SMPTE RP 2019-2:2016
Revision of
SMPTE RP 2019-2:2014

SMPTE RECOMMENDED PRACTICE

VC-3 Decoder and Bitstream Conformance

Table of Contents

Page

Approved
May 18, 2016

Foreword

SMPTE (the Society of Motion Picture and Television Engineers) is an internationally-recognized standards developing organization. Headquartered and incorporated in the United States of America, SMPTE has members in over 80 countries on six continents. SMPTE's Engineering Documents, including Standards, Recommended Practices and Engineering Guidelines, are prepared by SMPTE's Technology Committees. Participation in these Committees is open to all with a bona fide interest in their work. SMPTE cooperates closely with other standards-developing organizations, including ISO, IEC and ITU.

SMPTE Engineering Documents are drafted in accordance with the rules given in its Standards Operations Manual.

SMPTE RP 2019-2 was prepared by Technology Committee 10E.

Intellectual Property

SMPTE draws attention to the fact that it is claimed that compliance with this Standard may involve the use of one or more patents or other intellectual property rights (collectively, "IPR"). The Society takes no position concerning the evidence, validity, or scope of this IPR.

Each holder of claimed IPR has assured the Society that it is willing to License all IPR it owns, and any third party IPR it has the right to sublicense, that is essential to the implementation of this Standard to those (Members and non-Members alike) desiring to implement this Standard under reasonable terms and conditions, demonstrably free of discrimination. Each holder of claimed IPR has filed a statement to such effect with SMPTE. Information may be obtained from the Director, Standards & Engineering at SMPTE Headquarters.

Attention is also drawn to the possibility that elements of this Standard may be subject to IPR other than those identified above. The Society shall not be responsible for identifying any or all such IPR.

1 Scope

This document specifies the testing procedures to be used to verify the conformance of SMPTE VC-3 video decoders and the conformance of VC-3 bitstreams. A description of the test data files is provided. Additionally, information regarding the VC-3 reference decoder is provided.

2 Conformance Notation

Normative text is text that describes elements of the design that are indispensable or contains the conformance language keywords: "shall", "should", or "may". Informative text is text that is potentially helpful to the user, but not indispensable, and can be removed, changed, or added editorially without affecting interoperability. Informative text does not contain any conformance keywords.

All text in this document is, by default, normative, except: the Introduction, any section explicitly labeled as "Informative" or individual paragraphs that start with "Note:”

The keywords "shall" and "shall not" indicate requirements strictly to be followed in order to conform to the document and from which no deviation is permitted.

The keywords, "should" and "should not" indicate that, among several possibilities, one is recommended as particularly suitable, without mentioning or excluding others; or that a certain course of action is preferred but not necessarily required; or that (in the negative form) a certain possibility or course of action is deprecated but not prohibited.

The keywords "may" and "need not" indicate courses of action permissible within the limits of the document.

The keyword “reserved” indicates a provision that is not defined at this time, shall not be used, and may be defined in the future. The keyword “forbidden” indicates “reserved” and in addition indicates that the provision will never be defined in the future.

A conformant implementation according to this document is one that includes all mandatory provisions ("shall") and, if implemented, all recommended provisions ("should") as described. A conformant implementation need not implement optional provisions ("may") and need not implement them as described.

3 Normative References

The following standards contain provisions which, through reference in this text, constitute provisions of this standard. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this standard are encouraged to investigate the possibility of applying the most recent edition of the standards indicated below.

SMPTE ST 2019-1:2016, VC-3 Picture Compression and Data Stream Format

SMPTE VC-3 Test Materials — Reference Decoder Software and Bitstreams

ISO/IEC 23002-1:2006, Information Technology — MPEG Video Technologies — Part 1: Accuracy Requirements for Implementation of Integer-Output 8×8 Inverse Discrete Cosine Transform

4 Overview (Informative)

Figure 1 depicts the VC-3 decoder verification process diagram. There are two test sequences which the decoder will need to pass in order for a target decoder to successfully conform to SMPTE VC-3.

The first test sequence is called Decoder Front-End Conformance Testing and is detailed in Section 6.1.1.

The second test sequence is called the Inverse Discrete Cosine Transform Statistical Test. It is described in Section 6.1.2.

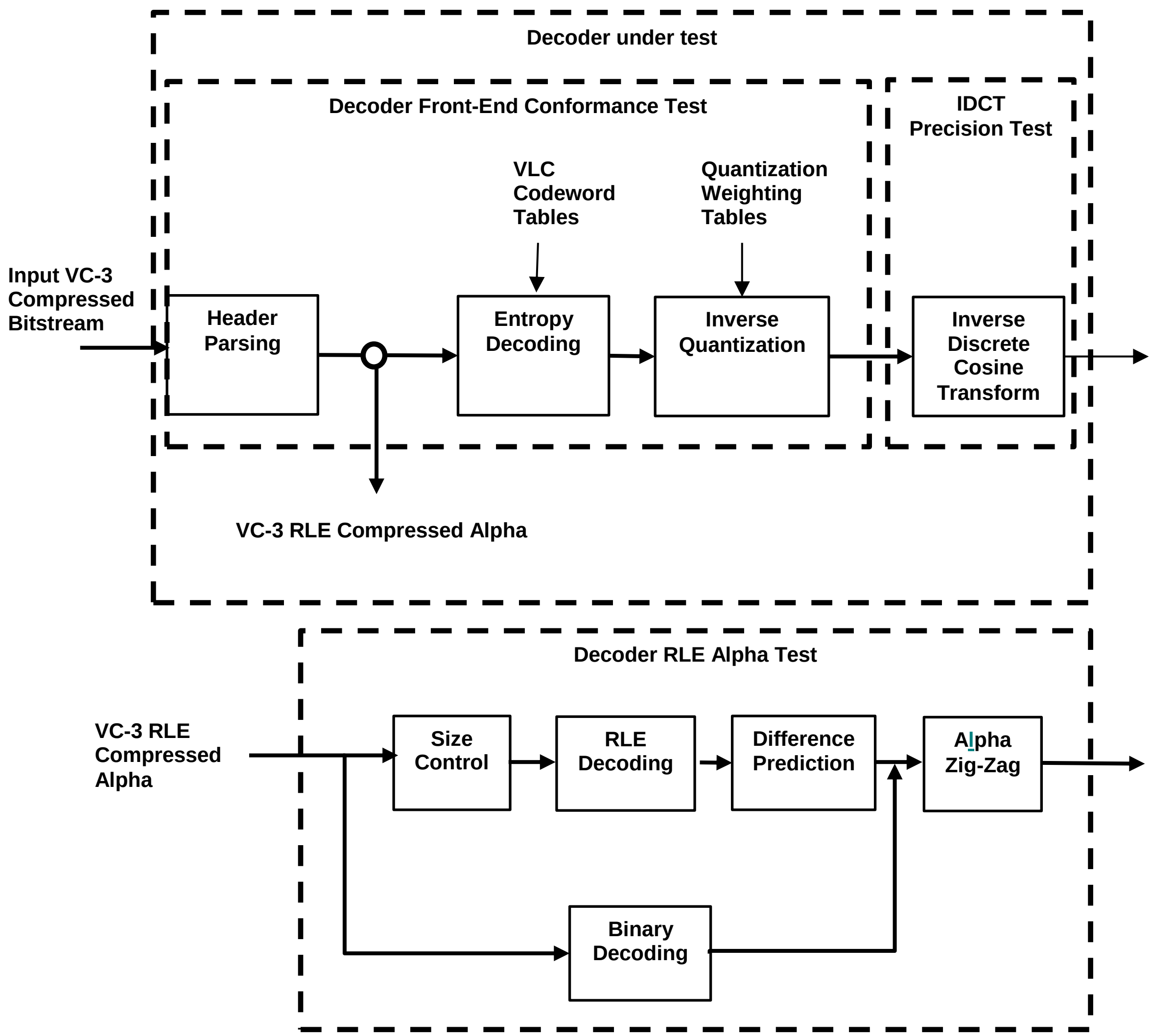

Figure 1 – Target Decoder Conformance Testing

The purpose of the first test (Decoder Front-End Conformance Test) is to test the front-end of the VC-3 decoding stages; i.e. the header parsing, entropy decoding, inverse quantization stages of the decoding process and decoding process of compressed alpha channel. This test compares the output of a reference decoder's diagnostic output against the output of the target decoder. The reference decoder and testing materials are described in Section 5. VC-3 test files and the software reference decoder are available at the

SMPTE Store at http://smpte.org. Navigate to the SMPTE Store, then to Test Materials, to locate the VC-3 materials and the corresponding license. The reference decoder conforms to the complete decoding process specified in SMPTE ST 2019-1. Since the Entropy Decoding, Inverse Quantization and RLE decoding stages are mathematically specified, a conforming decoder's output from this processing chain will match exactly with another conforming decoder's output. This enables the Decoder Front-End Conformance Test to be specified as a binary comparison between the reference decoder's output and the target decoder's output.

The purpose of the second test (Inverse Discrete Cosine Transform Statistical Test) is to measure the statistical properties of an Inverse Discrete Cosine Transform (IDCT) implementation. Since the implementation of the IDCT is not completely specified in SMPTE ST 2019-1, there can be small variations depending upon a decoder's implementation. That is why the statistical measures listed in Section 6.1.2 are applied — the precision and accuracy of a decoder's IDCT are measured to verify that it meets the required statistical tolerance.

The purpose of the third test is to ensure the lossless Alpha decoding works according to specification and that the decoder will return a result which can be used for binary comparison of the overall decoding results.

The International Standards Organization (ISO/IEC) has written a document (ISO/IEC 23002-1) describing accuracy requirements for IDCT operations in video compression systems. This Recommended Practice refers to the ISO/IEC documents when detailing the requirements of the Inverse Discrete Cosine Transform Statistical Tests.

Note: The use of "YC_BC_R" terminology in this document is consistent with its use in ST 2019-1

Finally, Section 6.2 describes the bitstream conformance testing procedure.

5 Description of Testing Materials

5.1 Reference Decoder Software

The reference decoder which implements the complete decoding process specified in SMPTE ST 2019-1 shall be used for testing bitstream conformance. This sub-section describes the operation of the reference decoder.

The reference decoder, in the form of C++ source code and associated documentation, provides decoding functionality for VC-3 conformant bitstreams. It compiles under common "Windows, Linux and Mac OS X" operating systems. The source code will compile to a library, using a generic make file.

The reference decoder supports two modes of operation: "Diagnostic Output" and "Reference Channel Output". These are described further in following sub-sections.

5.1.1 Diagnostic Mode Output

The diagnostic mode output of the reference decoder shall be utilized to verify a candidate decoder's conformance to the VC-3 standard as described in Section 6.1.1. When operating in this mode, the reference decoder will perform the header parsing, the entropy decoding, and the inverse quantization processing steps as shown in Figure 2.

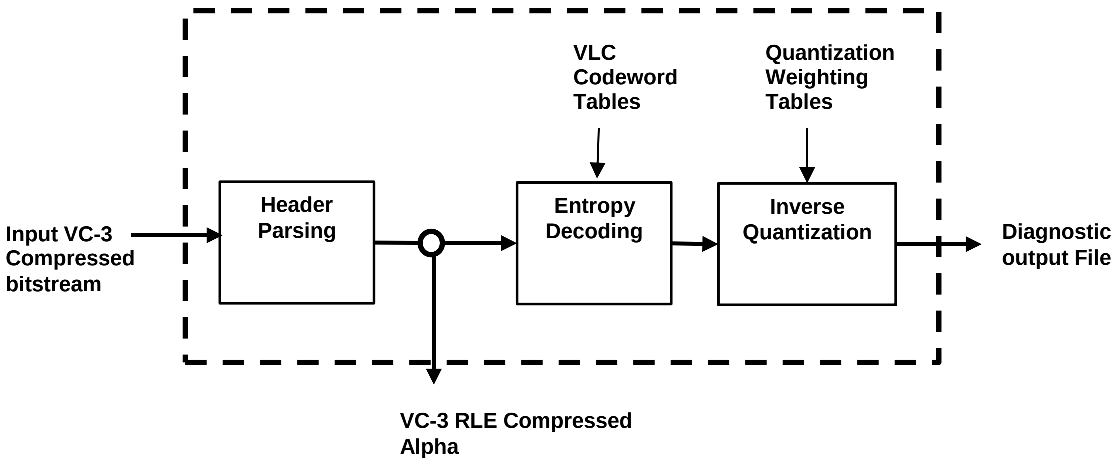

Figure 2 – Reference Decoder in Diagnostic Mode

When the reference decoder is in the diagnostic mode, it will generate diagnostic output files containing RLE Compressed Alpha if present in the bitstream and the 2-D Discrete Cosine Transform (DCT) coefficients $X_k^l(u,v)$, prior to the IDCT.[1] The variable, *l*, indicates the macroblock index. The variable, *k*, indicates the DCT block index. The variables u and v represent the horizontal and vertical indices, respectively.

The ordering of the output data shall follow the raster subdivision into macroblocks (see Figure 2 to Figure 8 as appropriate in SMPTE ST 2019-1). That is, $l = 0,1,...,N_T$, where N_T is the number of macroblocks in a frame or field. Within a macroblock, the ordering shall be according to the DCT order as per Figure 9 to Figure 11 and Figure 15 in SMPTE ST 2019-1. In other words, $k = 0,1,...,3$, $k = 0,1,...,5$, $k = 0,1,...,7$ or $k = 0,1,...,11$.

Within a DCT block, the ordering of the DCT coefficients shall be from top to bottom, left to right as a raster scan would display the image. That is the first coefficient is $X_k^l(0,0)$, followed by $X_k^l(1,0)$. The last coefficient within a DCT block is $X_k^l(7,7)$.

The DCT coefficient data within the diagnostic output files shall be represented by two's complement, 16-bit values (i.e. short) in little endian format. A set of VC-3 compressed bitstreams and the corresponding diagnostic output files are provided with the SMPTE VC-3 test materials.

5.1.2 Reference Channel Output

When operating in the "Reference Channel Output" mode, the reference decoder will perform the entire decoding process described by SMPTE ST 2019-1. This mode shall be used to verify bitstream conformance as detailed in Section 6.2. The "Reference Channel Output" mode will verify the contents of the VC-3 header, compressed payload, and EOF area. If a non-conforming element is found in these regions, the reference decoder will output a diagnostic error message and terminate operation.

[1] The 2-D DCT coefficients should not be confused with the Cosine Basis Function values that are part of the IDCT. These coefficients are the 2-D, 8x8 block DCT transform representation of the encoded video raster.

When operating in the "Reference Channel Output" mode, the reference decoder will utilize a floating point implementation of the IDCT to avoid any problems associated with numerical precision such as arithmetic underflow or overflow.

Conforming decoder implementations need not use a floating point IDCT implementation. Suitable integer IDCT formulations exist.

The output file format for the reference decoder in this mode will depend on the Compression ID. For Compression ID other than 1256 and 1270, format will be header-less binary YC_BC_R(A) 4:2:2(:4) or 4:2:0(:4), with components written in order as per Figure 3 and Figure 4. The samples will be presented in an interleaved order C_B, Y, (A), C_R, Y, (A) or planar order Plane1(Y), Plane2(C_B), Plane3(C_r), (Plane4(A)) . For interlaced video, full-height frames will be produced as field one followed by field two. For compressions ID 1256 and 1270, the output file format will be header-less binary interleaved order C_B, Y, C_R, (A) 4:4:4(:4) or RGB(A) with components written in order as per Figure 5 and Figure 6. Note that alpha-components (shown in {…} and (…)) are optional and will be skipped (not 0 padded) if not present.

The output samples of the reference decoder will be arranged from left to right, top to bottom. The first sample will be the leftmost sample of the top row. This is the same order as a raster scan would display an image.

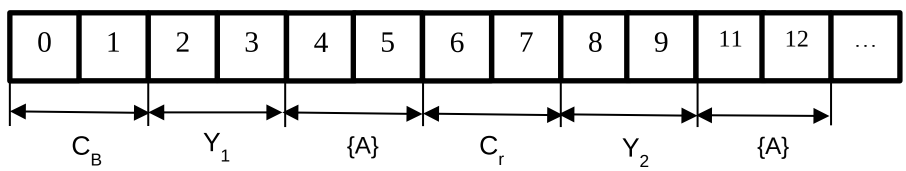

Figure 3 – Decoder output sequence (YC_BC_R(A) 4:2:2(:4))

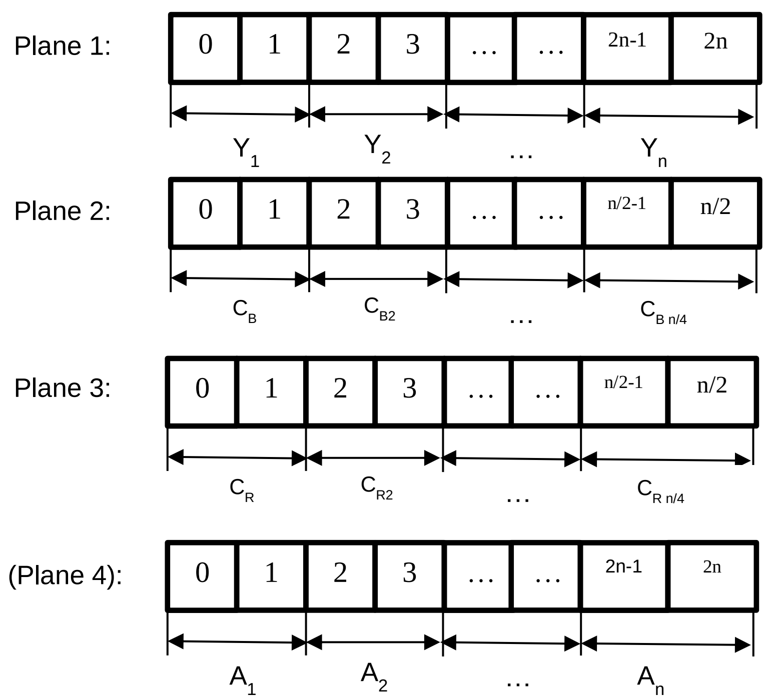

Figure 4 – Decoder output sequence (YC_BC_R(A) 4:2:0(:4))

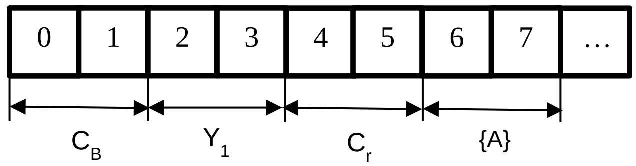

Figure 5 – Decoder output sequence (YC_BC_R(A) 4:4:4(:4))

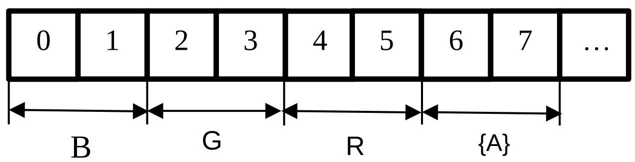

Figure 6 – Decoder output sequence (RGB(A))

For 10-bit or 12-bit components, each will be represented by 2 bytes in ”big endian” order according to Figure 7 and Figure 8; within the 16-bit field, the most significant 10 bits or 12 bit will be valid and the other six or four lower bits will be ignored. Ignored bits will have a value of 0. The most significant 8 bits of a component will be mapped into the first byte (byte 0) and the lower 2/4 bits will be mapped into top 2/4 bits of the second byte (byte 1).

The VC-3 reference decoder will support 8-bit,10-bit and 12-bit decoding modes. The exact decoding mode will be determined by the SBD flag (see ST 2019-1, Figure 20 for more information on the Sample Bit Depth flag). For either bit length, the output of the decoder will be one of described above formats with the lower 8 bits (byte 1) of 16 bits being zero for the 8-bit mode.

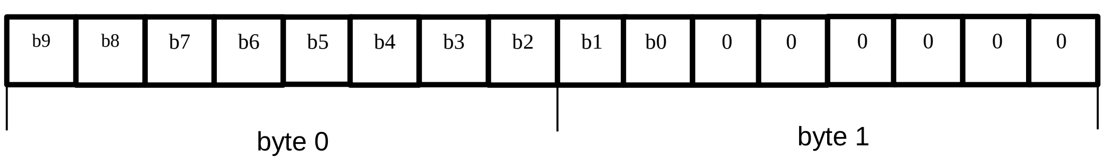

Figure 7 – Layout of the 10-bit data within the 16-bit data field

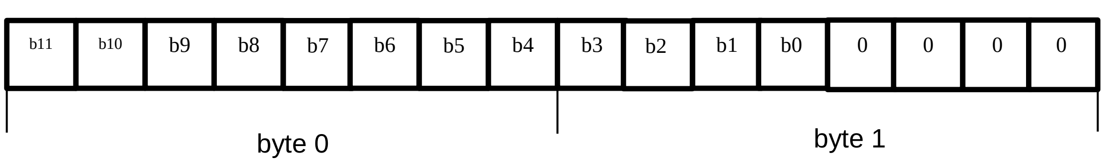

Figure 8 – Layout of the 12-bit data within the 16-bit data field

5.2 Test Bitstreams and Data Files

The set of bitstreams and data files that shall be used for testing a target decoder's conformance are specified in Annex A.1. The bitstreams and data files are provided with the reference decoder. The set of test bitstreams is not intended to represent *all* possible combinations of variable length codewords and DCT coefficients. Rather, the set of bitstreams was developed according to the following guidelines:

- The test bitstreams will follow the bitstream syntax and range of permissible values as defined in SMPTE ST 2019-1.
- Parameters specific to the display process will not be tested, as the display process is outside the scope of the SMPTE ST 2019-1 standard.
- There will be no tests that are frame rate dependent.
- The test bitstreams will contain both "typical video" and edge case conditions in order to better cover the range of possible run/length and DCT coefficient values.

6 Conformance Testing Procedure

The following sub-sections specify the normative tests for verifying conformance of a VC-3 video bitstream and the decoder to SMPTE ST 2019-1. These tests make use of the bitstream test suites (specified in Annex A of this document) and of the reference software decoder described in Section 5.1.

The motivation to apply the decoder testing process is to verify that a candidate decoder implementation is conformant to the decoding process as specified in SMPTE ST 2019-1. If a candidate decoder fails either the Decoder Front-End Conformance Test or the IDCT Statistical test, the candidate decoder does not conform to SMPTE ST 2019-1. On the other hand, successful completion of both tests indicates that the candidate decoder is highly likely to be conformant, but does guarantee conformance.

6.1 Decoder Conformance Requirements

In order to test decoder conformance to SMPTE ST 2019-1, a target decoder shall be tested with both the Decoder Front-End Conformance Test described in Section 6.1.1 and the Inverse Discrete Transform Statistical Tests detailed in Section 6.1.2.

6.1.1 Decoder Front-End Conformance Testing

The decoding process defined by VC-3 is precisely defined for processing steps prior to the IDCT. The RLE decoding, entropy decoding and inverse quantization stages are specified in such a way that the input to the IDCT stage of any conformant decoder shall be identical to any other conformant decoder's input.

Figure 9 and Figure 10 depict the procedure that shall be used to test the front end of the decoding process. The output of the inverse quantization stage of the decoder under test shall be compared to the diagnostic output files obtained by operating the reference decoder in Diagnostic Output mode. Any binary mismatch between the two outputs shall indicate failure and shall mean that the decoder under test is not VC-3 conformant.

The steps for performing a Decoder Front-End Conformance Test for a VC-3 bitstream shall consist of:

1. Present each VC-3 bitstream defined in Table A.1 to the decoder under test.
2. Record the output of the decoder under test after the inverse quantization step (the diagnostic output), but prior to the IDCT.

3. Record the output of the decoder under test after the decoding of lossless alpha (if present)

4. Perform a bit-level, binary comparison of the output of step 2 to the diagnostic output file created by the reference decoder.

5. Perform a bit-level, binary comparison of the output of step 3 to the diagnostic output file created by the reference decoder.

6. Any mismatch between the output of the decoder under test and the diagnostic output of the reference decoder shall indicate a failed test.

7. If no mismatch between the outputs is observed, then the test shall be considered to have passed.

The Front End Conformance Test shall be performed for all 20 VC-3 Compression IDs using the single frame VC-3 bitstream files tabulated in Table A.1. If the test fails for any of the listed bitstreams, the decoder shall not be deemed conformant with SMPTE ST 2019-1.

Informational Notes:

1. It is recommended that the decoder supplier test with additional conformant VC-3 bitstreams and use the reference decoder to generate diagnostic output files of their own.

2. The decoder under test can be “probed” during the development process in order to obtain the values of the 2-D DCT coefficients $X_k^l(u,v)$, prior to the inverse DCT, if such data is not readily accessible in the normal mode of operation. Depending upon the implementation of the decoder under test, the DCT coefficients can be formatted to match the format of the reference decoder’s diagnostic output (see Section 5.1.1). When the formats of the two sets of DCT coefficients are the same, the sets are compared.

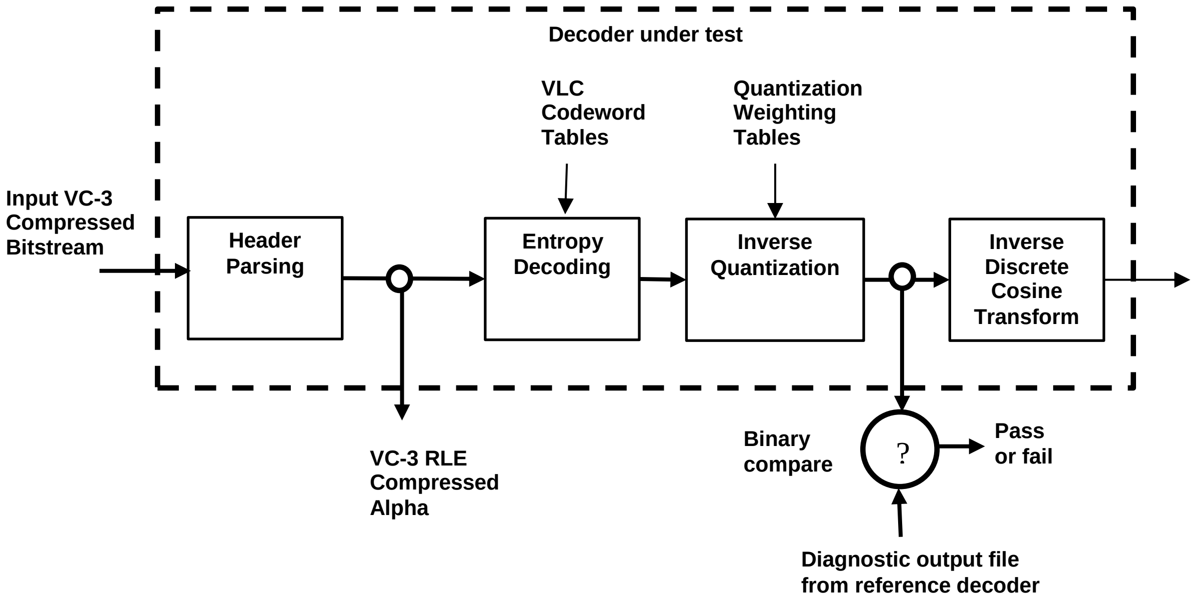

Figure 9 – Decoder front-end conformance testing configuration

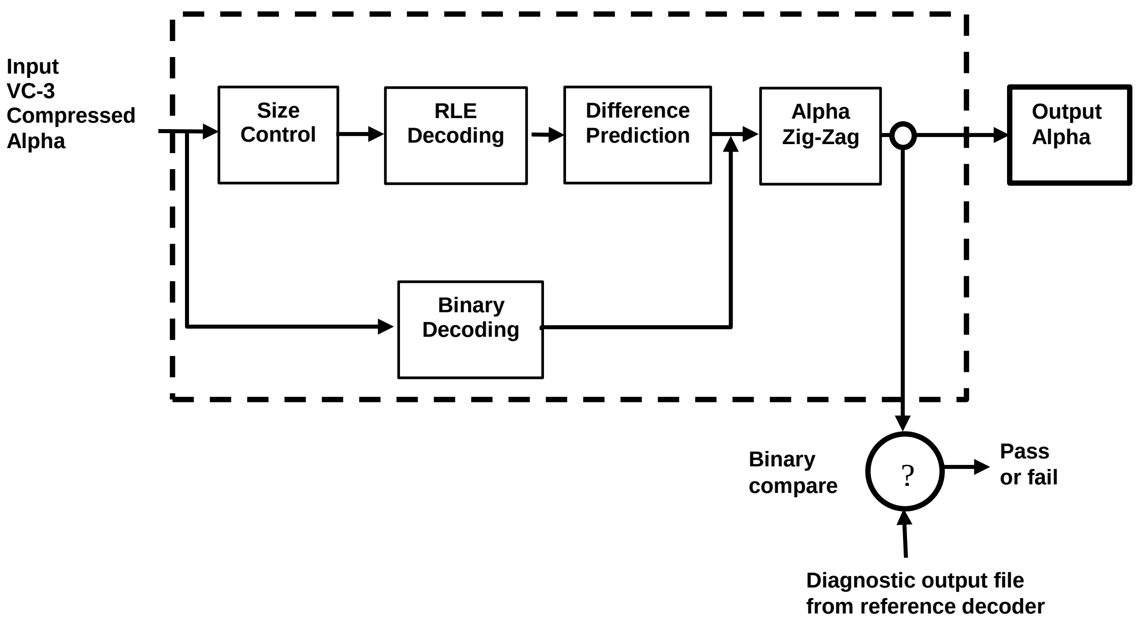

Figure 10 – Decoder front-end conformance testing configuration for lossless alpha

6.1.2 Inverse Discrete Cosine Transform Statistical Tests

In order to verify that the IDCT of a conforming decoder performs with appropriate numerical accuracy and precision, a statistical testing approach shall be taken. The approach shall follow the one presented in the ISO/IEC 23002-1 standard. The ISO/IEC 23002-1 standard requires presenting randomly generated data to the transform under test and comparing its output to the output of a reference implementation with at least 64 bits of floating point accuracy. The comparison is made in terms of a number of statistical measurements.

Note: The first amendment to ISO/IEC 23002-1 describes a reference software implementation for integer IDCT testing. This software implementation is suitable for performing the tests described in this section.

6.1.2.1 Statistical Measures

The ISO/IEC 23002-1 standard defines a set of six statistical measures. The measures are:

- Peak Absolute Error (PAE)
- Absolute Mean Error (AME)
- Mean Squared Error (MSE)
- Overall Mean Error (OME)
- Overall Mean Squared Error (OMSE)
- All Zero Input Case

6.1.2.2 IDCT Statistical Testing Procedure

Sets of random data shall be generated according to the method of random number generation described in ISO/IEC 23002-1. The data shall be presented to the IDCT and the results analyzed in the manner described by 23002-1. The analysis shall yield the values of the 6 statistical measures for each set of random data.

6.1.2.2.1 8-Bit Pixel Data IDCT Test

In order to test the IDCT's statistical performance for 8-bit pixel data, the following sets of random data shall be generated and presented to the IDCT accuracy test. The parameters Q, L, and H are defined in ISO/IEC 23002-1.

- 10,000 random blocks generated with the range of (-128 to 127); i.e. (Q = 10,000, L = 128, and H = 127).
- 10,000 random blocks generated with the range of (-5 to 5); i.e. (Q = 10,000, L = 5, and H = 5).
- 10,000 random blocks generated with the range of (-150 to 150); i.e. (Q = 10,000, L =150, and H = 150).
- Rerun all three of the random data tests, but change the sign on each data value.

For each set of 10,000 blocks of random data, the statistical measures shall be calculated.

6.1.2.2.2 10-Bit Pixel Data IDCT Test

In order to test the IDCT's statistical performance for 10-bit pixel data, the following sets of random data shall be generated and presented to the IDCT accuracy test. The parameters Q, L, and H are defined in ISO/IEC 23002-1.

- 10,000 random blocks generated with the range of (-512 to 511); i.e. (Q = 10,000, L = 512, and H = 511).

- 10,000 random blocks generated with the range of (-5 to 5); i.e. (Q = 10,000, L = 5, and H = 5).
- 10,000 random blocks generated with data in the range of (-600 to 600); i.e. (Q = 10,000, L = 600, and H = 600).
- Re-run all three of the random data tests, but change the sign on each data value.

For each set of 10,000 blocks of random data, the statistical measures shall be calculated.

6.1.2.2.3 12-Bit Pixel Data IDCT Test

In order to test the IDCT's statistical performance for 12-bit pixel data, the following sets of random data shall be generated and presented to the IDCT accuracy test. The parameters Q, L, and H are defined in ISO/IEC 23002-1.

- 10,000 random blocks generated with the range of (-2048 to 2047); i.e. (Q = 10,000, L = 2048, and H = 2047).
- 10,000 random blocks generated with the range of (-5 to 5); i.e. (Q = 10,000, L = 5, and H = 5).
- 10,000 random blocks generated with data in the range of (-2400 to 2400); i.e. (Q = 10,000, L = 2400, and H = 2400).
- Re-run all three of the random data tests, but change the sign on each data value.

For each set of 10,000 blocks of random data, the statistical measures shall be calculated.

6.1.2.3 Test Acceptance Criteria

The outcome of each test shall be an acceptance or rejection of the test transform. The acceptance criterion shall be based on comparing the output of the six statistics to a set of tolerances. If the statistics are within the tolerances for all the presented data (i.e. all sets of 8-, 10- and 12-bit test data), then the transform passes and shall be declared conformant. The tolerance limits for each test are listed in Table 1.

The accuracy and precision of a transform under test shall be considered conformant, if it passes all tests listed in Section 6.1.2.2.

Table 1 – Tolerance Limits for IDCT Testing

Statistic	Limit for 8-Bit Test Data	Limit for 10-Bit Test Data	Limit for 12-Bit Test Data
Peak Absolute Error (PAE)	<= 1	<= 1	<= 1
Absolute Mean Error (AME)	<= 0.015	<= 0.015	<= 0.015
Mean Squared Error (MSE)	<= 0.06	<= 0.06	<= 0.06
Overall Mean Error (OME)	<= 0.0015	<= 0.0015	<= 0.0015
Overall Mean Squared Error (OMSE)	<= 0.02	<= 0.035	<= 0.07
All Zero Input	All zero output	All zero output	All zero output

6.2 Bitstream Conformance Testing

A bitstream shall be conformant with SMPTE ST 2019-1, if all of the following criteria are met.

1. The bitstream conforms to the syntax and specifications of SMPTE ST 2019-1.

2. The reference decoder, when operating in “Reference Channel Output” mode successfully decodes the bitstream with no errors. As described in Section 5.1.2, the reference decoder shall identify non-conforming elements in the VC-3 bitstream such as badly formed headers, invalid compressed data and inconsistent bitstream parameters. When an error occurs, the reference decoder will issue a diagnostic error message and terminate operation before completion of the decoding process. In this case, the bitstream shall be declared non-conformant.

3. When operating in “Reference Channel Output” mode, the reference decoder produces data according to the format described in Section 5.1.2.

Annex A Test Bitstreams

A.1 Overview (Informative)

A.1.1 HD Raster Sizes

The filenames with a ".raw16" suffix contain data in the format specified in Figure 3, Figure 6, and Figure 7 without alpha. The filenames with a ".bin" suffix contain DCT coefficient data in the format specified in Section 5.1.1.

As shown in the table, there is a column labeled "Uncompressed Original Frames". These frames consist of video data that has not been compressed according to the VC-3 compression standard. For each HD raster size (1920x1080, 1280x720, and the two thin rasters: 1440x1080, 960x720), there are four unique uncompressed video frames. These frames covered a wide range of video quality, dynamic range, and color values. The other two video frames in the uncompressed set consist of natural scenes of variable complexity and tone.

The Uncompressed Original Frames were encoded according to SMPTE ST 2019-1. Their filenames are specified in the column labeled "VC-3 Bitstream File." There are 60 unique compressed files because each of four uncompressed frames was encoded for each of the fifteen compression IDs. The content of the compressed files follows the VC-3 bitstream syntax as specified in SMPTE ST 2019-1.

The VC3 bitstream files were decoded by the reference decoder in both the Reference Channel Output mode and Diagnostic Output mode of operation. The files produced by the Reference Channel Output mode are tabulated in the "Decoded Reference File" column, while the Diagnostic Output mode files are listed in the "Diagnostic Output File" column. The diagnostic output files can be used during decoder conformance verification through the Decoder Front End Test as described in Section 6.1.1. The decoded reference files are recommended to perform additional quality verification tests as the vendor sees fit.

A.1.2 Resolution Independent (RI) Raster Sizes

For RI raster size there are 8 uncompressed video frames for testing and a number of unique compressed files which correspond to different combinations of compression parameters. RI uncompressed test media files contain data in the format specified in Figure 3, Figure 4, Figure 5, Figure 6, Figure 7 and Figure 8. The Uncompressed Original Frames were encoded according to SMPTE ST 2019-1.

A.2 Bitstream List (Normative)

A.2.1 For HD Bitstreams

In order to test video sequence decoding, a series of multi-frame bitstreams shall be created from the single frame bitstream files tabulated in Table A.1. For each of the 15 VC-3 Compression IDs, new sequences shall be created by concatenating the four files associated with a compression ID. All 24 possible permutations of concatenation of the four files shall be created. The Decoder Front-End Conformance Test shall be performed for all 360 concatenated bitstreams. If the test fails for any of the listed bitstreams, the decoder shall not be deemed conformant with SMPTE ST 2019-1.

A.2.2 For RI Raster Sizes

For RI raster size there are 8 uncompressed video frames for testing and a number of unique compressed and uncompressed files which correspond to different combinations of compression parameters.

Format of the original filenames:

<OriginalName>_SDI.<Width> x <Height>.<CompFormat>[.<StorageFormat>].<ByteOrder>.<CompOrder><Bit depth>

Example: Challenge_SDI.1920 x 1080.420.Planar.LE.YUV16

OriginalName: Challenge
_SDI: Video components obey SDI-safe range, Alpha uses full range (unclipped)
Width: 1920
Height: 1080
ComponentFormat: 420 (alternates: 422, 444, RGB, 4204, 4224, 4444, RGBA)
StorageFormat: Planar. The StorageFormat is omitted for interleaved formats
ByteOrder: LE (Little Endian) or BE (Big Endian)
ComponentOrder: Y, U, V
BitDepth: 16 bit per component

Tested bit depths are obtained by padding the LSB bits with 0 bits according to the desired bit depth.

The RI diagnostic test files contain data in the format specified in Figure 3, Figure 4, Figure 5, Figure 6, Figure 7 and Figure 8. The names of the diagnostic files are derived from the Original filename as:

<OriginalFilename>_<CompID>_<BitDepth>b_<AlphaPresence>.<ext>

Example: Sunset Alpha_SDI.1920 x 1080.RGBA.LE.BGRA16_1270_12b_lossless_alpha.alpha

Original Filename: As listed in Table A.2
BitDepth: 8, 10, 12
CompID:Compression ID: 1270, 1271, 1272, 1273, 1274
AlphaPresence: lossy_alpha, lossless_alpha, no_alpha
raw16: uncompressed file, **vc3:** Reference Channel output, **bin** and **alpha** for Diagnostic Output mode generated files

Note: **alpha** diagnostic files are only created for lossless alpha.

In Table A.2, there are 8 different columns. The right most 6 columns are labeled using 3 different column label types. These column types are; (1) Original Filename, (2) decorator part and (3) extension part. The “Original Filename” is the (1) file name for uncompressed source videos (column 3 from left) and (2) the preamble portion of the normative conformance files of the form <Original Filename> <decorator part> <extension part>. So the Original Filename is appended, by the testing user, with the “decorator part” and the “extension part” to create the full file name. The form of the extension part depends on the test file usage.

For example, use Table A.2 to create the file: Challenge_SDI.1920 x 1080.420.Planar.LE.YUV16_1271_10b_no_alpha.vc3. This is the decoder input file to test 10 bit CID = 1271 with no alpha. This file name is found in the library of actual test files.

All test files referenced in Tables A.1 and A.2 are provided outside of this Recommended Practice and are available in the SMPTE Store for test materials.

The test files are contained in 4 folders. The folders are named as follows;

- RP2019a-AnnexA.2.RI.OriginalFiles.BEordering(informative)
- RP2019b-AnnexA.2.RI.OriginalFiles.LEordering(informative)
- RP2019c-AnnexA.2.Test-bitstreams(normative).zip
- RP2019d-VC3-reference-source-code(informative).zip

Table A.1 – Test Data Provided with Reference Decoder (HD raster)

Compression ID	Source scan type and encode bit length	Samples per line	Active lines per frame	Uncompressed Original Frames (informative)	VC-3 Bitstream File	Decoded Reference File (informative)	Diagnostic Output File
1235	Progressive 10 bits	1920	1080	1080p422_00.raw16	CID_1235_00.vc3	CID_1235_dec_00.raw16	CID_1235_diag_00.bin
1235	Progressive 10 bits	1920	1080	1080p422_01.raw16	CID_1235_01.vc3	CID_1235_dec_01.raw16	CID_1235_diag_01. bin
1235	Progressive 10 bits	1920	1080	1080p422_02.raw16	CID_1235_02.vc3	CID_1235_dec_02.raw16	CID_1235_diag_02. bin
1235	Progressive 10 bits	1920	1080	1080p422_03.raw16	CID_1235_03.vc3	CID_1235_dec_03.raw16	CID_1235_diag_03. bin
1237	Progressive 8 bits	1920	1080	1080p422_00.raw16	CID_1237_00.vc3	CID_1237_dec_00.raw16	CID_1237_diag_00. bin
1237	Progressive 8 bits	1920	1080	1080p422_01.raw16	CID_1237_01.vc3	CID_1237_dec_01.raw16	CID_1237_diag_01. bin
1237	Progressive 8 bits	1920	1080	1080p422_02.raw16	CID_1237_02.vc3	CID_1237_dec_02.raw16	CID_1237_diag_02. bin
1237	Progressive 8 bits	1920	1080	1080p422_03.raw16	CID_1237_03.vc3	CID_1237_dec_03.raw16	CID_1237_diag_03. bin
1238	Progressive 8 bits	1920	1080	1080p422_00.raw16	CID_1238_00.vc3	CID_1238_dec_00.raw16	CID_1238_diag_00. bin
1238	Progressive 8 bits	1920	1080	1080p422_01.raw16	CID_1238_01.vc3	CID_1238_dec_01.raw16	CID_1238_diag_01. bin
1238	Progressive 8 bits	1920	1080	1080p422_02.raw16	CID_1238_02.vc3	CID_1238_dec_02.raw16	CID_1238_diag_02. bin
1238	Progressive 8 bits	1920	1080	1080p422_03.raw16	CID_1238_03.vc3	CID_1238_dec_03.raw16	CID_1238_diag_03. bin
1241	Interlaced 10 bits	1920	1080	1080i422_00.raw16	CID_1241_00.vc3	CID_1241_dec_00.raw16	CID_1241_diag_00. bin
1241	Interlaced 10 bits	1920	1080	1080i422_01.raw16	CID_1241_01.vc3	CID_1241_dec_01.raw16	CID_1241_diag_01. bin
1241	Interlaced 10 bits	1920	1080	1080i422_02.raw16	CID_1241_02.vc3	CID_1241_dec_02.raw16	CID_1241_diag_02. bin
1241	Interlaced 10 bits	1920	1080	1080i422_03.raw16	CID_1241_03.vc3	CID_1241_dec_03.raw16	CID_1241_diag_03. bin
1242	Interlaced 8 bits	1920	1080	1080i422_00.raw16	CID_1242_00.vc3	CID_1242_dec_00.raw16	CID_1242_diag_00. bin
1242	Interlaced 8 bits	1920	1080	1080i422_01.raw16	CID_1242_01.vc3	CID_1242_dec_01.raw16	CID_1242_diag_01. bin
1242	Interlaced 8 bits	1920	1080	1080i422_02.raw16	CID_1242_02.vc3	CID_1242_dec_02.raw16	CID_1242_diag_02. bin
1242	Interlaced 8 bits	1920	1080	1080i422_03.raw16	CID_1242_03.vc3	CID_1242_dec_03.raw16	CID_1242_diag_03. bin
1243	Interlaced 8 bits	1920	1080	1080i422_00.raw16	CID_1243_00.vc3	CID_1243_dec_00.raw16	CID_1243_diag_00. bin
1243	Interlaced 8 bits	1920	1080	1080i422_01.raw16	CID_1243_01.vc3	CID_1243_dec_01.raw16	CID_1243_diag_01. bin
1243	Interlaced 8 bits	1920	1080	1080i422_02.raw16	CID_1243_02.vc3	CID_1243_dec_02.raw16	CID_1243_diag_02. bin
1243	Interlaced 8 bits	1920	1080	1080i422_03.raw16	CID_1243_03.vc3	CID_1243_dec_03.raw16	CID_1243_diag_03. bin
1244	Interlaced 8 bits	1440	1080	1080i422_TR_0.raw16	CID_1244_00.vc3	CID_1244_dec_00.raw16	CID_1244_diag_00.bin
1244	Interlaced 8 bits	1440	1080	1080i422_TR_1.raw16	CID_1244_01.vc3	CID_1244_dec_01.raw16	CID_1244_diag_01.bin
1244	Interlaced 8 bits	1440	1080	1080i422_TR_2.raw16	CID_1244_02.vc3	CID_1244_dec_02.raw16	CID_1244_diag_02.bin
1244	Interlaced 8 bits	1440	1080	1080i422_TR_3.raw16	CID_1244_03.vc3	CID_1244_dec_03.raw16	CID_1244_diag_03.bin
1250	Progressive 10 bits	1280	720	720p422_00.raw16	CID_1250_00.vc3	CID_1250_dec_00.raw16	CID_1250_diag_00. bin

1250	Progressive 10 bits	1280	720	720p422_01.raw16	CID_1250_01.vc3	CID_1250_dec_01.raw16	CID_1250_diag_01. bin
1250	Progressive 10 bits	1280	720	720p422_02.raw16	CID_1250_02.vc3	CID_1250_dec_02.raw16	CID_1250_diag_02. bin
1250	Progressive 10 bits	1280	720	720p422_03.raw16	CID_1250_03.vc3	CID_1250_dec_03.raw16	CID_1250_diag_03. bin
1251	Progressive 8 bits	1280	720	720p422_00.raw16	CID_1251_00.vc3	CID_1251_dec_00.raw16	CID_1251_diag_00. bin
1251	Progressive 8 bits	1280	720	720p422_01.raw16	CID_1251_01.vc3	CID_1251_dec_01.raw16	CID_1251_diag_01. bin
1251	Progressive 8 bits	1280	720	720p422_02.raw16	CID_1251_02.vc3	CID_1251_dec_02.raw16	CID_1251_diag_02. bin
1251	Progressive 8 bits	1280	720	720p422_03.raw16	CID_1251_03.vc3	CID_1251_dec_03.raw16	CID_1251_diag_03. bin
1252	Progressive 8 bits	1280	720	720p422_00.raw16	CID_1252_00.vc3	CID_1252_dec_00.raw16	CID_1252_diag_00. bin
1252	Progressive 8 bits	1280	720	720p422_01.raw16	CID_1252_01.vc3	CID_1252_dec_01.raw16	CID_1252_diag_01. bin
1252	Progressive 8 bits	1280	720	720p422_02.raw16	CID_1252_02.vc3	CID_1252_dec_02.raw16	CID_1252_diag_02. bin
1252	Progressive 8 bits	1280	720	720p422_03.raw16	CID_1252_03.vc3	CID_1252_dec_03.raw16	CID_1252_diag_03. bin
1253	Progressive 8 bits	1920	1080	1080p422_00.raw16	CID_1253_00.vc3	CID_1253_dec_00.raw16	CID_1253_diag_00. bin
1253	Progressive 8 bits	1920	1080	1080p422_01.raw16	CID_1253_01.vc3	CID_1253_dec_01.raw16	CID_1253_diag_01. bin
1253	Progressive 8 bits	1920	1080	1080p422_02.raw16	CID_1253_02.vc3	CID_1253_dec_02.raw16	CID_1253_diag_02. bin
1253	Progressive 8 bits	1920	1080	1080p422_03.raw16	CID_1253_03.vc3	CID_1253_dec_03.raw16	CID_1253_diag_03. Bin
1256	Progressive 10 bits	1920	1080	1080p444_00.raw16	CID_1256_00.vc3	CID_1256_dec_00.raw16	CID_1256_diag_00. bin
1256	Progressive 10 bits	1920	1080	1080p444_01.raw16	CID_1256_01.vc3	CID_1256_dec_01.raw16	CID_1256_diag_01. bin
1256	Progressive 10 bits	1920	1080	1080p444_02.raw16	CID_1256_02.vc3	CID_1256_dec_02.raw16	CID_1256_diag_02. bin
1256	Progressive 10 bits	1920	1080	1080p444_03.raw16	CID_1256_03.vc3	CID_1256_dec_03.raw16	CID_1256_diag_03. bin
1258	Progressive 10 bits	960	720	720p422_TR 00.raw16	CID_1258_00.vc3	CID_1258_dec_00.raw16	CID_1258_diag_00. bin
1258	Progressive 10 bits	960	720	720p422_TR 01.raw16	CID_1258_01.vc3	CID_1258_dec_01.raw16	CID_1258_diag_01. bin
1258	Progressive 10 bits	960	720	720p422_TR 02.raw16	CID_1258_02.vc3	CID_1258_dec_02.raw16	CID_1258_diag_02. bin
1258	Progressive 10 bits	960	720	720p422_TR 03.raw16	CID_1258_03.vc3	CID_1258_dec_03.raw16	CID_1258_diag_03. bin
1259	Progressive 10 bits	1440	1080	1080p422_TR 00.raw16	CID_1259_00.vc3	CID_1259_dec_00.raw16	CID_1259_diag_00. bin
1259	Progressive 10 bits	1440	1080	1080p422_TR 01.raw16	CID_1259_01.vc3	CID_1259_dec_01.raw16	CID_1259_diag_01. bin
1259	Progressive 10 bits	1440	1080	1080p422_TR 02.raw16	CID_1259_02.vc3	CID_1259_dec_02.raw16	CID_1259_diag_02. bin
1259	Progressive 10 bits	1440	1080	1080p422_TR 03.raw16	CID_1259_03.vc3	CID_1259_dec_03.raw16	CID_1259_diag_03. bin
1260	Interlaced 10 bits	1440	1080	1080i422_TR 00.raw16	CID_1260_00.vc3	CID_1260_dec_00.raw16	CID_1260_diag_00. bin
1260	Interlaced 10 bits	1440	1080	1080i422_TR 01.raw16	CID_1260_01.vc3	CID_1260_dec_01.raw16	CID_1260_diag_01. bin
1260	Interlaced 10 bits	1440	1080	1080i422_TR 02.raw16	CID_1260_02.vc3	CID_1260_dec_02.raw16	CID_1260_diag_02. bin
1260	Interlaced 10 bits	1440	1080	1080i422_TR 03.raw16	CID_1260_03.vc3	CID_1260_dec_03.raw16	CID_1260_diag_03. bin

Table A.2 – Test Data Provided with Reference Decoder (RI raster)

The files listed are for LE (little endian) byte ordering. However, the repository of test files also includes equivalent BE (big endian) byte ordered files.

Compression ID	Encode bit length	Original Filename (Informative)	Decoder diagnostic filename (decorator part)	Decoder Uncompressed output file extension (Informative)	Decoder Diagnostic output file extension	Decoder input VC-3 bitstream file extension	Lossless alpha file extension
1271	10b	Challenge_SDI.1920 x 1080.420.Planar.LE.YUV16	_1271_10b_no_alpha	.raw16	.bin	.vc3	-
1271	12b	Challenge_SDI.1920 x 1080.420.Planar.LE.YUV16	_1271_12b_no_alpha	.raw16	.bin	.vc3	-
1272	8b	Challenge_SDI.1920 x 1080.420.Planar.LE.YUV16	_1272_8b_no_alpha	.raw16	.bin	.vc3	-
1273	8b	Challenge_SDI.1920 x 1080.420.Planar.LE.YUV16	_1273_8b_no_alpha	.raw16	.bin	.vc3	-
1274	8b	Challenge_SDI.1920 x 1080.420.Planar.LE.YUV16	_1274_8b_no_alpha	.raw16	.bin	.vc3	-
1271	10b	Challenge_SDI.1920 x 1080.422.LE.UYVY16	_1271_10b_no_alpha	.raw16	.bin	.vc3	-
1271	12b	Challenge_SDI.1920 x 1080.422.LE.UYVY16	_1271_12b_no_alpha	.raw16	.bin	.vc3	-
1272	8b	Challenge_SDI.1920 x 1080.422.LE.UYVY16	_1272_8b_no_alpha	.raw16	.bin	.vc3	-
1273	8b	Challenge_SDI.1920 x 1080.422.LE.UYVY16	_1273_8b_no_alpha	.raw16	.bin	.vc3	-
1274	8b	Challenge_SDI.1920 x 1080.422.LE.UYVY16	_1274_8b_no_alpha	.raw16	.bin	.vc3	-
1270	10b	Challenge_SDI.1920 x 1080.444.LE.UYV16	_1270_10b_no_alpha	.raw16	.bin	.vc3	-
1270	12b	Challenge_SDI.1920 x 1080.444.LE.UYV16	_1270_12b_no_alpha	.raw16	.bin	.vc3	-
1270	10b	Challenge_SDI.1920 x 1080.RGB.LE.BGR16	_1270_10b_no_alpha	.raw16	.bin	.vc3	-
1270	12b	Challenge_SDI.1920 x 1080.RGB.LE.BGR16	_1270_12b_no_alpha	.raw16	.bin	.vc3	-
1271	10b	ColoredLoops_SDI.1920 x 1282.420.Planar.LE.YUV16	_1271_10b_no_alpha	.raw16	.bin	.vc3	-
1271	12b	ColoredLoops_SDI.1920 x 1282.420.Planar.LE.YUV16	_1271_12b_no_alpha	.raw16	.bin	.vc3	-
1272	8b	ColoredLoops_SDI.1920 x 1282.420.Planar.LE.YUV16	_1272_8b_no_alpha	.raw16	.bin	.vc3	-
1273	8b	ColoredLoops_SDI.1920 x 1282.420.Planar.LE.YUV16	_1273_8b_no_alpha	.raw16	.bin	.vc3	-
1274	8b	ColoredLoops_SDI.1920 x 1282.420.Planar.LE.YUV16	_1274_8b_no_alpha	.raw16	.bin	.vc3	-
1271	10b	ColoredLoops_SDI.1920 x 1282.422.LE.UYVY16	_1271_10b_no_alpha	.raw16	.bin	.vc3	-
1271	12b	ColoredLoops_SDI.1920 x 1282.422.LE.UYVY16	_1271_12b_no_alpha	.raw16	.bin	.vc3	-
1272	8b	ColoredLoops_SDI.1920 x 1282.422.LE.UYVY16	_1272_8b_no_alpha	.raw16	.bin	.vc3	-
1273	8b	ColoredLoops_SDI.1920 x 1282.422.LE.UYVY16	_1273_8b_no_alpha	.raw16	.bin	.vc3	-
1274	8b	ColoredLoops_SDI.1920 x 1282.422.LE.UYVY16	_1274_8b_no_alpha	.raw16	.bin	.vc3	-

1270	10b	ColoredLoops_SDI.1920 x 1282.444.LE.UYV16	_1270_10b_no_alpha	.raw16	.bin	.vc3	-
1270	12b	ColoredLoops_SDI.1920 x 1282.444.LE.UYV16	_1270_12b_no_alpha	.raw16	.bin	.vc3	-
1270	10b	ColoredLoops_SDI.1920 x 1282.RGB.LE.BGR16	_1270_10b_no_alpha	.raw16	.bin	.vc3	-
1270	12b	ColoredLoops_SDI.1920 x 1282.RGB.LE.BGR16	_1270_12b_no_alpha	.raw16	.bin	.vc3	-
1271	10b	LensFlare_SDI.2000 x 1500.420.Planar.LE.YUV16	_1271_10b_no_alpha	.raw16	.bin	.vc3	-
1271	12b	LensFlare_SDI.2000 x 1500.420.Planar.LE.YUV16	_1271_12b_no_alpha	.raw16	.bin	.vc3	-
1272	8b	LensFlare_SDI.2000 x 1500.420.Planar.LE.YUV16	_1272_8b_no_alpha	.raw16	.bin	.vc3	-
1273	8b	LensFlare_SDI.2000 x 1500.420.Planar.LE.YUV16	_1273_8b_no_alpha	.raw16	.bin	.vc3	-
1274	8b	LensFlare_SDI.2000 x 1500.420.Planar.LE.YUV16	_1274_8b_no_alpha	.raw16	.bin	.vc3	-
1271	10b	LensFlare_SDI.2000 x 1500.422.LE.UYVY16	_1271_10b_no_alpha	.raw16	.bin	.vc3	-
1271	12b	LensFlare_SDI.2000 x 1500.422.LE.UYVY16	_1271_12b_no_alpha	.raw16	.bin	.vc3	-
1272	8b	LensFlare_SDI.2000 x 1500.422.LE.UYVY16	_1272_8b_no_alpha	.raw16	.bin	.vc3	-
1273	8b	LensFlare_SDI.2000 x 1500.422.LE.UYVY16	_1273_8b_no_alpha	.raw16	.bin	.vc3	-
1274	8b	LensFlare_SDI.2000 x 1500.422.LE.UYVY16	_1274_8b_no_alpha	.raw16	.bin	.vc3	-
1270	10b	LensFlare_SDI.2000 x 1500.444.LE.UYV16	_1270_10b_no_alpha	.raw16	.bin	.vc3	-
1270	12b	LensFlare_SDI.2000 x 1500.444.LE.UYV16	_1270_12b_no_alpha	.raw16	.bin	.vc3	-
1270	10b	LensFlare_SDI.2000 x 1500.RGB.LE.BGR16	_1270_10b_no_alpha	.raw16	.bin	.vc3	-
1270	12b	LensFlare_SDI.2000 x 1500.RGB.LE.BGR16	_1270_12b_no_alpha	.raw16	.bin	.vc3	-
1270	10b	LinearGradients_12bit_SDI.1920 x 1080.444.LE.UYV16	_1270_10b_no_alpha	.raw16	.bin	.vc3	-
1270	12b	LinearGradients_12bit_SDI.1920 x 1080.444.LE.UYV16	_1270_12b_no_alpha	.raw16	.bin	.vc3	-
1270	10b	LinearGradients_12bit_SDI.1920 x 1080.RGB.LE.BGR16	_1270_10b_no_alpha	.raw16	.bin	.vc3	-
1270	12b	LinearGradients_12bit_SDI.1920 x 1080.RGB.LE.BGR16	_1270_12b_no_alpha	.raw16	.bin	.vc3	-
1270	10b	RadialGradients_12bit_SDI.1920 x 1080.444.LE.UYV16	_1270_10b_no_alpha	.raw16	.bin	.vc3	-
1270	12b	RadialGradients_12bit_SDI.1920 x 1080.444.LE.UYV16	_1270_12b_no_alpha	.raw16	.bin	.vc3	-
1270	10b	RadialGradients_12bit_SDI.1920 x 1080.RGB.LE.BGR16	_1270_10b_no_alpha	.raw16	.bin	.vc3	-
1270	12b	RadialGradients_12bit_SDI.1920 x 1080.RGB.LE.BGR16	_1270_12b_no_alpha	.raw16	.bin	.vc3	-
1271	10b	Rainbow Girls_SDI.1920 x 1236.420.Planar.LE.YUV16	_1271_10b_no_alpha	.raw16	.bin	.vc3	-
1271	12b	Rainbow Girls_SDI.1920 x 1236.420.Planar.LE.YUV16	_1271_12b_no_alpha	.raw16	.bin	.vc3	-
1272	8b	Rainbow Girls_SDI.1920 x 1236.420.Planar.LE.YUV16	_1272_8b_no_alpha	.raw16	.bin	.vc3	-
1273	8b	Rainbow Girls_SDI.1920 x 1236.420.Planar.LE.YUV16	_1273_8b_no_alpha	.raw16	.bin	.vc3	-
1274	8b	Rainbow Girls_SDI.1920 x 1236.420.Planar.LE.YUV16	_1274_8b_no_alpha	.raw16	.bin	.vc3	-

1271	10b	Rainbow Girls_SDI.1920 x 1237.422.LE.UYVY16	_1271_10b_no_alpha	.raw16	.bin	.vc3	-
1271	12b	Rainbow Girls_SDI.1920 x 1237.422.LE.UYVY16	_1271_12b_no_alpha	.raw16	.bin	.vc3	-
1272	8b	Rainbow Girls_SDI.1920 x 1237.422.LE.UYVY16	_1272_8b_no_alpha	.raw16	.bin	.vc3	-
1273	8b	Rainbow Girls_SDI.1920 x 1237.422.LE.UYVY16	_1273_8b_no_alpha	.raw16	.bin	.vc3	-
1274	8b	Rainbow Girls_SDI.1920 x 1237.422.LE.UYVY16	_1274_8b_no_alpha	.raw16	.bin	.vc3	-
1270	10b	Rainbow Girls_SDI.1920 x 1237.444.LE.UYV16	_1270_10b_no_alpha	.raw16	.bin	.vc3	-
1270	12b	Rainbow Girls_SDI.1920 x 1237.444.LE.UYV16	_1270_12b_no_alpha	.raw16	.bin	.vc3	-
1270	10b	Rainbow Girls_SDI.1920 x 1237.RGB.LE.BGR16	_1270_10b_no_alpha	.raw16	.bin	.vc3	-
1270	12b	Rainbow Girls_SDI.1920 x 1237.RGB.LE.BGR16	_1270_12b_no_alpha	.raw16	.bin	.vc3	-
1271	10b	Sunset Alpha_SDI.1920 x 1080.4204.Planar.LE.YUVA16	_1271_10b_lossy_alpha	.raw16	.bin	.vc3	-
1271	10b	Sunset Alpha_SDI.1920 x 1080.4204.Planar.LE.YUVA16	_1271_10b_no_alpha	.raw16	.bin	.vc3	-
1271	12b	Sunset Alpha_SDI.1920 x 1080.4204.Planar.LE.YUVA16	_1271_12b_lossy_alpha	.raw16	.bin	.vc3	-
1271	12b	Sunset Alpha_SDI.1920 x 1080.4204.Planar.LE.YUVA16	_1271_12b_no_alpha	.raw16	.bin	.vc3	-
1272	8b	Sunset Alpha_SDI.1920 x 1080.4204.Planar.LE.YUVA16	_1272_8b_lossy_alpha	.raw16	.bin	.vc3	-
1272	8b	Sunset Alpha_SDI.1920 x 1080.4204.Planar.LE.YUVA16	_1272_8b_no_alpha	.raw16	.bin	.vc3	-
1273	8b	Sunset Alpha_SDI.1920 x 1080.4204.Planar.LE.YUVA16	_1273_8b_lossy_alpha	.raw16	.bin	.vc3	-
1273	8b	Sunset Alpha_SDI.1920 x 1080.4204.Planar.LE.YUVA16	_1273_8b_no_alpha	.raw16	.bin	.vc3	-
1274	8b	Sunset Alpha_SDI.1920 x 1080.4204.Planar.LE.YUVA16	_1274_8b_no_alpha	.raw16	.bin	.vc3	-
1271	10b	Sunset Alpha_SDI.1920 x 1080.4224.LE.UYAVYA16	_1271_10b_lossy_alpha	.raw16	.bin	.vc3	-
1271	10b	Sunset Alpha_SDI.1920 x 1080.4224.LE.UYAVYA16	_1271_10b_no_alpha	.raw16	.bin	.vc3	-
1271	12b	Sunset Alpha_SDI.1920 x 1080.4224.LE.UYAVYA16	_1271_12b_lossy_alpha	.raw16	.bin	.vc3	-
1271	12b	Sunset Alpha_SDI.1920 x 1080.4224.LE.UYAVYA16	_1271_12b_no_alpha	.raw16	.bin	.vc3	-
1272	8b	Sunset Alpha_SDI.1920 x 1080.4224.LE.UYAVYA16	_1272_8b_lossy_alpha	.raw16	.bin	.vc3	-
1272	8b	Sunset Alpha_SDI.1920 x 1080.4224.LE.UYAVYA16	_1272_8b_no_alpha	.raw16	.bin	.vc3	-
1273	8b	Sunset Alpha_SDI.1920 x 1080.4224.LE.UYAVYA16	_1273_8b_lossy_alpha	.raw16	.bin	.vc3	-
1273	8b	Sunset Alpha_SDI.1920 x 1080.4224.LE.UYAVYA16	_1273_8b_no_alpha	.raw16	.bin	.vc3	-
1274	8b	Sunset Alpha_SDI.1920 x 1080.4224.LE.UYAVYA16	_1274_8b_no_alpha	.raw16	.bin	.vc3	-
1270	10b	Sunset Alpha_SDI.1920 x 1080.4444.LE.UYVA16	_1270_10b_lossless_alpha	.raw16	.bin	.vc3	.alpha
1270	10b	Sunset Alpha_SDI.1920 x 1080.4444.LE.UYVA16	_1270_10b_lossy_alpha	.raw16	.bin	.vc3	.alpha
1270	10b	Sunset Alpha_SDI.1920 x 1080.4444.LE.UYVA16	_1270_10b_no_alpha	.raw16	.bin	.vc3	-

1270	12b	Sunset Alpha_SDI.1920 x 1080.4444.LE.UYVA16	_1270_12b_lossless_alpha	.raw16	.bin	.vc3	.alpha
1270	12b	Sunset Alpha_SDI.1920 x 1080.4444.LE.UYVA16	_1270_12b_lossy_alpha	.raw16	.bin	.vc3	-
1270	12b	Sunset Alpha_SDI.1920 x 1080.4444.LE.UYVA16	_1270_12b_no_alpha	.raw16	.bin	.vc3	-
1270	10b	Sunset Alpha_SDI.1920 x 1080.RGBA.LE.BGRA16	_1270_10b_lossless_alpha	.raw16	.bin	.vc3	alpha
1270	10b	Sunset Alpha_SDI.1920 x 1080.RGBA.LE.BGRA16	_1270_10b_lossy_alpha	.raw16	.bin	.vc3	-
1270	10b	Sunset Alpha_SDI.1920 x 1080.RGBA.LE.BGRA16	_1270_10b_no_alpha	.raw16	.bin	.vc3	-
1270	12b	Sunset Alpha_SDI.1920 x 1080.RGBA.LE.BGRA16	_1270_12b_lossless_alpha	.raw16	.bin	.vc3	alpha
1270	12b	Sunset Alpha_SDI.1920 x 1080.RGBA.LE.BGRA16	_1270_12b_lossy_alpha	.raw16	.bin	.vc3	-
1270	12b	Sunset Alpha_SDI.1920 x 1080.RGBA.LE.BGRA16	_1270_12b_no_alpha	.raw16	.bin	.vc3	-

Annex B Bibliography (Informative)

ISO/IEC 23002-1:2006 / Amd 1:2008, Information Technology — MPEG Video Technologies — Part 1: Accuracy Requirements for Implementation of Integer-output 8x8 Inverse Discrete Cosine Transform, Amendment 1: Software for Integer IDCT Accuracy Testing