INTERNATIONAL TELECOMMUNICATION UNION

ITU-T

H.264.1

TELECOMMUNICATION STANDARDIZATION SECTOR
OF ITU

SERIES H: AUDIOVISUAL AND MULTIMEDIA SYSTEMS
Infrastructure of audiovisual services – Coding of moving video

Conformance specification for H.264 advanced video coding

CAUTION!

PREPUBLISHED RECOMMENDATION

This prepublication is an unedited version of a recently approved Recommendation. It will be replaced by the published version after editing. Therefore, there will be differences between this prepublication and the published version.
FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications. 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/had not] received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementors are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database.

© ITU 2005

All rights reserved. No part of this publication may be reproduced, by any means whatsoever, without the prior written permission of ITU.
Conformance specification for H.264 Advanced Video Coding

Summary
This Recommendation H.264.1 specifies tests designed to verify whether bitstreams and decoders meet the normative requirements specified in ITU-T Rec. H.264 | ISO/IEC 14496-10:

- An encoder can claim conformance to ITU-T Rec. H.264 | ISO/IEC 14496-10 if the bitstreams that it generates are conforming bitstreams.
- A decoder can claim conformance to a specified profile and level of ITU-T Rec. H.264 if it can properly decode all bitstreams obeying constraints specified in ITU-T Rec. H.264 | ISO/IEC 14496-10.

The tests specified in this Recommendation provide methods for (non-exhaustive) testing of whether encoders and decoders meet these requirements.

This twin text with ISO/IEC has been jointly developed in the context of JVT and has been submitted to the ISO/IEC JTC 1/SC 29/WG 11 (MPEG) as ISO/IEC 14496-4:2002 / Amendment 6 (2005 E).
Foreword

This Recommendation | International Standard establishes conformance test requirements for conformance to ITU-T Rec. H.264 | ISO/IEC 14496-10 Advanced Video Coding.

Introduction

This Recommendation | International Standard has been jointly developed by ITU-T Video Coding Experts Group (VCEG) and the ISO/IEC Moving Picture Experts Group. It is published as technically-aligned twin text in both organizations ITU-T and ISO/IEC.


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

For the pre-published version of this specification, as the accompanying bitstreams require a substantial amount of disk space, they can be found at the following web page:

http://ftp3.itu.int/av-arch/jvt-site/draft_conformance
1 Scope

This document specifies tests designed to verify whether bitstreams and decoders meet normative requirements specified in ITU-T Rec. H.264 | ISO/IEC 14496-10. An encoder can claim conformance to ITU-T Rec. H.264 | ISO/IEC 14496-10 if the bitstreams that it generates are conforming bitstreams.

Characteristics of coded bitstreams and decoders are defined for ITU-T Rec. H.264 | ISO/IEC 14496-10. The characteristics of a bitstream define the subset of the standard that is exploited in the bitstream. Examples are the applied values or range of the picture size and bit rate parameters. Decoder characteristics define the properties and capabilities of the applied decoding process. The capabilities of a decoder specify which bitstreams the decoder can decode and reconstruct, by defining the subset of the ITU-T Rec. H.264 | ISO/IEC 14496-10 standard that may be exploited in the bitstreams that it will decode. A bitstream can be decoded by a decoder if the characteristics of the bitstream are within the subset of the standard specified by the decoder capabilities.

Procedures are described for testing conformance of bitstreams and decoders to the requirements defined in ITU-T Rec. H.264 | ISO/IEC 14496-10. Given the set of characteristics claimed, the requirements that shall be met are fully determined by ITU-T Rec. H.264 | ISO/IEC 14496-10. This document summarizes the requirements, cross references them to characteristics, and defines how conformance with them can be tested. Guidelines are given on constructing tests to verify bitstream and decoder conformance. This document gives guidelines on how to construct bitstream test suites to check or verify decoder conformance. In addition, the test bitstreams implemented according to those guidelines are provided as an electronic annex to this document.

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

2.4 Additional references
   – None.

3 Definitions
For the purposes of this document, the terms, definitions, abbreviations and symbols specified in ITU-T Rec. H.264 | ISO/IEC 14496-10 (particularly in clause 3) apply. The following terms are further clarified for purposes herein as follows.


3.2 decoder: An ITU-T Rec. H.264 | ISO/IEC 14496-10 video decoder, i.e., an embodiment of the decoding process specified by ITU-T Rec. H.264 | ISO/IEC 14496-10. The decoder does not include the display process, which is outside the scope of this standard.


4 Abbreviations
For the purposes of this Recommendation | International Standard, relevant abbreviations are specified in clause 4 of ITU-T Rec. H.264 | ISO/IEC 14496-10.

5 Conventions
For the purposes of this Recommendation | International Standard, relevant conventions are specified in clause 5 in ITU-T Rec.264 | ISO/IEC 14496-10.

6 Conformance for ITU-T Rec. H.264 | ISO/IEC 14496-10

6.1 Introduction
The following subclauses specify the 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 document, and the reference software decoder specified in ITU-T Rec. H.264.2 | ISO/IEC 14496-5 with source code included in electronic format.

6.2 Bitstream conformance

6.3 Decoder conformance

6.4 Procedure to test bitstreams
A bitstream that claims conformance with this standard shall pass the following normative test:

The bitstream shall be decoded by processing it with the reference software decoder specified in ITU-T Rec. H.264.2 | ISO/IEC 14496-5. When processed by the reference software decoder, the bitstream shall not cause any error or non-conformance messages to be reported by the reference software decoder. This test should not be applied to bitstreams that are known to contain errors introduced by transmission, as such errors are highly likely to result in bitstreams that lack conformance to ITU-T Rec. H.264 | ISO/IEC 14496-10.
Successfully passing the reference software decoder test provides only a strong presumption that the bitstream under test is conforming to the video layer, i.e. that it does indeed meet all the requirements for the video layer (except Annexes C, D and E) specified in ITU-T Rec. H.264 | ISO/IEC 14496-10 that are tested by the reference software decoder.

Additional tests may be necessary to more thoroughly check that the bitstream properly meets all the requirements specified in ITU-T Rec. H.264 | ISO/IEC 14496-10 including the 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.

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

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

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

### 6.5 Procedure to test decoder conformance

#### 6.5.1 Conformance bitstreams

In this subclause, except where stated otherwise, the term "bitstream" refers to a conforming ITU-T Rec. H.264 | ISO/IEC 14496-10 video bitstream (as defined in this document), that has values of profile_idc, level_idc, and constraint_setX_flag values (where X is a number in the range of 0 to 2, inclusive) corresponding to a set of specified constraints on a bitstream for which a decoder conforming to a specified profile and level is required in Annex A of ITU-T Rec. H.264 | ISO/IEC 14496-10 to properly perform the decoding process.

#### 6.5.2 Contents of bitstream file

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

- ITU-T Rec. H.264 | ISO/IEC 14496-10 video bitstream
- Reconstructed pictures or hashes of decoded pictures
- Short description of the bitstream
- Trace file (the bitstream in ASCII format)

#### 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


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

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

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

#### 6.5.4 Recommendations (informative)

In addition to the requirements, it is desirable that conforming decoders implement various informative recommendations defined in ITU-T Rec. H.264 | ISO/IEC 14496-10 that are not an integral part of that Recommendation | International Standard. This subclause lists 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 macroblocks 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 reconstructed samples. This subclause will explain how this test can be accomplished when the reconstructed 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 reconstructed by the decoder under test shall be identical to the values of the reference samples attached to the bitstream file, or shall be identical to the values of the samples reconstructed by the reference decoder in cases where the values of the samples are not attached to the bitstream file. 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 reconstructed samples are output and that the timing of the output of the decoder's reconstructed samples conforms to the specification of clause 8 and Annex C of ITU-T Rec. H.264 | ISO/IEC 14496-10, and to verify that the HRD models (as defined by the CPB and DPB specification in Annex C of ITU-T Rec. H.264 | ISO/IEC 14496-10) are not violated when the bits are delivered at the proper rate.

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

If buffering period SEI and picture timing SEI are included in the test bitstream, HRD conformance shall be verified using the values of initial_cpb_removal_delay, initial_cpb_removal_delay_offset, cpb_removal_delay and dpb_removal_delay that are included in the bitstream.

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

- fixed_frame_rate_flag shall be inferred to be 1.
- low_delay_hrd_flag shall be inferred to be 0.
- cbr_flag shall be inferred to be 0.
- The frame rate of the stream shall be inferred to be the frame rate value specified in Table-1. If this is missing, then a frame rate of either 25 or 30000 ÷ 1001 can be inferred.
- time_scale shall be set to 90,000 and the value of num_units_in_tick shall be computed based on field rate (twice the frame rate).
- The bit rate of the bitstream shall be inferred to be the maximum value for the level defined in Table A-1 in ITU-T Rec. H.264 | ISO/IEC 14496-10.
- CPB and DPB sizes shall be inferred to be the maximum value for the level defined in Table A-1 in ITU-T Rec. H.264 | ISO/IEC 14496-10.

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

- The CPB is filled starting at time $t = 0$, until it is full, before removal of the first access unit. This means that the initial_cpb_removal_delay shall be inferred to be equal to the total CPB buffer size divided by the bit rate divided by 90000 (rounded downwards) and initial_cpb_removal_delay_offset shall be inferred to be equal to zero.
- The first access unit is removed at time $t = \text{initial_cpb_removal_delay} = 90000$ and subsequent access units are removed at intervals based on the frame distance, i.e. $\text{cpb_removal_delay} = 2 \times \left( \frac{90000}{\text{num_units_in_tick}} \right)$ or the field distance i.e. $\text{cpb_removal_delay} = \left( \frac{90000}{\text{num_units_in_tick}} \right)$, depending whether the access unit is coded as a frame picture or field picture.
- Using these inferences, the CPB will not overflow or underflow and the DPB will not overflow.
6.5.7 Decoder conformance test of a particular profile-and-level

In order for a decoder of a particular profile-and-level to claim output order conformance to the standard as described by this specification, the decoder shall successfully pass the static test defined in subclause 6.5.5 with all the bitstreams of the normative test suite specified for testing decoders of this particular profile-and-level.

In order for a decoder of a particular profile and level to claim output timing conformance to the standard as described by this specification, the decoder shall successfully pass both the static test defined in subclause 6.5.5 and the dynamic test defined in subclause 6.5.6 with all the bitstreams of the normative test suite specified for testing decoders of this particular profile-and-level. Table 1 and Table 2 define the normative test suites for each profile-and-level combination. The test suite for a particular profile-and-level combination is the list of bitstreams that are marked with an ‘X’ in the column corresponding to that profile-and-level combination.

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

The bitstream specification indicates the test bitstream specification used for each bitstream.

A decoder compliant with High, High 10, High 4:2:2, or High 4:4:4 shall be capable of decoding Main profile bitstreams. In addition to the streams defined in Table 2, a compliant decoder shall decode Main profile streams in Table 1.

6.6 Specification of the test bitstreams

Some characteristics of each bitstream listed in Table 1 and Table 2 are described in the subclauses of this subclause. In Table 1 and Table 2, the value "29.97" shall be interpreted as an approximation of an exact value of 30000 ÷ 1001.

6.6.1 Test Bitstreams – General

6.6.1.1 Test bitstream #AVCNL-1, #AVCNL-2

**Specification:** All slices are coded as I slices. Each picture contains only one slice. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Decoding of I slices.

**Purpose:** Check that decoder can properly decode I slices.

6.6.1.2 Test bitstream #AVCNL-3, #AVCNL-4

**Specification:** All slices are coded as I or P slices. Each picture contains only one slice. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Decoding of P slices.

**Purpose:** Check that decoder can properly decode P slices.

6.6.1.3 Test bitstream #AVCBA-1

**Specification:** All slices are coded as I slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Decoding of I slices with the deblocking filter process enabled.

**Purpose:** Check that the decoder can properly decode I slices with the deblocking filter process enabled.

6.6.1.4 Test bitstream #AVCBA-2

**Specification:** All slices are coded as I slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 2. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Decoding of I slices with the deblocking filter process enabled.

**Purpose:** Check that the decoder can properly decode I slices with the deblocking filter process enabled.
6.6.1.5 Test bitstream #AVCBA-3

**Specification:** All slices are coded as I or P slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Decoding of P slices with the deblocking filter process enabled.

**Purpose:** Check that the decoder can properly decode P slice with the deblocking filter process enabled.

6.6.1.6 Test bitstream #AVCBA-4

**Specification:** All slices are coded as I or P slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 2. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Decoding of P slices with the deblocking filter process enabled.

**Purpose:** Check that the decoder can properly decode P slices with the deblocking filter process enabled.

6.6.1.7 Test bitstream #AVCBA-5, #AVCBA-6

**Specification:** All slices are coded as I or P slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Decoding of P slices with the deblocking filter process enabled.

**Purpose:** Check that the decoder can properly decode P slices with the deblocking filter process enabled.

6.6.1.8 Test bitstream #AVCBA-7, #AVCBA-8

**Specification:** All slices are coded as I or P slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 2. MC block size is limited to 8x8 and above. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Decoding of P slices with the deblocking filter process enabled.

**Purpose:** Check that the decoder can properly decode P slices with the deblocking filter process enabled.

6.6.1.9 Test bitstream #AVCMQ-1

**Specification:** All slices are coded as I slices. Each picture contains only one slice. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 1. mb_qp_delta is equal to a non-zero value to change the quantizer scale at each MB. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Decoding of I slices with mb_qp_delta not equal to 0.

**Purpose:** Check that decoder can properly decode I slices with mb_qp_delta not equal to 0.

6.6.1.10 Test bitstream #AVCMQ-2

**Specification:** All slices are coded as I or P slices. Each picture contains only one slice. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 1. mb_qp_delta is equal to a non-zero value to change the quantizer scale at each MB. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Decoding of P slices with mb_qp_delta not equal to 0.

**Purpose:** Check that decoder can properly decode P slices with mb_qp_delta not equal to 0.

6.6.1.11 Test bitstream #AVCMQ-3

**Specification:** All slices are coded as I slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 1. Spatial direct prediction is used for direct prediction. mb_qp_delta is equal to a non-zero value to change the quantizer scale at each MB. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.
Functional stage: Decoding of I slices with mb_qp_delta not equal to 0.
Purpose: Check that decoder can properly decode I slices with mb_qp_delta not equal to 0.

6.6.1.12 Test bitstream #AVCMQ-4

Specification: All slices are coded as I or P slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 1. Spatial direct prediction is used for direct prediction. mb_qp_delta is equal to a non-zero value to change the quantizer scale at some MBs. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage: Decoding of P slices with mb_qp_delta not equal to 0.
Purpose: Check that decoder can properly decode P slices with mb_qp_delta not equal to 0.

6.6.1.13 Test bitstream #AVCSL-1

Specification: All slices are coded as I or P slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 1. Spatial direct prediction is used for direct prediction. mb_qp_delta is equal to a non-zero value to change the quantizer scale at some MBs. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage: Decoding of I and P slices.
Purpose: Check that decoder can properly decode pictures with multiple slices.

6.6.1.14 Test bitstream #AVCSL-2

Specification: All slices are coded as I or P slices. Each picture contains more than one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 2. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage: Decoding of I and P slices.
Purpose: Check that decoder can properly decode pictures with multiple slices.

6.6.1.15 Test bitstream #AVCSQ-1

Specification: All slices are coded as I slices. Each picture contains 20 slices. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. Spatial direct prediction is used for direct prediction. slice_qp_delta is equal to a non-zero value to change the quantizer scale at each slice. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage: Decoding of I slices with non-zero values of slice_qp_delta.
Purpose: Check that decoder can properly decode I slices with non-zero values of slice_qp_delta.

6.6.1.16 Test bitstream #AVCFM-1

Specification: All slices are coded as I or P slices. The number of slices and slice groups is greater than 1 in each picture. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10. Multiple parameter sets are included in the bitstream.

Functional stage: Slice groups.
Purpose: Check that decoder handles multiple slice groups and parameter sets.

6.6.1.17 Test bitstream #AVCFM-2

Specification: All slices are coded as I or P slices. The number of slices and slice groups is greater than 1 in each picture. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 1. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage: Slice groups.
Purpose: Check that decoder handles multiple slice groups and parameter sets.
6.6.1.18 Test bitstream #AVCFM-3

**Specification:** All slices are coded as I or P slices. The number of slices and slice groups is greater than 1 in each picture. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 2. Recovery point SEI is included in this bitstream. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Slice groups.

**Purpose:** Check that decoder handles multiple slice groups and parameter sets.

6.6.1.19 Test bitstream #AVCCI-1

**Specification:** All slices are coded as I or P slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. constrained_intra_pred_flag is equal to 1. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Constrained intra prediction.

**Purpose:** Check that decoder handles constrained intra prediction.

6.6.1.20 Test bitstream #AVCCI-2

**Specification:** All slices are coded as I or P slices. Each picture contains more than one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. constrained_intra_pred_flag is equal to 1. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Constrained intra prediction

**Purpose:** Check that decoder handles constrained intra prediction.

6.6.1.21 Test bitstream #AVCCI-3

**Specification:** All slices are coded as I or P slices. Each picture contains more than one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. constrained_intra_pred_flag is equal to 2. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Constrained intra prediction

**Purpose:** Check that decoder handles constrained intra prediction.

6.6.1.22 Test bitstream #AVCFC-1

**Specification:** All slices are coded as I or P slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. Decoded pictures are cropped with frame_cropping_flag equal to 1. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Decoding of I and P slices with frame cropping.

**Purpose:** Check that decoder can properly decode I and P slices with frame cropping.

6.6.1.23 Test bitstream #AVCAUD-1

**Specification:** All slices are coded as I slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. Access unit delimiter NAL units are included in the bitstream. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Decoding of I slices with Access unit delimiter NAL units.

**Purpose:** Check that decoder can properly decode I slices with Access unit delimiter NAL units.

6.6.1.24 Test bitstream #AVCMIDR-1

**Specification:** All slices are coded as I or P slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. IDR is inserted in every two frame.
All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage**: Decoding of I slices and more than one IDR.

**Purpose**: Check that decoder can properly decode I slices with more than IDR in bitstream.

### 6.6.1.25 Test bitstream #AVCNRF-1

**Specification**: All slices are coded as I or P slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. Two non-reference pictures are present. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage**: Decoding of I and P slices with non-reference pictures.

**Purpose**: Check that decoder can properly decode I and P slices with non-reference pictures.

### 6.6.1.26 Test bitstream #AVCMPS-1

**Specification**: All slices are coded as I or P slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. Multiple parameter sets are included in this bitstream. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage**: Decoding of I and P slices with multiple parameter set.

**Purpose**: Check that decoder can properly decode I and P slices with multiple parameter set.

### 6.6.1.27 Test bitstream #AVCBS-1

**Specification**: All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 1. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage**: Decoding of B slices with temporal direct prediction.

**Purpose**: Check that decoder can properly decode B slices with temporal direct prediction.

### 6.6.1.28 Test bitstream #AVCBS-2

**Specification**: All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. Spatial direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 1. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage**: Decoding of B slices with spatial direct prediction.

**Purpose**: Check that decoder can properly decode B slices with spatial direct prediction.

### 6.6.1.29 Test bitstream #AVCBS-3

**Specification**: All slices are coded as I, P or B slices. Each picture contains more than one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 0. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage**: Decoding of B slices with temporal direct prediction.

**Purpose**: Check that decoder can properly decode B slices with temporal direct prediction.

### 6.6.1.30 Test bitstream #AVCBS-4

**Specification**: All slices are coded as I, P or B slices. Each picture contains only one slice. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. Spatial direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 0. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage**: Decoding of B slices with spatial direct prediction.
Purpose: Check that decoder can properly decode B slices with spatial direct prediction.

6.6.1.31 Test bitstream #AVCBS-5

Specification: All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. Spatial direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 1. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage: Decoding of B slices with spatial direct prediction.

Purpose: Check that decoder can properly decode B slices with spatial direct prediction.

6.6.2 Test Bitstreams – I_PCM

6.6.2.1 Test bitstream #AVCPCM-1, AVCPCM-2

Specification: All slices are coded as I slices. Each picture contains only one slice. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. mb_type is equal to I_PCM for some macroblocks. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage: Decoding of macroblocks with mb_type equal to I_PCM

Purpose: Check that decoder can properly decode macroblocks with mb_type equal to I_PCM.

6.6.3 Test Bitstreams – Memory management control operation

6.6.3.1 Test bitstream #AVCMR-1

Specification: All slices are coded as I or P slices. Each picture contains more than one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 1. Reference picture list reordering and memory management control operations are used. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage: Reference picture list reordering and memory management control operations.

Purpose: Check that decoder handles reference picture list reordering and memory management control operations.

6.6.3.2 Test bitstream #AVCMR-2

Specification: All slices are coded as I or P slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 2. Reference picture list reordering and memory management control operations are used. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage: Reference picture list reordering and memory management control operations.

Purpose: Check that decoder handles reference picture list reordering and memory management control operations.

6.6.3.3 Test bitstream #AVCMR-3

Specification: All slices are coded as I or P slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 2. gaps_in_frame_num_value_allowed_flag is equal to 1. Reference picture list reordering and various memory management control operations are used. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage: Reference picture list reordering and memory management control operations.

Purpose: Check that decoder handles reference picture list reordering and memory management control operations.

6.6.3.4 Test bitstream #AVCMR-4

Specification: All slices are coded as I or P slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. gaps_in_frame_num_value_allowed_flag is equal to 1. Reference picture list reordering and various memory management control operations are used. The decoding order is different from the output order. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Purpose: Check that decoder handles gaps in frame_num, reference picture list reordering and memory management control operations.
**Functional stage:** Reference picture list reordering, memory management control operations and non-increasing PicOrderCnt values.

**Purpose:** Check that decoder handles reference picture list reordering and memory management control operations. Test output order conformance for non-increasing PicOrderCnt values.

### 6.6.3.5 Test bitstream #AVCMR-5

**Specification:** All slices are coded as I or P slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 1. gaps_in_frame_num_value_allowed_flag is equal to 1. Reference picture list reordering and various memory management control operation is used. The decoding order is different from the output order. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Reference picture list reordering, memory management control operation and non-increasing PicOrderCnt values.

**Purpose:** Check that decoder handles gaps_in_frame_num_value_allowed_flag equal to 1, reference picture list reordering and memory management control operation. Test output order conformance for non-increasing PicOrderCnt values.

### 6.6.3.6 Test bitstream #AVCMR-6

**Specification:** All slices are coded as I or P slices. Each picture contains more than one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. Reference picture list reordering is used. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Reference picture list reordering.

**Purpose:** Check that decoder handles reference picture list reordering.

### 6.6.3.7 Test bitstream #AVCMR-7

**Specification:** All slices are coded as I or P slices. Each picture contains more than one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. Memory management control operations are used. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Memory management control operations.

**Purpose:** Check that decoder handles memory management control operations.

### 6.6.3.8 Test bitstream #AVCMR-8, #AVCMR-9

**Specification:** All slices are coded as I or P slices. Each picture contains more than one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 1. Reference picture list reordering and memory management control operations are used. direct_8x8_inference_flag is equal to 1. Each slice is a coded field. VUI is included in the bitstream. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Reference picture list reordering and memory management control operations.

**Purpose:** Check that decoder handles reference picture list reordering and memory management control operations.

### 6.6.3.9 Test bitstream #AVCMR-10

**Specification:** All slices are coded as I, P or B slices. Each picture contains more than one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 1. Reference picture list reordering and memory management control operations are used. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 1. Each slice is a coded field. VUI is included in the bitstream. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Reference picture list reordering and memory management control operations.

**Purpose:** Check that decoder handles reference picture list reordering and memory management control operations.
6.6.4 Test Bitstreams – Weighted sample prediction process

6.6.4.1 Test bitstream #AVCWP-1

**Specification:** All slices are coded as I or P slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 2. weighted_pred_flag is equal to 1. Plural reference indices are assigned to each reference picture. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

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

**Purpose:** Check that decoder handles weighted sample prediction for P slices with plural reference indexes.

6.6.4.2 Test bitstream #AVCWP-2

**Specification:** All slices are coded as I or P slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 2. weighted_pred_flag is equal to 1. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Weighted sample prediction process for P slices.

**Purpose:** Check that decoder handles weighted sample prediction for P slices.

6.6.4.3 Test bitstream #AVCWP-3

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 2. weighted_bipred_idc is equal to 1. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Weighted sample prediction process for B slices with temporal direct prediction.

**Purpose:** Check that decoder handles weighted sample prediction for B slices with temporal direct prediction.

6.6.4.4 Test bitstream #AVCWP-4

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 2. Spatial direct prediction is used for direct prediction. weighted_bipred_idc is equal to 1. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Weighted sample prediction process for B slices with temporal direct prediction.

**Purpose:** Check that decoder handles weighted sample prediction for B slices with temporal direct prediction.

6.6.5 Test Bitstreams – Slice of coded field

6.6.5.1 Test bitstream #AVCFI-1

**Specification:** All slices are coded as I or P slices. Each picture contains only one slice. Each slice is a coded field. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Slices of coded fields.

**Purpose:** Check that decoder handles I and P slices of coded fields.

6.6.5.2 Test bitstream #AVCFI-2

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. Each slice is a coded field. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. Each slice is a coded field. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. Spatial direct prediction is used for direct prediction. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Slices of coded fields with spatial direct prediction.

**Purpose:** Check that decoder handles B slices of coded fields with spatial direct prediction.
6.6.5.3 Test bitstream #AVCFI-3

**Specification:** All slices are coded as I or P slices. Each picture contains only one slice. Each slice is a coded field. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 1. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Slices of coded fields

**Purpose:** Check that decoder handles I and P slices of coded fields.

6.6.5.4 Test bitstream #AVCFI-4

**Specification:** All slices are coded as I or P slices. Each picture contains only one slice. Each slice is a coded field. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 1. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Slices of coded fields.

**Purpose:** Check that decoder handles I and P slices of coded fields.

6.6.5.5 Test bitstream #AVCFI-5

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. Each slice is a coded field. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. Each slice is a coded field. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 1. Spatial direct prediction is used for direct prediction. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Slices of coded fields.

**Purpose:** Check that decoder handles B slices of coded fields.

6.6.5.6 Test bitstream #AVCFI-6

**Specification:** All slices are coded as I or P slices. Each picture contains more than one slice. Each slice is a coded field. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Slices of coded fields.

**Purpose:** Check that decoder handles I and P slices of coded fields.

6.6.5.7 Test bitstream #AVCFI-7

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. Each slice is a coded field. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Slices of coded fields with temporal direct prediction.

**Purpose:** Check that decoder handles B slices of coded fields with temporal direct prediction.

6.6.5.8 Test bitstream #AVCFI-8

**Specification:** All slices are coded as I slices. Only one slices is contained in each picture. Each slice is a coded field. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Slices of coded fields.

**Purpose:** Check that decoder handles I slices of coded fields.

6.6.5.9 Test bitstream #AVCFI-9

**Specification:** All slices are coded as I or P slices. Each picture contains only one slice. Each slice is a coded field. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.
**Functional stage:** Slices of coded fields.

**Purpose:** Check that decoder handles I and P slices of coded fields.

### 6.6.5.10 Test bitstream #AVCFI-10

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. Each slice is a coded field. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Slices of coded fields.

**Purpose:** Check that decoder handles I and P slices of coded fields.

### 6.6.5.11 Test bitstream #AVCFI-11

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. Each slice is a coded field. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. Spatial direct prediction is used for direct prediction. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Slices of coded fields.

**Purpose:** Check that decoder handles B slices of coded fields with spatial direct prediction.

### 6.6.5.12 Test bitstream #AVCFI-12

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. Each slice is a coded field. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. The number of motion vectors per two consecutive MBs is equal to the maximum value specified in Annex A.3.1.m in ITU-T Rec. H.264 | ISO/IEC 14496-10. No intra, skip and direct MBs are included in P and B slices. Each slice is a coded field. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Slices of coded fields.

**Purpose:** Check that decoder can properly decode slices of coded fields with maximum number of motion vectors per consecutive MBs.

### 6.6.6 Test Bitstreams – Frame/field coding

#### 6.6.6.1 Test bitstream #AVCPA-1

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. Spatial direct prediction is used for direct prediction. Each slice is either a coded frame or a coded field. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Slices of coded frames/fields.

**Purpose:** Check that decoder can properly decode slices of coded frames and fields.

#### 6.6.6.2 Test bitstream #AVCPA-2

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. Each slice is either a coded frame or a coded field. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Slices of coded frames/fields.

**Purpose:** Check that decoder can properly decode slices of coded frames and fields.

#### 6.6.6.3 Test bitstream #AVCPA-3

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. Spatial direct prediction is used for direct prediction. Each slice is either a coded frame or a coded field. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.
Functional stage: Slices of coded frames/fields.

Purpose: Check that decoder can properly decode slices of coded frames and fields.

6.6.7 Test bitstreams - Macroblock adaptive frame/field coding

6.6.7.1 Test bitstream #AVCMA-1

Specification: All slices are coded as I, P or B slices. Each picture contains only one slice. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. Spatial direct prediction is used for direct prediction. mb_adaptive_frame_field_coding is equal to 1. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage: Macroblock adaptive frame field decoding.

Purpose: Check that decoder can properly decode slices with mb_adaptive_frame_field_flag=1.

6.6.7.2 Test bitstream #AVCMA-2

Specification: All slices are coded as I, P or B slices. Each picture contains only one slice. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. mb_adaptive_frame_field_coding is equal to 1. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage: Macroblock adaptive frame field decoding.

Purpose: Check that decoder can properly decode slices with mb_adaptive_frame_field_flag=1.

6.6.7.3 Test bitstream #AVCMA-3

Specification: All slices are coded as I slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. mb_adaptive_frame_field_coding is equal to 1. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage: Macroblock adaptive frame field decoding.

Purpose: Check that decoder can properly decode slices with mb_adaptive_frame_field_flag=1.

6.6.7.4 Test bitstream #AVCMA-4

Specification: All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. Spatial direct prediction is used for direct prediction. mb_adaptive_frame_field_coding is equal to 1. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage: Macroblock adaptive frame field decoding.

Purpose: Check that decoder can properly decode slices with mb_adaptive_frame_field_flag=1.

6.6.7.5 Test bitstream #AVCMA-5

Specification: All slices are coded as I, or P slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. mb_adaptive_frame_field_coding is equal to 1. mb_qp_delta is equal to a non-zero value to change the quantizer scale at some MBs. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage: Macroblock adaptive frame field decoding.

Purpose: Check that decoder can properly decode slices with mb_adaptive_frame_field_flag=1.

6.6.7.6 Test bitstream #AVCMA-6

Specification: All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. mb_adaptive_frame_field_coding is equal to 1. mb_qp_delta is equal to a non-zero value to change the quantizer scale at some MBs. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.
**Functional stage:** Macroblock adaptive frame field decoding.

**Purpose:** Check that decoder can properly decode slices with mb_adaptive_frame_field_flag=1.

### 6.6.7.7 Test bitstream #AVCMA-7

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 1. Some slices are coded as a coded field. mb_adaptive_frame_field_coding equal to 1 in the rest of the frames. mb_qp_delta is equal to a non-zero value to change the quantizer scale at some MBs. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Macroblock adaptive frame field decoding and slices of a coded field.

**Purpose:** Check that decoder can properly decode both slices of a coded frame with mb_adaptive_frame_field_flag=1 and slices of a coded field.

### 6.6.7.8 Test bitstream #AVCMA-8

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. Spatial direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 1. mb_adaptive_frame_field_coding is equal to 1. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Macroblock adaptive frame field decoding.

**Purpose:** Check that decoder can properly decode slices with mb_adaptive_frame_field_flag=1.

### 6.6.7.9 Test bitstream #AVCMA-9

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. The number of motion vectors per two consecutive MBs is equal to the maximum value specified in Annex A.3.1.m in ITU-T Rec. H.264 | ISO/IEC 14496-10. No intra, skip and direct MBs are included in P and B slices. mb_adaptive_frame_field_coding is equal to 1. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Macroblock adaptive frame field decoding.

**Purpose:** Check that decoder can properly decode slices with mb_adaptive_frame_field_flag=1 and with maximum number of motion vectors per consecutive MBs.

### 6.6.8 Test Bitstreams – S picture

#### 6.6.8.1 Test bitstream #AVCSP-1

**Specification:** All slices are coded as I, P and SP slice. Each picture contains more than one slice. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 1. memory_management_operation is set to 5 on SP slice. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Decoding of SP slice.

**Purpose:** Check that decoder can properly decode SP slice.

#### 6.6.8.2 Test bitstream #AVCSP-2

**Specification:** All slices are coded as I, P and SP slice. Each picture contains more than one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 1. memory_management_operation is set to 5 on SP slice. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Decoding of SP slice.

**Purpose:** Check that decoder can properly decode SP slice with deblocking filter.
6.6.9  Test Bitstreams – Long sequence

6.6.9.1  Test bitstream #AVCLS-1

Specification:  All slices are coded as I or P slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage:  Decoding of POC for long sequence.

Purpose:  Check that the decoder can properly decode POC for long sequence.

6.6.10  Test Bitstreams – SEI/VUI

6.6.10.1  Test bitstream #AVCSE-1

Specification:  All slices are coded as I or P slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. SEI (Buffering period SEI and Picture timing SEI with pic_struct) and VUI are included in the bitstream. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage:  Decoding of SEI/VUI.

Purpose:  Check that the decoder can properly decode SEI/VUI.

6.6.10.2  Test bitstream #AVCSE-2

Specification:  All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 1. SEI (Buffering period SEI and Picture timing SEI with pic_struct) and VUI are included in the bitstream. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage:  Decoding of SEI/VUI.

Purpose:  Check that the decoder can properly decode SEI/VUI.

6.6.10.3  Test bitstream #AVCSE-3

Specification:  All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 1. SEI (Buffering period SEI and Picture timing SEI with pic_struct) and VUI are included in the bitstream. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage:  Decoding of SEI/VUI.

Purpose:  Check that the decoder can properly decode SEI/VUI.

6.6.11  Test Bitstreams – CABAC: Basic features

6.6.11.1  Test bitstream #AVCCANL-1

Specification:  All slices are coded as I slices. Each picture contains only one slice. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 2. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage:  Decoding of I slice with CABAC parsing.

Purpose:  Check that decoder can properly decode I slices with CABAC parsing.

6.6.11.2  Test bitstream #AVCCANL-2

Specification:  All slices are coded as I slices. Each picture contains only one slice. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage:  Decoding of I slice with CABAC parsing.
Purpose: Check that decoder can properly decode I slices with CABAC parsing.

6.6.11.3 Test bitstream #AVCCANL-3

Specification: All slices are coded as I or P slices. Each picture contains only one slice. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage: Decoding of P slices with CABAC parsing.

Purpose: Check that decoder can properly decode P slices with CABAC parsing.

6.6.11.4 Test bitstream #AVCCANL-4

Specification: All slices are coded as I, P or B slices. Each picture contains only one slice. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Spatial direct prediction is used for direct prediction. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage: Decoding of B slices with CABAC parsing.

Purpose: Check that decoder can properly decode B slices with CABAC parsing.

6.6.11.5 Test bitstream #AVCCANL-5

Specification: All slices are coded as I slices. Each picture contains only one slice. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 2. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage: Decoding of I slice with CABAC parsing.

Purpose: Check that decoder can properly decode I slices with CABAC parsing.

6.6.11.6 Test bitstream #AVCCANL-6

Specification: All slices are coded as I slices. Each picture contains only one slice. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage: Decoding of I slice with CABAC parsing.

Purpose: Check that decoder can properly decode I slices with CABAC parsing.

6.6.11.7 Test bitstream #AVCCANL-7

Specification: All slices are coded as I or P slices. Each picture contains only one slice. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage: Decoding of P slices with CABAC parsing.

Purpose: Check that decoder can properly decode P slices with CABAC parsing.

6.6.11.8 Test bitstream #AVCCANL-8

Specification: All slices are coded as I, P or B slices. Each picture contains only one slice. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Spatial direct prediction is used for direct prediction. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage: Decoding of B slices with CABAC parsing.

Purpose: Check that decoder can properly decode B slices with CABAC parsing.
6.6.11.9 Test bitstream #AVCCABA-1

**Specification:** All slices are coded as I slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Decoding of I slice with the deblocking filter process enabled and CABAC.

**Purpose:** Check that decoder can properly decode I slices with CABAC parsing.

6.6.11.10 Test bitstream #AVCCABA-2

**Specification:** All slices are coded as I or P slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Decoding of P slices with CABAC parsing.

**Purpose:** Check that decoder can properly decode P slices with CABAC parsing.

6.6.11.11 Test bitstream #AVCCABA-3

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Decoding of B slices with CABAC parsing.

**Purpose:** Check that decoder can properly decode B slices with CABAC parsing.

6.6.11.12 Test bitstream #AVCCABA-4

**Specification:** All slices are coded as I, or P slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Decoding of P slices with CABAC parsing.

**Purpose:** Check that decoder can properly decode P slices with CABAC parsing.

6.6.11.13 Test bitstream #AVCCABA-5

**Specification:** All slices are coded as I slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Decoding of I slice with the deblocking filter process enabled and CABAC.

**Purpose:** Check that decoder can properly decode I slices with CABAC parsing.

6.6.11.14 Test bitstream #AVCCABA-6

**Specification:** All slices are coded as I or P slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Decoding of P slices with CABAC parsing.

**Purpose:** Check that decoder can properly decode P slices with CABAC parsing.

6.6.11.15 Test bitstream #AVCCABA-7

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Decoding of B slices with CABAC parsing.

**Purpose:** Check that decoder can properly decode B slices with CABAC parsing.
6.6.11.16 Test bitstream #AVCCABA-8

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Decoding of B slices with CABAC parsing.

**Purpose:** Check that decoder can properly decode B slices with CABAC parsing.

### 6.6.12 Test Bitstreams – CABAC: Initialization

#### 6.6.12.1 Test bitstream #AVCCAIN-1

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 0. cabac_init_idc is equal to 0, 1, or 2 at slice header. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Initialization of CABAC.

**Purpose:** Check that decoder can initialize CABAC with cabac_init_idc=0, 1, or 2.

#### 6.6.13 Test Bitstreams – CABAC: MB QP Delta

#### 6.6.13.1 Test bitstream #AVCCAQP-1

**Specification:** All slices are coded as I slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 1. mb_qp_delta is equal to non-zero value to change the quantizer scale at each MB. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Decoding of I slices with mb_qp_delta not equal to 0.

**Purpose:** Check that decoder can properly decode I slices with mb_qp_delta not equal to 0.

#### 6.6.13.2 Test bitstream #AVCCAQP-2

**Specification:** All slices are coded as I, P or B slices. Each picture contains more than one slice. Each slice has different size. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. mb_qp_delta is equal to non-zero value to change the quantizer scale at each MB. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Decoding of I, P, and B slices with mb_qp_delta not equal to 0.

**Purpose:** Check that decoder can properly decode I slices with mb_qp_delta not equal to 0, disable_deblocking_filter_idc equal to 2, and non-zero chroma_qp_index_offset.

#### 6.6.14 Test Bitstreams – CABAC: Slice

#### 6.6.14.1 Test bitstream #AVCCASL-1

**Specification:** All slices are coded as I, P or B slices. Each picture contains more than one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 0. Each picture contains more than one slice. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Decoding of different slice types in a picture with CABAC parsing.

**Purpose:** Check that decoder can properly decode different slice types in a picture with CABAC parsing.

#### 6.6.14.2 Test bitstream #AVCCASL-2

**Specification:** All slices are coded as I, P or B slices. Each picture contains more than one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0.
Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 0. Slices with different slice types are included in a picture. Stored B slices are included in the bitstream. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Decoding of different slice types in a picture with CABAC parsing.

**Purpose:** Check that decoder can properly decode different slice types in a picture with CABAC parsing.

### 6.6.15 Test Bitstreams – CABAC: I_PCM

#### 6.6.15.1 Test bitstream #AVCCAPCM-1

**Specification:** All slices are coded as I slices. Each picture contains only one slice. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. mb_type is equal to I_PCM at some Macroblocks. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Decoding of Macroblock with mb_type equal to I_PCM.

**Purpose:** Check that decoder can properly decode Macroblock with mb_type equal to I_PCM.

#### 6.6.15.2 Test bitstream #AVCCAPCM-2

**Specification:** All slices are coded as I slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. mb_type is equal to I_PCM at some Macroblocks. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Decoding of Macroblock with mb_type equal to I_PCM.

**Purpose:** Check that decoder can properly decode Macroblock with mb_type equal to I_PCM.

#### 6.6.15.3 Test bitstream #AVCCAPCM-3

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 1. mb_type is equal to I_PCM at some Macroblocks. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Decoding of Macroblock with mb_type equal to I_PCM.

**Purpose:** Check that decoder can properly decode Macroblock with mb_type equal to I_PCM.

### 6.6.16 Test Bitstreams – CABAC: Memory management control operation

#### 6.6.16.1 Test bitstream #AVCCAMR-1

**Specification:** All slices are coded as I, P or B slices. Each picture contains more than one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 1. Reference picture list reordering and memory management control operations are used. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 1. Each slice is a coded frame. mb_adaptive_frame_field_coding equal to 1. VUI is included in the bitstream. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Reference picture list reordering and memory management control operations.

**Purpose:** Check that decoder handles reference picture list reordering and memory management control operations.

### 6.6.17 Test Bitstreams – CABAC: Weighted sample prediction process

#### 6.6.17.1 Test bitstream #AVCCAWP-1

**Specification:** All slices are coded as I or P slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 2. weighted_pred_flag is equal to 1. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Weighted sample prediction process for P slice

**Purpose:** Check that decoder handles weighted sample prediction for P slice.
6.6.17.2 Test bitstream #AVCCAWP-2

**Specification:** All slices are coded as I or P slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 2. weighted_pred_flag is equal to 1. Plural reference indices are assigned to each reference picture. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

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

**Purpose:** Check that decoder handles weighted sample prediction for P slices with plural reference indexes.

6.6.18 Test Bitstreams – CABAC: Field coding

6.6.18.1 Test bitstream #AVCCAFI-1

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 1. Each slice is a coded field. Stored B slices are included in the bitstream. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Decoding of coded fields.

**Purpose:** Check that decoder can properly decode slice of coded field including stored B slice.

6.6.18.2 Test bitstream #AVCCAFI-2

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Spatial direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 1. Each slice is a coded field. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Decoding of coded fields.

**Purpose:** Check that decoder can properly decode slice of coded field.

6.6.18.3 Test bitstream #AVCCAFI-3

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 1. Each slice is a coded field. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Decoding of coded fields.

**Purpose:** Check that decoder can properly decode slice of coded field.

6.6.19 Test Bitstreams – CABAC: Frame/field decoding

6.6.19.1 Test bitstream #AVCCAPA-1

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 1. Spatial direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 1. Each slice is either a coded frame or a coded field. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Picture adaptive frame/field decoding.

**Purpose:** Check that decoder can properly decode slices of coded frames and fields with direct_8x8_inference_flag=1.

6.6.19.2 Test bitstream #AVCCAPA-2

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 1. Each slice is either a coded frame or a coded field. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.
Functional stage: Picture adaptive frame/field decoding.

Purpose: Check that decoder can properly decode slices of coded frames and fields with direct_8x8_inference_flag=1.

6.6.19.3 Test bitstream #AVCCAPA-3

Specification: All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 1. Each slice is either a coded frame or a coded field. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage: Picture adaptive frame/field decoding.

Purpose: Check that decoder can properly decode slices of coded frames and fields with direct_8x8_inference_flag=1.

6.6.20 Test bitstreams - Macroblock adaptive frame/field decoding

6.6.20.1 Test bitstream #AVCCAMA-1

Specification: All slices are coded as I, P or B slices. Each picture contains only one slice. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Spatial direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 1. mb_adaptive_frame_field_coding is equal to 1. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage: Macroblock adaptive frame field decoding.

Purpose: Check that decoder can properly decode slices with mb_adaptive_frame_field_flag=1.

6.6.20.2 Test bitstream #AVCCAMA-2

Specification: All slices are coded as I, P or B slices. Each picture contains only one slice. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 1. mb_adaptive_frame_field_coding is equal to 1. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage: Macroblock adaptive frame field decoding.

Purpose: Check that decoder can properly decode slices with mb_adaptive_frame_field_flag=1.

6.6.20.3 Test bitstream #AVCCAMA-3

Specification: All slices are coded as I or P slices. Each picture contains only one slice. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. num_ref_frames is equal to 1. mb_adaptive_frame_field_coding is equal to 1. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage: Macroblock adaptive frame field decoding.

Purpose: Check that decoder can properly decode slices with mb_adaptive_frame_field_flag=1.

6.6.20.4 Test bitstream #AVCCAMA-4

Specification: All slices are coded as I or P slices. Each picture contains only one slice. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. mb_adaptive_frame_field_coding is equal to 1. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage: Macroblock adaptive frame field decoding.

Purpose: Check that decoder can properly decode slices with mb_adaptive_frame_field_flag=1.
6.6.20.5 Test bitstream #AVCCAMA-5

**Specification:** All slices are coded as I slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. mb_adaptive_frame_field_coding is equal to 1. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Macroblock adaptive frame field decoding.

**Purpose:** Check that decoder can properly decode slices with mb_adaptive_frame_field_flag=1.

6.6.20.6 Test bitstream #AVCCAMA-6

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Spatial direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 1. mb_adaptive_frame_field_coding is equal to 1. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Macroblock adaptive frame field decoding.

**Purpose:** Check that decoder can properly decode slices with mb_adaptive_frame_field_flag=1.

6.6.20.7 Test bitstream #AVCCAMA-7

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 1. mb_adaptive_frame_field_coding is equal to 1. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Macroblock adaptive frame field decoding.

**Purpose:** Check that decoder can properly decode slices with mb_adaptive_frame_field_flag=1.

6.6.20.8 Test bitstream #AVCCAMA-8

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 1. mb_adaptive_frame_field_coding is equal to 1. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Macroblock adaptive frame field decoding.

**Purpose:** Check that decoder can properly decode slices with mb_adaptive_frame_field_flag=1.

6.6.20.9 Test bitstream #AVCCAMA-9

**Specification:** All slices are coded as I, P or B slices. Each picture contains more than one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 1. mb_adaptive_frame_field_coding is equal to 1. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Macroblock adaptive frame field decoding.

**Purpose:** Check that decoder can properly decode slices with mb_adaptive_frame_field_flag=1.

6.6.20.10 Test bitstream #AVCCAMA-10

**Specification:** All slices are coded as I, P or B slices. Each picture contains more than one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. direct_8x8_inference_flag is equal to 1. mb_adaptive_frame_field_coding is equal to 1. constrained_intra_pred_flag is equal to 1. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Macroblock adaptive frame field decoding.

**Purpose:** Check that decoder can handle constrained intra prediction with mb_adaptive_frame_field_flag=1.
6.6.20.11 Test bitstream #AVCCAMA-11

**Specification:** All slices are coded as I, P or B slices. Each picture contains more than one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. direct_8x8_inference_flag is equal to 1. mb_adaptive_frame_field_coding is equal to 1. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Macroblock adaptive frame field decoding.

**Purpose:** Check that decoder can properly decode slices with mb_adaptive_frame_field_flag=1.

6.6.20.12 Test bitstream #AVCCAMA-12 and AVCCAMA-13

**Specification:** All slices are coded as I, P or B slices. Each picture contains more than one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. The number of motion vectors per tow consecutive MBs is equal to the maximum value specified in Annex A.3.1.m in ITU-T Rec. H.264 | ISO/IEC 14496-10. No intra, skip and direct MBs are included in P and B slices. mb_adaptive_frame_field_coding is equal to 1. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Macroblock adaptive frame field decoding.

**Purpose:** Check that decoder can properly decode slices with mb_adaptive_frame_field_flag=1 and with maximum number of motion vectors per consecutive MBs.

6.6.20.13 Test bitstream #AVCCAPAMA-1

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. Both coded frames and coded fields are included in the bitstream. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 1. mb_adaptive_frame_field_coding is equal to 1. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Macroblock adaptive frame field decoding and slices of a coded field.

**Purpose:** Check that decoder can properly decode both slice of a coded frame with mb_adaptive_frame_field_flag=1 and slices of a coded field.

6.6.20.14 Test bitstream #AVCCAPAMA-2

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 1. The first field of the first frame only contains I slice and the second field only contains P slice. mb_adaptive_frame_field_coding is equal to 1 in the rest of the frames. The indicated display of this bitstream is bottom field first. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Macroblock adaptive frame field decoding and slices of a coded field.

**Purpose:** Check that decoder can properly decode both slice of a coded frame with mb_adaptive_frame_field_flag=1 and slices of a coded field.

6.6.20.15 Test bitstream #AVCCAPAMA-3

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 1. The first field of the first frame only contains I slice and the second field only contains P slice. mb_adaptive_frame_field_coding is equal to 1 in the rest of the frames. The indicated display of this bitstream is top field first. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Macroblock adaptive frame field decoding and slices of a coded field.

**Purpose:** Check that decoder can properly decode both slices of a coded frame with mb_adaptive_frame_field_flag=1 and slices of a coded field.
6.6.20.16 Test bitstream #AVCCAPAMA-4

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 1. The first field of the first frame only contains I slice and the second field only contains P slice. mb_adaptive_frame_field_coding is equal to 1 in the rest of the frames. The indicated display of this bitstream is top field first. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Macroblock adaptive frame field decoding and slices of a coded field.

**Purpose:** Check that decoder can properly decode both slices of a coded frame with mb_adaptive_frame_field_flag=1 and slices of a coded field.

6.6.20.17 Test bitstream #AVCCAMV-1

**Specification:** The bitstream conforms to MP@L3.0. Frame size is 720x480. All slices are coded as I, P or B slices. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 1. In P slices, each macroblock is coded as sixteen 4x4 blocks. Each block has one motion vector in 1/4 sample position. In B slices, each macroblock is coded as eight 8x4 blocks. Each block has two motion vectors, one for list0 the other for list1. Both vectors are in 1/4 sample position. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Prediction bandwidth.

**Purpose:** Check that the decoder handles the worse case of prediction bandwidth. Prediction bandwidth is at maximum due to largest number of motion vectors (in 1/4 sample position) per macroblock pair (32 as defined in standard). Non-integer position motion vectors require using 6-tap filter always.

6.6.20.18 Test bitstream #AVCCVCANLMA-1

**Specification:** All slices are coded as I or P slices. Each picture contains only one slice. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. Both entropy_coding_mode_flag equal to 0, specifying the CAVLC parsing process, and entropy_coding_mode_flag equal to 1, specifying the CABAC parsing process are present within the bitstream. pic_order_cnt_type is equal to 0. mb_adaptive_frame_field_coding is equal to 1. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Macroblock adaptive frame field decoding using both CAVLC and CABAC.

**Purpose:** Check that decoder can properly decode slices with mb_adaptive_frame_field_flag=1 Check that the decoder can properly decode both CABAC and CAVLC.

6.6.20 Test bitstreams – Fidelity Range Extensions: 4:2:0 8 bit

6.6.20.1 Test bitstream #FREH-1, #FREH-28

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. Spatial direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 0. Transform mode is set to 8x8 block size only. seq_scaling_matrix_present_flag and pic_scaling_matrix_flag are set to 1. Scaling lists are included in the sequence parameter set and the picture parameter set. Each slice is a coded frame. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Tests loading of scaling list in the sequence parameter set and the picture parameter set. Tests 8x8 block size transform mode. Tests decoding of level prefix more than 16 bits in CAVLC entropy coding. Tests deblocking for 8x8 transform.

**Purpose:** Check that a decoder can properly decode slices of coded frames with 8x8 block size transform for CAVLC and check that scaling list is implemented correctly for frame only coding.

6.6.20.2 Test bitstream #FREH-2, #FREH-29

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is set equal to 0. Both 4x4 and 8x8 block size transform modes are used. seq_scaling_matrix_present_flag and pic_scaling_matrix_flag are set to 1. Scaling lists are included in the...
sequence parameter set and the picture parameter set. Each slice is a coded frame. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Tests 4x4 and 8x8 block size transform modes in CABAC entropy coding. Tests loading of scaling list in the sequence parameter set and the picture parameter set. Tests deblocking for 4x4 and 8x8 transform.

**Purpose:** Check that a decoder can properly decode slices of coded frames with both 4x4 and 8x8 block size transform modes and check that scaling list is implemented correctly for CABAC entropy coding for frame only coding.

### 6.6.20.3 Test bitstream #FREH-3, #FREH-30

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. The value of cabac_init_idc is adaptively changed in slice header. Spatial direct prediction is used for direct prediction. direct_8x8_inference_flag is set equal to 1. Both 4x4 and 8x8 block size transform modes are used. seq_scaling_matrix_present_flag is set to 1 and default scaling lists are used. Each slice is a coded frame. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Tests 4x4 and 8x8 block size transform modes in CABAC entropy coding.

**Purpose:** Check that a decoder can properly decode slices of coded frames with both 4x4 and 8x8 block size transform modes.

### 6.6.20.4 Test bitstream #FREH-4, #FREH-31

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. The value of cabac_init_idc is adaptively changed in slice header. Spatial direct prediction is used for direct prediction. direct_8x8_inference_flag is set equal to 1. Both 4x4 and 8x8 block size transform modes are used. seq_scaling_matrix_present_flag is set to 1 and default scaling lists are used. Each slice is either a coded frame or a coded field. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Tests 4x4 and 8x8 block size transform modes in CABAC entropy coding.

**Purpose:** Check that a decoder can properly decode slices of coded frames and fields with both 4x4 and 8x8 block size transform modes.

### 6.6.20.5 Test bitstream #FREH-5, #FREH-32

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. The value of cabac_init_idc is adaptively changed in slice header. Spatial direct prediction is used for direct prediction. direct_8x8_inference_flag is set equal to 1. Both 4x4 and 8x8 block size transform modes are used. seq_scaling_matrix_present_flag is set to 1 and default scaling lists are used. Each slice is a coded frame. mb_adaptive_frame_field_coding equal to 1. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Macroblock adaptive frame field decoding and slices of a coded frame with both 4x4 and 8x8 block size transform modes in CABAC entropy coding.

**Purpose:** Check that a decoder can properly decode slices of coded frames with mb_adaptive_frame_field_flag=1 and with both 4x4 and 8x8 block size transform modes.

### 6.6.20.6 Test bitstream #FREH-6, #FREH-33

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 1. Transform mode is set to 8x8 block size only. seq_scaling_matrix_present_flag and pic_scaling_matrix_flag are set to 1. Scaling lists are included in the sequence parameter set and the picture parameter set. Each slice is either a coded frame or a coded field. mb_adaptive_frame_field_coding equal to 1 in coded frames. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Tests loading of scaling list in the sequence parameter set and the picture parameter set. Tests 8x8 block size transform mode. Tests decoding of level prefix more than 16 bits in CAVLC entropy coding. Tests deblocking for 8x8 transform.
Purpose: Check that a decoder can properly decode slices of coded frames with 8x8 block size transform for CAVLC and check that scaling list is implemented correctly for both slices of a coded frame with mb_adaptive_frame_field_flag=1 and slices of a coded field.

6.6.20.7 Test bitstream #FREH-7, #FREH-34

Specification: All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is set equal to 1. Both 4x4 and 8x8 block size transform modes are used. seq_scaling_matrix_present_flag and pic_scaling_matrix_flag are set to 1. Scaling lists are included in the sequence parameter set and the picture parameter set. Each slice is either a coded frame or a coded field. mb_adaptive_frame_field_coding equal to 1 in coded frames. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage: Tests 4x4 and 8x8 block size transform modes in CABAC entropy coding. Tests loading of scaling list in the sequence parameter set and the picture parameter set. Tests deblocking for 4x4 and 8x8 transform.

Purpose: Check that a decoder can properly decode slices of coded frames with both 4x4 and 8x8 block size transform modes and check that scaling list is implemented correctly for CABAC entropy coding for both slices of a coded frame with mb_adaptive_frame_field_flag=1 and slices of a coded field.

6.6.20.8 Test bitstream #FREH-8

Specification: All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is set equal to 1. Both 4x4 and 8x8 block size transform modes are used. Both seq_scaling_matrix_present_flag and pic_scaling_matrix is set to 0. Each slice is a coded frame. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage: Tests 4x4 and 8x8 block size transform modes in CABAC entropy coding.

Purpose: Check that a decoder can properly decode slices of coded frames with both 4x4 and 8x8 block size transform modes.

6.6.20.9 Test bitstream #FREH-9

Specification: All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is set equal to 1. Both 4x4 and 8x8 block size transform modes are used. Both seq_scaling_matrix_present_flag and pic_scaling_matrix is set to 0. Each slice is a coded frame. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage: Tests 4x4 and 8x8 block size transform modes in CABAC entropy coding.

Purpose: Check that a decoder can properly decode slices of coded frames with both 4x4 and 8x8 block size transform modes.

6.6.20.10 Test bitstream #FREH-10

Specification: All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is set equal to 1. Both 4x4 and 8x8 block size transform modes are used. Both seq_scaling_matrix_present_flag and pic_scaling_matrix is set to 0. Each slice is a coded field. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage: Tests 4x4 and 8x8 block size transform modes in CABAC entropy coding.

Purpose: Check that a decoder can properly decode slices of coded fields with both 4x4 and 8x8 block size transform modes.

6.6.20.11 Test bitstream #FREH-11

Specification: All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is set equal to 1. Both 4x4 and 8x8 block size transform modes are
used. Both seq_scaling_matrix_present_flag and pic_scaling_matrix is set to 0. Each slice is a coded field. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Tests 4x4 and 8x8 block size transform modes in CABAC entropy coding.

**Purpose:** Check that a decoder can properly decode slices of coded fields with both 4x4 and 8x8 block size transform modes.

### 6.6.20.12 Test bitstream #FREH-12, #FREH-39

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is set equal to 0. Both 4x4 and 8x8 block size transform modes are used. seq_scaling_matrix_present_flag is set to 1 and default scaling lists are used. Each slice is a coded frame. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Tests 4x4 and 8x8 block size transform modes in CABAC entropy coding.

**Purpose:** Check that a decoder can properly decode slices of coded fields with both 4x4 and 8x8 block size transform modes.

### 6.6.20.13 Test bitstream #FREH-13, #FREH-14, #FRE-15

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is set equal to 1. Both 4x4 and 8x8 block size transform modes are used. seq_scaling_matrix_present_flag is set to 1 and default scaling lists are used. Each slice is a coded frame. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Tests 4x4 and 8x8 block size transform modes in CABAC entropy coding.

**Purpose:** Check that a decoder can properly decode slices of coded frames with both 4x4 and 8x8 block size transform modes.

### 6.6.20.14 Test bitstream #FREH-16

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is set equal to 1. Both 4x4 and 8x8 block size transform modes are used. seq_scaling_matrix_present_flag is set to 1. Scaling lists are included in the sequence parameter set. Each slice is a coded frame. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Tests loading of scaling list in the sequence parameter set. Tests 8x8 block size transform mode.

**Purpose:** Check that a decoder can properly decode slices of a coded frame with 8x8 block size transform for CABAC. Check that scaling list is implemented correctly for frame only coding. Check that a decoder can handle temporal direct mode with direct_inference_flag=1 for coded frames with 8x8 block size transform.

### 6.6.20.15 Test bitstream #FREH-17

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is set equal to 1. Both 4x4 and 8x8 block size transform modes are used. seq_scaling_matrix_present_flag is set to 1. Scaling lists are included in the sequence parameter set. Each slice is either a coded frame or a coded field. mb_adaptive_frame_field_coding equal to 1 in coded frames. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Tests loading of scaling list in the sequence parameter set. Tests 8x8 block size transform mode.

**Purpose:** Check that a decoder can properly decode slices of a coded frame with 8x8 block size transform for CABAC. Check that scaling list is implemented correctly for field coding and MBAFF. Check that a decoder can handle temporal direct mode with direct_inference_flag=1 for coded frames with 8x8 block size transform.

### 6.6.20.16 Test bitstream #FREH-18

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is set equal to 1. Both 4x4 and 8x8 block size transform modes are used.

---

ITU-T Rec. H.264.1 (03/2005) – Prepublished version
used. Both seq_scaling_matrix_present_flag and pic_scaling_matrix is set to 0. Each slice is a coded frame. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Tests 4x4 and 8x8 block size transform modes in CAVLC entropy coding.

**Purpose:** Check that a decoder can properly decode slices of coded frames with both 4x4 and 8x8 block size transform modes.

**6.6.20.17 Test bitstream #FREH-19**

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is set equal to 1. Both 4x4 and 8x8 block size transform modes are used. Both seq_scaling_matrix_present_flag and pic_scaling_matrix is set to 0. Each slice is a coded frame. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Tests 4x4 and 8x8 block size transform modes in CAVLC entropy coding.

**Purpose:** Check that a decoder can properly decode slices of coded frames with both 4x4 and 8x8 block size transform modes.

**6.6.20.18 Test bitstream #FREH-20**

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is set equal to 1. Both 4x4 and 8x8 block size transform modes are used. Both seq_scaling_matrix_present_flag and pic_scaling_matrix is set to 0. Each slice is a coded field. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Tests 4x4 and 8x8 block size transform modes in CAVLC entropy coding.

**Purpose:** Check that a decoder can properly decode slices of coded fields with both 4x4 and 8x8 block size transform modes.

**6.6.20.19 Test bitstream #FREH-21**

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is set equal to 1. Both 4x4 and 8x8 block size transform modes are used. Both seq_scaling_matrix_present_flag and pic_scaling_matrix is set to 0. Each slice is a coded field. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Tests 4x4 and 8x8 block size transform modes in CAVLC entropy coding.

**Purpose:** Check that a decoder can properly decode slices of coded fields with both 4x4 and 8x8 block size transform modes.

**6.6.20.20 Test bitstream #FREH-22**

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is set equal to 1. Both 4x4 and 8x8 block size transform modes are used. Both seq_scaling_matrix_present_flag and pic_scaling_matrix is set to 0. Each slice is a coded field. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Tests 4x4 and 8x8 block size transform modes in CAVLC entropy coding.

**Purpose:** Check that a decoder can properly decode slices of coded fields with both 4x4 and 8x8 block size transform modes.
coded field. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Tests 4x4 and 8x8 block size transform modes in CAVLC entropy coding.

**Purpose:** Check that a decoder can properly decode slices of coded fields with both 4x4 and 8x8 block size transform modes.

6.6.20.22 Test bitstream #FREH-24

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. Spatial direct prediction is used for direct prediction. direct_8x8_inference_flag is set equal to 1. Both 4x4 and 8x8 block size transform modes are used. seq_scaling_matrix_present_flag is set to 1 and default scaling lists are used. mb_adaptive_frame_field_coding equal to 1. Each slice is a coded frame. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Tests 4x4 and 8x8 block size transform modes in CAVLC entropy coding.

**Purpose:** Check that a decoder can properly decode slices of coded fields with both 4x4 and 8x8 block size transform modes.

6.6.20.23 Test bitstream #FREH-25

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is set equal to 1. Both 4x4 and 8x8 block size transform modes are used. seq_scaling_matrix_present_flag is set to 1. Scaling lists are included in the sequence parameter set. Each slice is a coded frame. chroma_format_idc is equal to 0, specifying monochrome chroma format. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Tests monochrome chroma format in CAVLC entropy coding.

**Purpose:** Check that a decoder can properly decode slices of coded frame for monochrome chroma format.

6.6.20.24 Test bitstream #FREH-26

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is set equal to 1. Both 4x4 and 8x8 block size transform modes are used. seq_scaling_matrix_present_flag is set to 1. Scaling lists are included in the sequence parameter set. Each slice is a coded frame. chroma_format_idc is equal to 0, specifying monochrome chroma format. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Tests monochrome chroma format in CABAC entropy coding.

**Purpose:** Check that a decoder can properly decode slices of coded frame for monochrome chroma format.

6.6.20.25 Test bitstream #FREH-27

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is set equal to 1. Both 4x4 and 8x8 block size transform modes are used. seq_scaling_matrix_present_flag is set to 1. Scaling lists are included in the sequence parameter set. Each slice is a coded frame. second_chroma_qp_index_offset is equal to 2. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Tests second_chroma_qp_index_offset.

**Purpose:** Check that a decoder can properly decode slices of coded frame with second_chroma_qp_index_offset.

6.6.20.26 Test bitstream #FREH-35

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction. direct_8x8_inference_flag is set equal to 1. Both 4x4 and 8x8 block size transform modes are used. Both seq_scaling_matrix_present_flag and pic_scaling_matrix_flag are set to 1. Scaling lists are included in the sequence parameter set.
parameter set and the picture parameter set. Each slice is a coded frame. All NAL units are encapsulated into the byte

**Functional stage:** Tests 4x4 and 8x8 block size transform modes in CABAC entropy coding.

**Purpose:** Check that a decoder can properly decode slices of coded frames with both 4x4 and 8x8 block size
transform modes.

### 6.6.20.27 Test bitstream #FREH-36

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag
is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Temporal direct prediction is
used for direct prediction. direct_8x8_inference_flag is set equal to 1. Both 4x4 and 8x8 block size transform modes are
used. Both seq_scaling_matrix_present_flag and pic_scaling_matrix_flag are set to 1. Scaling lists are included in the
sequence parameter set and the picture parameter set. Each slice is a coded frame. All NAL units are encapsulated into

**Functional stage:** Tests 4x4 and 8x8 block size transform modes in CABAC entropy coding.

**Purpose:** Check that a decoder can properly decode slices of coded frames with both 4x4 and 8x8 block size
transform modes.

### 6.6.20.28 Test bitstream #FREH-37

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag
is equal to 1, specifying the CABAC parsing process. disable_deblocking_filter_idc is equal to 1, specifying disabling of
the deblocking filter process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction.
direct_8x8_inference_flag is set equal to 1. Both 4x4 and 8x8 block size transform modes are used. Both seq_scaling_matrix_present_flag and pic_scaling_matrix_flag are set to 1. Scaling lists are included in the sequence parameter set and the picture parameter set. Each slice is a coded field. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Tests 4x4 and 8x8 block size transform modes in CABAC entropy coding.

**Purpose:** Check that a decoder can properly decode slices of coded frames with both 4x4 and 8x8 block size
transform modes.

### 6.6.20.29 Test bitstream #FREH-38

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag
is equal to 1, specifying the CABAC parsing process. disable_deblocking_filter_idc is equal to 1, specifying disabling of
the deblocking filter process. pic_order_cnt_type is equal to 0. Temporal direct prediction is used for direct prediction.
direct_8x8_inference_flag is set equal to 1. Both 4x4 and 8x8 block size transform modes are used. Both seq_scaling_matrix_present_flag and pic_scaling_matrix_flag are set to 1. Scaling lists are included in the sequence parameter set and the picture parameter set. Each slice is a coded field. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Tests 4x4 and 8x8 block size transform modes in CABAC entropy coding.

**Purpose:** Check that a decoder can properly decode slices of coded fields with both 4x4 and 8x8 block size transform
modes.

### 6.6.21 Test bitstreams – Fidelity Range Extensions: 4:2:0 10 bit

#### 6.6.21.1 Test bitstream #FREH10-1

**Specification:** All slices are coded as I slices. Each picture contains only one slice. disable_deblocking_filter_idc is
equal to 1, specifying disabling of the deblocking filter process. entropy_coding_mode_flag is equal to 0, specifying the
CAVLC parsing process. pic_order_cnt_type is equal to 0. frame_mbs_only_flag is equal to 1. chroma_format_idc is
equal to 1. Both bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 2. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Decoding of I slices for 4:2:0 10-bit.

**Purpose:** Check that a decoder can properly decode I slices for 4:2:0 10-bit.

#### 6.6.21.2 Test bitstream #FREH10-2

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. disable_deblocking_filter_idc is
equal to 1, specifying disabling of the deblocking filter process.
entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. frame_mbs_only_flag is equal to 1. chroma_format_idc is equal to 1. Both bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 2. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage: Decoding of I, P, and B slices for 4:2:0 10-bit.

Purpose: Check that a decoder can properly decode I, P and B slices for 4:2:0 10-bit.

6.6.22 Test bitstreams – Fidelity Range Extensions: 4:2:2 10 bit

6.6.22.1 Test bitstream #FREH422-1

Specification: All slices are coded as I or P slices. Each picture contains only one slice. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. chroma_format_idc is equal to 2, specifying 4:2:2 chroma format. Both bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 0. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage: Decoding of P slices for 4:2:2 8 bit.

Purpose: Check that a decoder can properly decode P slices for 4:2:2 8 bit.

6.6.22.2 Test bitstream #FREH422-2

Specification: All slices are coded as I, P or B slices. Each picture contains only one slice. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. Direct prediction is not used in this bitstream. chroma_format_idc is equal to 2, specifying 4:2:2 chroma format. Both bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 0. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage: Decoding of B slices for 4:2:2 8 bit.

Purpose: Check that a decoder can properly decode B slices for 4:2:2 8 bit.

6.6.22.3 Test bitstream #FREH422-3

Specification: All slices are coded as I or P slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. chroma_format_idc is equal to 2, specifying 4:2:2 chroma format. Both bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 0. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage: Decoding of P slices for 4:2:2 8 bit.

Purpose: Check that a decoder can properly decode P slices with deblocking filter for 4:2:2 8 bit.

6.6.22.4 Test bitstream #FREH422-4

Specification: All slices are coded as I slices. Each picture contains only one slice. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. chroma_format_idc is equal to 2, specifying 4:2:2 chroma format. Both bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 0. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage: Decoding of I slices for 4:2:2 8 bit.

Purpose: Check that a decoder can properly decode I slices for 4:2:2 8 bit without deblocking filter.

6.6.22.5 Test bitstream #FREH422-5

Specification: All slices are coded as I or P slices. Each picture contains only one slice. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. chroma_format_idc is equal to 2, specifying 4:2:2 chroma format. Both bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 0. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage: Decoding of P slices for 4:2:2 8 bit.

Purpose: Check that a decoder can properly decode P slices for 4:2:2 8 bit without deblocking filter.
6.6.22.6 Test bitstream #FREH422-6

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. entropy_coding_mode_flag is equal to 0, specifying the CAEVC parsing process. pic_order_cnt_type is equal to 0. Spatial direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 0. chroma_format_idc is equal to 2, specifying 4:2:2 chroma format. Both bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 0. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Decoding of B slices for 4:2:2 8 bit.

**Purpose:** Check that a decoder can properly decode B slices for 4:2:2 8 bit without deblocking filter.

6.6.22.7 Test bitstream #FREH422-7

**Specification:** All slices are coded as I or P slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAEVC parsing process. pic_order_cnt_type is equal to 0, chroma_format_idc is equal to 2, specifying 4:2:2 chroma format. Both bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 0. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Decoding of P slices for 4:2:2 8 bit.

**Purpose:** Check that a decoder can properly decode P slices for 4:2:2 8 bit with deblocking filter.

6.6.22.8 Test bitstream #FREH422-8

**Specification:** All slices are coded as I slices. Each picture contains only one slice. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. entropy_coding_mode_flag is equal to 0, specifying the CAEVC parsing process. pic_order_cnt_type is equal to 0, chroma_format_idc is equal to 2, specifying 4:2:2 chroma format. Both bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 2, specifying 10 bit video. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Decoding of I slices for 4:2:2 10 bit.

**Purpose:** Check that a decoder can properly decode I slices for 4:2:2 10 bit without deblocking filter.

6.6.22.9 Test bitstream #FREH422-9

**Specification:** All slices are coded as I or P slices. Each picture contains only one slice. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. entropy_coding_mode_flag is equal to 0, specifying the CAEVC parsing process. pic_order_cnt_type is equal to 0, chroma_format_idc is equal to 2, specifying 4:2:2 chroma format. Both bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 2, specifying 10 bit video. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Decoding of P slices for 4:2:2 8 bit.

**Purpose:** Check that a decoder can properly decode P slices for 4:2:2 10 bit without deblocking filter.

6.6.22.10 Test bitstream #FREH422-10

**Specification:** All slices are coded as I, P or B slices. Each picture contains only one slice. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. entropy_coding_mode_flag is equal to 0, specifying the CAEVC parsing process. pic_order_cnt_type is equal to 0. Spatial direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 0. chroma_format_idc is equal to 2, specifying 4:2:2 chroma format. Both bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 2, specifying 10 bit video. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Decoding of B slices for 4:2:2 10 bit.

**Purpose:** Check that a decoder can properly decode B slices for 4:2:2 10 bit without deblocking filter.

6.6.22.11 Test bitstream #FREH422-11

**Specification:** All slices are coded as I or P slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 0, specifying the CAEVC parsing process. pic_order_cnt_type is equal to 0, chroma_format_idc is equal to 2, specifying 4:2:2 chroma format. Both bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 2,
specifying 10 bit video. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage**: Decoding of P slices for 4:2:2 10 bit.

**Purpose**: Check that a decoder can properly decode P slices for 4:2:2 10 bit with deblocking filter.

### 6.6.22.12 Test bitstream #FREH422-12

**Specification**: All slices are coded as I slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. chroma_format_idc is equal to 2, specifying 4:2:2 chroma format. Both bit_depth_luma_minus8 and bit_depth_chromaMinus8 are set equal to 0. Both 4x4 and 8x8 block size transform modes are used. seq_scaling_matrix_present_flag is set to 1 and default scaling lists are used. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage**: Decoding of I slices for 4:2:2 8 bit.

**Purpose**: Check that a decoder can properly decode I slices for 4:2:2 8 bit without deblocking filter.

### 6.6.22.13 Test bitstream #FREH422-13

**Specification**: All slices are coded as I or P slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. chroma_format_idc is equal to 2, specifying 4:2:2 chroma format. Both bit_depth_luma_minus8 and bit_depth_chromaMinus8 are set equal to 0. Both 4x4 and 8x8 block size transform modes are used. seq_scaling_matrix_present_flag is set to 1 and default scaling lists are used. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage**: Decoding of P slices for 4:2:2 8 bit.

**Purpose**: Check that a decoder can properly decode P slices for 4:2:2 8 bit without deblocking filter.

### 6.6.22.14 Test bitstream #FREH422-14

**Specification**: All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Spatial direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 0. chroma_format_idc is equal to 2, specifying 4:2:2 chroma format. Both bit_depth_luma_minus8 and bit_depth_chromaMinus8 are set equal to 0. Both 4x4 and 8x8 block size transform modes are used. seq_scaling_matrix_present_flag is set to 1 and default scaling lists are used. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage**: Decoding of B slices for 4:2:2 8 bit.

**Purpose**: Check that a decoder can properly decode B slices for 4:2:2 8 bit without deblocking filter.

### 6.6.22.15 Test bitstream #FREH422-15

**Specification**: All slices are coded as I slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. 10 bit video. Both 4x4 and 8x8 block size transform modes are used. seq_scaling_matrix_present_flag is set to 1 and default scaling lists are used. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage**: Decoding of I slices for 4:2:2 10 bit.

**Purpose**: Check that a decoder can properly decode I slices for 4:2:2 10 bit without deblocking filter.

### 6.6.22.16 Test bitstream #FREH422-16

**Specification**: All slices are coded as I or P slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. chroma_format_idc is equal to 2, specifying 4:2:2 chroma format. Both bit_depth_luma_minus8 and bit_depth_chromaMinus8 are set equal to 2, specifying 10 bit video. Both 4x4 and 8x8 block size transform modes are used. seq_scaling_matrix_present_flag is set to 1 and default scaling lists are used. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage**: Decoding of P slices for 4:2:2 10 bit.
Purpose: Check that a decoder can properly decode P slices for 4:2:2 10 bit without deblocking filter.

6.6.22.17 Test bitstream #FREH422-17

Specification: All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Spatial direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 0. chroma_format_idc is equal to 2, specifying 4:2:2 chroma format. Both bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 2, specifying 10 bit video. Both 4x4 and 8x8 block size transform modes are used. seq_scaling_matrix_present_flag is set to 1 and default scaling lists are used. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage: Decoding of B slices for 4:2:2 10 bit.

Purpose: Check that a decoder can properly decode B slices for 4:2:2 10 bit without deblocking filter.

6.6.22.18 Test bitstream #FREH422-18

Specification: All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Spatial direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 0. chroma_format_idc is equal to 2, specifying 4:2:2 chroma format. Both bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 2, specifying 10 bit video. Both 4x4 and 8x8 block size transform modes are used. seq_scaling_matrix_present_flag is set to 1. Scaling lists are included in the sequence parameter set and the picture parameter set. Each slice is a coded field. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage: Decoding of B slices for 4:2:2 10 bit.

Purpose: Check that a decoder can properly decode B slices of coded fields for 4:2:2 10 bit.

6.6.22.19 Test bitstream #FREH422-19

Specification: All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Spatial direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 0. chroma_format_idc is equal to 2, specifying 4:2:2 chroma format. Both bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 2, specifying 10 bit video. Both 4x4 and 8x8 block size transform modes are used. seq_scaling_matrix_present_flag is set to 1. Scaling lists are included in the sequence parameter set and the picture parameter set. Each slice is a coded frame. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage: Decoding of B slices for 4:2:2 10 bit.

Purpose: Check that a decoder can properly decode B slices of coded fields for 4:2:2 10 bit.

6.6.22.20 Test bitstream #FREH422-20

Specification: All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Spatial direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 0. chroma_format_idc is equal to 2, specifying 4:2:2 chroma format. Both bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 2, specifying 10 bit video. Both 4x4 and 8x8 block size transform modes are used. seq_scaling_matrix_present_flag is set to 1. Scaling lists are included in the sequence parameter set and the picture parameter set. Each slice is either a coded frame or a coded field. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

Functional stage: Decoding of B slices for 4:2:2 10 bit.

Purpose: Check that a decoder can properly decode B slices of coded frames for 4:2:2 10 bit.

6.6.22.21 Test bitstream #FREH422-21

Specification: All slices are coded as I, P or B slices. Each picture contains only one slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. Spatial direct prediction is used for direct prediction. direct_8x8_inference_flag is equal to 0. chroma_format_idc is equal to 2, specifying 4:2:2 chroma format. Both bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 2, specifying 10 bit video. Both 4x4 and 8x8 block size transform modes are used. seq_scaling_matrix_present_flag is set to 1. Scaling lists are included in the sequence parameter set and the picture parameter set. Each slice is a coded frame. mb_adaptive_frame_field_coding is equal to 1. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.
**Functional stage:** Decoding of B slices for 4:2:2 10 bit.

**Purpose:** Check that a decoder can properly decode B slices with mb_adaptive_frame_field_flag=1 for 4:2:2 10 bit.

### 6.6.23 Test bitstreams – Fidelity Range Extensions: 4:4:4 12 bit

#### 6.6.23.1 Test bitstream #FREH444-1

**Specification:** All slices are coded as I slices. Each picture contains only one slice. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. entropy_coding_mode_flag is equal to 0, specifying the CAVLC parsing process. pic_order_cnt_type is equal to 0. frame_mbs_only_flag is equal to 1. chroma_format_idc is equal to 3. Both bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 4. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Decoding of I slices for 4:4:4 12-bit

**Purpose:** Check that the decoder can properly decode I slices for 4:4:4 12-bit.

#### 6.6.23.2 Test bitstream #FREH444-2

**Specification:** All slices are coded as IBBP slices. Each picture contains only one slice. disable_deblocking_filter_idc is equal to 1, specifying disabling of the deblocking filter process. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. pic_order_cnt_type is equal to 0. frame_mbs_only_flag is equal to 1. chroma_format_idc is equal to 3. residual_colour_transform_flag is equal to 1. Both bit_depth_luma_minus8 and bit_depth_chroma_minus8 are set equal to 4. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Decoding of I, P, and B slices for 4:4:4 12-bit

**Purpose:** Check that the decoder can properly decode I, P and B slices for 4:4:4 12-bit with residual colour transform.

#### 6.6.24 Auxiliary coded picture

#### 6.6.24.1 Test bitstream #FREAUX-1

**Specification:** Coded slices of an auxiliary coded picture are included in this bitstream. The rest of the slices are coded as either an I slice or a P slice. entropy_coding_mode_flag is equal to 1, specifying the CABAC parsing process. All NAL units are encapsulated into the byte stream format specified in Annex B in ITU-T Rec. H.264 | ISO/IEC 14496-10.

**Functional stage:** Decoding of coded slices of an auxiliary coded picture

**Purpose:** Check that the decoder can properly handle coded slices of an auxiliary coded picture

### 6.7 Normative Test Suites for ITU-T Rec. H.264 | ISO/IEC 14496-10

**Legend:**

X – Bitstream is for static and dynamic test

**Table 1 – Bitstreams for Baseline, Extended and Main profile**

<table>
<thead>
<tr>
<th>Categories</th>
<th>Bitstream</th>
<th>Donated by</th>
<th>File Name</th>
<th>Baseline</th>
<th>Extended</th>
<th>Main</th>
<th>Level</th>
<th>Frame Rate (Frame/Sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>General</td>
<td>AVCNL-1</td>
<td>Sony</td>
<td>NL1_Sony_D</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>1.2 and higher</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td>AVCNL-2</td>
<td>SVA</td>
<td>SVA_NL1_B</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>2.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCNL-3</td>
<td>Sony</td>
<td>NL2_Sony_H</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.1 and higher</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td>AVCNL-4</td>
<td>SVA</td>
<td>SVA_NL2_E</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>2.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCBA-1</td>
<td>Sony</td>
<td>BA1_Sony_D</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>1.2 and higher</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td>AVCBA-2</td>
<td>SVA</td>
<td>SVA_BA1_B</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>2.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td>Categories</td>
<td>Bitstream</td>
<td>Donated by</td>
<td>File Name</td>
<td>Baseline</td>
<td>Extended</td>
<td>Main</td>
<td>Level</td>
<td>Frame Rate (Frame/Sec)</td>
</tr>
<tr>
<td>------------</td>
<td>-----------</td>
<td>-----------------</td>
<td>-----------</td>
<td>----------</td>
<td>----------</td>
<td>------</td>
<td>-------------</td>
<td>-----------------------</td>
</tr>
<tr>
<td>AVCBA-3</td>
<td>Sony</td>
<td>BA2_Sony_F</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.1 and higher</td>
<td>15</td>
</tr>
<tr>
<td>AVCBA-4</td>
<td>SVA</td>
<td>SVA BA2_D</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>2.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td>AVCBA-5</td>
<td>MCubeworks</td>
<td>BA_MW_D</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>1.0 and higher</td>
<td>15</td>
</tr>
<tr>
<td>AVCBA-6</td>
<td>MCubeworks</td>
<td>BANM_MW_D</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>1.0 and higher</td>
<td>15</td>
</tr>
<tr>
<td>AVCBA-7</td>
<td>France Telecom</td>
<td>BA1_FT_C</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>2.0 and higher</td>
<td>25</td>
</tr>
<tr>
<td>AVCMQ-1</td>
<td>JVC</td>
<td>NLMQ1_JVC_C</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>2.0 and higher</td>
<td>25</td>
</tr>
<tr>
<td>AVCMQ-2</td>
<td>JVC</td>
<td>NLMQ2_JVC_C</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>2.0 and higher</td>
<td>25</td>
</tr>
<tr>
<td>AVCMQ-3</td>
<td>JVC</td>
<td>BAMQ1_JVC_C</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>2.0 and higher</td>
<td>25</td>
</tr>
<tr>
<td>AVCMQ-4</td>
<td>JVC</td>
<td>BAMQ2_JVC_C</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>2.0 and higher</td>
<td>25</td>
</tr>
<tr>
<td>AVCSSL-1</td>
<td>SVA</td>
<td>SVA Base B</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>2.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td>AVCSSL-2</td>
<td>SVA</td>
<td>SVA FM1_E</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>2.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td>AVCSQ-1</td>
<td>Sony</td>
<td>BASQP1_Sony_C</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>2.1 and higher</td>
<td>15</td>
</tr>
<tr>
<td>AVCFM-1</td>
<td>British Telecom</td>
<td>FM1_BT_B</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>1.0 and higher</td>
<td>5</td>
</tr>
<tr>
<td>AVCFM-2</td>
<td>SVA</td>
<td>FM2_SVA_C</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>2.1 and higher</td>
<td>15</td>
</tr>
<tr>
<td>AVCFM-3</td>
<td>France Telecom</td>
<td>FM1_FT_E</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>2.0 and higher</td>
<td>25</td>
</tr>
<tr>
<td>AVCSCI-1</td>
<td>MCubeworks</td>
<td>CI_MW_D</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>1.0 and higher</td>
<td>15</td>
</tr>
<tr>
<td>AVCSCI-2</td>
<td>SVA</td>
<td>SVA_CL1_E</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>2.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td>AVCSCI-3</td>
<td>France Telecom</td>
<td>CI1_FT_B</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>2.0 and higher</td>
<td>25</td>
</tr>
<tr>
<td>AVCFC-1</td>
<td>Sony</td>
<td>CVFC1_Sony_C</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td>AVCAUD-1</td>
<td>Mcubeworks</td>
<td>AUD_MW_E</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>1.0 and higher</td>
<td>15</td>
</tr>
<tr>
<td>AVCMIDR-1</td>
<td>Mcubeworks</td>
<td>MIDR_MW_D</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>1.0 and higher</td>
<td>15</td>
</tr>
<tr>
<td>AVCNRF-1</td>
<td>Mcubeworks</td>
<td>NRF_MW_E</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>1.0 and higher</td>
<td>15</td>
</tr>
<tr>
<td>AVCMPH-1</td>
<td>Mcubeworks</td>
<td>MPS_MW_A</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>1.1 and higher</td>
<td>15</td>
</tr>
<tr>
<td>AVCBS-1</td>
<td>Sony</td>
<td>CVBS3_Sony_C</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>1.2 and higher</td>
<td>15</td>
</tr>
<tr>
<td>AVCBS-2</td>
<td>SVA</td>
<td>BA3_SVA_C</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>2.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td>AVCBS-3</td>
<td>SVA</td>
<td>SL1_SVA_B</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>2.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td>AVCBS-4</td>
<td>SVA</td>
<td>NL3_SVA_E</td>
<td></td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>1.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td>AVCBS-5</td>
<td>Motorola</td>
<td>cavlc mot_frm0_full B</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>2.2 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td>I PCM</td>
<td>AVCPCM-1</td>
<td>SVA CVPCMNL1_SVA_C</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>4.0 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td>I PCM</td>
<td>AVCPCM-2</td>
<td>SVA CVPCMNL2_SVA_C</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>4.0 and higher</td>
<td>60</td>
</tr>
<tr>
<td>Categories</td>
<td>Bitstream</td>
<td>Donated by</td>
<td>File Name</td>
<td>Baseline</td>
<td>Extended</td>
<td>Main</td>
<td>Level</td>
<td>Frame Rate (Frame/Sec)</td>
</tr>
<tr>
<td>------------</td>
<td>-----------</td>
<td>-------------------</td>
<td>----------------</td>
<td>----------</td>
<td>----------</td>
<td>-------------</td>
<td>------------------------------</td>
<td>------------------------</td>
</tr>
<tr>
<td>MMCO</td>
<td>AVCMR-1</td>
<td>British Telecom</td>
<td>MR1_BT_A</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>1.1 and higher</td>
<td>20</td>
</tr>
<tr>
<td></td>
<td>AVCMR-2</td>
<td>Tandberg</td>
<td>MR2_Tandberg_E</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCMR-3</td>
<td>Tandberg</td>
<td>MR3_Tandberg_B</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCMR-4</td>
<td>Tandberg</td>
<td>MR4_Tandberg_C</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCMR-5</td>
<td>Tandberg</td>
<td>MR5_Tandberg_C</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCMR-6</td>
<td>Mucubeworks</td>
<td>MR1_MW_A</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>1.1 and higher</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td>AVCMR-7</td>
<td>Mucubeworks</td>
<td>MR2_MW_A</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>1.1 and higher</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td>AVCMR-8</td>
<td>British Telecom</td>
<td>MR6_BT_B</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>2.1 and higher</td>
<td>25</td>
</tr>
<tr>
<td></td>
<td>AVCMR-9</td>
<td>British Telecom</td>
<td>MR7_BT_B</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>2.1 and higher</td>
<td>25</td>
</tr>
<tr>
<td></td>
<td>AVCMR-10</td>
<td>British Telecom</td>
<td>MR8_BT_B</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>2.1 and higher</td>
<td>25</td>
</tr>
<tr>
<td>WP</td>
<td>AVCWP-1</td>
<td>Toshiba</td>
<td>CVWP5_TOSHIWA_E</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>2.0 and higher</td>
<td>7.5</td>
</tr>
<tr>
<td></td>
<td>AVCWP-2</td>
<td>Toshiba</td>
<td>CVWP1_TOSHIWA_E</td>
<td>X</td>
<td></td>
<td>X</td>
<td>2.0 and higher</td>
<td>7.5</td>
</tr>
<tr>
<td></td>
<td>AVCWP-3</td>
<td>Toshiba</td>
<td>CVWP2_TOSHIWA_E</td>
<td>X</td>
<td></td>
<td>X</td>
<td>2.0 and higher</td>
<td>7.5</td>
</tr>
<tr>
<td></td>
<td>AVCWP-4</td>
<td>Toshiba</td>
<td>CVWP3_TOSHIWA_E</td>
<td>X</td>
<td></td>
<td>X</td>
<td>2.0 and higher</td>
<td>7.5</td>
</tr>
<tr>
<td>Field coding</td>
<td>AVCFI-1</td>
<td>Sony</td>
<td>CVNLFI1_Sony_C</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCFI-2</td>
<td>Sony</td>
<td>CVNLFI2_Sony_H</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCFI-3</td>
<td>Sharp Labs</td>
<td>Sharp_MP Field1_B</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.0 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCFI-4</td>
<td>Sharp Labs</td>
<td>Sharp_MP Field2_B</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.0 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCFI-5</td>
<td>Sharp Labs</td>
<td>Sharp_MP Field3_B</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.0 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCFI-6</td>
<td>Sony</td>
<td>CVFI1_Sony_D</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCFI-7</td>
<td>Sony</td>
<td>CVFI2_Sony_H</td>
<td>X</td>
<td></td>
<td>X</td>
<td>3.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCFI-8</td>
<td>Sony</td>
<td>FI1_Sony_E</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>2.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCFI-9</td>
<td>SVA</td>
<td>CVFI1_SVA_C</td>
<td>X</td>
<td></td>
<td>X</td>
<td>3.0 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCFI-10</td>
<td>SVA</td>
<td>CVFI2_SVA_C</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.0 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCFI-11</td>
<td>Motorola</td>
<td>cvlc_mot fld0_full_B</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>2.2 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCFI-12</td>
<td>Motorola</td>
<td>CVMP_MOT FLD_L30_B</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.0 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td>Frame/field coding</td>
<td>AVCPA-1</td>
<td>Sharp Labs</td>
<td>Sharp_MP_PAFF_1r2</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.0 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCPA-2</td>
<td>Toshiba</td>
<td>CVPA1_TOSHIWA_B</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>2.1 and higher</td>
<td>25</td>
</tr>
<tr>
<td></td>
<td>AVCPA-3</td>
<td>Motorola</td>
<td>cvlc_mot_picafl0_full_B</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>2.2 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td>MBAFF</td>
<td>AVCMA-1</td>
<td>Toshiba</td>
<td>CVMANL1_TOSHIWA_B</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>2.1 and higher</td>
<td>25</td>
</tr>
<tr>
<td>Categories</td>
<td>Bitstream</td>
<td>Donated by</td>
<td>File Name</td>
<td>Baseline</td>
<td>Extended</td>
<td>Main</td>
<td>Level</td>
<td>Frame Rate (Frame/Sec)</td>
</tr>
<tr>
<td>----------------</td>
<td>---------------</td>
<td>------------</td>
<td>--------------------------------</td>
<td>----------</td>
<td>----------</td>
<td>-----</td>
<td>-------------</td>
<td>------------------------</td>
</tr>
<tr>
<td></td>
<td>AVCMA-2</td>
<td>Toshiba</td>
<td>CVMANL2_TOSHIBA_B</td>
<td>X</td>
<td>X</td>
<td></td>
<td>2.1 and higher</td>
<td>25</td>
</tr>
<tr>
<td></td>
<td>AVCMA-3</td>
<td>Sony</td>
<td>CVMA1_Sony_D</td>
<td>X</td>
<td>X</td>
<td></td>
<td>3.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCMA-4</td>
<td>Toshiba</td>
<td>CVMA1_TOSHIBA_B</td>
<td>X</td>
<td>X</td>
<td></td>
<td>2.1 and higher</td>
<td>25</td>
</tr>
<tr>
<td></td>
<td>AVCMA-5</td>
<td>Sony</td>
<td>CVMAQP2_Sony_G</td>
<td>X</td>
<td>X</td>
<td></td>
<td>3.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCMA-6</td>
<td>Sony</td>
<td>CVMAQP3_Sony_D</td>
<td>X</td>
<td>X</td>
<td></td>
<td>2.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCMA-7</td>
<td>Sony</td>
<td>CVMAQP3_Sony_E</td>
<td>X</td>
<td>X</td>
<td></td>
<td>3.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCMA-8</td>
<td>Motorola</td>
<td>cavlc_mot_mbafl0_full_B</td>
<td>X</td>
<td>X</td>
<td></td>
<td>2.2 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCMA-9</td>
<td>Motorola</td>
<td>CVMP_MOT_FRM_L31_B</td>
<td>X</td>
<td>X</td>
<td></td>
<td>3.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td>S Picture</td>
<td>AVCSP-1</td>
<td>British Telecom</td>
<td>SP1_BT_A</td>
<td>X</td>
<td></td>
<td></td>
<td>1.0 and higher</td>
<td>10</td>
</tr>
<tr>
<td></td>
<td>AVCSP-2</td>
<td>British Telecom</td>
<td>SP2_BT_B</td>
<td>X</td>
<td></td>
<td></td>
<td>1.0 and higher</td>
<td>20</td>
</tr>
<tr>
<td>Long Sequence</td>
<td>AVCLS-1</td>
<td>SVA</td>
<td>LS_SVA_D</td>
<td>X</td>
<td>X</td>
<td></td>
<td>1.3 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td>SEI/VUI</td>
<td>AVCSE-1</td>
<td>Sony</td>
<td>CVSE2_Sony_B</td>
<td>X</td>
<td>X</td>
<td></td>
<td>2.1 and higher</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td>AVCSE-2</td>
<td>Sony</td>
<td>CVSE3_Sony_H</td>
<td>X</td>
<td>X</td>
<td></td>
<td>2.1 and higher</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td>AVCSE-3</td>
<td>Sony</td>
<td>CVSEFDFT3_Sony_E</td>
<td>X</td>
<td>X</td>
<td></td>
<td>2.1 and higher</td>
<td>15</td>
</tr>
<tr>
<td>CABAC</td>
<td>AVCCANL-1</td>
<td>Toshiba</td>
<td>CANL1_TOSHIBA_G</td>
<td>X</td>
<td></td>
<td></td>
<td>1.2 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCCANL-2</td>
<td>Sony</td>
<td>CANL1_Sony_E</td>
<td>X</td>
<td></td>
<td></td>
<td>2.1 and higher</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td>AVCCANL-3</td>
<td>Sony</td>
<td>CANL2_Sony_E</td>
<td>X</td>
<td></td>
<td></td>
<td>2.1 and higher</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td>AVCCANL-4</td>
<td>Sony</td>
<td>CANL3_Sony_C</td>
<td>X</td>
<td></td>
<td></td>
<td>1.2 and higher</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td>AVCCANL-5</td>
<td>SVA</td>
<td>CANL1_SVA_B</td>
<td>X</td>
<td></td>
<td></td>
<td>2.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCCANL-6</td>
<td>SVA</td>
<td>CANL2_SVA_B</td>
<td>X</td>
<td></td>
<td></td>
<td>2.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCCANL-7</td>
<td>SVA</td>
<td>CANL3_SVA_B</td>
<td>X</td>
<td></td>
<td></td>
<td>2.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCCANL-8</td>
<td>SVA</td>
<td>CANL4_SVA_B</td>
<td>X</td>
<td></td>
<td></td>
<td>2.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCCABA-1</td>
<td>Sony</td>
<td>CABA1_Sony_D</td>
<td>X</td>
<td></td>
<td></td>
<td>2.1 and higher</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td>AVCCABA-2</td>
<td>Sony</td>
<td>CABA2_Sony_E</td>
<td>X</td>
<td></td>
<td></td>
<td>2.1 and higher</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td>AVCCABA-3</td>
<td>Sony</td>
<td>CABA3_Sony_C</td>
<td>X</td>
<td></td>
<td></td>
<td>1.2 and higher</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td>AVCCABA-4</td>
<td>Toshiba</td>
<td>CABA3_TOSHIBA_E</td>
<td>X</td>
<td></td>
<td></td>
<td>1.2 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCCABA-5</td>
<td>SVA</td>
<td>CABA1_SVA_B</td>
<td>X</td>
<td></td>
<td></td>
<td>2.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCCABA-6</td>
<td>SVA</td>
<td>CABA2_SVA_B</td>
<td>X</td>
<td></td>
<td></td>
<td>2.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCCABA-7</td>
<td>SVA</td>
<td>CABA3_SVA_B</td>
<td>X</td>
<td></td>
<td></td>
<td>2.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCCABA-8</td>
<td>Motorola</td>
<td>cabac_mot_frm0_full</td>
<td>X</td>
<td></td>
<td></td>
<td>2.2 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td>CABAC:</td>
<td>AVCCAIN-1</td>
<td>Sony</td>
<td>CABC13_Sony_B</td>
<td>X</td>
<td></td>
<td></td>
<td>2.1 and higher</td>
<td>15</td>
</tr>
</tbody>
</table>

ITU-T Rec. H.264.1 (03/2005) – Prepublished version
<table>
<thead>
<tr>
<th>Categories</th>
<th>Bitstream</th>
<th>Donated by</th>
<th>File Name</th>
<th>Baseline</th>
<th>Extended</th>
<th>Main</th>
<th>Level</th>
<th>Frame Rate (Frame/Sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CABAC: MB QP Delta</td>
<td>AVCCAQP-1</td>
<td>Sony</td>
<td>CAQP1_Sony_B</td>
<td>X</td>
<td></td>
<td>1.2</td>
<td>and higher</td>
<td>15</td>
</tr>
<tr>
<td></td>
<td>AVCCAQP-2</td>
<td>Sony</td>
<td>CACQP3_Sony_D</td>
<td>X</td>
<td></td>
<td>2.1</td>
<td>and higher</td>
<td>15</td>
</tr>
<tr>
<td>CABAC: Slice</td>
<td>AVCCASL-1</td>
<td>Sony</td>
<td>CABAST3_Sony_E</td>
<td>X</td>
<td></td>
<td>2.1</td>
<td>and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCCASL-2</td>
<td>Sony</td>
<td>CABASTBR3_Sony_B</td>
<td>X</td>
<td></td>
<td>2.1</td>
<td>and higher</td>
<td>29.97</td>
</tr>
<tr>
<td>CABAC: I PCM</td>
<td>AVCCAPCM-1</td>
<td>Broadcom</td>
<td>CAPCMNL1_Sand_E</td>
<td>X</td>
<td></td>
<td>4.0</td>
<td>and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCCAPCM-2</td>
<td>Broadcom</td>
<td>CAPCM1_Sand_E</td>
<td>X</td>
<td></td>
<td>4.0</td>
<td>and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCCAPCM-3</td>
<td>Sony</td>
<td>CAPM3_Sony_D</td>
<td>X</td>
<td></td>
<td>2.1</td>
<td>and higher</td>
<td>15</td>
</tr>
<tr>
<td>CABAC: MMCO</td>
<td>AVCCAMR-1</td>
<td>British Telecom</td>
<td>MR9_BT_B</td>
<td>X</td>
<td></td>
<td>2.1</td>
<td>and higher</td>
<td>25</td>
</tr>
<tr>
<td>CABAC: WP</td>
<td>AVCCAWP-1</td>
<td>Toshiba</td>
<td>CAWP1_TOSHIBA_E</td>
<td>X</td>
<td></td>
<td>2.0</td>
<td>and higher</td>
<td>7.5</td>
</tr>
<tr>
<td></td>
<td>AVCCAWP-2</td>
<td>Toshiba</td>
<td>CAWP5_TOSHIBA_E</td>
<td>X</td>
<td></td>
<td>2.0</td>
<td>and higher</td>
<td>7.5</td>
</tr>
<tr>
<td>CABAC: Field coding</td>
<td>AVCCAFI-1</td>
<td>Broadcom</td>
<td>CABREF3_Sand_D</td>
<td>X</td>
<td></td>
<td>4.0</td>
<td>and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCCAFI-2</td>
<td>SVA</td>
<td>CAFI_SVA_C</td>
<td>X</td>
<td></td>
<td>3.0</td>
<td>and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCCAFI-3</td>
<td>Motorola</td>
<td>cabac_mot fld0_full</td>
<td>X</td>
<td></td>
<td>2.2</td>
<td>and higher</td>
<td>29.97</td>
</tr>
<tr>
<td>CABAC: Frame/Field Coding</td>
<td>AVCCAPA-1</td>
<td>Sharp Labs</td>
<td>Sharp_MP_PAFF_2r</td>
<td>X</td>
<td></td>
<td>3.0</td>
<td>and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCCAPA-2</td>
<td>Toshiba</td>
<td>CAPA1_TOSHIBA_B</td>
<td>X</td>
<td></td>
<td>2.1</td>
<td>and higher</td>
<td>25</td>
</tr>
<tr>
<td></td>
<td>AVCCAPA-3</td>
<td>Motorola</td>
<td>cabac_mot_paff0_full</td>
<td>X</td>
<td></td>
<td>2.2</td>
<td>and higher</td>
<td>29.97</td>
</tr>
<tr>
<td>CABAC: MBAFF</td>
<td>AVCCAMA-1</td>
<td>Toshiba</td>
<td>CAMANL1_TOSHIBA_B</td>
<td>X</td>
<td></td>
<td>2.1</td>
<td>and higher</td>
<td>25</td>
</tr>
<tr>
<td></td>
<td>AVCCAMA-2</td>
<td>Toshiba</td>
<td>CAMANL2_TOSHIBA_B</td>
<td>X</td>
<td></td>
<td>2.1</td>
<td>and higher</td>
<td>25</td>
</tr>
<tr>
<td></td>
<td>AVCCAMA-3</td>
<td>Sony</td>
<td>CANLMA2_Sony_C</td>
<td>X</td>
<td></td>
<td>3.1</td>
<td>and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCCAMA-4</td>
<td>Sony</td>
<td>CANLMA3_Sony_C</td>
<td>X</td>
<td></td>
<td>3.1</td>
<td>and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCCAMA-5</td>
<td>Sony</td>
<td>CAM1_Sony_C</td>
<td>X</td>
<td></td>
<td>3.1</td>
<td>and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCCAMA-6</td>
<td>Toshiba</td>
<td>CAM1_TOSHIBA_B</td>
<td>X</td>
<td></td>
<td>2.1</td>
<td>and higher</td>
<td>25</td>
</tr>
<tr>
<td></td>
<td>AVCCAMA-7</td>
<td>Broadcom</td>
<td>CAMANL3_Sand_E</td>
<td>X</td>
<td></td>
<td>4.0</td>
<td>and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCCAMA-8</td>
<td>Broadcom</td>
<td>CAMA3_Sand_E</td>
<td>X</td>
<td></td>
<td>4.0</td>
<td>and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCCAMA-9</td>
<td>Sony</td>
<td>CAMASL3_Sony_B</td>
<td>X</td>
<td></td>
<td>2.1</td>
<td>and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCCAMA-10</td>
<td>Sony</td>
<td>CAMAC13_Sony_C</td>
<td>X</td>
<td></td>
<td>2.1</td>
<td>and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCCAMA-11</td>
<td>Motorola</td>
<td>cabac_mot_maff0_full</td>
<td>X</td>
<td></td>
<td>2.2</td>
<td>and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCCAMA-12</td>
<td>Motorola</td>
<td>CAMP_MOT_MAFF_L30</td>
<td>X</td>
<td></td>
<td>3.0</td>
<td>and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCCAMA-13</td>
<td>Motorola</td>
<td>CAMP_MOT_MAFF_L31</td>
<td>X</td>
<td></td>
<td>3.1</td>
<td>and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>AVCCAPAMA-1</td>
<td>Broadcom</td>
<td>CAPAMA3_Sand_F</td>
<td>X</td>
<td></td>
<td>4.0</td>
<td>and higher</td>
<td>29.97</td>
</tr>
</tbody>
</table>
### Table 2 – Bitstreams for High, High 10, High 4:2:2, and High 4:4:4 profile

<table>
<thead>
<tr>
<th>Categories</th>
<th>Bitstream</th>
<th>Donated by</th>
<th>File Name</th>
<th>Baseline</th>
<th>Extended</th>
<th>Main</th>
<th>Level</th>
<th>Frame Rate (Frame/Sec)</th>
</tr>
</thead>
<tbody>
<tr>
<td>4:2:0 8 bit</td>
<td>FREH-1</td>
<td>Panasonic Singapore Lab.</td>
<td>FRExt1_Panasonic_C</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>2.1 and higher</td>
</tr>
<tr>
<td>FREH-2</td>
<td>Panasonic Singapore Lab.</td>
<td>FRExt3_Panasonic_D</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>2.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td>FREH-3</td>
<td>HHI</td>
<td>HCAFR1_HHI_C</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.0 and higher</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>FREH-4</td>
<td>HHI</td>
<td>HCAFF1_HHI_B</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.0 and higher</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>FREH-5</td>
<td>HHI</td>
<td>HCAMFF1_HHI_B</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.0 and higher</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>FREH-6</td>
<td>Panasonic Singapore Lab.</td>
<td>FRExt2_Panasonic_B</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>2.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td>FREH-7</td>
<td>Panasonic Singapore Lab.</td>
<td>FRExt4_Panasonic_A</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>2.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td>FREH-8</td>
<td>Broadcom</td>
<td>HPCANL_BRCM_C</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>4.0 and higher</td>
<td>29.97</td>
<td></td>
</tr>
<tr>
<td>FREH-9</td>
<td>Broadcom</td>
<td>HPCA_BRCM_C</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>4.0 and higher</td>
<td>29.97</td>
<td></td>
</tr>
<tr>
<td>FREH-10</td>
<td>Broadcom</td>
<td>HPCAFLNL_BRCM_C</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>4.0 and higher</td>
<td>29.97</td>
<td></td>
</tr>
<tr>
<td>FREH-11</td>
<td>Broadcom</td>
<td>HPCAFL_BRCM_C</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>4.0 and higher</td>
<td>29.97</td>
<td></td>
</tr>
<tr>
<td>Categories</td>
<td>Bitstream</td>
<td>Donated by</td>
<td>File Name</td>
<td>High 10</td>
<td>High 4:2:2</td>
<td>High 4:4:4</td>
<td>Level</td>
<td>Frame Rate (Frames/Sec)</td>
</tr>
<tr>
<td>------------</td>
<td>-----------</td>
<td>------------</td>
<td>--------------------</td>
<td>--------</td>
<td>-----------</td>
<td>------------</td>
<td>----------------</td>
<td>-------------------------</td>
</tr>
<tr>
<td>FREH-12</td>
<td>HHI</td>
<td></td>
<td>HCAFR2_HHI_A</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>2.0 and higher</td>
<td>15</td>
</tr>
<tr>
<td>FREH-13</td>
<td>HHI</td>
<td></td>
<td>HCAFR3_HHI_A</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.0 and higher</td>
<td>15</td>
</tr>
<tr>
<td>FREH-14</td>
<td>HHI</td>
<td></td>
<td>HCAFR4_HHI_A</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.0 and higher</td>
<td>15</td>
</tr>
<tr>
<td>FREH-15</td>
<td>Broadcom</td>
<td></td>
<td>HPCADQ_BRCM_B</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>4.0 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td>FREH-16</td>
<td>Broadcom</td>
<td></td>
<td>HPCALQ_BRCM_B</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>4.0 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td>FREH-17</td>
<td>Broadcom</td>
<td></td>
<td>HPCAMAPALQ_BRCM_B</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>4.0 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td>FREH-18</td>
<td>Broadcom</td>
<td></td>
<td>HPCV_BRCM_A</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>4.0 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td>FREH-19</td>
<td>Broadcom</td>
<td></td>
<td>HPCVNL_BRCM_A</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>4.0 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td>FREH-20</td>
<td>Broadcom</td>
<td></td>
<td>HPCVFL_BRCM_A</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>4.0 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td>FREH-21</td>
<td>Broadcom</td>
<td></td>
<td>HPCVFLNL_BRCM_A</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>4.0 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td>FREH-22</td>
<td>Sony</td>
<td></td>
<td>HVLCFI0_Sony_A</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td>FREH-23</td>
<td>Sony</td>
<td></td>
<td>HVLCPFF0_Sony_A</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td>FREH-24</td>
<td>Sony</td>
<td></td>
<td>HVLCMFF0_Sony_A</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td>FREH-25</td>
<td>Broadcom</td>
<td></td>
<td>HPCVMOLQ_BRCM_B</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>4.0 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td>FREH-26</td>
<td>Broadcom</td>
<td></td>
<td>HPCAMOLQ_BRCM_B</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>4.0 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td>FREH-27</td>
<td>Broadcom</td>
<td></td>
<td>HPCAQ2LQ_BRCM_B</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>4.0 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td>FREH-28</td>
<td>Broadcom</td>
<td></td>
<td>bcm_freh1_B</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.0 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td>FREH-29</td>
<td>Broadcom</td>
<td></td>
<td>bcm_freh2_B</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.0 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td>FREH-30</td>
<td>Broadcom</td>
<td></td>
<td>bcm_freh3</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.0 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td>FREH-31</td>
<td>Broadcom</td>
<td></td>
<td>bcm_freh4</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.0 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td>Categories</td>
<td>Bitstream</td>
<td>Donated by</td>
<td>File Name</td>
<td>High 4:2:0</td>
<td>High 4:2:2</td>
<td>High 4:4:4</td>
<td>Level</td>
<td>Frame Rate (Frame/Sec)</td>
</tr>
<tr>
<td>------------</td>
<td>-----------</td>
<td>------------</td>
<td>-------------</td>
<td>-------------</td>
<td>-------------</td>
<td>-------------</td>
<td>---------</td>
<td>----------------------------</td>
</tr>
<tr>
<td></td>
<td>FREH-32</td>
<td>Broadcom</td>
<td>bcm_freh5</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.0 and higher</td>
</tr>
<tr>
<td></td>
<td>FREH-33</td>
<td>Broadcom</td>
<td>bcm_freh6</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.0 and higher</td>
</tr>
<tr>
<td></td>
<td>FREH-34</td>
<td>Broadcom</td>
<td>bcm_freh7_B</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.0 and higher</td>
</tr>
<tr>
<td></td>
<td>FREH-35</td>
<td>Broadcom</td>
<td>bcm_freh8</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.0 and higher</td>
</tr>
<tr>
<td></td>
<td>FREH-36</td>
<td>Broadcom</td>
<td>bcm_freh9</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.0 and higher</td>
</tr>
<tr>
<td></td>
<td>FREH-37</td>
<td>Broadcom</td>
<td>bcm_freh10</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.0 and higher</td>
</tr>
<tr>
<td></td>
<td>FREH-38</td>
<td>Broadcom</td>
<td>bcm_freh11</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.0 and higher</td>
</tr>
<tr>
<td></td>
<td>FREH-39</td>
<td>Broadcom</td>
<td>bcm_freh12_B</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.0 and higher</td>
</tr>
<tr>
<td>4:2:0 10 bit</td>
<td>FREH10-1</td>
<td>Dolby</td>
<td>FREH10-1</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>4 and higher</td>
</tr>
<tr>
<td></td>
<td>FREH10-2</td>
<td>Dolby</td>
<td>FREH10-2</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>4 and higher</td>
</tr>
<tr>
<td>4:2:2 10 bit</td>
<td>FREH422-1</td>
<td>Tandberg</td>
<td>FREXT3_TANDBERG_A</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>2.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>FREH422-2</td>
<td>Tandberg</td>
<td>FREXT5_TANDBERG_A</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>2.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>FREH422-3</td>
<td>Tandberg</td>
<td>FREXT6_TANDBERG_A</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>2.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>FREH422-4</td>
<td>Sony</td>
<td>Hi422FREXT1_Sony_A</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>FREH422-5</td>
<td>Sony</td>
<td>Hi422FREXT2_Sony_A</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>FREH422-6</td>
<td>Sony</td>
<td>Hi422FREXT3_Sony_A</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>FREH422-7</td>
<td>Sony</td>
<td>Hi422FREXT4_Sony_A</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>FREH422-8</td>
<td>Sony</td>
<td>Hi422FREXT6_Sony_A</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>FREH422-9</td>
<td>Sony</td>
<td>Hi422FREXT7_Sony_A</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td></td>
<td>FREH422-10</td>
<td>Sony</td>
<td>Hi422FREXT8_Sony_A</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>3.1 and higher</td>
<td>29.97</td>
</tr>
<tr>
<td>Categories</td>
<td>Bitstream</td>
<td>Donated by</td>
<td>File Name</td>
<td>High 10</td>
<td>High 4:2:2</td>
<td>High 4:4:4</td>
<td>Level</td>
<td>Frame Rate (Frame/Sec)</td>
</tr>
<tr>
<td>-------------------</td>
<td>-----------</td>
<td>------------</td>
<td>-----------------------------</td>
<td>---------</td>
<td>------------</td>
<td>------------</td>
<td>-------------------</td>
<td>------------------------</td>
</tr>
<tr>
<td>FREH422-11</td>
<td>Sony</td>
<td>Hi422FREXT9_Sony_A</td>
<td>X</td>
<td>X</td>
<td>3.1 and higher</td>
<td>29.97</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FREH422-12</td>
<td>Sony</td>
<td>Hi422FREXT10_Sony_A</td>
<td>X</td>
<td>X</td>
<td>3.1 and higher</td>
<td>29.97</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FREH422-13</td>
<td>Sony</td>
<td>Hi422FREXT11_Sony_A</td>
<td>X</td>
<td>X</td>
<td>3.1 and higher</td>
<td>29.97</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FREH422-14</td>
<td>Sony</td>
<td>Hi422FREXT12_Sony_A</td>
<td>X</td>
<td>X</td>
<td>3.1 and higher</td>
<td>29.97</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FREH422-15</td>
<td>Sony</td>
<td>Hi422FREXT13_Sony_A</td>
<td>X</td>
<td>X</td>
<td>3.1 and higher</td>
<td>29.97</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FREH422-16</td>
<td>Sony</td>
<td>Hi422FREXT14_Sony_A</td>
<td>X</td>
<td>X</td>
<td>3.1 and higher</td>
<td>29.97</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FREH422-17</td>
<td>Sony</td>
<td>Hi422FREXT15_Sony_A</td>
<td>X</td>
<td>X</td>
<td>3.1 and higher</td>
<td>29.97</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FREH422-18</td>
<td>Sony</td>
<td>Hi422FREXT16_Sony_A</td>
<td>X</td>
<td>X</td>
<td>4 and higher</td>
<td>29.97</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FREH422-19</td>
<td>Sony</td>
<td>Hi422FREXT17_Sony_A</td>
<td>X</td>
<td>X</td>
<td>4 and higher</td>
<td>29.97</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FREH422-20</td>
<td>Sony</td>
<td>Hi422FREXT18_Sony_A</td>
<td>X</td>
<td>X</td>
<td>4 and higher</td>
<td>29.97</td>
<td></td>
<td></td>
</tr>
<tr>
<td>FREH422-21</td>
<td>Sony</td>
<td>Hi422FREXT19_Sony_A</td>
<td>X</td>
<td>X</td>
<td>4 and higher</td>
<td>29.97</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4:4:4 12 bit</td>
<td>FREH444-1</td>
<td>FREXT9_Dolby_C</td>
<td>X</td>
<td>4 and higher</td>
<td>24</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FREH444-2</td>
<td>Samsung AIT</td>
<td>FREXT10_Samsung_A</td>
<td>X</td>
<td>4 and higher</td>
<td>24</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Auxiliary codec</td>
<td>FREAUX-1</td>
<td>Apple alphaconformanceA</td>
<td>X</td>
<td>X</td>
<td>2.1 and higher</td>
<td>29.97</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>