		AMVP_B	AMVP_B_MTK_4	X	X			4.0 and higher	50
		AMVP_C	AMVP_C_Samsung_7	X	X			5.1 and higher	30
	Temporal motion vector prediction	TMVP_A	TMVP_A_MS_3	X	X			2.0 and higher	30
	mvd_l1_zero_flag	MVDL1ZERO_A	MVDL1ZERO_A _docomo_4	X	X			4.0 and higher	50
	Motion vector prediction clipping	MVCLIP_A	MVCLIP_A_qualcomm_3	X	X			2.0 and higher	30
	Motion vector pointing to picture edge	MVEDGE_A	MVEDGE_A_qualcomm_3	X	X			2.0 and higher	30
	Weighted prediction	WP_A	WP_A_Toshiba_3	X	X			2.0 and higher	60
		WP_B	WP_B_Toshiba_3	X	X			2.0 and higher	60
Transform and quantization	Residual quadtree	RQT_A	RQT_A_HHI_4	X	X			3.1 and higher	60
		RQT_B	RQT_B_HHI_4	X	X			3.1 and higher	60
		RQT_C	RQT_C_HHI_4	X	X			3.1 and higher	60
		RQT_D	RQT_D_HHI_4	X	X			3.1 and higher	60
		RQT_E	RQT_E_HHI_4	X	X			3.1 and higher	60
		RQT_F	RQT_F_HHI_4	X	X		X	3.1 and higher	60
		RQT_G	RQT_G_HHI_4	X	X		X	3.1 and higher	60
		TUSIZE_A	TUSIZE_A_Samsung_1	X	X		X	5.0 and higher	30

Table 1 – Bitstreams for Main, Main Still Picture, and Main 10 profiles

Categories	Subcategory	Bitstream	File name	Main	Main 10	Main Still Picture	Main tier	Level	Frame rate (Frames/s)
	Quantization	DELTAQP_A	DELTAQP_A_BRCM_4	X	X		X	5.0 and higher	24
		DELTAQP_B	DELTAQP_B_SONY_3	X	X			4.0 and higher	30
		DELTAQP_C	DELTAQP_C_SONY_3	X	X			4.0 and higher	30
		INITQP_A	INITQP_A_Sony_1	X	X			4.0 and higher	30
	Scaling list	SLIST_A	SLIST_A_Sony_5	X	X		X	4.0 and higher	60
		SLIST_B	SLIST_B_Sony_9	X	X		X	4.0 and higher	60
		SLIST_C	SLIST_C_Sony_4	X	X		X	4.0 and higher	60
		SLIST_D	SLIST_D_Sony_9	X	X		X	4.0 and higher	60
In-loop filter	Deblocking filter	DBLK_A	DBLK_A_SONY_3	X	X			4.0 and higher	30
		DBLK_B	DBLK_B_SONY_3	X	X			4.0 and higher	30
		DBLK_C	DBLK_C_SONY_3	X	X			4.0 and higher	30
		DBLK_D	DBLK_D_VIXS_2	X	X		X	4.1 and higher	60
		DBLK_E	DBLK_E_VIXS_2	X	X		X	4.1 and higher	60
		DBLK_F	DBLK_F_VIXS_2	X	X		X	4.1 and higher	60
		DBLK_G	DBLK_G_VIXS_2	X	X		X	4.1 and higher	60
	Sample adaptive offset (SAO)	SAO_A	SAO_A_MediaTek_4	X	X		X	4.0 and higher	60
		SAO_B	SAO_B_MediaTek_5	X	X		X	4.0 and higher	60
		SAO_C	SAO_C_Samsung_5	X	X		X	4.1 and higher	60
		SAO_D	SAO_D_Samsung_5	X	X		X	4.1 and higher	60
		SAO_E	SAO_E_Canon_4	X	X		X	4.0 and higher	50
		SAO_F	SAO_F_Canon_3	X	X		X	4.0 and higher	50
		SAO_G	SAO_G_Canon_3	X	X		X	6.2	50
		SAO_H	SAO_H_Parabola_1	X	X		X	6.0 and higher	30
		SAODBLK_A	SAODBLK_A_MainConcept_4	X	X		X	4.1 and higher	29.97
		SAODBLK_B	SAODBLK_B_MainConcept_4	X	X		X	4.1 and higher	29.97

Table 1 – Bitstreams for Main, Main Still Picture, and Main 10 profiles

Categories	Subcategory	Bitstream	File name	Main	Main 10	Main Still Picture	Main tier	Level	Frame rate (Frames/s)
Entropy coding	Maximum bins	MAXBINS_A	MAXBINS_A_TI_4	X	X		X	2.0 and higher	30
		MAXBINS_B	MAXBINS_B_TI_4	X	X		X	2.0 and higher	30
		MAXBINS_C	MAXBINS_C_TI_4	X	X		X	2.0 and higher	30
	CABAC initialization	CAINIT_A	CAINIT_A_SHARP_4	X	X		X	3.0 and higher	50
		CAINIT_B	CAINIT_B_SHARP_4	X	X		X	3.0 and higher	50
		CAINIT_C	CAINIT_C_SHARP_3	X	X		X	3.0 and higher	50
		CAINIT_D	CAINIT_D_SHARP_3	X	X		X	3.0 and higher	50
		CAINIT_E	CAINIT_E_SHARP_3	X	X		X	3.0 and higher	50
		CAINIT_F	CAINIT_F_SHARP_3	X	X		X	3.0 and higher	50
		CAINIT_G	CAINIT_G_SHARP_3	X	X		X	3.1 and higher	50
		CAINIT_H	CAINIT_H_SHARP_3	X	X		X	3.1 and higher	50
	Sign data hiding	SDH_A	SDH_A_Orange_4	X	X			4.1 and higher	50
Temporal scalability	Temporal scalability	TSCL_A	TSCL_A_VIDYO_5	X	X		X	2.1 and higher	50
		TSCL_B	TSCL_B_VIDYO_4	X	X		X	2.1 and higher	50
Parallel processing tools	Tiles	TILES_A	TILES_A_Cisco_2	X	X		X	4.1 and higher	60
		TILES_B	TILES_B_Cisco_1	X	X		X	4.1 and higher	60
	Entropy coding synchronization	WPP_A	WPP_A_ericsson_ MAIN_2	X	X		X	2.0 and higher	50
		WPP_B	WPP_B_ericsson_ MAIN_2	X	X		X	2.0 and higher	50
		WPP_C	WPP_C_ericsson_ MAIN_2	X	X		X	2.0 and higher	50
		WPP_D	WPP_D_ericsson_ MAIN_2	X	X		X	2.0 and higher	50
		WPP_E	WPP_E_ericsson_ MAIN_2	X	X		X	2.0 and higher	50
		WPP_F	WPP_F_ericsson_ MAIN_2	X	X		X	2.0 and higher	50
	Entry point	ENTP_A	ENTP_A_QUALCOMM_1	X	X		X	4.1 and higher	60
		ENTP_B	ENTP_B_Qualcomm_1	X	X		X	4.1 and higher	60
		ENTP_C	ENTP_C_Qualcomm_1	X	X		X	4.1 and higher	60

Table 1 – Bitstreams for Main, Main Still Picture, and Main 10 profiles

Categories	Subcategory	Bitstream	File name	Main	Main 10	Main Still Picture	Main tier	Level	Frame rate (Frames/s)
Other coding tools	Pulse-code modulation (PCM)	IPCM_A	IPCM_A_NEC_3	X	X		X	2.0 and higher	30
		IPCM_B	IPCM_B_NEC_3	X	X		X	2.0 and higher	30
		IPCM_C	IPCM_C_NEC_3	X	X		X	2.0 and higher	30
		IPCM_D	IPCM_D_NEC_3	X	X		X	2.0 and higher	30
		IPCM_E	IPCM_E_NEC_2	X	X		X	2.0 and higher	30
	Transform skip	TS_A	TSKIP_A_MS_3	X	X		X	3.1 and higher	30
	Asymmetric motion partition (AMP)	AMP_A	AMP_A_Samsung_7	X	X		X	5.1 and higher	30
		AMP_B	AMP_B_Samsung_7	X	X		X	5.1 and higher	30
		AMP_D	AMP_D_Hisilicon_3	X	X		X	6.2 and higher	24
		AMP_E	AMP_E_Hisilicon_3	X	X		X	6.2 and higher	50
		AMP_F	AMP_F_Hisilicon_3	X	X		X	6.2 and higher	60
	Transform/quantization/filtering bypass	LS_A	LS_A_Orange_2	X	X		X	5.0 and higher	30
		LS_B	LS_B_Orange_4	X	X		X	5.0 and higher	30
High level syntax	NAL unit types	NUT_A	NUT_A_ericsson_5	X	X		X	3.0 and higher	30
		FILLER_A	FILLER_A_Sony_1	X	X			4.0 and higher	30
	Video Parameter Set (VPS)	VPSID_A	VPSID_A_VIDYO_2	X	X		X	3.1 and higher	50
		PS_B	PS_B_VIDYO_3	X	X		X	2.1 and higher	50
		VPSSPSPPS_A	VPSSPSPPS_A_MainConcept_1	X	X		X	4.1 and higher	29.97
	Picture parameter set (PPS)	PPS_A	PPS_A_qualcomm_7	X	X		X	6.2 and higher	30
	Sub layer	SLPPLP_A	SLPPLP_A_VIDYO_2	X	X		X	3.1 and higher	50

Table 1 – Bitstreams for Main, Main Still Picture, and Main 10 profiles

Categories	Subcategory	Bitstream	File name	Main	Main 10	Main Still Picture	Main tier	Level	Frame rate (Frames/s)
	Picture output control	OPFLAG_A	OPFLAG_A_Qualcomm_1	X	X		X	2.1 and higher	50
		OPFLAG_B	OPFLAG_B_Qualcomm_1	X	X		X	3.1 and higher	60
		OPFLAG_C	OPFLAG_C_Qualcomm_1	X	X		X	3.1 and higher	60
		NoOutPrior_A	NoOutPrior_A_Qualcomm_1	X	X		X	3.1 and higher	60
		NoOutPrior_B	NoOutPrior_B_Qualcomm_1	X	X		X	3.1 and higher	60
	Picture size	PICSIZE_A	PICSIZE_A_Bossen_1	X	X		X	5.1 and higher	50
		PICSIZE_B	PICSIZE_B_Bossen_1	X	X		X	5.1 and higher	50
		PICSIZE_C	PICSIZE_C_Bossen_1	X	X		X	4.1 and higher	50
		PICSIZE_D	PICSIZE_D_Bossen_1	X	X		X	4.1 and higher	50
	Picture order count	POC_A	POC_A_Bossen_3	X	X		X	4.0 and higher	50
	Random access	RAP_A	RAP_A_docomo_6	X	X		X	2.0 and higher	30
		RAP_B	RAP_B_Bossen_2	X	X		X	6.2	50
	Reference Picture Set (RPS)	RPS_A	RPS_A_docomo_5	X	X		X	2.0 and higher	30
		RPS_B	RPS_B_qualcomm_5	X	X		X	3.0 and higher	30
		RPS_C	RPS_C_ericsson_5	X	X		X	3.0 and higher	30
		RPS_D	RPS_D_ericsson_6	X	X		X	3.0 and higher	30
		RPS_E	RPS_E_qualcomm_5	X	X		X	3.0 and higher	30
		RPS_F	RPS_F_docomo_2	X	X		X	6.2	30
	Long term reference	LTRPSPS	LTRPSPS_A_Qualcomm_1	X	X		X	2.1 and higher	50
	Reference picture list modification	RPLM_A	RPLM_A_qualcomm_4	X	X		X	2.0 and higher	30
		RPLM_B	RPLM_B_qualcomm_4	X	X		X	2.0 and higher	30
	Slice type	SLICES_A	SLICES_A_Rovi_3	X	X		X	6.2	30
	Dependent slice	DSLICE_A	DSLICE_A_HHI_5	X	X		X	3.1 and higher	24
		DSLICE_B	DSLICE_B_HHI_5	X	X		X	3.1 and higher	24
		DSLICE_C	DSLICE_C_HHI_5	X	X		X	3.1 and higher	24
	Decoded picture buffer (DPB)	BUMPING_A	BUMPING_A_ericsson_1	X	X		X	3.0 and higher	30

Table 1 – Bitstreams for Main, Main Still Picture, and Main 10 profiles

Categories	Subcategory	Bitstream	File name	Main	Main 10	Main Still Picture	Main tier	Level	Frame rate (Frames/s)
	Conformance window	CONFWIN_A	CONFWIN_A_Sony_1	X	X			4.0 and higher	30
	Hypothetical reference decoder (HRD)	HRD_A	HRD_A_Fujitsu_3	X	X		X	6.2	50
	Extensions	EXT_A	EXT_A_ericsson_4	X	X		X	3.0 and higher	30
10 bit	Weighted prediction	WP_A_MAIN10	WP_A_MAIN10_ Toshiba_3		X		X	2.0 and higher	60
		WP_B_MAIN10	WP_B_MAIN10_ Toshiba_3		X		X	2.0 and higher	60
	Transform Skip	TSUNEQBD_A_ MAIN10	TSUNEQBD_A_MAIN10_ Technicolor_2		X		X	5.1 and higher	30
	Deblocking filter	DBLK_A_MAIN1 0	DBLK_A_MAIN10_ VIXS_4		X		X	4.0 and higher	30
	Quantization	INITQP_B_Main1 0	INITQP_B_Main10_Sony_1		X			4.0 and higher	30
	Entropy coding synchronization	WPP_A_MAIN10	WPP_A_ericsson_MAIN10_ _2		X		X	2.0 and higher	50
		WPP_B_MAIN10	WPP_B_ericsson_MAIN10_ _2		X		X	2.0 and higher	50
		WPP_C_MAIN10	WPP_C_ericsson_MAIN10_ _2		X		X	2.0 and higher	50
		WPP_D_MAIN10	WPP_D_ericsson_MAIN10_ _2		X		X	2.0 and higher	50
		WPP_E_MAIN10	WPP_E_ericsson_MAIN10_ _2		X		X	2.0 and higher	50
		WPP_F_MAIN10	WPP_F_ericsson_MAIN10_ _2		X		X	2.0 and higher	50

6.7.2 Bitstreams for Multiview Main profile

Legend:

X – Bitstream that a decoder conforming to the Main tier needs to decode for static and dynamic test

Table 2 – Bitstreams for Multiview Main profile

Categories	Subcategory	Bitstream	File name	Base layer profile: Main	Base layer profile: Main 10	Base layer profile: Main Still Picture	Base layer profile: Main tier	Level	Frame rate (Frames/sec)
Prediction Structure	Inter-view prediction	MVHEVCS-A	MVHEVCS_A_Qualcomm_3	X			X	4 and higher	30
(2-view)	All intra	MVHEVCS-B	MVHEVCS_B_Sharp_3	X			X	4 and higher	30
	Simulcast	MVHEVCS-C	MVHEVCS_C_Sony_3	X			X	4 and higher	30
	Simulcast with asymmetric resolutions	MVHEVCS-D	MVHEVCS_D_NTT_3	X			X	5.1 and higher	30
	Inter-view prediction and hierarchical B	MVHEVCS-E	MVHEVCS_E_Qualcomm_3	X			X	4 and higher	30
	Inter-view prediction for IRAP AUs only	MVHEVCS-F	MVHEVCS_F_Qualcomm_3	X			X	4 and higher	30
Prediction Structure (3-view)	Inter-view prediction PIP view structure	MVHEVCS-G	MVHEVCS_G_NTT_3	X			X	5.1 and higher	30
	Inter-view prediction with IBP view structure	MVHEVCS-H	MVHEVCS_H_LGE_3	X			X	5.1 and higher	30
	Inter-view prediction with IBP view structure and auxiliary depth	MVHEVCS-I	MVHEVCS_I_Nokia_3	X			X	5.1 and higher	30

6.7.3 Bitstreams for 3D Main profile

Legend:

X – Bitstream that a decoder conforming to the Main tier needs to decode for static and dynamic test

Table 3 – Bitstreams for 3D Main profile

Categories	Subcategory	Bitstream	File name	Base layer profile: Main	Main10	Main Still Picture	Main tier	Level	Frame rate (Frames/sec)
Texture tool	ARP	3DHC_T_A	3DHC_T_A_Qualcomm_3	X			X	4 and higher	30
	Sub-PU inter-view motion prediction	3DHC_T_B	3DHC_T_B_MediaTek_3	X			X	5.1 and higher	30
	Illumination compensation	3DHC_T_C	3DHC_T_C_Sharp_3	X			X	4 and higher	30
	Combined texture	3DHC_T_D	3DHC_T_D_Sharp_3	X			X	4 and higher	30
	Combined texture only bitstream	3DHC_T_E	3DHC_T_E_HHI_3	X			X	5.1 and higher	30
Depth tool	Depth intra wedge	3DHC_D1_A	3DHC_D1_A_HHI_3	X			X	5.1 and higher	30
	Depth intra wedge	3DHC_D1_B	3DHC_D1_B_HHI_3	X			X	5.1 and higher	30
	Intra SDC	3DHC_D1_C	3DHC_D1_C_RWTH_3	X			X	5.1 and higher	30
	Intra SDC	3DHC_D1_D	3DHC_D1_D_RWTH_3	X			X	5.1 and higher	30
	Depth intra skip	3DHC_D1_E	3DHC_D1_E_MediaTek_3	X			X	5.1 and higher	30
	Depth intra skip	3DHC_D1_F	3DHC_D1_F_MediaTek_3	X			X	5.1 and higher	30
	Combined depth	3DHC_D1_G	3DHC_D1_G_Hisilicon_3	X			X	5.1 and higher	30
	Combined depth	3DHC_D1_H	3DHC_D1_H_Hisilicon_3	X			X	5.1 and higher	30
	Inter-view motion	3DHC_D2_A	3DHC_D2_A_Samsung_3	X			X	5.1 and higher	30
	Inter SDC	3DHC_D2_B	3DHC_D2_B_LGE_3	X			X	5.1 and higher	30
Depth dependent texture tool	DoNBDV	3DHC_DT_A	3DHC_DT_A_MediaTek_3	X			X	5.1 and higher	30
	VSP	3DHC_DT_B	3DHC_DT_B_NTT_3	X			X	5.1 and higher	30
	DBBP	3DHC_DT_C	3DHC_DT_C_Hisilicon_3	X			X	5.1 and higher	30

Table 3 – Bitstreams for 3D Main profile

Categories	Subcategory	Bitstream	File name	Base layer profile: Main	Base layer profile: Main10	Base layer profile: Main Still Picture	Main tier	Level	Frame rate (Frames/sec)
	Combined	3DHC_DT_D	3DHC_DT_D_NTT_3	X			X	5.1 and higher	30
Texture dependent depth tool	Sub-PU MPI	3DHC_TD_A	3DHC_TD_A_Qualcomm_3	X			X	4 and higher	30
	MPI	3DHC_TD_B	3DHC_TD_B_MediaTek_3	X			X	5.1 and higher	30
	QTL	3DHC_TD_C	3DHC_TD_C_MediaTek_3	X			X	5.1 and higher	30
	Depth intra contour prediction	3DHC_TD_D	3DHC_TD_D_HHI_3	X			X	5.1 and higher	30
	Depth intra contour prediction	3DHC_TD_E	3DHC_TD_E_HHI_3	X			X	5.1 and higher	30
Others	3-view random access	3DHC_C_A	3DHC_C_A_HHI_3	X			X	5.1 and higher	30
	3-view all Intra	3DHC_C_B	3DHC_C_B_HHI_3	X			X	5.1 and higher	30
	2-view random access	3DHC_C_C	3DHC_C_C_Sharp_3	X			X	4 and higher	30

6.7.4 Bitstreams for format range extensions and high throughput profiles

Legend:

X – Bitstream that a decoder conforming to the Main tier needs to decode for static and dynamic test

Table 4 – Bitstreams for format range extensions and high throughput profiles

Categories	Subcategory	Bitstream	File name	Profile	Main tier	Level	Frame rate (Frames/sec)
Intra coding	Intra chroma prediction angle	ADJUST_IPRED_ANGLE_A	ADJUST_IPRED_ANGLE_A_RExt_Mitsubishi_2	Main 4:2:2 10	X	6.2	24
Inter coding	Cross component prediction	CCP_8bit_RExt	CCP_8bit_RExt_QCOM_1	Main 4:4:4	X	4.1 and higher	30
		CCP_10bit_RExt	CCP_10bit_RExt_QCOM_1	Main 4:4:4 10	X	4.1 and higher	24
		CCP_12bit_RExt	CCP_12bit_RExt_QCOM_1	Main 4:4:4 12	X	4.1 and higher	30
Bit depth	Different bit depth for luma and chroma	Bitdepth_A_RExt	Bitdepth_A_RExt_Sony_1	Main 4:4:4 12	X	4.1 and higher	60
		Bitdepth_B_RExt	Bitdepth_B_RExt_Sony_1	Main 4:4:4 12	X	4.1 and higher	60
Quantization	Scaling list	QMATRIX_A_RExt	QMATRIX_A_RExt_Sony_1	Main 4:4:4	X	4.0 and higher	20
Loop filter	SAO	SAO_A_RExt	SAO_A_RExt_MediaTek_1	Main 4:4:4 12	X	6.2	30
Entropy coding	Persistent Rice parameter tool	PERSIST_RPARAM_A_RExt	PERSIST_RPARAM_A_RExt_Sony_3	Main 4:4:4 12 Intra	X	3.0 and higher	
Precision	Extended precision	HIGH_TP_8BIT_RExt	EXTPREC_HIGHTHROUGHPUT_444_16_INTRA_8BIT_RExt_Sony_1	High Throughput 4:4:4 16 Intra	X	3.0 and higher	
		HIGH_TP_10BIT_RExt	EXTPREC_HIGHTHROUGHPUT_444_16_INTRA_10BIT_RExt_Sony_1	High Throughput 4:4:4 16 Intra	X	3.0 and higher	
		HIGH_TP_12BIT_RExt	EXTPREC_HIGHTHROUGHPUT_444_16_INTRA_12BIT_RExt_Sony_1	High Throughput 4:4:4 16 Intra	X	3.0 and higher	
		HIGH_TP_16BIT_RExt	EXTPREC_HIGHTHROUGHPUT_444_16_INTRA_16BIT_RExt_Sony_1	High Throughput 4:4:4 16 Intra	X	3.0 and higher	
		HIGH_TP_8BIT_RExt	EXTPREC_MAIN_444_16_INTRA_8BIT_RExt_Sony_1	Main 4:4:4 16 Intra	X	3.0 and higher	
		HIGH_TP_10BIT_RExt	EXTPREC_MAIN_444_16_INTRA_10BIT_RExt_Sony_1	Main 4:4:4 16 Intra	X	3.0 and higher	
		HIGH_TP_12BIT_RExt	EXTPREC_MAIN_444_16_INTRA_12BIT_RExt_Sony_1	Main 4:4:4 16 Intra	X	3.0 and higher	
		HIGH_TP_16BIT_RExt	EXTPREC_MAIN_444_16_INTRA_16BIT_RExt_Sony_1	Main 4:4:4 16 Intra	X	3.0 and higher	
Others	PCM	IPCM_A_RExt	IPCM_A_RExt_NEC_2	Main 4:2:2 10	X	6.0 and higher	30
		IPCM_B_RExt	IPCM_B_RExt_NEC_1	Main 4:2:2 10	X	6.0 and higher	30
	Transform skip context	TSCTX_8bit_I_RExt	TSCTX_8bit_I_RExt_SHARP_1	Main 4:4:4	X	6.2	30
		TSCTX_8bit_RExt	TSCTX_8bit_RExt_SHARP_1	Main 4:4:4	X	6.2	30

Table 4 – Bitstreams for format range extensions and high throughput profiles

Categories	Subcategory	Bitstream	File name	Profile	Main tier	Level	Frame rate (Frames/sec)
		TSCTX_10bit_I_RExt	TSCTX_10bit_I_RExt_SHARP_1	Main 4:4:4 10	X	6.2	30
		TSCTX_10bit_RExt	TSCTX_10bit_RExt_SHARP_1	Main 4:4:4 10	X	6.2	30
		TSCTX_12bit_I_RExt	TSCTX_12bit_I_RExt_SHARP_1	Main 4:4:4 12	X	6.2	30
		TSCTX_12bit_RExt	TSCTX_12bit_RExt_SHARP_1	Main 4:4:4 12	X	6.2	30
	RDPCM	ExplicitRdpcm_A_RExt	ExplicitRdpcm_A_BBC_1	Main 4:4:4 12	X	6.2	60
		ExplicitRdpcm_B_RExt	ExplicitRdpcm_B_BBC_2	Main 4:4:4 12	X	6.2	30
	Various combination	Main_422_10_A_RExt	Main_422_10_A_RExt_Sony_2	Main 4:2:2 10	X	4.0 and higher	24
		Main_422_10_B_RExt	Main_422_10_B_RExt_Sony_2	Main 4:2:2 10	X	5.0 and higher	30
		GENERAL_8b_400_RExt	GENERAL_8b_400_RExt_Sony_1	Monochrome	X	3.0 and higher	
		GENERAL_8b_420_RExt	GENERAL_8b_420_RExt_Sony_1	Main Intra	X	3.0 and higher	
		GENERAL_8b_444_RExt	GENERAL_8b_444_RExt_Sony_2	Main 4:4:4 Intra	X	3.0 and higher	
		GENERAL_10b_420_RExt	GENERAL_10b_420_RExt_Sony_1	Main 10 Intra	X	3.0 and higher	
		GENERAL_10b_422_RExt	GENERAL_10b_422_RExt_Sony_1	Main 4:2:2 10 Intra	X	3.0 and higher	
		GENERAL_10b_444_RExt	GENERAL_10b_444_RExt_Sony_2	Main 4:4:4 10 Intra	X	3.0 and higher	
		GENERAL_12b_400_RExt	GENERAL_12b_400_RExt_Sony_1	Monochrome 12	X	3.0 and higher	
		GENERAL_12b_420_RExt	GENERAL_12b_420_RExt_Sony_1	Main 12 Intra	X	3.0 and higher	
		GENERAL_12b_422_RExt	GENERAL_12b_422_RExt_Sony_1	Main 4:2:2 12 Intra	X	3.0 and higher	
		GENERAL_12b_444_RExt	GENERAL_12b_444_RExt_Sony_4	Main 4:4:4 12 Intra	X	3.0 and higher	
		GENERAL_16b_400_RExt	GENERAL_16b_400_RExt_Sony_1	Monochrome 16	X	3.0 and higher	
		GENERAL_16b_444_RExt	GENERAL_16b_444_RExt_Sony_2	Main 4:4:4 16 Intra	X	3.0 and higher	
		GENERAL_16b_444_ighThroughput_RExt	GENERAL_16b_444_highThroughput_RExt_Sony_2	High Throughput 4:4:4 16 Intra	X	3.0 and higher	
		WAVETILES_RExt	WAVETILES_RExt_Sony_2	HighThroughput 4:4:4 16b Intra	X	3.0 and higher	

6.7.5 Bitstreams for Scalable Main and Scalable Main 10 profiles

Legend:

X – Bitstream that a decoder conforming to the Main tier needs to decode for static and dynamic test

Table 5 – Bitstreams for Scalable Main and Scalable Main 10 profiles

				Base layer profile					
Categories	**Subcategory**	**Bitstream**	**File name**	**Main**	**Main 10**	**Main Still**	**Main tier**	**Level**	**Frame rate (Frames/sec)**
Layer dependencies	Layer ID	LAYERID_A	LAYERID_A_NOKIA_2	X			X	3.1 and higher	24
	Motion vector dependency	MVD_A_IDCC	MVD_A_IDCC_1	X			X	3.1 and higher	24
		MVD_A_NOKIA	MVD_A_NOKIA_1	X			X	3.1 and higher	24
	Maximum temporal ID	MAXTID_A	MAXTID_A_ETRI_2	X			X	3.1 and higher	24
		MAXTID_B	MAXTID_B_ETRI_2	X			X	3.1 and higher	24
		MAXTID_C	MAXTID_C_ETRI_2	X			X	3.1 and higher	24
	Inactive reference layers	INACTIVE_A	INACTIVE_A_QCOM_1	X			X	2.1 and higher	50
	Reference layers	REFLAYER_A	REFLAYER_A_VIDYO_2	X			X	3 and higher	24
		REFLAYER_B	REFLAYER_B_VIDYO_2	X			X	3 and higher	24
		REFLAYER_C	REFLAYER_C_VIDYO_2	X			X	3 and higher	24
		REFLAYER_D	REFLAYER_D_VIDYO_2	X			X	3 and higher	24
VPS syntax	Split flag	SPLITFLAG_A	SPLITFLAG_A_HHI_1	X			X	3.1 and higher	24
	VUI	VUI_A	VUI_A_QUALCOMM_1	X			X	3.1 and higher	50
		VUI_B	VUI_B_QUALCOMM_1	X			X	3.1 and higher	50
		VUI_C	VUI_C_QUALCOMM_1	X			X	3.1 and higher	50
	Non-VUI	NONVUI_A	NONVUI_A_QUALCOMM_1	X			X	3.1 and higher	50
		NONVUI_B	NONVUI_B_QUALCOMM_1	X			X	3.1 and higher	50
		NONVUI_C	NONVUI_C_QUALCOMM_1	X			X	3.1 and higher	50
	DPB	DPB_A	DPB_A_VIDYO_2	X			X	3 and higher	30
		DPB_B	DPB_B_VIDYO_2	X			X	3 and higher	30
Picture resolution	Scalability ratios	SRATIOS_A	SRATIOS_A_SAMSUNG_3	X			X	3.1 and higher	60
		SRATIOS_B	SRATIOS_B_SAMSUNG_2	X			X	3.1 and higher	50
	SNR scalability	SNR_A	SNR_A_IDCC_1	X			X	4 and higher	24

Table 5 – Bitstreams for Scalable Main and Scalable Main 10 profiles

Categories	Subcategory	Bitstream	File name	Base layer profile				Level	Frame rate (Frames/sec)
				Main	**Main 10**	**Main Still**	**Main tier**		
		SNR_B	SNR_B_IDCC_1	X			X	4 and higher	24
		SNR_C	SNR_C_IDCC_1	X			X	4 and higher	24
	VPS representation format	REPFMT_A	REPFMT_A_VIDYO_2	X			X	3 and higher	30
		REPFMT_B	REPFMT_B_VIDYO_2	X			X	3 and higher	30
		REPFMT_C	REPFMT_C_VIDYO_2	X			X	3 and higher	30
	Resolution change	RESCHANGE_A	RESCHANGE_A_VIDYO_1	X			X	3.1 and higher	30
	Adaptive resolution	ADAPTRES_A	ADAPTRES_A_ERICCSON_1	X			X	3.1 and higher	24
	SPS representation format	SPSREPFMT_A	SPSREPFMT_A_Sony_2	X			X	5.1 and higher	24
	Conformance cropping window	CONFCROP_A	CONFCROP_A_VIDYO_2	X			X	3.1 and higher	30
		CONFCROP_B	CONFCROP_B_VIDYO_2	X			X	3 and higher	30
		CONFCROP_C	CONFCROP_C_VIDYO_3	X			X	3 and higher	30
Offsets and phase adjustments	Scaled reference layer offsets	SCREFFOFF_A	SCREFFOFF_A_QCOM_1	X			X	3 and higher	50
	Reference region offsets	REFREGOFF_A	REFREGOFF_A_SHARP_1	X			X	4.1 and higher	30
	Resample phase	RESPHASE_A	RESPHASE_A_SAMSUNG_2	X			X	3.1 and higher	60
Output layers and pictures	Output layer sets	OLS_A	OLS_A_NOKIA_1	X			X	3.1 and higher	24
		OLS_B	OLS_B_NOKIA_1	X			X	3.1 and higher	24
		OLS_C	OLS_C_NOKIA_1	X			X	3.1 and higher	24
	Discardable pictures	DISFLAG_A	DISFLAG_A_QUALCOMM_1	X			X	3.1 and higher	50
Scaling list	PPS scaling list	PPSLIST_A	PPSLIST_A_Sony_2	X			X	4.1 and higher	N/A
	SPS scaling list	SPSLIST_A	SPSLIST_A_Sony_2	X			X	4.1 and higher	24
Colour gamut scalability	CGS	CGS_A	CGS_A_TECHNICOLOR_1	X			X	4.1 and higher	50
		CGS_B	CGS_B_TECHNICOLOR_1	X			X	4.1 and higher	50
		CGS_C	CGS_C_TECHNICOLOR_1		X		X	4.1 and higher	50
		CGS_D	CGS_D_TECHNICOLOR_1		X		X	4.1 and higher	50
		CGS_E	CGS_E_TECHNICOLOR_1	X			X	4.1 and higher	60

Table 5 – Bitstreams for Scalable Main and Scalable Main 10 profiles

Categories	Subcategory	Bitstream	File name	Base layer profile: Main	Main 10	Main Still	Main tier	Level	Frame rate (Frames/sec)
		CGS_F	CGS_F_TECHNICOLOR_1	X			X	4.1 and higher	60
		CGS_G	CGS_G_TECHNICOLOR_1	X			X	4.1 and higher	50
		CGS_H	CGS_H_TECHNICOLOR_1		X		X	4.1 and higher	50
		CGS_I	CGS_I_TECHNICOLOR_1	X			X	4.1 and higher	60
Additional extensibility	Parameter set extension	PSEXT_A	PSEXT_A_VIDYO_2	X			X	3.1 and higher	30
	Layer ID 63	LAYERID63_A	LAYERID63A_HHI_1	X			X	N/A	N/A
Picture Order Count	Unaligned POC	POC_A	POC_A_Ericsson_1	X			X	3.1 and higher	24
		POC_B	POC_B_Ericsson_1	X			X	3.1 and higher	24
Base layer type	Hybrid scalability	HYBRID_A	HYBRID_A_QUALCOMM_1	N/A			N/A	N/A	30
	Base layer unavailable (INBLD)	INBLD_A	INBLD_A_NOKIA_2	N/A			X	3.1 and higher	24
	Simulcast	SIM_A	SIM_A_IDCC_1	X			X	3.1 and higher	24
		SIM_B	SIM_B_IDCC_1	X			X	3.1 and higher	24
Level signalling	Sub-layer level signalling	SLLEV_A	SLLEV_A_VIDYO_1	X			X	3.1 and higher	60
Auxiliary pictures	Alpha	ALPHA_A_BBC	ALPHA_A_BBC_1	X			X	4.1 and higher	30
	Depth	DEPTH_A	DEPTH_A_NOKIA_1	X			X	3.1 and higher	30
Scalable Range Extensions	Monochrome SNR layers	SREXT_A	SREXT_A_FUJITSU_1					4 and higher	30
		SREXT_B	SREXT_B_FUJITSU_1					4 and higher	30
		SREXT_C	SREXT_C_FUJITSU_1					4 and higher	30
		SREXT_D	SREXT_D_FUJITSU_1					4 and higher	30
	SNR layers	SREXT_E	SREXT_E_FUJITSU_1					4 and higher	30
	Spatial scalable layers	SREXT_F	SREXT_F_FUJITSU_1					4 and higher	30

6.7.6 Bitstreams for Scalable Monochrome, Scalable Monochrome 12, Scalable Monochrome 16, and Scalable Main 4:4:4 profiles

Legend:

X – Bitstream that a decoder conforming to the Main tier needs to decode for static and dynamic test

Table 6 – Bitstreams for Scalable Monochrome, Scalable Monochrome 12, Scalable Monochrome 16 and Scalable Main 4:4:4 profiles

Categories	Subcategory	Bitstream	File name	Base layer profie			
				Profile	Main tier	Level	Frame rate (Frames/sec)
Scalable Range Extensions	Monochrome SNR layers	SREXT_A	SREXT_A_FUJITSU_1	Monochrome	X	4 and higher	30
		SREXT_B	SREXT_B_FUJITSU_1	Monochrome 12	X	4 and higher	30
		SREXT_C	SREXT_C_FUJITSU_1	Monochrome 16	X	4 and higher	30
		SREXT_D	SREXT_D_FUJITSU_1	Monochrome 12	X	4 and higher	30
	SNR layers	SREXT_E	SREXT_E_FUJITSU_1	Main 4:4:4	X	4 and higher	30
	Spatial scalable layers	SREXT_F	SREXT_F_FUJITSU_1	Main 4:4:4	X	4 and higher	30

SERIES OF ITU-T RECOMMENDATIONS

Series A	Organization of the work of ITU-T
Series D	Tariff and accounting principles and international telecommunication/ICT economic and policy issues
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	Transmission systems and media, digital systems and networks
Series H	**Audiovisual and multimedia systems**
Series I	Integrated services digital network
Series J	Cable networks and transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Environment and ICTs, climate change, e-waste, energy efficiency; construction, installation and protection of cables and other elements of outside plant
Series M	Telecommunication management, including TMN and network maintenance
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Telephone transmission quality, telephone installations, local line networks
Series Q	Switching and signalling, and associated measurements and tests
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	Data networks, open system communications and security
Series Y	Global information infrastructure, Internet protocol aspects, next-generation networks, Internet of Things and smart cities
Series Z	Languages and general software aspects for telecommunication systems

Printed in Switzerland
Geneva, 2017