INTERNATIONAL ORGANISATION FOR STANDARDISATION

ORGANISATION INTERNATIONALE DE NORMALISATION

ISO/IEC JTC1/SC29/WG11

CODING OF MOVING PICTURES AND AUDIO INFORMATION

ISO/IEC JTC1/SC29/WG11 N1216

28 March 1996

Conformance

INFORMATION TECHNOLOGY -

GENERIC CODING OF MOVING PICTURES AND ASSOCIATED AUDIO: CONFORMANCE

ISO/IEC 13818-4

International Standard

Draft of: 16:33 29 March 1996

Contents

ISO/IEC Copyright Office • Case Postale 56 • CH1211 Genèva 20 • Switzerland

Printed in Switzerland.

Foreword

Foreword to be provided by ISO/CS.

Introduction

Parts 1, 2 and 3 of ISO/IEC 13818 specify a multiplex structure and coded representations of audio-visual information. Parts 1, 2 and 3 of ISO/IEC 13818 allow for large flexibility, achieving suitability of ISO/IEC 13818 for many different applications. The flexibility is obtained by including parameters in the bitstream that define the characteristics of coded bitstreams. Examples are the audio sampling frequency, picture size, picture rate and bitrate parameters.

This part of ISO/IEC 13818 specifies how tests can be designed to verify whether bitstreams and decoders meet the requirements as specified in parts 1, 2 and 3 of ISO/IEC 13818. These tests can be used for various purposes such as:

- manufacturers of encoders, and their customers, can use the tests to verify whether the encoder produces valid bitstreams.

- manufacturers of decoders and their customers can use the tests to verify whether the decoder meets the requirements specified in parts 1,2 and 3 of ISO/IEC 13818 for the claimed decoder capabilities.

Information technology -- Generic Coding of Moving Pictures and Associated Audio

Part 4:
Conformance Testing

1 General

1.1 Scope

This part of ISO/IEC 13818 specifies how tests can be designed to verify whether bitstreams and decoders meet requirements specified in parts 1, 2 and 3 of ISO/IEC 13818. In this part of ISO/IEC 13818, encoders are not addressed specifically. An encoder may be said to be an ISO/IEC 13818 encoder if it generates bitstreams compliant with the syntactic and semantic bitstream requirements specified in parts 1, 2 and 3 of ISO/IEC 13818.

Characteristics of coded bitstreams and decoders are defined for parts 1, 2 and 3 of ISO/IEC 13818. 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 bitrate parameters. Decoder characteristics define the properties and capabilities of the applied decoding process. An example of a property is the applied arithmetic accuracy. The capabilities of a decoder specify which coded bitstreams the decoder can decode and reconstruct, by defining the subset of the standard that may be exploited in decodable bitstreams. A bitstream can be decoded by a decoder if the characteristics of the coded 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 parts 1, 2 and 3 of ISO/IEC 13818. Given the set of characteristics claimed, the requirements that must be met are fully determined by parts 1, 2 and 3 of ISO/IEC 13818. This part of ISO/IEC 13818 summarises 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, some test bitstreams implemented according to those guidelines are provided as an electronic annex to this document.

1.2 Normative references

The following standards contain provisions which, through reference in this text, constitute provisions of this part of ISO/IEC 13818. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this part of ISO/IEC 13818 are encouraged to investigate the possibility of applying the most recent editions of the standards indicated below. Members of IEC and ISO maintain registers of currently valid International Standards.

ISO 639:1988, *Code for the representation of names of languages*

ISO 8859-1:1987, *Information processing-8-bit single-byte coded graphic character sets-Part 1: Latin alphabet No. 1*

ISO/IEC 10918-1, *Information technology-Digital compression and coding of continuous-tone still images: Requirements and guidelines.* (See also ITU-T Rec. T.81.)

ISO/IEC 11172-1:1993, *Information technology-Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s-Part 1: Systems.*

ISO/IEC 11172-2:1993, *Information technology-Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s-Part 2: Video*.

ISO/IEC 11172-3:1993, *Information technology-Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s-Part 3: Audio*.

ISO/IEC 11172-4:1993, *Information technology-Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s-Part 4: Conformance*.

ISO/IEC 13818-1:1996, *Information technology-Generic coding of moving pictures and associated audio information-Part 1: Systems.*

ISO/IEC 13818-2:1996, *Information technology-Generic coding of moving pictures and associated audio information-Part 2: Video*.

ISO/IEC 13818-3:1996, *Information technology-Generic coding of moving pictures and associated audio information-Part 3: Audio*.

Recommendations and reports of the CCIR, 1990

XVIIth Plenary Assembly, Dusseldorf, 1990 Volume XI - Part 1

Broadcasting Service (Television) Rec. 601-2, *Encoding parameters of digital television for studios.*

CCIR Volume X and XI Part 3 Recommendation 648: *Recording of audio signals*.

CCIR Volume X and XI Part 3 Report 955-2: *Sound broadcasting by satellite for portable and mobile receivers, including Annex IV Summary description of advanced digital system II.*

IEEE Standard Specifications for the Implementations of 8 by 8 Inverse Discrete Cosine Transform, IEEE Std 1180-1990, December 6, 1990.

IEC 461:1986, *Time and control code for video tape recorders.*

IEC 908:198, *Compact disk digital audio system.*

ITU-T Recommendation H.261 (Formerly CCITT Recommendation H.261) “Codec for audiovisual services at px64 kbit/s” Geneva, 1990.

2 Technical elements

2.1 Definitions

For the purposes of this part of ISO/IEC 13818, the following definitions apply.

2.1.1 16x8 prediction [video]: A prediction mode similar to field-based prediction but where the predicted block size is 16x8 luminance samples.

2.1.2 AC coefficient [video]: Any DCT coefficient for which the frequency in one or both dimensions is non-zero.

2.1.3 access unit [systems]: A coded representation of a presentation unit. In the case of audio, an access unit is the coded representation of an audio frame.
In the case of video, an access unit includes all the coded data for a picture, and any stuffing that follows it, up to but not including the start of the next access unit. If a picture is not preceded by a group_start_code or a sequence_header_code, the access unit begins with the picture start code. If a picture is preceded by a group_start_code and/or a sequence_header_code, the access unit begins with the first byte of the first of these start codes. If it is the last picture preceding a sequence_end_code in the bitstream all bytes between the last byte of the coded picture and the sequence_end_code (including the sequence_end_code) belong to the access unit.

2.1.4 adaptive bit allocation [audio]: The assignment of bits to subbands in a time and frequency varying fashion according to a psychoacoustic model.

2.1.5 adaptive multichannel prediction [audio]: A method of multichannel data reduction exploiting statistical inter-channel dependencies.

2.1.6 adaptive noise allocation [audio]: The assignment of coding noise to frequency bands in a time and frequency varying fashion according to a psychoacoustic model.

2.1.7 adaptive segmentation [audio]: A subdivision of the digital representation of an audio signal in variable segments of time.

2.1.8 alias [audio]: Mirrored signal component resulting from sub-Nyquist sampling.

2.1.9 analysis filterbank [audio]: Filterbank in the encoder that transforms a broadband PCM audio signal into a set of subsampled subband samples.

2.1.10 ancillary data [audio]: part of the bitstream that might be used for transmission of ancillary data.

2.1.11 audio access unit [audio]: For Layers I and II, an audio access unit is defined as the smallest part of the encoded bitstream which can be decoded by itself, where decoded means "fully reconstructed sound". For Layer III, an audio access unit is part of the bitstream that is decodable with the use of previously acquired main information.

2.1.12 audio buffer [audio]: A buffer in the system target decoder for storage of compressed audio data.

2.1.13 audio sequence [audio]: A non-interrupted series of audio frames (base frames plus optional extension frames) in which the following parameters are not changed:

- ID
- Layer
- Sampling Frequency

For Layer I and II, a decoder is not required to support a continuously variable bitrate (change in the bitrate index) of the base stream. Such a relaxation of requirements does not apply to the extension stream.

2.1.14 B-field picture [video]: A field structure B-Picture.

2.1.15 B-frame picture [video]: A frame structure B-Picture.

2.1.16 B-picture; **bidirectionally predictive-coded picture [video]**: A picture that is coded using motion compensated prediction from past and/or future reference fields or frames.

2.1.17 backward compatibility: A newer coding standard is backward compatible with an older coding standard if decoders designed to operate with the older coding standard are able to continue to operate by decoding all or part of a bitstream produced according to the newer coding standard.

2.1.18 backward motion vector [video]: A motion vector that is used for motion compensation from a reference frame or reference field at a later time in display order.

2.1.19 backward prediction [video]: Prediction from the future reference frame (field).

2.1.20 Bark [audio]: Unit of critical band rate. The Bark scale is a non-linear mapping of the frequency scale over the audio range closely corresponding with the frequency selectivity of the human ear across the band.

2.1.21 base layer [video]: First, independently decodable layer of a scalable hierarchy.

2.1.22 big picture [video]: A coded picture that would cause VBV buffer underflow as defined in C.7 Annex C of ISO/IEC 13818-2. Big pictures can only occur in sequences where low_delay is equal to 1. “Skipped picture” is a term that is sometimes used to describe the same concept.

2.1.23 bitrate [audio]: The rate at which the compressed bitstream is delivered to the input of a decoder.

2.1.24 bitstream; stream: An ordered series of bits that forms the coded representation of the data.

2.1.25 bitstream verifier [video]: A process by which it is possible to test and verify that all the requirements specified in ISO/IEC 13818-2 are met by the bitstream.

2.1.26 block [video]: An 8-row by 8-column matrix of samples, or 64 DCT coefficients (source, quantised or dequantised).

2.1.27 block companding [audio]: Normalising of the digital representation of an audio signal within a certain time period.

2.1.28 bottom field [video]: One of two fields that comprise a frame. Each line of a bottom field is spatially located immediately below the corresponding line of the top field.

2.1.29 bound [audio]: The lowest subband in which intensity stereo coding is used.

2.1.30 byte aligned: A bit in a coded bitstream is byte-aligned if its position is a multiple of 8-bits from the first bit in the stream.

2.1.31 byte: Sequence of 8-bits.

2.1.32 centre channel [audio]: An audio presentation channel used to stabilise the central component of the frontal stereo image.

2.1.33 channel [audio]: A sequence of data representing an audio signal being transported.

2.1.34 chroma simulcast [video]: A type of scalability (which is a subset of SNR scalability) where the enhancement layer (s) contain only coded refinement data for the DC coefficients, and all the data for the AC coefficients, of the chrominance components.

2.1.35 chrominance format [video]: Defines the number of chrominance blocks in a macroblock.

2.1.36 chrominance component [video]: A matrix, block or single sample representing one of the two colour difference signals related to the primary colours in the manner defined in the bitstream. The symbols used for the chrominance signals are Cr and Cb.

2.1.37 coded audio bitstream [audio]: A coded representation of an audio signal as specified in part 3 of ISO/IEC 13818.

2.1.38 coded B-frame [video]: A B-frame picture or a pair of B-field pictures.

2.1.39 coded frame [video]: A coded frame is a coded I-frame, a coded P-frame or a coded B-frame.

2.1.40 coded I-frame [video]: An I-frame picture or a pair of field pictures, where the first field picture is an I-picture and the second field picture is an I-picture or a P-picture.

2.1.41 coded order [video]: The order in which the pictures are transmitted and decoded. This order is not necessarily the same as the display order.

2.1.42 coded P-frame [video]: A P-frame picture or a pair of P-field pictures.

2.1.43 coded picture [video]: A coded picture is made of a picture header, the optional extensions immediately following it, and the following picture data. A coded picture may be a coded frame or a coded field.

2.1.44 coded representation: A data element as represented in its encoded form.

2.1.45 coded video bitstream [video]: A coded representation of a series of one or more pictures as defined in ISO/IEC 13818-2.

2.1.46 coding parameters [video]: The set of user-definable parameters that characterise a coded bitstream. Bitstreams are characterised by coding parameters. Decoders are characterised by the bitstreams that they are capable of decoding.

2.1.47 component [video]: A matrix, block or single sample from one of the three matrices (luminance and two chrominance) that make up a picture.

2.1.48 compression: Reduction in the number of bits used to represent an item of data.

2.1.49 constant bitrate: Operation where the bitrate is constant from start to finish of the coded bitstream.

2.1.50 constrained parameters [video]: The values of the set of coding parameters defined in 2.4.3.2 of ISO/IEC 11172-2.

2.1.51 constrained system parameter stream; CSPS [systems]: A Program Stream for which the constraints defined in subclause 2.7.9 of ISO/IEC 13818-1 apply.

2.1.52 CRC: The Cyclic Redundancy Check to verify the correctness of data.

2.1.53 critical band [audio]: Psychoacoustic measure in the spectral domain which corresponds to the frequency selectivity of the human ear. This selectivity is expressed in Bark.

2.1.54 critical band rate [audio]: Psychoacoustic function of frequency. At a given audible frequency, it is proportional to the number of critical bands below that frequency. The units of the critical band rate scale are Barks.

2.1.55 data element: An item of data as represented before encoding and after decoding.

2.1.56 data partitioning [video]: A method for dividing a bitstream into two separate bitstreams for error resilience purposes. The two bitstreams have to be recombined before decoding.

2.1.57 DC coefficient [video]: The DCT coefficient for which the frequency is zero in both dimensions.

2.1.58 DCT coefficient [video]: The amplitude of a specific cosine basis function.

2.1.59 de-emphasis [audio]: Filtering applied to an audio signal after storage or transmission to undo a linear distortion due to emphasis.

2.1.60 decoded stream: The decoded reconstruction of a compressed bitstream.

2.1.61 decoder input buffer [video]: The first-in first-out (FIFO) buffer specified in the video buffering verifier.

2.1.62 decoder: An embodiment of a decoding process.

2.1.63 decoder sub-loop [video]: Stages within encoder which produce numerically identical results to the decode process described in ISO/IEC 13818-2 Chapter 7. Encoders capable of producing more than just I-pictures embed a decoder sub-loop to create temporal predictions and to model the behaviour of downstream decoders.

2.1.64 decoding (process): The process defined in ISO/IEC 13818 parts 1, 2 and 3 that reads an input coded bitstream and outputs decoded pictures or audio samples.

2.1.65 decoding time-stamp; DTS [systems]: A field that may be present in a PES packet header that indicates the time that an access unit is decoded in the system target decoder.

2.1.66 dequantisation: The process of rescaling the quantised DCT coefficients after their representation in the bitstream has been decoded and before they are presented to the inverse DCT.

2.1.67 digital storage media; DSM: A digital storage or transmission device or system.

2.1.68 discrete cosine transform; DCT: Either the forward discrete cosine transform or the inverse discrete cosine transform. The DCT is an invertible, discrete orthogonal transformation.

2.1.69 display aspect ratio [video]: The ratio height/width (in SI units) of the intended display.

2.1.70 display order [video]: The order in which the decoded pictures are displayed. Normally this is the same order in which they were presented at the input of the encoder.

2.1.71 display process [video]: The (non-normative) process by which reconstructed frames are displayed.

2.1.72 downmix [audio]: A matrixing of n channels to obtain less than n channels.

2.1.73 drift [video]: Accumulation of mismatch between the reconstructed output produced by the hypothetical decoder sub-loop embedded within an encoder (see definition of "decoder sub-loop") and the reconstructed outputs produced by a (downstream) decoder.

2.1.74 DSM-CC: digital storage media command and control.

2.1.75 dual channel mode [audio]: A mode, where two audio channels with independent programme contents (e.g. bilingual) are encoded within one bitstream. The coding process is the same as for the stereo mode.

2.1.76 dual-prime prediction [video]: A prediction mode in which two forward field-based predictions are averaged. The predicted block size is 16x16 luminance samples. Dual-prime prediction is only used in interlaced P-pictures.

2.1.77 dynamic crosstalk [audio]: A method of multichannel data reduction in which stereo-irrelevant signal components are copied to another channel.

2.1.78 dynamic transmission channel switching [audio]: A method of multichannel data reduction by allocating the most orthogonal signal components to the transmission channels.

2.1.79 editing: The process by which one or more coded bitstreams are manipulated to produce a new coded bitstream. Conforming edited bitstreams must meet the requirements defined in parts 1, 2, and 3 of ISO/IEC 13818.

2.1.80 Elementary Stream Clock Reference; ESCR [systems]: A time stamp in the PES Stream from which decoders of PES streams may derive timing.

2.1.81 elementary stream; ES [systems]: A generic term for one of the coded video, coded audio or other coded bitstreams in PES packets. One elementary stream is carried in a sequence of PES packets with one and only one stream_id.

2.1.82 emphasis [audio]: Filtering applied to an audio signal before storage or transmission to improve the signal-to-noise ratio at high frequencies.

2.1.83 encoder: An embodiment of an encoding process.

2.1.84 encoding (process): A process, not specified in ISO/IEC 13818, that reads a stream of input pictures or audio samples and produces a valid coded bitstream as defined in parts 1, 2, and 3 of ISO/IEC 13818.

2.1.85 enhancement layer [video]: A relative reference to a layer (above the base layer) in a scalable hierarchy. For all forms of scalability, its decoding process can be described by reference to the lower layer decoding process and the appropriate additional decoding process for the enhancement layer itself.

2.1.86 entitlement control message; ECM [systems]: Entitlement Control Messages are private conditional access information which specify control words and possibly other, typically stream-specific, scrambling and/or control parameters.

2.1.87 entitlement management message; EMM [systems]: Entitlement Management Messages are private conditional access information which specify the authorisation levels or the services of specific decoders. They may be addressed to single decoders or groups of decoders.

2.1.88 entropy coding: Variable length lossless coding of the digital representation of a signal to reduce redundancy.

2.1.89 event [systems]: An event is defined as a collection of elementary streams with a common time base, an associated start time, and an associated end time.

2.1.90 evil bitstreams: Bitstreams orthogonal to reality.

2.1.91 extension bitstream [audio]: Information contained in an optional additional bit stream related to the audio base bit stream at the system level, to support bit rates beyond those defined in ISO/IEC 11172-3. The optional extension bit stream contains the remainder of the multichannel and multilingual data.

2.1.92 fast reverse playback [video]: The process of displaying the picture sequence in the reverse of display order faster than real-time.

2.1.93 fast forward playback [video]: The process of displaying a sequence, or parts of a sequence, of pictures in display-order faster than real-time.

2.1.94 FFT: Fast Fourier Transformation. A fast algorithm for performing a discrete Fourier transform (an orthogonal transform).

2.1.95 field [video]: For an interlaced video signal, a “field” is the assembly of alternate lines of a frame. Therefore an interlaced frame is composed of two fields, a top field and a bottom field.

2.1.96 field period [video]: The reciprocal of twice the frame rate.

2.1.97 field picture; field structure picture [video]: A field structure picture is a coded picture with picture_structure is equal to "Top field" or "Bottom field".

2.1.98 field-based prediction [video]: A prediction mode using only one field of the reference frame. The predicted block size is 16x16 luminance samples. Field-based prediction is not used in progressive frames.

2.1.99 filterbank [audio]: A set of band-pass filters covering the entire audio frequency range.

2.1.100 fixed segmentation [audio]: A subdivision of the digital representation of an audio signal into fixed segments of time.

2.1.101 flag: A variable which can take one of only the two values defined in this specification.

2.1.102 FLC: Fixed Length Code.

2.1.103 forbidden: The term "forbidden", when used in the clauses defining the coded bitstream, indicates that the value shall never be used. This is usually to avoid emulation of start codes.

2.1.104 forced updating [video]: The process by which macroblocks are intra-coded from time-to-time to ensure that mismatch errors between the inverse DCT processes in encoders and decoders cannot build up excessively.

2.1.105 forward compatibility: A newer coding standard is forward compatible with an older coding standard if decoders designed to operate with the newer coding standard are able to decode bitstreams of the older coding standard.

2.1.106 forward motion vector [video]: A motion vector that is used for motion compensation from a reference frame or reference field at an earlier time in display order.

2.1.107 forward prediction [video]: Prediction from the past reference frame (field).

2.1.108 frame [audio]: A part of the audio bit stream that corresponds to audio PCM samples from an Audio Access Unit.

2.1.109 frame [video]: A frame contains lines of spatial information of a video signal. For progressive video, these lines contain samples starting from one time instant and continuing through successive

lines to the bottom of the frame. For interlaced video a frame consists of two fields, a top field and a bottom field. One of these fields may be temporally located one field period later than the other.

2.1.110 frame period [video]: The reciprocal of the frame rate.

2.1.111 frame picture; frame structure picture [video]: A frame structure picture is a coded picture with picture_structure is equal to "Frame".

2.1.112 frame rate [video]: The rate at which frames are be output from the decoding process.

2.1.113 frame reordering [video]: The process of reordering the reconstructed frames when the coded order is different from the display order. Frame reordering occurs when B-frames are present in a bitstream. There is no frame reordering when decoding low delay bitstreams.

2.1.114 frame-based prediction [video]: A prediction mode using both fields of the reference frame.

2.1.115 free format [audio]: Any bitrate other than the defined bitrates that is less than the maximum valid bitrate for each layer.

2.1.116 future reference frame (field) [video]: A future reference frame(field) is a reference frame(field) that occurs at a later time than the current picture in display order.

2.1.117 granules [Layer II] [audio]: The set of 3 consecutive subband samples from all 32 subbands that are considered together before quantisation. They correspond to 96 PCM samples.

2.1.118 granules [Layer III] [audio]: 576 frequency lines that carry their own side information.

2.1.119 group of pictures [video]: A notion defined only in ISO/IEC 11172-2 (MPEG-1 Video). In ISO/IEC 13818-2, a similar functionality can be achieved by the mean of inserting group of pictures headers.

2.1.120 Hann window [audio]: A time function applied sample-by-sample to a block of audio samples before Fourier transformation.

2.1.121 header: A block of data in the coded bitstream containing the coded representation of a number of data elements pertaining to the coded data that follow the header in the bitstream.

2.1.122 Huffman coding: A specific method for entropy coding.

2.1.123 hybrid filterbank [audio]: A serial combination of subband filterbank and MDCT.

2.1.124 hybrid scalability [video]: Hybrid scalability is the combination of two (or more) types of scalability.

2.1.125 I-field picture [video]: A field structure I-Picture.

2.1.126 I-frame picture [video]: A frame structure I-Picture.

2.1.127 I-picture; intra-coded picture [video]: A picture coded using information only from itself.

2.1.128 IDCT: Inverse Discrete Cosine Transform.

2.1.129 IMDCT [audio]: Inverse Modified Discrete Cosine Transform.

2.1.130 intensity stereo [audio]: A method of exploiting stereo irrelevance or redundancy in stereophonic audio programmes based on retaining at high frequencies only the energy envelope of the right and left channels.

2.1.131 interlace [video]: The property of conventional television frames where alternating lines of the frame represent different instances in time. In an interlaced frame, one of the field is meant to be displayed first. This field is called the first field. The first field can be the top field or the bottom field of the frame.

2.1.132 intra coding [video]: Coding of a macroblock or picture that uses information only from that macroblock or picture.

2.1.133 ITU-T Rec. H.222.0 | ISO/IEC 13818 (multiplexed) stream [systems]: A bitstream composed of 0 or more elementary streams combined in the manner defined in ITU-T Rec. H.222.0 | ISO/IEC 13818-1.

2.1.134 joint stereo coding [audio]: Any method that exploits stereophonic irrelevance or stereophonic redundancy.

2.1.135 joint stereo mode [audio]: A mode of the audio coding algorithm using joint stereo coding.

2.1.136 layer [audio]: One of the levels in the coding hierarchy of the audio system defined in ISO/IEC 13818-3.

2.1.137 layer [systems]: One of the levels in the data hierarchy of the video and system specifications defined in ISO/IEC 13818 parts 1 and 2.

2.1.138 layer [video]: In a scalable hierarchy denotes one out of the ordered set of bitstreams and (the result of) its associated decoding process (implicitly including decoding of **all** layers below this layer).

2.1.139 layer bitstream [video]: A single bitstream associated to a specific layer (always used in conjunction with layer qualifiers, e. g. "enhancement layer bitstream").

2.1.140 level [video]: A defined set of constraints on the values which may be taken by the parameters of this specification within a particular profile. A profile may contain one or more levels. In a different context, level is the absolute value of a non-zero coefficient (see “run”).

2.1.141 LFE [audio]: Low Frequency Enhancement channel. A limited bandwidth channel for low frequency audio effects in a multichannel system.

2.1.142 low frequency enhancement channel [audio]: A limited bandwidth channel for low frequency audio effects in a multichannel system.

2.1.143 lower layer [video]: A relative reference to the layer immediately below a given enhancement layer (implicitly including decoding of **all** layers below this enhancement layer).

2.1.144 luminance component [video]: A matrix, block or single sample representing a monochrome representation of the signal and related to the primary colours in the manner defined in the bitstream. The symbol used for luminance is Y.

2.1.145 macroblock [video]: The four 8 by 8 blocks of luminance data and the two (for 4:2:0 chrominance format), four (for 4:2:2 chrominance format) or eight (for 4:4:4 chrominance format) corresponding 8 by 8 blocks of chrominance data coming from a 16 by 16 section of the luminance component of the picture. Macroblock is sometimes used to refer to the sample data and sometimes to the coded representation of the sample values and other data elements defined in the macroblock header of the syntax defined in this part of this specification. The usage is clear from the context.

2.1.146 mapping [audio]: Conversion of an audio signal from time to frequency domain by subband filtering and/or by MDCT.

2.1.147 masking [audio]: A property of the human auditory system by which an audio signal cannot be perceived in the presence of another audio signal.

2.1.148 masking threshold [audio]: A function in frequency and time below which an audio signal cannot be perceived by the human auditory system.

2.1.149 Mbit [video]: 1 000 000 bits.

2.1.150 MCP [video]: Motion Compensated Predictor.

2.1.151 MDCT [audio]: Modified Discrete Cosine Transform which corresponds to the Time Domain Aliasing Cancellation Filter Bank.

2.1.152 mismatch [video]: Numerical discrepancy between the data reconstructed from the same coded bitstream by two decoding processes. With the exception of IDCT, the specification of ISO/IEC 13818-2 defines the decoding process absolutely unambiguously. Therefore, if both decoding processes are implemented according the specifications ISO/IEC 13818-2, mismatch can only be caused by different implementations of IDCT.

2.1.153 motion compensation [video]: The use of motion vectors to improve the efficiency of the prediction of sample values. The prediction uses motion vectors to provide offsets into the past and/or future reference frames or reference fields containing previously decoded sample values that are used to form the prediction error.

2.1.154 motion estimation [video]: The process of estimating motion vectors during the encoding process.

2.1.155 motion vector [video]: A two-dimensional vector used for motion compensation that provides an offset from the coordinate position in the current picture or field to the coordinates in a reference frame or reference field.

2.1.156 MS stereo [audio]: A method of exploiting stereo irrelevance or redundancy in stereophonic audio programmes based on coding the sum and difference signal instead of the left and right channels.

2.1.157 multichannel [audio]: A combination of audio channels used to create a spatial sound field.

2.1.158 multilingual [audio]: A presentation of dialogue in more than one language.

2.1.159 NIT [systems]: Network Information Table as defined in table 2-23 of ISO/IEC 13818-1.

2.1.160 non-intra coding [video]: Coding of a macroblock or picture that uses information both from itself and from macroblocks and pictures occurring at other times.

2.1.161 non-tonal component [audio]: A noise-like component of an audio signal.

2.1.162 Nyquist sampling: Sampling at or above twice the maximum bandwidth of a signal.

2.1.163 opposite parity [video]: The opposite parity of top is bottom, and vice versa.

2.1.164 P-field picture [video]: A field structure P-Picture.

2.1.165 P-frame picture [video]: A frame structure P-Picture.

2.1.166 P-picture; predictive-coded picture [video]: A picture that is coded using motion compensated prediction from past reference fields or frame.

2.1.167 pack [systems]: A pack consists of a pack header followed by zero or more packets. It is a layer in the system coding syntax described in 2.5.3.3 on page 51 of ISO/IEC 13818-1.

2.1.168 packet [systems]: A packet consists of a header followed by a number of contiguous bytes from an elementary data stream. It is a layer in the system coding syntax described in 2.4.3 of ISO/IEC 13818-1.

2.1.169 packet data [systems]: Contiguous bytes of data from an elementary stream present in a packet.

2.1.170 packet identifier; PID [systems]: A unique integer value used to associate elementary streams of a program in a single or multi-program Transport Stream as described in 2.4.3 of ISO/IEC 13818-1.

2.1.171 padding [audio]: A method to adjust the average length of an audio frame in time to the duration of the corresponding PCM samples, by conditionally adding a slot to the audio frame.

2.1.172 parameter: A variable within the syntax of this specification which may take one of a range of values. A variable which can take one of only two values is a flag or indicator and not a parameter.

2.1.173 parity (of field) [video]: The parity of a field can be top or bottom.

2.1.174 parser: Functional stage of a decoder which extracts from a coded bitstream series of bits representing coded elements (FLC or VLC).

2.1.175 past reference frame (field) [video]: A past reference frame(field) is a reference frame(field) that occurs at an earlier time than the current picture in display order.

2.1.176 PAT [systems]: Program Association Table as defined in clause 2.4.4.3 of ISO/IEC 13818-1.

2.1.177 payload [systems]: Payload refers to the bytes which follow the header bytes in a packet. For example, the payload of a Transport Stream packet includes the PES_packet_header and its PES_packet_data_bytes, or pointer_field and PSI sections, or private data; but a PES_packet_payload consists of only PES_packet_data_bytes. The Transport Stream packet header and adaptation fields are not payload.

2.1.178 PES [systems]: An abbreviation for Packetized Elementary Stream.

2.1.179 PES packet [systems]: The data structure used to carry elementary stream data. It consists of a PES packet header followed by PES packet payload and is described in 2.4.3.6 and 2.4.3.7 of ISO/IEC 13818-1.

2.1.180 PES packet header[systems]: The leading fields in a PES packet up to and not including the PES_packet_data_byte fields, where the stream is not a padding stream. In the case of a padding stream the PES packet header is similarly defined as the leading fields in a PES packet up to and not including padding_byte fields.

2.1.181 PES Stream [systems]: A PES Stream consists of PES packets, all of whose payloads consist of data from a single elementary stream, and all of which have the same stream_id. Specific semantic constraints apply.

2.1.182 picture [video]: Source, coded or reconstructed image data. A source or reconstructed picture consists of three rectangular matrices of 8-bit numbers representing the luminance and two chrominance signals. A “coded picture” is defined in ISO/IEC 13818-2. For progressive video, a picture is identical to a frame, while for interlaced video, a picture can refer to a frame, or the top field or the bottom field of the frame depending on the context.

2.1.183 picture data [video]: In the VBV operations, picture data is defined as all the bits of the coded picture, all the header(s) and user data immediately preceding it if any (including any stuffing between them) and all the stuffing following it, up to (but not including) the next start code, except in the case where the next start code is an end of sequence code, in which case it is included in the picture data.

2.1.184 polyphase filterbank [audio]: A set of equal bandwidth filters with special phase interrelationships, allowing for an efficient implementation of the filterbank.

2.1.185 prediction [audio]: The use of a predictor to provide an estimate of the subband sample in one channel from the subband samples in other channels.

2.1.186 prediction error: The difference between the actual value of a sample or data element and its predictor.

2.1.187 prediction: The use of a predictor to provide an estimate of the sample value or data element currently being decoded.

2.1.188 predictor: A linear combination of previously decoded sample values or data elements.

2.1.189 presentation channel [audio]: audio channels at the output of the decoder corresponding to the loudspeaker positions left, centre, right, left surround and right surround.

2.1.190 presentation time-stamp; PTS [systems]: A field that may be present in a PES packet header that indicates the time that a presentation unit is presented in the system target decoder.

2.1.191 presentation unit; PU [systems]: A decoded Audio Access Unit or a decoded picture.

2.1.192 profile [video]: A defined subset of the syntax of this specification.

2.1.193 profile-and-level combination [video]: Point of conformance for video bitstreams and decoders. Defined profile-and-level combinations are defined in Chapter 8 of ISO/IEC 13818-2. In the case of a bitstream, the profile-and-level combination is derived from the profile_and_level_indication. A decoder may comply with several profile-and level combinations.

2.1.194 program [systems]: A program is a collection of program elements. Program elements may be elementary streams. Program elements need not have any defined time base; those that do, have a common time base and are intended for synchronised presentation.

2.1.195 Program Clock Reference; PCR [systems]: A time stamp in the Transport Stream from which decoder timing is derived.

2.1.196 program element[systems]: A generic term for one of the elementary streams or other data streams that may be included in a program.

2.1.197 Program Specific Information; PSI [systems]: PSI consists of normative data which is necessary for the demultiplexing of Transport Streams and the successful regeneration of programs

and is described in 2.4.4 of ISO/IEC 13818-1. One case of PSI, the non-mandatory network information table, is privately defined.

2.1.198 progressive [video]: The property of film frames where all the samples of the frame represent the same instances in time.

2.1.199 psychoacoustic model [audio]: A mathematical model of the masking behaviour of the human auditory system.

2.1.200 quantisation matrix [video]: A set of sixty-four 8-bit values used by the dequantiser.

2.1.201 quantised DCT coefficients [video]: DCT coefficients before dequantisation. A variable length coded representation of quantised DCT coefficients is transmitted as part of the coded video bitstream.

2.1.202 quantiser scale [video]: A scale factor coded in the bitstream and used by the decoding process to scale the dequantisation.

2.1.203 random access: The process of beginning to read and decode the coded bitstream at an arbitrary point.

2.1.204 reconstructed frame [video]: A reconstructed frame consists of three rectangular matrices of 8-bit numbers representing the luminance and two chrominance signals. A reconstructed frame is obtained by decoding a coded frame.

2.1.205 reconstructed picture [video]: A reconstructed picture is obtained by decoding a coded picture. A reconstructed picture is either a reconstructed frame (when decoding a frame picture), or one field of a reconstructed frame (when decoding a field picture). If the coded picture is a field picture, then the reconstructed picture is the top field or the bottom field of the reconstructed frame.

2.1.206 reference decoder [video]: A decoder that implements precisely the decoding process as specified in ISO/IEC 13818-2 and uses a reference IDCT. The reference decoder is capable of decoding compliant bitstreams of any defined profile-and-level.

2.1.207 reference IDCT [video]: Embodiment of the saturated mathematical integer-number IDCT specified in Annex A of ISO/IEC 13818-2.

2.1.208 reference field [video]: A reference field is one field of a reconstructed frame. Reference fields are used for forward and backward prediction when P-pictures and B-pictures are decoded. Note that when field P-pictures are decoded, prediction of the second field P-picture of a coded frame uses the first reconstructed field of the same coded frame as a reference field.

2.1.209 reference frame [video]: A reference frame is a reconstructed frame that was coded in the form of a coded I-frame or a coded P-frame. Reference frames are used for forward and backward prediction when P-pictures and B-pictures are decoded.

2.1.210 reordering delay [video]: A delay in the decoding process that is caused by frame reordering.

2.1.211 requantisation [audio]: Decoding of coded subband samples in order to recover the original quantised values.

2.1.212 reserved: The term "reserved" when used in the clauses defining the coded bitstream indicates that the value may be used in the future for ISO/IEC defined extensions.

2.1.213 run [video]: The number of zero coefficients preceding a non-zero coefficient, in the scan order. The absolute value of the non-zero coefficient is called “level”.

2.1.214 sample aspect ratio [video]: (abbreviated to **SAR**). This specifies the distance between samples. It is defined (for the purposes of this specification) as the vertical displacement of the lines of luminance samples in a frame divided by the horizontal displacement of the luminance samples. Thus its units are (metres per line) ÷ (metres per sample).

2.1.215 saturation [video]: Limiting a value that exceeds a defined range by setting its value to the maximum or minimum of the range as appropriate.

2.1.216 scalability [video]: Scalability is the ability of a decoder to decode an ordered set of bitstreams to produce a reconstructed sequence. Moreover, useful video is output when subsets are

decoded. The minimum subset that can thus be decoded is the first bitstream in the set which is called the base layer. Each of the other bitstreams in the set is called an enhancement layer. When addressing a specific enhancement layer, "lower layer" refer to the bitstream which precedes the enhancement layer.

2.1.217 scalable hierarchy [video]: coded video data consisting of an ordered set of more than one video bitstream.

2.1.218 scalefactor [audio]: Factor by which a set of values is scaled before quantisation.

2.1.219 scalefactor band [audio]: A set of frequency lines in Layer III which are scaled by one scalefactor.

2.1.220 scalefactor index [audio]: A numerical code for a scalefactor.

2.1.221 scrambling [systems]: The alteration of the characteristics of a video, audio or coded data stream in order to prevent unauthorised reception of the information in a clear form. This alteration is a specified process under the control of a conditional access system.

2.1.222 side information: Information in the bitstream necessary for controlling the decoder.

2.1.223 skipped macroblock [video]: A macroblock for which no data is encoded.

2.1.224 slice [video]: A series of consecutive macroblocks that start, in the coded bitstream, by a slice_start_code and that continue up to the next start code (or to the first stuffing bytes if the next start_code is preceded by stuffing bytes)..

2.1.225 slot [audio]: A slot is an elementary part in the audio bit stream. In Layer I a slot equals four bytes, in Layers II and III one byte.

2.1.226 SNR scalability [video]: A type of scalability where the enhancement layer (s) contain only coded refinement data for the DCT coefficients of the lower layer.

2.1.227 source stream: A single non-multiplexed stream of samples before compression coding.

2.1.228 source; input [video]: Term used to describe the video material or some of its attributes before encoding.

2.1.229 spatial prediction [video]: prediction derived from a decoded frame of the lower layer decoder used in spatial scalability.

2.1.230 spatial scalability [video]: A type of scalability where an enhancement layer also uses predictions from sample data derived from a lower layer without using motion vectors. The layers can have different frame sizes, frame rates or chrominance formats.

2.1.231 splicing[systems]: The concatenation, performed on the system level, of two different elementary streams. The resulting system stream conforms totally to ISO/IEC 13818-1. The splice may result in discontinuities in timebase, continuity counter, PSI, and decoding.

2.1.232 spreading function [audio]: A function that describes the frequency spread of masking effects.

2.1.233 start codes [systems]: 32-bit codes embedded in the coded bitstream that are unique. They are used for several purposes including identifying some of the layers in the coding syntax.

2.1.234 STD input buffer [systems]: A first-in first-out buffer at the input of a system target decoder for storage of compressed data from elementary streams before decoding.

2.1.235 stereo mode [audio]: Mode, where two audio channels which form a stereo pair (left and right) are encoded within one bitstream. The coding process is the same as for the dual channel mode.

2.1.236 stereo-irrelevant [audio]: a portion of a stereophonic audio signal which does not contribute to spatial perception.

2.1.237 still picture [systems]: A coded still picture consists of a video sequence containing exactly one coded picture which is intra-coded. This picture has an associated PTS and the presentation time of succeeding pictures, if any, is later than that of the still picture by at least two picture periods.

2.1.238 stuffing (bits); stuffing (bytes): Code-words that may be inserted at particular locations in the coded bitstream that are discarded in the decoding process. Their purpose is to increase the bitrate of the stream which would otherwise be lower than the desired bitrate.

2.1.239 subband [audio]: Subdivision of the audio frequency band.

2.1.240 subband filterbank [audio]: A set of band filters covering the entire audio frequency range. In ISO/IEC 13818-3, the subband filterbank is a polyphase filterbank.

2.1.241 subband samples [audio]: The subband filterbank within the audio encoder creates a filtered and subsampled representation of the input audio samples. The filtered samples are called subband samples. From 32 time-consecutive input audio samples, one subband sample is generated within each of the 32 subbands.

2.1.242 surround channel [audio]: An audio presentation channel added to the front channels (L and R or L, R, and C) to enhance the spatial perception.

2.1.243 syncword [audio]: A 12-bit code embedded in the audio bit stream that identifies the start of a base frame or an extension frame.

2.1.244 synthesis filterbank [audio]: Filterbank in the decoder that reconstructs a PCM audio signal from subband samples.

2.1.245 System Clock Reference; SCR [systems]: A time stamp in the Program Stream from which decoder timing is derived.

2.1.246 system header [systems]: The system header is a data structure defined in subclause 2.5.3.5 of ISO/IEC 13818-1, that carries information summarising the system characteristics of the ITU-T Rec. H.222.0 | ISO/IEC 13818 multiplexed Program Stream.

2.1.247 system target decoder; STD [systems]: A hypothetical reference model of a decoding process used to describe the semantics of an ITU-T Rec. H.222.0 | ISO/IEC 13818 multiplexed bitstream.

2.1.248 temporal prediction [video]: prediction derived from reference frames or fields other than those defined as spatial prediction.

2.1.249 temporal scalability [video]: A type of scalability where an enhancement layer also uses predictions from sample data derived from a lower layer using motion vectors. The layers have identical frame size, and chrominance formats, but can have different frame rates.

2.1.250 time-stamp [systems]: A term that indicates the time of a specific action such as the arrival of a byte or the presentation of a Presentation Unit.

2.1.251 tonal component [audio]: A sinusoid-like component of an audio signal.

2.1.252 top field [video]: One of two fields that comprise a frame. Each line of a top field is spatially located immediately above the corresponding line of the bottom field.

2.1.253 top layer [video]: the topmost layer (with the highest layer_id) of a scalable hierarchy.

2.1.254 Transport Stream packet header [systems]: The leading fields in a Transport Stream packet, up to and including the continuity_counter field.

2.1.255 triplet [audio]: A set of 3 consecutive subband samples from one subband. A triplet from each of the 32 subbands forms a granule.

2.1.256 variable bitrate: Operation where the bitrate varies with time during the decoding of a coded bitstream.

2.1.257 variable length coding: A reversible procedure for coding that assigns shorter code-words to frequent events and longer code-words to less frequent events.

2.1.258 VLC: Variable Length Code.

2.1.259 VLD: Variable Length Decoder.

2.1.260 video buffering verifier; VBV [video]: A hypothetical decoder that is conceptually connected to the output of the encoder. Its purpose is to provide a constraint on the variability of the data rate that an encoder or editing process may produce.

2.1.261 video sequence [video]: The highest syntactic structure of coded video bitstreams. It contains a series of one or more coded frames.

2.1.262 xxx profile bitstream [video]: a bitstream of a scalable hierarchy with a profile indication corresponding to xxx. Note that this bitstream is only decodable together with all its lower layer bitstreams (unless it is a base layer bitstream).

2.1.263 xxx profile decoder [video]: decoder able to decode one or a scalable hierarchy of bitstreams of which the top layer conforms to the specifications of the xxx profile (with xxx being any of the defined Profile names).

2.1.264 xxx profile scalable hierarchy [video]: set of bitstreams of which the top layer conforms to the specifications of the xxx profile.

2.1.265 zig-zag scanning order [video]: A specific sequential ordering of the DCT coefficients from (approximately) the lowest spatial frequency to the highest.

2.2 Abbreviations and symbols

The mathematical operators used to describe this part of ISO/IEC 13818 are similar to those used in the C programming language. However, integer divisions with truncation and rounding are specifically defined. Numbering and counting loops generally begin from zero.

2.2.1 Arithmetic operators

+	Addition.
-	Subtraction (as a binary operator) or negation (as a unary operator).
++	Increment.
- -	Decrement.
×	Multiplication.
*	Multiplication.
^	Power.
/	Integer division with truncation of the result toward zero. For example, 7/4 and -7/-4 are truncated to 1 and -7/4 and 7/-4 are truncated to -1.
//	Integer division with rounding to the nearest integer. Half-integer values are rounded away from zero unless otherwise specified. For example 3//2 is rounded to 2, and -3//2 is rounded to -2.
DIV	Integer division with truncation of the result toward minus infinity. For example 3 DIV 2 is rounded to 1, and -3 DIV 2 is rounded to -2.
| |	Absolute value. | x | = x when x > 0 | x | = 0 when x == 0 | x | = −x when x < 0
÷	Used to denote division in mathematical equations where no truncation or rounding is intended.
%	Modulus operator. Defined only for positive numbers.
Sign( )	Sign(x) = 1 x > 0 0 x == 0 -1 x < 0

NINT ( ) Nearest integer operator. Returns the nearest integer value to the real-valued argument. Half-integer values are rounded away from 0.

sin Sine.

cos Cosine.

exp Exponential.

√ Square root.

$\log_{10}$ Logarithm to base ten.

$\log_e$ Logarithm to base e.

$\log_2$ Logarithm to base 2.

Abs( ) Absolute value.
Abs(x) = x when x > 0
Abs(x) = 0 when x == 0
Abs(x) = –x when x < 0

$\sum_{i=a}^{i<b} f(i)$ The summation of the *f*(*i*) with *i* taking integral values from *a* up to, but not including *b*.

2.2.2 Logical operators

|| Logical OR.

&& Logical AND.

! Logical NOT.

2.2.3 Relational operators

> Greater than.

>= Greater than or equal to.

≥ Greater than or equal to.

< Less than.

<= Less than or equal to.

≤ Less than or equal to.

== Equal to.

!= Not equal to.

max [,...,] the maximum value in the argument list.

min [,...,] the minimum value in the argument list.

2.2.4 Bitwise operators

& AND

| OR

>> Shift right with sign extension.

<< Shift left with zero fill.

2.2.5 Assignment

= Assignment operator.

2.2.6 Mnemonics

The following mnemonics are defined to describe the different data types used in the coded bitstream.

bslbf	Bit string, left bit first, where "left" refers to the order in which bit strings are written in ISO/IEC 13818. Bit strings are written as a string of 1s and 0s within single quote marks, e.g. '1000 0001'. Bit strings are also written in hexadecimal notation, e.g. 0x4F. Blanks within a bit string are for ease of reading and have no significance.
center_chan	Index of center channel. (Audio)
center_limited	Variable which indicates whether a subband of the center is not transmitted. It is used in the case of phantom coding of center channel. (Audio)
ch	Channel. If ch has the value 0, the left channel of a stereo signal or the first of two independent signals is indicated. (Audio)
dyn_cross	dyn_cross means that dynamic crosstalk is used for a certain transmission channel and a certain subband. (Audio)
gr	Granule of 3 * 32 subband samples in audio Layer II, 18 * 32 subband samples in audio Layer III. (Audio)
Lo, Ro	Compatible stereo audio signals. (Audio)
L, C, R, LS, RS	Left, center, right, left surround and right surround audio signals. (audio)
L^W, C^W, R^W, LS^W, RS^W	Weighted left, center, right, left surround and right surround audio signals. The weighting is necessary for two reasons: 1) All signals have to be attenuated prior to encoding to avoid overload when calculating the compatible stereo signal. 2)The weighted and processed signals are actually coded and transmitted, and denormalised in the decoder. (Audio)
left_sur_chan	Index of left surround channel. (Audio)
main_data	The main_data portion of the bitstream contains the scale factors, Huffman encoded data, and ancillary information. (Audio)
main_data_beg	This gives the location in the bitstream of the beginning of the main_data for the frame. The location is equal to the ending location of the previous frame's main_data plus 1 bit. It is calculated from the main_data_end value of the previous frame. (Audio)
mono_sur_chan	Index of the mono surround channel. This index is identical to the index of the left surround channel. (Audio)
msblimit	Maximum used subband. (Audio)
nch	Number of channels; equal to 1 for single_channel mode, 2 in other modes. (Audio)
nmch	Number of channels in the multichannel extension part. (Audio)
npred	Number of allowed predictors according to the tables in subclause 2.5.2.10 of ISO/IEC 13818-3. (Audio)
npredcoeff	Number of prediction coefficients used. (Audio)
part2_length	The number of main_data bits used for scale factors. (Audio)
pci	index of predictor[0, 1, 2] (Audio)
px	index of predictor[0, 1, ..., npred-1] (Audio)
right_sur_chan	Index of right surround channel. (Audio)
rpchof	Remainder polynomial coefficients, highest order first. (Audio)

sb	Subband. (Audio)
sbgr	Groups of individual subband according to subbandgroup table in subclause 3.5.2.10. (Audio)
sblimit	The number of the lowest subband for which no bits are allocated. (Audio)
scfsi	Scalefactor selection information. (Audio)
simsbf	Signed integer, in two’s complement format, most significant (sign) bit first. (Audio)
switch_point_l	Number of scalefactor band (long block scalefactor band) from which point on window switching is used. (Audio)
switch_point_s	Number of scalefactor band (short block scalefactor band) from which point on window switching is used. (Audio)
T0, T1, T2, T3, T4	Audio transmission channels. The assignment of audio signals to transmission channels is determined by the dematrixing procedure and the transmission channel allocation information. (Audio)
tc	Transmitted channel. (Audio)
tcimsbf	two's complement integer, msb (sign) bit first. (Systems)
uimsbf	Unsigned integer, most significant bit first.
vlclbf	Variable length code, left bit first, where "left" refers to the order in which the VLC codes are written.
window	Number of the actual time slot in case of block_type==2, 0 <= window <= 2. (Audio)

The byte order of multi-byte words is most significant byte first.

2.2.7 Constants

π	3,14159265358...
e	2,71828182845...

2.3 Systems

2.3.1 System bitstream characteristics

2.3.1.1 General system bitstream characteristics

Transport Stream and Program Stream encoders may apply restrictions to the following parameters of Transport and Program Streams (see ITU-T Rec. H.222.0 | ISO/IEC 13818-1) :

- constraints for STD Model
- use of smoothing buffer parameters
- delay caused by system target decoder input buffering
- length of PES packets
- presence of time stamps in PES packet headers (DTS, PTS)
- frequency of encoding time stamps in PES packet headers (DTS, PTS)
- use of private streams
- fixed or variable bitrate operation
- number of multiplexed audio streams in a program
- number of multiplexed video streams in a program
- decoding of trick modes
- use of PES Packet CRCs
- carriage of Program Stream data
- carriage of private data
- use of descriptors
- constraints for sending descriptors

2.3.1.2 Transport Stream specific characteristics

System encoders that produce Transport Streams may apply restrictions to the following parameters of Transport Streams (see ITU-T Rec. H.222.0 | ISO/IEC 13818-1) :

- use of transport priority indication
- application of scrambling
- frequency of opening an adaptation header field
- frequency of flagging random access points using the random_access_indicator
- use of elementary_stream_priority_indicator
- frequency of encoding PCRs
- use of option to encode OPCRs
- occurrence of splice points
- carriage of private data
- use of LTW concept
- support of seamless splicing
- encoding of a valid value in the PES_packet_length field in the case of a PES packet carrying MPEG-2 video data
- frequency of transmitting PSI tables
- use of Network Information Table
- use of Conditional Access Table
- use of private sections

2.3.1.3 Program Stream specific characteristics

Program Stream encoders may apply restrictions to the following parameters of Transport and Program Streams (see ITU-T Rec. H.222.0 | ISO/IEC 13818-1) :

- program mux_rate
- rate_bound
- P-STD_buffer_size
- P-STD_buffer_size_bound
- difference between two SCRs in successive packs
- length of a pack
- length of a PES packet
- number of PES packets in a pack
- presence of time stamps in packet headers (DTS, PTS)
- CSPS_flag
- use of private streams
- packet rate
- fixed or variable bitrate operation (fixed_flag parameter)
- number of multiplexed audio streams (audio_bound parameter)
- number of multiplexed video streams (video_bound parameter)
- locking of audio sampling frequency and frequency of system clock
- locking of video picture rate and frequency of system clock
- constraints for use of Program Stream Map Table
- constraints for use of Program Stream Directory Table

2.3.2 System bitstream tests

Each bitstream shall meet the syntactic and semantic requirements specified in ISO/IEC 13818-1. This subclause describes a set of semantic tests to be performed on bitstreams. To verify whether the syntax is correct is straight forward and therefore not defined in this subclause. In the description of the semantic tests it is assumed that the tested bitstream contains no errors due to transmission or other causes. For each test the condition or conditions that must be satisfied are given, as well as the prerequisites or conditions in which the test can be applied. Note that the application of these tests requires parsing of the bitstream to the appropriate levels, which in some cases goes as far as the slice layer of video. Parsing and interpretation of PSI is also required. In some cases of scrambled data descrambling is required before the tests can be performed.

2.3.2.1 Tests of Transport Streams

2.3.2.1.1 Tests of transport packet header

The following tests are applicable to Transport Stream packets in which the transport_error_indicator is set to ‘0’.

payload_unit_start_indicator: If the PID is 0x1FFF test that the payload_unit_start_indicator is ‘0’. If a PID is carrying PES packets or PSI sections, and payload_unit_start_indicator is set to 1, then test that adaptation_field_control is ‘01’ or ‘11’.

PID: Test that the PID does not take any of the reserved values described in ISO/IEC 13818-1 table 2-4.

transport_scrambling_control: If the PID is 0x0000 or 0x0001 or 0x1FFF, or if the Transport Stream Packet contains PMT sections, test that the transport_scrambling_control is ‘00’.

adaptation_field_control: If the PID is 0x1FFF, test that adaptation_field_control is ‘01’, else test that adaptation_field_control is one of ‘01’, ‘10’, or ‘11’.

Transport Stream Continuity: Duplicate packets are packets which are identical except for the PCR field, if present. The following tests do not apply to null packets.

When (1) adaptation_field_control is ‘01’ or ‘11’,
and (2) discontinuity_indicator is ‘0’,
and (3) the Transport Stream packet is not a duplicate packet,
then test that the continuity counter is one greater (modulo 16) than the continuity counter of the previous Transport Stream packet of the same PID.

When (1) adaptation_field_control is ‘00’ or ‘10’,
and (2) discontinuity indicator is ‘0’,
or when the Transport Stream packet is a duplicate packet,
then test that the continuity_counter is equal to the continuity counter of the previous Transport Stream packet of the same PID.

duplicate_packet: If the previous Transport Stream packet is a duplicate, test that the present Transport Stream packet is not a duplicate packet.

If the Transport Stream packet is a duplicate, test that the adaptation_field_control is equal to ‘01’ or ‘11’.

2.3.2.1.2 Tests of adaptation field

adaptation_field_length: If the adaptation_field_control is ‘10’, then test that adaptation_field_length is 183. If the adaptation_field_control is ‘11’, then test that adaptation_field_length is in the range 0 to 182, inclusive.

random_access_indicator: if the Transport Stream packet contains PES data and the random_access_indicator is set to '1' and the payload_unit_start_indicator is set to '1', then test that the PES packet starts in the Transport Stream packet.

random_access_PCR: If random_access_indicator is set to ‘1’ and if the PID is a PCR_PID, then test that there is a PCR in the Transport Stream packet.

random_access_PTS: If random_access_indicator is set to ‘1’, and if the corresponding video stream type is 1 or 2 or the corresponding audio stream type is 3 or 4, then test that there is a PTS for the next picture or audio frame.

elementary_stream_priority_indicator: If elementary_stream_priority_indicator is set to ‘1’ and the Transport Stream packet carries video, then test that the Transport Packet payload contains one or more bytes from a slice that contains only an intra-coded macroblock.

PCR_OPCR_flags: If OPCR_flag is ‘1’, then test that the PCR_flag is ‘1’.

transport_private_data: If transport_private_data_flag is ‘1’ then test that transport_private_data_length is greater than 1 and less than the adapation_field_length less all existing fields in the adaptation field as indicated by the various flags and by adaptation_field_extension_length.

legal time window: If the ltw_valid_flag is set to '1', the following test can be applied on one PID at a time.

It can start as soon as relevant values for the ltw_offset field and piecewise_rate field are defined.

The ltw_offset is expressed in units of (300/fs) seconds, where fs is the system clock frequency of the program that this PID belongs to. The piecewise_rate is expressed in bits/second.

The arrival times of each Transport Packet of the chosen PID are continuously plotted.

The arrival time of the Transport Packet enabling the validation of the two values of the ltw_offset field and of the piecewise_rate field is noted.

The legal_time_windows of the incoming Transport Packets of the same PID are, at that moment defined.

The delivery of the incoming Transport Packets at the end of their legal_time_windows shall satisfy the following conditions:

For video: At the time the first byte of the payload of this transport packet enters MB, MB contains less than 184 bytes and no buffer violation occurs in the T-STD. MB is the multiplex buffer as defined in clause 2.4.2.3 of ISO/IEC 13818-1.

For audio: At the time the first byte of the payload of this transport packet enters B, B contains less than BSdec + 1 bytes and no buffer violation occurs in the T-STD. B is the main buffer as defined in clause 2.4.2.3 of ISO/IEC 13818-1.

If the delivery of all the incoming transport packets satisfy the mentioned conditions, the legal_time_window mechanism is successfully implemented.

If the delivery of one transport packet does not satisfy the mentioned conditions, the legal_time_window mechanism is not well implemented.

OPCR: If it is known (outside of examination of the bitstream) that the Transport Stream is the original single program Transport Stream, then test that the OPCR is equal to the PCR.

splice_countdown: If the adaptation_field_control is '11' and the Transport Stream packet is not a duplicate packet and if the splicing_point_flag is '1', then for all consecutive packets with splicing_point_flag set to '1' test that the splice_countdown field decrements with each packet containing payload. At the Transport Stream packet at which the splice_countdown reaches or would reach -1, test that the packet payload starts with the first byte of a PES packet which starts with the first byte of an access point or a sequence end code followed by an access point as defined in the semantic description of splice_countdown in ISO/IEC 13818-1. If any Transport Stream packet contains a splice_countdown, then test that a Transport Stream packet at which the splice_countdown reaches or would reach zero contains the splice_countdown field.

2.3.2.1.3 Discontinuity tests

In order to validate that the discontinuity indicator is properly used in a Transport Stream, a number of factors need to be tested. These include the proper utilisation of the continuity counter, the proper sequence of packets containing discontinuity indicators, the occurrence of elementary stream access points corresponding to both timebase discontinuities and continuity counter discontinuities, the detection of the minimum number of PCRs between timebase discontinuities, and the proper redefinition of PSI information.

2.3.2.1.3.1 Discontinuity indicator test

if the discontinuity indicator is "1" and the PID of the received packet is the designated PCR_PID,
then designate the current state of this PID as the discontinuity_state and test for time base discontinuity (clause 2.3.2.1.3.2).

else if the discontinuity indicator is '1' and the PID of the received packet is not the designated PCR_PID,
then test that the two previous Transport Stream packets of the same PID did not have discontinuity indicator set to '1', and designate the current state of this PID as the discontinuity_state and test for continuity counter discontinuity (clause 2.3.2.1.3.3).

else if the discontinuity indicator is '0', and the current state of this PID is designated as the discontinuity state,
then clear the discontinuity state for the current PID and test for termination of the discontinuity state (clause 2.3.2.1.3.4).

else if there is no adaptation field, and the current state of this PID is designated as the discontinuity state,
then clear the discontinuity state for the current PID and test for termination of the discontinuity state (clause 2.3.2.1.3.4).

else if the discontinuity indicator is '1' and the PID of the received packet contains PSI information,

then designate the current state of this PID as the discontinuity_state and test for version_number discontinuity (clause 2.3.2.1.3.5).

endif

2.3.2.1.3.2 Time base discontinuity test

(Use for PCR_PID packets when the discontinuity indicator is '1'.)

if the adaptation_field_control is '11' and the continuity counter is not one greater (modulo 16) than the continuity counter of the previous Transport Stream packet of the same PID (continuity counter discontinuity), and this packet is not a duplicate packet,
then (1) test that the packet contains a PCR and, (2) test that at least two PCRs have been received from the previous timebase, and (3) test that the first byte of elementary stream data in this packet is the first byte of an elementary stream access point (or alternatively, for video, a sequence_end_code followed by an access point), or (4) if no elementary stream data occurs in this packet, test that the first byte of elementary stream data in a Transport Stream packet of the same PID shall be the first byte of an elementary stream access point (or alternatively, for video, a sequence_end_code followed by an access point).

else if the adaptation_field_control is '10' and the continuity counter is not equal to the continuity counter of the previous Transport Stream packet of the same PID (continuity counter discontinuity),
then (1) test that the packet contains a PCR and, (2) test that at least two PCRs have been received from the previous timebase, and (3) test that the first byte of elementary stream data in a Transport Stream packet of the same PID is the first byte of an elementary stream access point (or alternatively, for video, a sequence_end_code followed by an access point).

else if the adaptation_field_control is '11' and the continuity counter is one greater (modulo 16) than the continuity counter of the previous Transport Stream packet of the same PID,
then test that at least two PCRs have been received from the previous timebase.

else if the adaptation_field_control is '10' and the continuity counter is equal to the continuity counter of the previous Transport Stream packet of the same PID, and this packet is not a duplicate packet,
then test that at least two PCRs have been received from the previous timebase.

endif

2.3.2.1.3.3 Continuity counter discontinuity test

(Use for non PCR_PID packets when the discontinuity indicator is '1'.)

if the adaptation_field_control is '11' and the continuity counter is not one greater (modulo 16) than the continuity counter of the previous Transport Stream packet of the same PID (continuity counter discontinuity) and this packet is not a duplicate packet,
then (1) test that a continuity counter discontinuity did not occur in the previous Transport Stream packet of the same PID, and
(2) test that the first byte of elementary stream data in this packet is the first byte of an elementary stream access point (or alternatively, for video, a sequence_end_code followed by an access point), or
(3) if no elementary stream data occurs in this packet, test that the first byte of elementary stream data in a Transport Stream packet of the same PID is the first byte of an elementary stream access point (or alternatively, for video, a sequence_end_code followed by an access point).

else if the adaptation_field_control is '10' and the continuity counter is not equal to the continuity counter of the previous Transport Stream packet of the same PID (continuity counter discontinuity),
then (1) test that a continuity counter discontinuity did not occur in the previous Transport Stream packet of the same PID, and
(2) test that the first byte of elementary stream data in a Transport Stream packet of the same PID is the first byte of an elementary stream access point (or alternatively, for video, a sequence_end_code followed by an access point).

endif

2.3.2.1.3.4 Discontinuity state termination test

(Use when the current state is designated as discontinuity state and the discontinuity indicator is ‘0’ or when there is no adaptation field.)

if the PID of the received packet is the designated PCR_PID,
then test that the previous packet with the same PID contains a PCR.

endif

2.3.2.1.3.5 Version number discontinuity test

(Use for PSI PIDs when the discontinuity indicator is ‘1’.)

if the section_length is “13” for the Transport Stream_program_map_sections,
then (1) test that the current_next_indicator is ‘1’,
(2) test that no program_descriptors exist in this packet,
(3) test that no elementary streams are described in this packet,
(4) test that following packet contain updated TS_program_map_sections for each affected program,
(5) with the version_number incremented by one and with the value of current_next_indicator set at ‘1’, test that these sections contain a complete program definition.

endif

2.3.2.1.4 Random access tests

2.3.2.1.4.1 Random access test

if the random access indicator is ‘1’ and the PID of the received packet is the designated PCR_PID,
then test for the existence of a PCR field, and
test for start of PES packet.

else if the random access indicator is ‘1’,
then test for start of PES packet.

else if (1) the random access indicator is ‘0’, and
(2) the random access indicator was ‘1’ in a previous Transport Stream packet of the same PID, and
(3) the beginning of a PES packet has not been detected in a packet of the same PID since the occurrence of the packet of the same PID which contained the random access indicator set to ‘1’,
then test for start of PES packet.

endif

2.3.2.1.4.2 Random access PES start test

if the start of a PES packet exists in the payload of the current Transport Stream Packet and the PES Stream type is ‘1’ or ‘2’,
then (1) test for the first byte of a video sequence header in the PES packet payload, or
(2) if no PES packet payload data occurs in this packet, test that the first Transport Stream packet of the same PID which contains PES packet payload data contains the first byte of a video sequence header, and
(3) test that a presentation time stamp is present in the first PES packet which contains the first picture following the sequence header.

else if the start of a PES packet exists in the payload of the current Transport Stream Packet and the PES Stream type is ‘3’ or ‘4’,
then (1) test for the first byte of an audio frame in the PES packet payload, or
(2) if no PES packet payload data occurs in this packet, test that the first Transport Stream

packet of the same PID which contains PES packet payload data contains the first byte of an audio frame, and
(3) test that a presentation timestamp is present in the PES packet which contains the first byte of the audio frame.

endif

2.3.2.1.5 Tests of PES header

stream_id: Test that the stream_id is consistent with the stream_type of the PID as assigned in the PSI.

PES_packet_length: In a Transport Stream, test that the PES_packet_length is not zero in any PES packet that does not contain video. In a Program Stream, test that the value is not zero. If the value is not equal to zero, test that the length addresses the next PES packet startcode prefix.

data_alignment_indicator: If the data_alignment_indicator is set to 1 and if there is a data stream alignment descriptor for the elementary stream, test that the alignment is as indicated by the descriptor as described in ISO/IEC 13818-1 table 2-47 for video streams or table 2-48 for audio streams. If there is no alignment descriptor, the alignment must be as indicated by the alignment_type '01' as given in ISO/IEC 13818-1 table 2-47 for video or table 2-48 for audio.

PTS_DTS_flags: Test that the PTS_DTS_flags is not 01.

PES_header_data_length: Test that the PES_header_data_length value is consistent with the optional data fields of the PES header, such that the indicated length is greater that or equal to the total of the number of optional bytes as indicated by the flags preceding the length field, from PTS_DTS_flags to PES_extension_flag, and including the bytes in PES extensions, if present. Also test that the length is not more than 32 longer than the number of bytes indicated to be present in the optional fields.

Timestamps: For the timestamps that are present, as indicated by PTS_DTS_flags, test for timestamp consistency within the context of the STD. Timestamp consistency means that the difference between any two timestamps of the same elementary stream equals the time that corresponds to the interval between the access units referred to by the timestamps to within ± 1 count. In the case of video picture reordering, this reordering must be taken into account. Note that timestamps consistency cannot be tested across a discontinuity. For I and P pictures in non-low-delay sequences and in the case when there is no decoding discontinuity between access units k and k´ and if the relevant PTS and DTS are present, test that the presentation time $tp_n(k)$ is equal to the decoding time $tp_n(k')$ of the next transmitted I or P picture. Test that there is a PTS at least every 700 msec for each ES.

intra_slice_refresh: If intra_slice_refresh is set to '0' and if the stream type is video, then test that there are no missing macroblocks between coded slices.

rep_cntrl: Test that rep_cntrl is not equal to 0.

previous_PES_packet_CRC: Test that the encoded value is correct.

pack_header_field_flag: If in a Program Stream, test that this bit is '0'.

program_packet_sequence_counter: If in a Program Stream, test that the program_packet_sequence_counter increments with each PES packet.

original_stuff_length: If MPEG1_MPEG2_identifier is 1, test that this value is not greater than 16. Otherwise test that this value is not greater than 32.

P-STD_buffer_scale: If in a Program Stream, then if audio test that the value is '0', else if video test that the value is '1'.

2.3.2.1.6 General tests for sections

section_length: Test that section_length is consistent with the pointer_fields.

version_number; current_next_indicator; section_number; last_section_number: These fields are analysed together. The same rules apply for this as for program_association_sections (clause 2.3.2.1.7)

A section is denoted with the following attributes: S(v, c, s, l) where
v is the value of the version_number field,
c is the value of the current_next_indicator flag,
s is the value of the section_number field of the section, and
l is the value of the last_section_number field.
Test that for every section S(v,'1', s, l) that is currently valid, v+1 is the next version to have the current_next_indicator flag set to 1.

CRC_32: Test that this is correctly calculated such that the field value gives a zero output of the registers in the decoded defined in Annex B of part 1 of ISO/IEC 13818 after processing the entire section, excluding the CRC field itself.

Test that the first byte after the last byte of the CRC_32 field in any Transport Packet with this PID takes either the appropriate table_id value (in which case this is the start of the following section) or the start of the next Transport Stream packet (0x47) or the value 0xFF. If the value of 0xFF is found then test that all following bytes to the end of the Transport Packet are also 0xFF.

If the value 0xFF was present then test that the following Transport Packet of the same PID either
(1) consists only of adaptation_field, or
(2) has the payload_unit_start_indicator set to '1', has the pointer_field set to '0' and then starts a new section.

If no section started in the following Transport Packet, then the same test should be carried out until a Transport Packet is found in which a new section starts (above condition 2).

2.3.2.1.7 Program Association Table tests

PID0: Confirm that in the Transport Stream under test , there is at least one Transport Stream packet with PID set to 0x0000.

transport_scrambling_control: Test that it is set to '00' - i.e. not scrambled.

payload_unit_start_indicator: If this is set to a value of '1' then the first data byte of the Transport Packet shall be understood to be a pointer_field. If the payload_unit_start_indicator is set to '0' then no pointer_field shall be present in that Transport Stream packet and it is not compliant for a section to start in that Transport Stream packet.

table_id: Test that the byte pointed to by the pointer_field takes the value '0x00'.

section_syntax_indicator: Test that this bit is '1'.

section_length: Test that the first two bits are set to '00'.

9 <= section_length <= 1021

if section_length == n then byte n+1 == 0x00 or 0xFF (or 0x47 if start of next Transport Stream packet).

The nth byte of the section shall be tested to be the last byte of the CRC_32.

transport_stream_id: No test on the value but it should be same in all sections of that version number.

Test that all sections of the PAT that are currently valid have the same version_number.

program_number: Test that the program_number field does not take any single value more than once within all sections of any version of the program association table.

If program_number == 0x0000 then

network_PID: Test that this field does not take any value from 0x0000 - 0x000F, or 0x1FFF. Test also that it does not take any values identified as Transport Packets carrying video or audio data. Note that it is permitted for the network_PID to be the same as a program_map_PID. Test that if there are any Transport Stream packets with the PID denoted the network_PID then these packets contain private_sections.

If program_number != 0x0000 then

program_map_PID: Test that this field does not take any value from 0x0000 - 0x000F and also does not take the value 0x1FFF. Test also that it does not take any values identified as Transport Packets carrying video or audio data.

Then test that the PID pointed to contains a TS_program_map_section, with the program_number field coded the same as the program_number field in the PAT.

Test that there are no sections with table_id == 0x00 in any Transport Packet with a PID value other than 0x0000.

If there is a change in the program_number fields of the programs present in the tested Transport Stream then it should be tested that a new version of the PAT is present in the tested Transport Stream prior to the change, giving the new configuration. An example of such a change would be adding a new program, or deleting a program in order to be able to re-use the program_number.

Test that all the sections of the PAT for a given version_number contain a complete list of all the programs present within the tested Transport Stream. This is done by testing that every section in the tested Transport Stream with table_id set to 0x02 are listed through their program_numbers in the PAT. i.e. no sections with table_id set to 0x02 should be found in the tested Transport Stream unless their program number is listed in the PAT.

2.3.2.1.8 Program Map Table tests

Program Map Table PIDs The entries of the program_association_table should be used to obtain the values of PMT PIDs. Each separate PMT PID should then be tested with this procedure to ensure the PSI is compliant. Test that the payload of the packets indicated by the PMT PID consists of sections, including any private sections.

transport_scrambling_control_flags: If table_id == 0x02 then test that the transport_scrambling_control_flags are set to '00', i.e. not scrambled.

payload_unit_start_indicator: If set to a value of '1' in a Transport Stream packet of PMT_PID then the first data byte of the Transport Packet shall be assumed to be a pointer_field. If set to '0' then no pointer_field shall be present in that Transport Stream packet and it is not compliant for a section to start in that Transport Stream packet**.**

table_id: Test that this does not take a value 0x00, 0x01 or 0x03-0x3F. If table_id 0xFF appears then test that stuffing occurs to the end of the Transport Packet. Only sections with table_id value 0x02 are then to be subject to the following tests, as only they are TS_program_map_sections. (Other sections may be private sections).

section_syntax_indicator: Test that this is set to '1'**.**

'0': Test that this bit is '0'.

section_length: Test that the first two bits are coded '00'.

9 <= section_length <= 1021
if section_length == n then byte n+1 == next table_id value, (or 0x47 if new Transport Stream packet)

The nth byte of the section shall be tested to be the last byte of the CRC_32.

program_number: Test that this is coded to be the same as one of the values of program_number listed in the PAT as having the PMT_PID value which is the same as the PID value in which the section is found. Test that this field does not take the value 0x0000, which is reserved for the NIT.

PCR_PID: Test that in the adaptation_fields in the Transport Stream packets having a PID the same as the PCR_PID, a PCR occurs with the required regularity - see the part of this document which defines how to test for correct PCRs.

program_info_length: Test that the first two bits are coded '00'. Test that this length field is consistent with the immediately following descriptor length fields, and that the CRC_32 field is correct. This length only codes the length of the immediately following descriptors.

descriptor: To test each individual descriptor, see 2.3.2.1.12 on page 30, however, at this point test that only descriptors listed as being permitted at 'program level' are included here, or private descriptors (those with tags 64-255).

stream_type: Test that the stream_type field does not take the following values:

0x00, 0x0F-0x7F.

- If the field takes the value 0x01, test that the Transport Stream packets of the PID listed in the next elementary_PID contain ISO/IEC 11172-2 data.
- If the field takes the value 0x02, test that the Transport Stream packets of the PID listed in the next elementary_PID contain ISO/IEC 13818-2 data.
- If the field takes the value 0x03, test that the Transport Stream packets of the PID listed in the next elementary_PID contain ISO/IEC 11172-3 data.
- If the field takes the value 0x04, test that the Transport Stream packets of the PID listed in the next elementary_PID contain ISO/IEC 13818-3 data.
- If the field takes the value 0x05, test that the Transport Stream packets of the PID listed in the next elementary_PID contain ISO/IEC 13818-1 data in private_sections.
- If the field takes the value 0x06, test that the Transport Stream packets of the PID listed in the next elementary_PID contain data in ISO/IEC 13818-1 PES packet form.
- If the field takes the value 0x07, test that the Transport Stream packets of the PID listed in the next elementary_PID contain ISO/IEC 13522 data.
- If the field takes the value 0x08, test that the Transport Stream packets of the PID listed in the next elementary_PID contain ISO/IEC 13818-1 DSM-CC data as defined in Annex A of ISO/IEC 13818-1, or ISO/IEC 13818-6 DSM-CC data.
- If the field takes the value 0x0E, test that the Transport Stream packets of the PID listed in the next elementary_PID contain ISO/IEC 13818-1 auxiliary data.

elementary_PID: Test that this field never takes the values 0x0000- 0x000F or 0x1FFF.

ES_info_length: Test that the first two bits are coded '00'. Test that this length field is consistent with the immediately following descriptor length fields, and that the CRC_32 field is correct.

descriptor: To test each individual descriptor, see 2.3.2.1.12 on page 30, however, at this point test that only descriptors listed as being permitted at 'ES level' are included here, or private descriptors (those with tags 64-255).

2.3.2.1.9 Conditional Access Table tests

PID1: Test that there are no sections with table_id set to 0x01 in any Transport Packet with a PID value other than 0x0001.

transport_scrambling_control: Test that transport_scrambling_control is set to '00', i.e. not scrambled.

payload_unit_start_indicator: If set to a value of '1' in a Transport Stream packet of PID set to 0x0001 then the first data byte of the Transport Packet shall be assumed to be a pointer_field. If the payload_unit_start_indicator is set to '0' then no pointer_field shall be present in that Transport Stream packet and it is not compliant for a section to start in that Transport Stream packet.

table_id: Test that the byte pointed to by the pointer_field takes the value '0x01' i.e. table_id set to 0x01.

section_syntax_indicator: Test that this bit is '1'.

'0': Test that this bit is '0'.

section_length: Test that the first two bits are set to '00'.

$9 <= section\_length <= 1021$

if section_length == n then byte n+1 == 0x01 or 0xFF (or 0x47 in the case of this being the start of the next Transport Stream packet)

The nth byte of the section shall be tested to be the last byte of the CRC_32.

Test that all sections of the CA Table that are currently valid, all have the same version_number.

descriptor: To test each individual descriptor, see 2.3.2.1.12 on page 30, however, at this point test that only descriptors listed as being permitted here are included here,(or private descriptors (those with tags 64-255).

There should be sufficient cases of the CA_descriptor to identify program related access control information for all scrambled streams. Since the content of the descriptor is primarily private, it is not possible to verify whether this private data is 'correct' or not.

If no section started in the following Transport Packet, then the same test should be carried out until a Transport Packet is found with the table_id set to 0x01.

2.3.2.1.10 Network Information Table tests

If in the PAT, a program_number set to 0x0000 occurs, then a test should be made for a network information table.

Test for any Transport Stream packets with a PID value of NIT as indicated in the PAT.

transport_scrambling_control: If the flags are set to a value other than '00' then no further tests are required. Otherwise, the assumption should be made that such packets are in section format.

payload_unit_start_indicator: When set to '1' then the first byte of the payload should be assumed to be a pointer field. This pointer should be used to identify the first byte of a section (table_id)

table_id: Test that it takes a value other than 0x00- 0x3F. If table_id is 0xFF then test that the rest of the Transport Stream packet is also filled with 0xFF bytes too. Otherwise the next test should be carried out on section_length field.

section_length: Assume this is correct. Using this information, skip over the full length of the section. The next byte should not take a value 0x00-0x3F. Continue to repeat the procedure of payload_unit_start_indicator and section_length tests until either a value of 0xFF is found as a first section byte (table_id), or until a section is found to end at the end of a Transport Stream packet.

Note:
If it is known through means outside of the bitstream that the NIT consists only of long private_sections, then it can be tested that the private_sections that make up the NIT conform to the long private_section syntax. If it is not known that the NIT consists only of long private_sections, then only the first part of the private_section (which is common to both the short and long syntax) need be tested for conformance.

2.3.2.1.11 Private section

If Transport Stream packet PIDs are listed in a program definition as containing private data in section form, then the following test should be performed.

payload_unit_start_indicator: If set to '1' in such a Transport Stream packet, then assume that the first byte of the payload of that Transport Stream packet is a pointer field. Use this pointer field to find the first byte of a section.

transport_scrambling_control: If set to anything other than '00', then no further test is necessary. If these sections occur in Transport Stream packets which are listed in the PAT as being PMT_PIDs then the transport_scrambling_control flags shall be set to '00' to be compliant.

table_id: Test that this does not take one of the table_id values 0x00-0x3F. If coded to 0xFF then test that all other bytes to the end of the Transport Stream packet also take the value 0xFF.

Consistency Test: Assume the private_section_length field is correct, and use it to skip to the table_id of the next section. If this is in the following Transport Stream packet test that the

payload_unit_start_indicator is set to '1' and that the pointer field in that Transport Stream packet also points to the start of the next section. Repeat procedure on next section.

2.3.2.1.12 Descriptor tests

2.3.2.1.12.1 General descriptor tests

Certain descriptors can only apply to certain stream types, and may be some only apply to either programs or elementary streams. Some descriptor tags are reserved.

Test that descriptors are not present in PID’s defined as reserved in ISO/IEC 13818-1 table 2-4.

descriptor_tag: Test that the descriptor’s tag does not have any of the reserved values described in ISO/IEC 13818-1 table 2-39.

descriptor_length: The test of descriptor’s length is particular to each descriptor, i.e. that no additional byte (not described in the descriptor’s syntax) can be added. The descriptor’s length shall reflect only the fields described in the descriptor’s syntax.

A global test has also to be performed, verifying that the total length of descriptors described in their associated PSI structures is equal to the sum of the lengths of all the included descriptors plus 2 times the number of descriptors.

Note : The fields which appear in the various descriptors and which are related to fields already present in the video and audio elementary streams structures shall be cross tested with these fields.

2.3.2.1.12.2 Video stream descriptor

This descriptor can be tested only if it applies to a video elementary stream.

The stream_id in the corresponding PES headers must be the stream_id of a video elementary stream (first 4 bits set to ‘1110’, ISO/IEC 13818-1 table 2-18).

In the PMT and PSM tables, this descriptor is allowed, at the elementary level and is not allowed at the program level.

Descriptor_length: Test that the length is 1 or 3.

Multiple_frame_rate_flag and frame_rate_code: When the value of the multiple_frame_rate_flag is zero, only a single frame rate may be assigned to the video elementary stream. That frame rate must correspond to the value indicated by the frame_rate_code following the multiple_frame_rate_flag flag (see first column of ISO/IEC 13818-1, table 2-41).

When the value of the multiple_frame_rate_flag is one, different frame rates may be assigned to the video elementary stream, and these frame rates must correspond to the rates allowed by the value indicated by the frame_rate_code following the multiple_frame_rate_flag (see the two columns of ISO/IEC 13818-1, table 2-41).

These indications have to be cross verified and must be coherent with the different definitions (current and next) of the Video elementary stream itself, as defined with the frame_rate_code field.

MPEG_1_only_flag:
If set to ‘0’:
The descriptor length should be equal to 3.
The stream_type should be 02 according to the ISO/IEC 13818-1, table 2-29.
The video elementary stream should only contain ISO/IEC 13818-2 video or ISO/IEC 11172 constrained parameter video.
The constrained_parameter_flag should be equal to 1.

If set to ‘1’:
The descriptor length should be 1.
The stream_type should be 01 according to ISO/IEC 13818-1, table 2-29,
The video elementary stream should only contain ISO/IEC 11172 video.

Constrained_parameter_flag: If the MPEG_1_only_flag is set to '0' test that the constrained_parameter_flag is set to '1'.
If set to '1', the corresponding elementary stream shall not contain unconstrained ISO/IEC 11172-2 video data.
If set to '0', the corresponding elementary stream may contain both constrained parameters and unconstrained ISO/IEC 11172-2 video data.

The explanation of the possible video elementary streams, according to the 4 different possible values of the MPEG_1_only and constrained_parameter flags, is the following:

	Constrained_parameter_flag = 1	Constrained_parameter_flag = 0
MPEG_1_only_flag= 1	Only constrained ISO/IEC 11172-2 video	Constrained ISO/IEC 11172-2 video and unconstrained ISO/IEC 11172-2 video
MPEG_1_only_flag= 0	Constrained ISO/IEC 11172-2 video and ISO/IEC 13818-2 video	Not allowed

Still_picture_flag: If still_picture_flag is set to '1', test that the video elementary stream contains only I-pictures.

Profile_and_level_indication: When the elementary stream is ISO/IEC 13818-2 video, test for consistency with the profile and level indicated in the video elementary stream.

chroma_format: When the elementary stream is ISO/IEC 13818-2 video, test for consistency with the chroma format indicated in the video elementary stream.

frame_rate_extension_flag: If set to 1 , and when the elementary stream is ISO/IEC 13818-2 video, test that at least one of the video frame rate extension fields is non zero.

2.3.2.1.12.3 Audio stream descriptor

This descriptor can be tested only if it applies to an audio elementary stream.

The stream_id in the corresponding PES headers must be the stream_id of an audio elementary stream (first 3 bits set to '110', ISO/IEC 13818-1, table 2-18).

In the PMT and PSM tables, this descriptor is allowed, at the elementary level and is not allowed at the program level.

Descriptor_length: Test that the length is equal to 1.

free_format_flag: Test for consistency with audio bitrate_index field.
If set to 1 the audio stream has a free format.
If set to 0, the bitrate_index in the audio structure gives the audio bitrate.

ID: Test for consistency with audio ID field.

layer: Test for consistency with audio layer field in the audio elementary stream.

variable_rate_audio_indicator: This field has no corresponding field in the audio headers.
When set to 1, test that no bit rate changes occur between consecutive audio frames.

2.3.2.1.12.4 Hierarchy descriptor

This descriptor can be tested only if it applies to an audio or video or private ES.

In the PMT and PSM tables, this descriptor is allowed at the elementary level and is not allowed at the program level.

Descriptor_length: Test that the length is equal to 4.

Hierarchy_type: Test that the hierarchy_type is not one of the reserved values presented in the ISO/IEC 13818-1, table 2-45.
Test for consistency with video/audio elementary streams as appropriate.
Test for consistency with profile_and_level fields in case of a video ES.
Test for consistency with audio descriptors in case of an audio ES.

Hierarchy_layer_index: Test that the index is unique in the program definition.

Hierarchy_embedded_layer_index: Test for the presence of the ES described by a hierarchy descriptor where the hierarchy_layer_index is the same as this hierarchy_embedded_layer_index. Test that the decoding of the ES this descriptor is attached to is impossible unless the decoding of the ES described by a hierarchy descriptor where the hierarchy_layer_index is the same as this hierarchy_embedded_layer_index is performed.

2.3.2.1.12.5 Registration descriptor

This descriptor can be tested only if it applies to an audio, video, or private ES, or to a program.

In the PMT and PSM tables, this descriptor is allowed, at the elementary level and at the program level.

Descriptor_length: Test that it is greater than or equal to 4.

Format_identifier: Test by verification with the Registration Authority.

2.3.2.1.12.6 Data stream alignment descriptor

This descriptor can be tested only if it applies to a video or an audio elementary stream.

Descriptor_length: Test that the length is 1.

alignment_type: In the PES packet test that the alignment exists (dependent on Video or Audio)

2.3.2.1.12.7 Target background grid descriptor

This descriptor can be tested only if it applies to a video elementary stream.

Descriptor_length: Test that the length is 4.

pel_aspect_ratio: Test for consistency with the video sequence header in the video ES.

2.3.2.1.12.8 Video window descriptor

This descriptor can be tested only if it applies to a video elementary stream.

This can only be present if target_background_grid_descriptor present.

Descriptor_length: Test that the length is 4.

horizontal_offset: This value should not exceed the value of horizontal_size as coded in the target_background_grid_descriptor when present.

vertical_offset: This value should not exceed the value of vertical_size as coded in the target_background_grid_descriptor when present.

2.3.2.1.12.9 Conditional access descriptor

This descriptor can be tested only if it is in PMT or CAT in Transport Stream, or refers to stream_id in a Program Stream map.

CA_PID: Test that this is not a reserved PID value as defined in ISO/IEC 13818-1 table 2-4 or PID 0x1FFF.

2.3.2.1.12.10 ISO 639 language descriptor

language_code: Test that values are allocated/permitted according to ISO 639 and ISO 8859-1, and in order.

audio_type: Test that this does not take a value outside 0x00 to 0x03.

2.3.2.1.12.11 System clock descriptor

This descriptor can be tested only if it refers to a Program.

In the PMT and PSM tables, this descriptor is allowed at the program level.

Descriptor_length: Test that the length is 2.

When the External_clock_indicator = 1, if by means external to the bitstream the accuracy of the applied system clock is known, then cross check with the encoded values in this descriptor.

2.3.2.1.12.12 Multiplex buffer utilization descriptor

Descriptor_length: Test that the length is 3.

mdv_valid_flag: If set to '0' then no test on other fields.

multiplex_delay_variation: See ltw test method in 2.3.2.1.2 on page 21.

multiplex_strategy: Test that this is not set to any of values 4 through 7.

2.3.2.1.12.13 Copyright descriptor

This descriptor can be tested only if it applies to an audio, video, or private ES or to a program.

In the PMT and PSM tables, this descriptor is allowed, at the elementary level and at the program level.

Descriptor_length: Test that this is greater than or equal to 4.

copyright_identifier: cross check with the Registration Authority.

2.3.2.1.12.14 Maximum bitrate descriptor

This descriptor can be tested if it is used at program or ES level.

Descriptor_length: Test that the length is 3.

maximum_bitrate: test against the measured rate of the ES or program; test against the bitrate fields of video and audio if this descriptor is applied to ES.

2.3.2.1.12.15 Private data indicator descriptor

Descriptor_length: Test that the length is 4.

Other fields of this descriptor can not be tested.

2.3.2.1.12.16 Smoothing buffer descriptor

This descriptor can be tested only if it refers to an audio, video or private ES, or to a program.

Only used in the PMT and PSM tables, this descriptor is allowed at the elementary level and at the program level.

Descriptor_length: Test that the length is 6.

sb_leak_rate: See test on the sb_size field.

sb_size: Test that the sb buffer defined with its sb_size and its sb_leak_rate does not overflow.

Test that during the continuous existence of a program, the values of the fields in this descriptor are encoded with the same values.

2.3.2.1.12.17 STD descriptor

This descriptor can be tested only if it refers to a video ES under the Transport Stream syntax.

In the PMT table, this descriptor is allowed, at the elementary level.

Descriptor_length: Test that the length is 1.

reserved: set to '1111 111'.

leak_valid_flag: no direct test on the value of that flag which may be either ‘0’ or ‘1’.

If this leak_valid_flag is 1, the tests conducted to verify the T-STD conformance will verify if the leak method (as defined in ISO/IEC 13818-1, paragraph 2.4.2.3) has been correctly applied to the corresponding ES.

If this leak_valid_flag is 0, and if the vbv_delay fields present in the associated video stream do not have the value 0xFFFF, the tests conducted to verify the T-STD conformance will verify, for the corresponding ES, that the transfer of data from the buffer MBn to the buffer EBn uses the vbv_method (as defined in ISO/IEC 13818-1, subclause 2.4.2.3).

2.3.2.1.12.18 IBP descriptor

This descriptor can be tested only if it applies to a video elementary stream.

Only used in the PMT and PSM tables, this descriptor is allowed at the elementary level.

Descriptor_length: Test that the length is 2.

closed_gop_flag: If set to ‘1’, test for consistency with the video headers, that a Group of picture is encoded before every I-frame, and that the closed_gop flag is set in all picture headers in the video sequence. If set to ‘0’, no test.

identical_gop_flag: If set to ‘1’, cross check with the video headers that the number of P-frames and B-frames between I-frames, and that the picture coding types and sequence of picture types between I-pictures is the same throughout the sequence, except possibly for the pictures up to the second I-picture. If set to ‘0’, no test

max_gop_length: The value ‘0’ shall not be used. When the value is not ‘0’, any number of pictures between two consecutive I-pictures shall be smaller or equal to that value.

2.3.2.2 Tests of Program Streams

2.3.2.2.1 Tests on the pack level

pack_start_code: Test that the pack_start_code is 0x000001BA.

system_clock_reference: In successive packs, the system_clock_reference field contains encoded values which are samples of a nominal 27 MHz system clock. The maximum interval between system_clock_reference fields is limited by the difference between encoded values in successive packs. Test that this difference does not exceed 0,7*90 000, as specified in ISO/IEC 13818-1.

program_mux_rate: Test that the value encoded in the mux_rate field is sufficiently large that, if all bytes in the pack are transmitted at that rate, they are delivered to the system target decoder before the time the first byte of the subsequent pack is delivered. The time that the first byte of the subsequent pack is delivered may be derived from the system_clock_reference and mux_rate fields in that subsequent pack.

program_mux_rate: Test that the mux_rate field is not encoded with the value zero.

2.3.2.2.2 Tests on the system header

general test (1): Test that the first pack of a Program Stream contains a system header.

general test (2): Test that system headers, if present in the pack, immediately follow the pack header.

general test (3): If the Program Stream contains more than one system header, test that the values encoded in all the headers are identical.

system_header_start_code: Test that the system_header_start_code is equal to 0x000001BB.

header_length: Test that the header_length field is encoded with a value equal to the number of bytes in the system header that follow the header_length field.

rate_bound: Test that the bitrate indicated by the rate_bound field is greater than or equal to the maximum bitrate value encoded in any mux_rate field in the same Program Stream.

audio_bound: Test that the value encoded in the audio_bound field is greater than or equal to the maximum number of simultaneously active ISO/IEC 13818-3 audio streams in the Program Stream. For the purpose of this subclause, an ISO/IEC 13818-3 audio stream is active if:

a) the input buffer of the system target decoder of that ISO/IEC 13818-3 audio stream is not empty, or if

b) one of the presentation units decoded from that ISO/IEC 13818-3 audio stream is being presented.

audio_bound: Test that the value encoded in the audio_bound field is less than or equal to 32.

fixed_flag: If the fixed_flag is set to '1', then test that the values encoded in all system_clock_reference fields satisfy 2.5.3.6 in ISO/IEC 13818-1.

CSPS_flag: If the CSPS_flag is set to '1', then test that the packet rate and the system target decoder buffer size satisfy 2.5.3.6 in ISO/IEC 13818-1.

system_audio_lock_flag: If the system_audio_lock_flag is set to '1', then test that the difference between the values encoded in any two presentation_time_stamp fields in audio packets of the same ISO/IEC 13818-3 audio stream correspond to the duration of the decoded audio access units in that ISO/IEC 13818-3 audio stream. For this purpose the duration (in terms of units of the system clock frequency) shall be derived from the number of samples and the sampling frequency of the decoded access units and the ratio SCASR as specified in 2.5.3.6 of ISO/IEC 13818-1. This assumes that no discontinuities occurred in the ISO/IEC 13818-3 audio stream in the presentation of the access units during the presentation period defined by both fields. See 2.7.5 of ISO/IEC 13818-1 for the definition of discontinuities.

system_video_lock_flag: If the system_video_lock_flag is set to '1', then test that the difference between the values encoded in any two presentation_time_stamp fields in video packets of the same ISO/IEC 13818-2 video stream shall correspond to the duration of the decoded pictures in that ISO/IEC 13818-2 video stream. For this purpose the duration (in terms of units of the system clock frequency) shall be derived from the frame rate of the decoded pictures and the ratio SCFR as specified in 2.5.3.6 of ISO/IEC 13818-1. This assumes that no discontinuities occurred in the ISO/IEC 13818-2 video stream in the presentation of the access units during the presentation period defined by both fields. See 2.7.5 of ISO/IEC 13818-1 for the definition of discontinuities.

video_bound: Test that the value encoded in the video_bound field is greater than or equal to the maximum number of simultaneously active ISO/IEC 13818-2 video streams in the ISO/IEC 13818 stream. For the purpose of this subclause, an ISO/IEC 13818-2 video stream is active if:

a) the input buffer of the system target decoder of that ISO/IEC 13818-2 video stream is not empty, or if

b) the reorder buffer of the system target decoder of that ISO/IEC 13818-2 video stream is not empty, or if

c) one of the presentation units decoded from that ISO/IEC 13818-2 video stream is being presented.

video_bound: Test that the value encoded in the video_bound field is less than or equal to 16.

stream_id: Test that the value encoded in the stream_id field is one of the values permitted by table 2-18 (stream_id) in 2.4.3.7 in ISO/IEC 13818-1.

stream_id: Test that the stream_id mechanism refers exactly once to each elementary stream in the multiplex.

STD_buffer_bound_scale: If the stream_id refers to an ISO/IEC 13818-3 audio stream, test that the STD_buffer_bound_scale is set to '0'. If the stream_id refers to an ISO/IEC 13818-2 video stream, test that the STD_buffer_bound_scale is set to '1'.

STD_buffer_size_bound: Test that the value encoded in the STD_buffer_size_bound field is greater than or equal to the maximum value encoded in any of the STD_buffer_size fields in packets of the same elementary stream.

2.3.2.2.3 Tests on Program Stream map

packet_start_code_prefix: Test that it has the value '0x000001'.

map_stream_id: Test that it is coded '0xBC'.

program_stream_map_length: Test that is does not contain a value greater than 1018. Test that this length field is consistent with the following length fields program_stream_info_length and elementary_stream_map_length, and that the CRC_32 field is correct.

current_next_indicator: The value of this bit is considered with program_stream_map_version.

program_stream_map_version: This is analysed in conjunction with the current_next_indicator. Any value of either field is permitted within the Program Stream, but the order in which they occur is restricted, and should be tested as follows.

If version number v with current_next_indicator set to '0' is encountered then test that it is only followed by a PSM with version number v and current_next_indicator set to '1', with identical data.

If version_number v with current_next_indicator set to '1' is encountered then test that it is only followed by one of the following:

i) an identical copy of itself

ii) a PSM with version number v+1 with current_next_indicator set to '1' or '0'

program_stream_info_length: Test that when this length field is assumed to be correct, then along with the descriptor lengths and other length fields, the CRC_32 field is shown to be correct. This length only codes the length of the immediately following descriptors.

descriptor: Test that only descriptors are included which are permitted in Program Streams; see table 2-39 in ISO/IEC 13818-1.

elementary_stream_map_length: Assume this length to be correct each time it occurs.

stream_type: Test that the stream_type field does not take the following values: 0x00, 0x05, 0x0A-0x7F.

elementary_stream_id: Test that this does not take one of the following values: '1011 1100'; '1111 0100'-'1111 1111'.

- Test that there is no contradiction between the elementary_stream_id and the stream_type fields:
- If the stream_type field takes the value 0x01, test that the elementary_stream_id takes a value '1110 xxxx' and that PES packets with a stream_id equal to the elementary_stream_id contain ISO/IEC 11172-2 data.
- If the stream_type field takes the value 0x02, test that the elementary_stream_id takes a value '1110 xxxx' and that PES packets with a stream_id equal to the elementary_stream_id contain ISO/IEC 13818-2 data.
- If the stream_type field takes the value 0x03, test that the elementary_stream_id takes a value '110x xxxx' and that PES packets with a stream_id equal to the elementary_stream_id contain ISO/IEC 11172-3 data.
- If the stream_type field takes the value 0x04, test that the elementary_stream_id takes a value '110x xxxx' and that PES packets with a stream_id equal to the elementary_stream_id contain ISO/IEC 13818-3 data.
- If the stream_type field takes the value 0x06, test that the elementary_stream_id takes a value '1011 1101' or '1011 1111' and that PES packets with a stream_id equal to the

elementary_stream_id contain ISO/IEC 11172-1 or ISO/IEC 13818-1 private_stream_1 or private_stream_2 data.

- If the stream_type field takes the value 0x07, test that the elementary_stream_id takes a value '1111 0011' and that PES packets with a stream_id equal to the elementary_stream_id contain ISO/IEC 13522 data.
- If the stream_type field takes the value 0x08, test that the elementary_stream_id takes a value '1111 0010' and that PES packets with a stream_id equal to the elementary_stream_id contain ISO/IEC 13818-1 DSM-CC data as defined in Annex A of ISO/IEC 13818-1, or ISO/IEC 13818-6 DSM-CC data.
- If the stream_type field takes the value 0x09, test that, if the elementary_stream_id takes a value '1111 1111' then PES packets with a stream_id equal to the elementary_stream_id contain a ISO/IEC 13818-1 program_stream_directory, or if the elementary_stream_id takes a value '1011 1011' then PES packets with a stream_id equal to the elementary_stream_id contain a ISO/IEC 13818-1 or 11172-1 system_header, or if the elementary_stream_id takes a value '1011 1010' then PES packets with a stream_id equal to the elementary_stream_id contain a ISO/IEC 13818-1 or 11172-1 pack_header.

elementary_stream_info_length: Test that when this length field is assumed to be correct, then along with the descriptor lengths and other length fields, the CRC_32 field is shown to be correct. This length only codes the length of the immediately following descriptors.

descriptors: Test that only descriptors are included which are permitted in Program Streams; see table 2-39 in ISO/IEC 13818-1.

CRC_32: Test that the four bytes after the last descriptor byte in the last loop of ES descriptors are the same as a CRC_32 calculated over the whole section excluding these bytes.

2.3.2.2.4 Tests on Program Stream directory

packet_start_code_prefix: Test that it is coded '0x000001'.

directory_stream_id: Test that it is coded 'FF'

PES_packet_length: Test that this length field is consistent with the number_of_access_units field such that PES_packet_length is equal to 14 + (18 x number_of_access_units).

prev_directory_offset: If the encoded value is zero, test that there is no previous program stream directory in the Program Stream. If the encoded value is not zero, then test that the first byte of the previous program stream directory is addressed.

next_directory_offset: If the encoded value is zero, test that there is no next program stream directory in the Program Stream. If the encoded value is not zero, then test that the first byte of the next program stream directory is addressed.

The following 18 bytes all refer to a single access unit, and the fields should be repeated once for each access unit signalled in the number_of_access_units field.

packet_stream_id: Test that it takes one of the following values: ‘110x xxxx'; '1110 xxxx'.

PES_header_position_offset_sign: May be set to any value. It is considered in conjunction with the PES header position offset fields which follow.

PES_header_position_offset: Test that the encoded value correctly gives the byte offset of a PES packet of stream_id = packet_stream_id containing the first byte of an access unit. The PES packet follows the directory if the PES_header_position_offset_sign is coded '0', and precedes it if the sign is coded '1'. If the PES_header_position_offset is coded to '0' then there is no test.

reference_offset: Test that it points to the first byte of the access unit relative to the first byte of the PES packet (i.e. packet_start_code_prefix) signalled to contain the access unit.

marker_bit: Test that this bit is '1'.

PTS: Test that the coded PTS value is consistent with the PTS's encoded in the elementary stream referred to.

bytes_to_read: Test that the encoded number of bytes includes all of the bytes of the access unit referred to.

marker_bit: Test that this bit is '1'.

intra_coded_indicator: In the case of an ISO/IEC 13818-2 stream or an ISO 11172-2 stream, then if intra_coded_indicator is set to '1', test that the access unit pointed to is predictively coded, and if coded to '0' then test that the access unit is not be predictively coded.

coding_parameters_indicator: Test that the encoding of this field is consistent with the coding parameters applied in the access unit referred to.

2.3.2.3 Tests of timing accuracy

PCR timing accuracy is verified by testing conformance with all of the following elements:

1.) Do the computations required to verify that the bitstream complies with the constraints of the T-STD model of ISO/IEC 13818-1.

2.) For a constant bitrate Transport Stream, test that all PCR's of a given PID satisfy the inequality given in the expression that follows, except under those conditions in which i and i" span a PCR discontinuity or PCR(i) is indicated as discontinuous. In order for the stream to be compliant, there must be at least one value of "k" that satisfies all PCR samples for all PCR PIDs.

$$\frac{(i - i'' - 1)}{PCR(i) - PCR(i'') + \delta} \leq k \leq \frac{(i - i'' + 1)}{PCR(i) - PCR(i'') - \delta}$$

Where $\delta = 27 + 810*(PCR(i)-PCR(i''))/27e6$

Where i and i" are the indices of bytes containing the last bit of the corresponding program_clock_reference_base fields of the PCRs being tested.

The -1 and +1 terms in the numerators allow for quantization uncertainty, since the i' and i" are bytes indices.

This test verifies that all PCRs were generated by system clocks that are within +/- 30 ppm of each other. The two terms in the expression for δ are: (1) 27, which arises from the +/- 500 nsec PCR tolerance allowed by part 1 of ISO/IEC 13818, and (2) a term which allows for 810 Hz (30 ppm) frequency error in the system clock allowed by part 1.

3.) For a constant bitrate Transport Stream, test that any two PCR's of a given PCR_PID satisfy the inequality given in the expression that follows, except under those conditions in which i' and i" span a PCR discontinuity or PCR(i') is indicated as discontinuous. In order for the stream to be compliant, there must be at least one value of "k" that satisfies all PCR samples for all PCR PIDs.

$$\frac{(i' - i'')}{PCR(i') - PCR(i'') + \delta} \leq k \leq \frac{(i' - i'')}{PCR(i') - PCR(i'') - \delta}$$

where

i' is the index of the byte containing the last bit of a program_clock_reference_base field applicable to the program being tested.

i" is the index of the byte containing the last bit of a program_clock_reference_base field previous to i' applicable to the program being tested.

The value for δ is determined by the maximum rate of change of the system clock frequency (slew rate), the length of time between the two PCR values being considered, and the acceptable tolerance of the PCR value (given as 500 ns in ISO/IEC 13818-1). Consider a bitstream created by an encoder experiencing the maximum allowable system clock slew of 0,075 Hz/sec. The amount of time required to change from the minimum acceptable system clock frequency to the maximum acceptable system clock frequency (a total change of 1620 Hz) is 6 hours or $5,832 \times 10^{11}$ system clock cycles.

Therefore, given two PCR values such that $(PCR(i') - PCR(i'')) \leq 5,832 \times 10^{11}$, the value of δ is determined as follows:

$$\delta = \text{inaccuracy of PCR values} + \text{inaccuracy of system clock from slew}$$

$$\delta = (500x10^{-9})*(2)*(27x10^{6}) + (0,075/2)*(PCR(i') - PCR(i''))^{2}/(27x10^{6})^{2}$$

The term representing the worst case inaccuracy of the system clock from slew has been derived by integrating the area under the line formed by plotting time vs. system clock frequency between the two PCR time values.

The PCR accuracy is not compliant if the system clock used to generate the bitstream continues to change at the maximum allowable slew rate for more than 6 hours. Given two PCR values such that $(PCR(i') - PCR(i'') > 5,832 \times 10^{11}$, the value of δ is determined as follows:

$$\delta = \text{inaccuracy of PCR values} + \text{inaccuracy of system clock from 6 hours of slew} + \text{maximum frequency offset}$$

$$\delta = (500x10^{-9})*(2)*(27x10^{6}) + (0,075/2)*(5,832 \times 10^{11})^{2}/(27x10^{6})^{2} + (1620)*(PCR(i') - PCR(i'') - 5,832 \times 10^{11})/(27x10^{6})$$

The time of 6 hours used in the above calculations is based on a default system clock frequency accuracy of +/- 30 ppm. If a system clock descriptor is present and indicates a system clock frequency accuracy other than 30 ppm, then the value of 6 hours used in this calculation must be adjusted accordingly.

4.) If the intended transport rate is known by means outside of the bitstream, then test that the value of k found from **2**) above is consistent with this known value.

5.) For a constant bitrate Transport Stream, test that any four PCR's of a given PCR_PID satisfy the inequality given in the expression that follows, except under those conditions in which i' and i"'' span a PCR discontinuity or PCR(i'''') is indicated as discontinuous.

$$\frac{(i' - i'')}{PCR(i') - PCR(i'') + \delta} \le \frac{(i'''-i'''')}{PCR(i''')-PCR(i'''')} \le \frac{(i' - i'')}{PCR(i') - PCR(i'') - \delta}$$

where

i’, I’’, I’’’, i’’’’ are the indices of the bytes containing the last bit of a program_clock_reference_base field of the PCR’s being tested, and

i < i’’ < i’’’” (Note: normally i’ < i’’ << i’’’ < i’’’’)

i’ < i’’’ < i’’’’

The value for δ is determined by the maximum rate of change of the system clock frequency (slew rate), the length of time between the two PCR values being considered, and the acceptable tolerance of the PCR value (given as 500 ns in ISO/IEC 13818-1). Consider a bitstream created by an encoder experiencing the maximum allowable system clock slew of 0.075 Hz/sec. The amount of time required to change from the minimum acceptable system clock frequency to the maximum acceptable system clock frequency (a total change of 1620 Hz) is 6 hours or 5.832×10^{11} system clock cycles.

Therefore, given two PCR values such that (PCR(i’) - PCR(i’’”)) ≤ 5.832×10^{11}, the value of δ is determined as follows:

δ = inaccuracy of PCR values + inaccuracy of system clock from slew

$$\delta = (500\text{x}10^{-9})*(2)*(27\text{x}10^{6}) + (0.075/2)*(PCR(i') - PCR(i'''))^{2}/(27\text{x}10^{6})^{2}$$

The term representing the worst case inaccuracy of the system clock from slew has been derived by integrating the area under the line formed by plotting time vs. system clock frequency between the two PCR time values.

The bitstream is not compliant if the system clock used to generate the bitstream continues to change at the maximum allowable slew rate for more than 6 hours. Given two PCR values such that (PCR(i’) - PCR(i’’”) > 5.832×10^{11}, the value of δ is determined as follows:

δ = inaccuracy of PCR values + inaccuracy of system clock from 6 hours of slew
+ maximum frequency offset

$$\delta = (500\text{x}10^{-9})*(2)*(27\text{x}10^{6}) + (0.075/2)*(5.832 \times 10^{11})^{2}/(27\text{x}10^{6})^{2} + (1620)*(PCR(i') - PCR(i''') - 5.832 \times 10^{11})/(27\text{x}10^{6})$$

The time of 6 hours used in the above calculations is based on a default system clock frequency accuracy of +/- 30 ppm. If a system clock descriptor is present and indicates a system clock frequency accuracy other than 30 ppm, then the value of 6 hours used in this calculation must be adjusted accordingly.

2.3.2.4 Buffer overflow/underflow tests for Transport Streams

Transport Stream testing of buffer underflow and overflow in the T-STD is done for one program at a time.

For each elementary stream of a program, a TB buffer test shall be performed, in order to verify:

(1) that the TB never overflows, and

(2) that the TB gets empty at least once a second

This test can be implemented by continuously monitoring the fullness of TB, based on the delivery schedules of data entering and leaving the buffer. The input to the buffer is defined by the bytes of the Transport Stream that enter the buffer at the rate derived from the bitstream by applying the transport rate formulae from ISO/IEC 13818-1. The output of the buffer is defined by the output rate defined in ISO 13818-1. If there is no data in TB, then the output rate equals zero.

For each video elementary stream of a program, an MB buffer and an EB buffer test shall be performed, in order to verify the constraints of these two buffers:

```
if (Leak method)
        {       test that MB never overflows;
                test that MB gets empty at least once a second;
                test that EB  never goes to underflow, except when low_delay == 1 or in trick mode;
        }


if (VBV Delay method)
        {       test that MB never overflows;
                test that MB never underflows;
                test that EB  never overflows;
                test that EB  never goes to underflow, except when low_delay == 1 or in trick mode;
        }
```

In this test, the only data that enters the system transport buffer, TB_{sys}, is complete Transport Stream packets containing system information, for the program under test. The PCR's to be used in equations (2-4) and (2-5) of ISO/IEC 13818-1 are the PCRs of the program under test.

This test can be implemented by continuously monitoring the fullness of MB and EB, based on the delivery schedules of data entering and leaving both buffers. The input to the buffer MB is defined by the output rate of TB. The output of buffer EB is defined by the decoding process of pictures, which instantaneously removes data from EB at times defined by the time stamps encoded in the bitstream. The output of MB enters EB instantaneously upon leaving MB. If the VBV Delay method is applied, the transfer of data between MB and EB is defined by emulating the input rate of the VBV buffer, with the timing of the decoding process defined by the time stamps encoded in the bitstream. If the Leak method is applied, the transfer rate between MB and EB is defined by the formulae defined for the leak method in ISO 13818-1.

2.3.3 General system decoder capabilities

In this subclause System Decoder capabilities are provided. Most of these capabilities cannot be tested in a mathematical way. These capabilities should therefore be considered as a functional checklist that can be used as a focus point for application and decoder developers. These capabilities are required for each compliant decoder.

2.3.3.1 Handling of decoder discontinuities

2.3.3.1.1 Sequence concatenation; decoding discontinuities; splicing; format changes

In compliant Transport Streams, at any audio access unit boundary or any video sequence boundary, the following discontinuities in the decoding process parameters can occur:

- for video, any parameter set in the sequence header or lower layer headers, such as profile/level, frame rate, bitrate, GOP parameters, picture format, etc.;
- for audio, any parameter such as audio layer, bitrate, sample rate, etc.;

- for both video and audio, the decoding time of the first access unit after the boundary can be larger than would have been expected had the boundary not been present. This can happen independently for all, some, or one of the elementary streams of a program. It may or may not be indicated by the presence of extra information referring to a seamless or non-seamless splice point.

Assuming any combination of change(s) in decoding process parameter(s) which lead(s) to parameter values that are supported by the decoder under test, the decoder under test shall:

- maintain correct presentation synchronisation between the different elementary streams of the program;
- not produce unacceptable audio or video artifacts, such as chirps, blocking etc. However, when a decoding discontinuity occurs, there may not be any data to present during some time interval. At such instants, audio decoders are recommended to mute and video decoders to freeze frame/field.

2.3.3.1.2 Time base synchronisation

Transport Stream decoders shall synthesise a decoder STC which is a local replica of the STC used to encode PCRs in the PCR_PID of the program being decoded. PCR discontinuities which are indicated by the discontinuity_indicator must produce corresponding discontinuities in the decoder STC. The decoder STC shall be used to derive the video and audio sampling rates

Program Stream decoders shall synthesise a decoder STC which is a local replica of the STC used to encode the SCRs in the pack_headers. If either the system_audio_lock_flag or the system_video_lock_flag is set, the decoder STC shall be used to derive the corresponding sampling rate.

In both Transport Stream and Program Stream decoders, the decoder STC shall be used as a basis for processing DTS and PTS timestamps in the program.

The amount of jitter on the PCR (SCR) samples is application dependent. The decoder STC must track the received (jittered) PCRs, which may have discontinuities indicated, (or SCRs) with sufficient accuracy that the program can be decoded and presented without artifacts, such as dropping or repeating presentation units, or malfunctions, such as buffer overflow, due to poor synchronisation. The output quality (frequency stability and phase noise) of the decoder STC must be sufficient to support all functions for which it is used e.g. chroma subcarrier generation if a composite video signal is being reconstructed.

2.3.3.2 Presentation timing

Timing of the presentation of decoded elementary streams shall be based on the encoded and interpolated PTSs for each presentation unit, as specified in ISO/IEC 13818-1. The timebase used shall be the replica of the STC synthesised in the decoder. The delay due to the implementation of the decoder must be accounted for when calculating the actual presentation time from the PTS. In normal operation (not trick modes) all presentation units in the program shall be presented, and the variation in the difference between the PTS and the actual presentation times of presentation units shall not be perceivable. A decoder manufacturer should provide the applied difference between the actual presentation time of an access unit and the associated PTS value.

2.3.3.3 Presentation synchronisation

Decoders shall synchronise the audio and video presentation units within +/- 40 ms regardless of the number of audio or video streams in the program. The delay between video and audio shall remain essentially constant after synchronisation is achieved. This shall be maintained under all of the following conditions:

- Permissible PCR jitter
- Permissible STC variations

- After any splice
- After channel change or program change
- After any change of video sequence (i.e., concatenation)
- After descrambling of audio, video or both

Applications may require better synchronisation than specified herein.

2.3.3.3.1 Program acquisition

Decoders shall be able to use PSI information to support acquiring a program. This entails:

1. Accessing Transport Stream packets with PID value of 0x0000, and finding the start of a section, parsing the sections in order to read the program_map_PID value of the wanted program. Since during the continuous existence of a program, the program_map_PID shall not change, it is not strictly necessary to continuously monitor Transport Stream packets of PID value 0x0000.
2. Accessing Transport Stream packets with the PID value listed in the PAT as carrying the program_map_PID for the wanted program; identifying sections within this PID which have table_id 0x02, and which have the program_number field of the wanted program.
3. Reading the data within any such sections, to identify the Transport Stream packets which may contain PCRs for the program, to identify video and audio streams which compose the program, and if necessary using descriptors, to identify which stream(s) is/are chosen for decoding. Note that it is possible within the duration of a program for a program definition to change, for example for the required video PID to change value, therefore it is a desirable feature to continuously monitor such sections.

2.3.3.4 Support of variable bitrate within a program

From a video stream with a variable bitrate in a program, a compliant decoder shall decode and present all presentation units without any decoding or presentation artifacts, such as blocking or unintended dropping or freezing of frames. The decoder shall also maintain presentation synchronisation between audio and video.

2.3.3.5 General capabilities for program acquisition

2.3.3.5.1 Change in program definition

A change in program definition may be noticed from a change in the information contained in a program_map_section defining a specific program. A change can be assumed to have occurred when a section with the current_next_indicator set to current and the program_number of the wanted program is detected, with a version number different from that of the previous version_number being used. A change in PID filtering should be accomplished within 1 second of the time that the last byte of the CRC_32 of the program_map_section carrying the change exits the buffer Bsys.

Audio: Decoders should be able to follow such a change, muting between streams if necessary. No audible artifacts should occur when such a change has been signalled in a program_map_section. If different languages are offered within one program, then the decoder should continue to present the same language after a change as before it, provided that the initial language is still available after the change and is supported by descriptors.

Video: Decoders should be able to follow such a change, freezing the frame if necessary. No other visible artifacts should occur when such a change has been signalled in a program_map_section.

Note on Conditional Access
Decoders which do not support conditional access or descrambling functionality do not need the ability to access Transport Stream packets with PID value of 0x0001. For decoders which do claim this functionality it may be desirable to continuously monitor such Transport Stream packets.

2.3.3.6 Private data handling

The normal operation of compliant MPEG decoders shall not be affected by the presence of private data in a Transport Stream or a Program Stream.

2.3.3.6.1 Private data in Transport Streams

Decoders shall at a minimum be capable of recognising and discarding all packets for which the stream_type is private (stream_type = 0x05, 0x06, 0x80-0xFF). These include the NIT. The presence of such packets in a Transport Stream shall not cause interruption or degradation of service in a decoder processing that Transport Stream.

Decoders shall at a minimum be capable of recognising and ignoring all private fields. These fields are listed in ISO/IEC 13818-1 annex H.

2.3.3.6.2 Private data in Program Streams

Decoders shall at a minimum be capable of recognising and discarding all packets for which:

- stream_id = private_stream_1 (‘1011 1101’)
- stream_id = private_stream_2 (‘1011 1111’)
- stream_id = ECM_stream (‘1111 0000’)
- stream_id = EMM_stream (‘1111 0001’)

2.3.3.7 Support of trick modes

Each decoder shall accept trick mode streams without producing unacceptable video artifacts. (Field or frame repeat or blanking shall not be considered unacceptable artifacts.) In addition, the following requirements apply to decoders which claim to support trick modes.

Decoders shall continue to display meaningful video while performing the trick mode functions. The displayed video should also be indicative of the function it is attempting to represent. In addition, the following constraints are placed upon the individual trick mode functions themselves.

Discontinuities may occur upon entering, changing, or leaving the trick mode state, as well as during trick mode operation; therefore conformance requirements for the handling of discontinuities must also be met by the decoder. Audio/video synchronisation conformance may not be applicable while in the trick mode state.

Fast Forward

Decoders shall display only the field(s) indicated by *field_id*.

If *intra_slice_refresh* is set, then no constraints are placed upon the display of missing macroblocks between coded slices. However, it is strongly recommended that decoders display visually acceptable material in these locations, e.g., co-sited macroblocks from previous pictures.

Slow Motion

For interlaced pictures decoders shall first display the top field for *field_rep_cntrl* periods, followed by the bottom field for *field_rep_cntrl* periods. For progressive pictures, the entire frame shall be displayed for *field_rep_cntrl* periods.

Freeze Frame

Decoders shall display only the field(s) indicated by *field id*.

The field(s) displayed are from the first video presentation unit in the PES packet that contains the *field_id* parameter. If no video presentation unit is contained in that PES packet, then the field(s) displayed are from the most recent previous video presentation unit.

Fast Reverse

Decoders shall display only the field(s) indicated by *field_id*.

If *intra_slice_refresh* is set, then no constraints are placed upon the display of missing macroblocks between coded slices. However, it is strongly recommended that decoders display visually acceptable material in these locations, e.g., co-sited macroblocks from previous pictures.

Slow Reverse

For interlaced pictures decoders shall first display the bottom field for *field_rep_cntrl* periods, followed by the top field for *field_rep_cntrl* periods. For progressive pictures, the entire frame shall be displayed for *field_rep_cntrl* periods.

2.3.3.8 Systems decoder requirements for forward compatibility

Wherever practical, decoders should be designed to allow for future compatible extensions to the bitstream syntax. All "reserved" bits or fields in MPEG-2 ISO/IEC 13818-1 Transport Streams or Program Streams shall be ignored by compliant decoders. “Ignored” means that the presence and/or content of these fields has no effect on the decoder. This includes the fact that “reserved” fields may in the future be defined as extensions of the bitstream syntax, which extensions should have no effect on decoders presently compliant. All “private” bits or fields shall be ignored by compliant decoders. The actions of decoders which are intended to make use of private definitions of private fields are outside the scope of ISO/IEC 13818 and no compliance tests are applied to verify if or how such decoders make use of private data.

2.3.4 Procedures to test system decoder conformance

This subclause describes procedures to verify conformance of system decoders. Each compliant decoder shall be able to decode all compliant ISO/IEC 13818-1 streams within the subset of the standard defined by the specified capabilities of the decoder.

All tests are performed using error free bitstreams. To test for correct interpretation of syntax and semantics, test sequences covering a wide range of parameters shall be supplied to the decoder under test and its output sequence shall be compared with the known expected output as described for the specific test sequence or bitstream. The comparison can be done, for example, by performing subjective evaluation, by verification of the expected result, or by comparing the timing performance. Such tests are necessary but not sufficient to prove conformance. They are helpful for discovering non-compliant implementations.

To test for decoder forward compatibility (i.e., that decoders ignore reserved and private data), the decoder under test shall be supplied test bitstreams which contain both “reserved” fields that contain data other than 1’s, and “private” data. ISO/IEC 13818-1 annex H explains the various methods by

which private data can be carried. Note that bitstreams containing “reserved” fields with data other than 1’s are not presently compliant bitstreams, but must be used to verify decoder compliance.

Tests are expected to be used for testing ISO/IEC 13818 decoders, including video and audio decoding, as it is generally not practical to test system decoders (or ISO/IEC 13818-1 decoders) alone. Practical test results depend on successful (or expected) output of the entire ISO/IEC 13818 decoder (systems, video, and audio).

Annex C lists an example test suite designed to test specific decoder capabilities. Examples of the test bitstreams corresponding to the descriptions of annex C are included as an informative electronic annex.

2.4 Video

In the video sections of this part of ISO/IEC 13818, except where stated otherwise, the following terms are used for practical purposes:

The term 'bitstream' means ISO/IEC 13818 video bitstream.

The term 'decoder' means ISO/IEC 13818 video decoder, i.e., an embodiment of the decoding process specified by ISO/IEC 13818-2. The video decoder does not include the display process, which is outside the scope of this standard. The output of a video decoder is specified in subclause 7.12 of ISO/IEC 13818-2.

The term 'Chapter 8' means Chapter 8 of ISO/IEC 13818-2 (definition of authorized profile-and-level combinations), as well as any future amendments of the standard which defines additional profile-and-level combinations not currently defined in Chapter 8.

The term 'verifier' means a ISO/IEC 13818 video bitstream verifier, i.e., a process by which it is possible to test and verify that all the requirements specified in ISO/IEC 13818-2 are met by the bitstream.

If any statement stated in this section accidentally contradicts a statement or requirement defined in ISO/IEC 13818-2, the text of ISO/IEC 13818-2 prevails.

The following subclauses specify the normative tests for verifying compliance of video bitstreams and video decoders. Those normative tests make use of test data (bitstream test suites) provided as an electronic annex to this document, and of a software verifier specified in ISO/IEC 13818-5 with source code available in electronic format.

Conformance of scalable hierarchies of bitstreams and scalable profile decoders is defined in 2.4.5.

2.4.1 Definition of video bitstream compliance

An ISO/IEC 13818 video bitstream is a bitstream that implements the specification defined by the normative sections of ISO/IEC 13818-2 (including Annexes A, B and C).

A compliant bitstream shall meet all the requirements and implement all the restrictions defined in the generic syntax defined by the ISO/IEC 13818-2 specification, including the restrictions defined in Chapter 8 for the profile-and-level specified in the sequence_extension() of the bitstream.

Subclause 2.4.2 of this part of ISO/IEC 13818 defines the normative test that a bitstream shall pass successfully in order to be claimed compliant with this specification.

A compliant bitstream of a given profile-and-level may be called an "ISO/IEC 13818-2 *Profile@Level* bitstream" or simply a "*Profile@Level* bitstream" (e.g. an MP@ML bitstream).

The following subclauses clarify some of the important requirements on bitstreams and on encoders producing bitstreams:

2.4.1.1 Requirements and restrictions related to profile-and-level

The profile_and_level_indication shall be one of the valid codes defined in Chapter 8. The profile-and-level derived from the profile_and_level_indication indicates that additional restrictions and constraints have been applied to several syntactic and semantic elements, as defined in Chapter 8.

The restrictions defined for a given profile-and-level are aimed at reducing the cost of decoder implementation and at facilitating interoperability. A compliant bitstream shall be decodable by any compliant ISO/IEC 13818 video decoder that supports the profile-and-level combination specified in the bitstream.

2.4.1.2 Additional restrictions on bitstream applied by the encoder

The video encoder may apply any additional restrictions on the parameters of the video bitstream, in addition to restrictions defined in the generic video syntax and the restrictions defined for the specified profile-and-level in Chapter 8. Not all additional restrictions can be known a priori without analyzing

or decoding the entire bitstream, since the syntax does not provide explicit mechanisms which signal such restrictions in advance for all cases.

2.4.1.3 Encoder requirements and recommendations

2.4.1.3.1 Encoder requirements

Although encoders are not directly addressed by ISO/IEC 13818-2, an encoder is said to be an ISO/IEC 13818-2 Profile@Level encoder if it satisfies the following requirements:

1) The bitstreams generated by the encoder are compliant Profile@Level bitstreams.

2) For encoding methods which include embedded decoding operations to produce the coded bitstream, these decoding operations shall be performed with the full arithmetic precision specified in ISO/IEC 13818-2.

This second requirement is necessary to guarantee that only compliant decoders will produce images that have optimum quality.

With this requirement on ISO/IEC 13818-2 encoders, any compliant decoder decoding a bitstream generated by a compliant encoder will normally reconstruct images of higher quality, compared to the images reconstructed from the same bitstream by a non-compliant decoder.

2.4.1.3.2 Encoder recommendations

It is strongly recommended that video encoders capable of producing P-pictures implement Note 2 of subclause 7.4.4 of ISO/IEC 13818-2. Failure to implement this recommendation may cause significant accumulation of mismatch between the reconstructed samples produced by the hypothetical decoder sub-loop embedded within an encoder and those produced by a (downstream) decoder using the coded bitstream produced by the encoder.

It is strongly recommended that video encoders capable of generating concealment motion vectors produce such concealment vectors that will help the concealment process for decoders which implement the specific concealment technique described in subclauses D.13.1.1.2, D.13.1.1.3, and 7.6.3.9 of ISO/IEC 13818-2.

The concealment motion vector transmitted with a macroblock should be a frame motion vector that would provide a "good" frame prediction for predicting the macroblock that lies vertically below the macroblock in which the motion vector occurs.

If the encoder is capable of generating bitstreams with slice_picture_id, it is recommended that temporal interval between pictures using the same value for slice_picture_id be as large as possible, so that error bursts causing loss of several consecutive pictures can be detected.

2.4.2 Procedure for testing bitstream compliance

The technical report (ISO/IEC 13818-5) contains the source code of a software video verifier that checks that a bitstream implements properly most of the normative requirements defined in ISO/IEC 13818-2.

A bitstream that claims compliance with this standard shall pass the following normative test:

When processed by the technical report verifier, the bitstream shall not cause any error or non-conformance messages to be reported by the verifier. This test shall be applied only to bitstreams that are known to be free of errors introduced by transmission.

Successfully passing the technical report verifier test only provides a strong presumption that the bitstream under test is compliant, i.e. that it does indeed meet all the requirements specified in ISO/IEC 13818-2.

Additional tests may be necessary to check more thoroughly that the bitstream implements properly all the requirements specified in ISO/IEC 13818-2. These complementary tests may be performed using other video bitstream verifiers that perform more complete tests than those implemented by the technical report software.

ISO/IEC 13818-2 contains several informative recommendations. When testing a bitstream for compliance, it is useful to tests 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 ISO/IEC 13818-2 (e.g. r_size, macroblock_pattern).

A verifier does not necessarily perform all stages of the decoding process described in ISO/IEC 13818-2 in order to verify bitstream correctness. Many tests are performed on syntax elements in a state prior to their use in some processing stages. However, some arithmetic may need to be performed on combinations of syntax elements. For example, motion vectors used for prediction in the motion compensation stage (subclause 7.6 in ISO/IEC 13818-2) may need to be fully reconstructed in order to verify that predictions do not make reference to samples outside coded boundaries of the reference picture.

A verifier which does perform the IDCT transform and calculates the reconstructed samples must comply with all the arithmetic precision requirements specified in ISO/IEC 13818-2. In addition, the IDCT of such a verifier shall be an embodiment of the saturated mathematical integer-number IDCT specified in Annex A of ISO/IEC 13818-2 (a software implementation using 64-bit double-precision floating-point is sufficient).

Performing the IDCT transform and calculating the reconstructed samples in a verifier, although not necessary, is useful for several reasons:

- It allows to test the subjective quality of the reconstructed frames. ISO/IEC 13818-2 does not put any requirement on subjective quality, but it is desirable that an encoder generates bitstreams for which the subjective quality of reconstructed frames is as good as possible.

- Checking the output of the IDCT can provide an indication of whether or not the encoder that produced the bitstream observed the recommendation of Note 2 in subclause 7.4.4 of ISO/IEC 13818-2).

If a bitstream contains a P-picture with many occurrences of coded blocks of DCT coefficients (i.e., blocks that are not all zeros) for which the output of the reference IDCT is all zeros, then the encoder that produced the bitstream can be suspected of not implementing this important recommendation.

The best chance to discover this problem is when a still image (with no motion at all and no noise) is encoded.

2.4.3 Definition of video decoder compliance

In this subclause, except where stated otherwise, the term 'bitstream' means compliant ISO/IEC 13818 video bitstream (as defined in this part of ISO/IEC 13818) that has the profile_and_level_indication corresponding to the profile-and-level combination considered for the decoder.

Compliance of an ISO/IEC 13818-2 decoder is defined only with regard to a legal profile-and-level combination, as specified in Chapter 8.

The normative tests that a decoder shall pass in order to claim compliance with a given profile-and-level combination are specified in 2.4.4.6. A decoder can claim compliance with regard to several profile-and-level combinations if and only if it passes the normative tests defined for each of the profile-and-level combinations.

Only a decoder that passes the conformance test for a given profile-and-level may be called “ISO/IEC 13818-2 *Profile@Level* decoder” or simply “*Profile@Level* decoder” (e.g., an ISO/IEC 13818-2 MP@ML decoder).

A decoder that fails the normative tests defined by this specification may only claim limited accuracy compliance to the standard. A limited accuracy decoder shall be accompanied upon request by a technical description of the results of each of the normative tests. This technical description of the results shall include at least the result of the IDCT accuracy test, the peak error for all static tests, and for the all dynamic tests, all the peak temporal errors or jitters when presenting the reconstructed samples, the description of the reconstructed samples not presented, etc. Real-time software decoders

may have to use limited accuracy if the resources of the processor are not sufficient to achieve real-time performances with the full accuracy.

A limited accuracy decoder is not a compliant decoder and may only be called “ISO/IEC 13818-2 limited accuracy *Profile@Level* decoder” or simply “limited accuracy *Profile@Level* decoder”.

In the following text, decoder compliance is always considered with regard to a particular profile-and-level combination, even when this is not specifically mentioned.

A compliant decoder shall implement a decoding process that is equivalent to the one specified in ISO/IEC 13818-2 and meets all the general requirements defined in ISO/IEC 13818-2 that apply for the profile-and-level combination considered, and if it can decode bitstreams with any options or parameters with values permitted for that profile-and-level combination. The permitted options and parameter range for each profile-and-level combinations are defined in Chapter 8.

A decoder which implements only a subset of the options or ranges of syntax and semantics for a given profile-and-level combination is not a compliant decoder for that profile-and-level, even if it passes the normative tests specified in 2.4.4.6. In effect such a decoder would not be capable of decoding all compliant bitstreams of the considered profile-and-level combination.

In the following subclauses the term ‘reference decoder’ means the technical report software verifier (ISO/IEC 13818-5).

The reference decoder is a decoder that implements precisely the decoding process as specified in ISO/IEC 13818-2. The IDCT function that shall be used when running the reference decoder is the very accurate approximation of the mathematical saturated integer-number IDCT f ’’ (x, y) specified in Annex A of ISO/IEC 13818-2 obtained by implementing f ’’ (x, y) with double-precision arithmetic.

Except for possible mismatches caused by ambiguous half-values rounding at the output of the IDCT function, the output of the reference decoder (reconstructed samples) is defined unambiguously by ISO/IEC 13818-2.

Fundamental requirement areas for decoders are listed in the following subclauses.

2.4.3.1 Requirement on arithmetic accuracy (without IDCT)

With the exception of IDCT, the specification of ISO/IEC 13818-2 defines the decoding process absolutely unambiguously. Only the IDCT may yield different results among different implementations. The requirements on the accuracy of the IDCT used by a compliant decoder are specified in Annex A of ISO/IEC 13818-2.

There is a requirement that for a block that contains no coefficient data (i.e. if pattern_code[i] is zero, or if the macroblock is skipped) the sample domain coefficients f[x][y] for the block shall all take the value zero (Cf. subclause 7.5.1 of ISO/IEC 13818-2).

Therefore, the following is a the requirement on the arithmetic accuracy of the decoder:

When a coded picture is decoded from a bitstream, for each 8x8 block of the coded picture that is "not-coded" or that contains only zero DCT coefficients, a compliant decoder shall produce reconstructed samples numerically identical to those produced by the reference decoder when the reference frames used by both decoders are numerically identical. A decoder that reconstructs one sample with a value different from that reconstructed by the reference decoder for the same sample is not a compliant decoder.

In other words, all compliant decoders shall produce numerically identical reconstructed samples when the IDCT is applied only to blocks of zero coefficients (assuming that they use numerically identical reference frames).

2.4.3.2 Requirement on arithmetic accuracy (with IDCT)

When a bitstream contains some 8x8 blocks with non-zero DCT coefficients, the output of a compliant decoder may differ from the output of the reference decoder. However, because of the accuracy requirements on the IDCT transform used by the decoder, there exist some accuracy requirements on the output of a compliant ISO/IEC 13818 video decoder.

The IDCT used in a compliant decoder shall meet all the requirements defined in Annex A of ISO/IEC 13818-2.

Annex A of ISO/IEC 13818-2 defines additional requirements above those defined by the IEEE Std 1180-1990 standard. In order to claim that the IDCT transform used by the decoder conforms to the specification of Annex A, the IDCT transform shall comply with the IEEE Std 1180-1990 standard and pass successfully the following test:

The test is derived from the specification given in the IEEE Std 1180-1990 standard, with the following modifications:

1) In item (1) of subclause 3.2 of the IEEE specification, the last sentence is replaced by: <<Data sets of 1 000 000 (one million) blocks each should be generated for (L=256, H=255), (L=H=5) and (L=384, H=383). >>

2) The text of subclause 3.3 of the IEEE specification is replaced by : <<For any pixel location, the peak error shall not exceed 2 in magnitude. There is no other accuracy requirement for this test.>>

3) Let F be the set of 4096 blocks Bi[y][x] (i=0..4095) defined as follows :

a) Bi[0][0] = i - 2048

b) Bi[7][7] = 1 if Bi[0][0] is even, Bi[7][7] = 0 if Bi[0][0] is odd

c) All other coefficients Bi[y][x] other than Bi[0][0] and Bi[7][7] are equal to 0

For each block Bi[y][x] that belongs to set F defined above, an IDCT that claims to conform to the specification of Annex A of ISO/IEC 13818-2 | ITU-T Rec. H.262 shall output a block f[y][x] that as a peak error of 1 or less compared to the reference saturated mathematical integer-number IDCT f’’(x,y). In other words, | f[y][x] - f’’(x,y)| shall be <= 1 for all x and y.

Successfully passing the conformance test defined in this document only provides a strong presumption that the IDCT transform is compliant, i.e. that it does meet all the requirements specified in Annex A of ISO/IEC 13818-2.

Additional tests may be necessary to check more thoroughly that the IDCT implements properly all the requirements and recommendations specified in Annex A of ISO/IEC 13818-2.

2.4.3.3 Requirement on output of the decoding process and timing

The output of the decoding process is specified by subclause 7.12 of ISO/IEC 13818-2.

It is a requirement that all the reconstructed samples of all the coded frames be output by a compliant decoder to the display process. For example, a decoder that occasionally does not output some of the reconstructed B-frames or that occasionally outputs incomplete reconstructed frames to the display process is not compliant. The actual output of the display process is not specified by this standard.

It is a requirement that a compliant decoder outputs the reconstructed samples at the rates specified in subclause 7.12 of ISO/IEC 13818-2.

For example, when decoding an interlaced sequence, there is a requirement that the samples of each field be output to the display process at intervals of 1/(2 * frame_rate).

2.4.3.4 Requirement for compatibility with ISO/IEC 11172-2 (MPEG-1 video)

The requirements for compatibility with ISO/IEC 11172-2 (MPEG-1 video) are specified in subclause 8.1 of ISO/IEC 13818-2.

It is a requirement that a compliant ISO/IEC 13818-2 decoder shall decode all compliant ISO/IEC 11172-2 constrained parameters bitstreams. It should be noted that the permitted ranges for the parameters of ISO/IEC 11172-2 constrained parameters bitstreams are different and not necessarily a subset of the permitted ranges for equivalent parameters of ISO/IEC 13818-2 bitstreams.

For example ISO/IEC 11172-2 constrained parameters bitstreams can have horizontal_size up to 768 samples, and vertical size > 480 is possible with a frame_rate different from 25 Hz. A compliant

decoder should decode such ISO/IEC 11172-2 constrained parameters bitstreams (i.e. constrained_parameter_flag = 1).

In addition, it is a requirement that a compliant decoder shall decode D-pictures-only ISO/IEC 11172-2 bitstreams which are within the level constraints of the decoder including some that may have constrained_parameter_flag set to 0.

2.4.3.5 Requirements for compatibility between various profile-and-level combinations

Chapter 8 defines additional requirements for compatibility between various profile-and-level combinations. Those requirements are defined by Table 8-15 in Chapter 8. The decoder shall meet all those compatibility requirements.

For example, a compliant Main Profile at Main Level decoder shall also be a compliant Main Profile at Low Level and Simple Profile at Main Level decoder.

2.4.3.6 Requirement for forward compatibility of future extensions

ISO/IEC 13818-2 defines several requirements on decoder that are needed for allowing forward compatibility of future extension to ISO/IEC 13818-2 with existing compliant decoders.

A compliant decoder that encounters an extension with an extension start code described as "reserved" in ISO/IEC 13818-2 shall discard and ignore all subsequent data until the next start code.

A compliant decoder that encounters the syntactic element extra_information_picture described as "reserved" in ISO/IEC 13818-2 shall discard this syntactic element and any subsequent one until it encounters extra_bit_picture with the value 0.

A compliant decoder that encounters the syntactic element extra_information_slice described as "reserved" in ISO/IEC 13818-2 shall discard this syntactic element and any subsequent one until it encounters extra_bit_slice with the value 0.

2.4.3.7 Requirements related to zero byte stuffing, user data and reserved extensions

A compliant decoder shall be able to decode bitstreams with any permitted amount of zero byte stuffing, user data and reserved extensions, at any place where those can legally occur. The maximum permitted amount of these data is limited by VBV requirements specified in Annex C of ISO/IEC 13818-2.

The output of a compliant decoder shall be identical between two bitstreams which differ only in the amount of user_data, extra_information_slice, extra_information_picture, and start code stuffing present in each respective bitstream. For example, a compliant decoder shall produce the same output when decoding a bitstream that contains user data and when decoding the bitstream derived by replacing all user data by zero byte stuffing.

Note that it is permitted in ISO/IEC 13818-2 that a majority of coded data in a video sequence be in the form of zero stuffing bytes, user data and/or reserved extensions.

It should be noted that the output and behavior of the display process (which is not part of the conformance test) may depend on the presence or contents of user data. Therefore bitstreams that differ only by the amount or contents of user data may be displayed differently by a decoder system, even though the video decoders reconstruct identical samples.

2.4.3.8 Recommendations

In addition to the requirements, it is desirable that compliant decoders implement various recommendations defined in ISO/IEC 13818-2.

This subclause lists some of the recommendations.

It is recommended that a compliant decoder be able to resume the decoding process as soon as possible after an error (or the occurrence of a sequence_error_code). In most cases it is possible to resume decoding at the next start code.

It is recommended that a compliant decoder be able to perform concealment for the macroblocks or slices for which all the coded data has not been received.

2.4.4 Procedure to test decoder compliance

In this subclause, except where stated otherwise, the term 'bitstream' means compliant ISO/IEC 13818 video bitstream (as defined in this document), that has the profile_and_level_indication corresponding to the profile-and-level combination for which conformance of the decoder is considered.

There are two types of tests for decoders: static tests and dynamic tests.

2.4.4.1 Static tests

Static tests of a video decoder requires 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. In that case this test should be performed by the manufacturer during the design and development phase.

Static tests are used for testing the arithmetic accuracy used in the decoding process.

There are two sorts of static tests.

- The static tests that do not involve the use of IDCT, in which case the test will check that the values of the samples reconstructed by the decoder under test shall be identical to the values of the samples reconstructed by the reference decoder when the reference frames used by both decoders are numerically identical.

- The static tests that involve the use of IDCT, in which case the test will check that the peak absolute error between the values of the samples reconstructed by the decoder under test and the values of the samples reconstructed by the reference decoder shall not be larger than 2 when the reference frames used by both decoders are numerically identical.

2.4.4.2 Dynamic tests

Dynamic tests are applied to check that all the reconstructed samples are output to the display process and that the timing of the output of the decoder's reconstructed samples to the display process conforms to the specification of subclause 7.12 of ISO/IEC 13818-2, and to verify that the decoder buffer (as defined by Annex C of ISO/IEC 13818-2, VBV specification) does not underflow or overflow when the bits are delivered at the proper rate.

2.4.4.3 Specification of the test bitstreams

This subclause provides the list of specifications that are used to produce the bitstream test suites for testing decoder compliance.

Not all the decoder requirements are covered by these tests, but tests for the most fundamental decoder requirements are believed to be covered by this test suite specification. These tests include :

1. General static tests:

   Bitstreams using all the possible coding options permitted by ISO/IEC 13818-2.

2. Memory bandwidth dynamic tests:

   Bitstreams with all macroblocks predicted with average (bi-directional) prediction or dual-prime, with half-sample interpolation in both the horizontal and vertical directions, for both the luminance and chrominance blocks if possible, using smallest possible prediction blocks and accessing as many different samples of the reference pictures as possible.

3. VLC/FLC decoding static tests:

   Bitstream using all the possible events within a table.

4. Bits and Symbol distribution (burst) dynamic tests:

   Bitstream containing very irregular distribution of bits or symbols.

5. ISO/IEC 11172-2 compatibility tests:

   ISO/IEC 11172-2 Bitstreams.

To test a decoder for conformance with regard to a particular profile-and-level combination, a bitstream test suite can be made according to this specification. Each bitstream of the test suite must have its profile_and_level_indication corresponding to the profile-and-level combination considered for the decoder, and must be fully compliant with ISO/IEC 13818-2, or with ISO/IEC 11172-2 when specified.

When a bitstream requires the use of an option or parameter value not permitted with the profile-and-level combination considered (e.g., B-pictures in the case of Simple Profile at Main Level), the test bitstream must be omitted from the bitstream test suite.

All the bitstreams in the test suite must be such that the output of the non-saturated integer number mathematical IDCT f ’ (x, y), as defined in Annex A of ISO/IEC 13818-2, has values within the range [-384, 383] for each coded block.

A set of test bitstreams constructed according to those specifications is provided in an electronic annex that forms an integral part of this part of 13818. These bitstreams constitute normative test suites that must be used to verify conformance of decoders. The test suites are described in section 2.4.4.6.

2.4.4.3.1 Test bitstream #1

Specification: A series of consecutive frame B-pictures with all macroblocks using bi-directional field-based prediction. Luminance sample rate and bitrate are the maximum allowed for the profile-and-level combination. Half-sample interpolation in both the horizontal and vertical directions, for all luminance and chrominance blocks.

Functional stage: prediction bandwidth

Purpose: Check that the decoder handles the worst case of prediction bandwidth. Field-based prediction in frame pictures have the largest prediction bandwidth overhead. Picture buffers organized as frames (interleaved fields) and macroblocks stored in contiguous address page segments would have the greatest penalty. Effective filtered block size is 16x8.

2.4.4.3.2 Test bitstream #2

Specification: A bitstream with a B-picture as large as the maximum vbv_buffer_size allowed for the profile-and-level combination, using long VLC’s (not via escapes) as much as possible. Luminance sample rate and bitrate are the maximum allowed for the profile-and-level combination.

Functional stage: VLD

Purpose: Check that decoder works in this situation. A large B-picture located after several smaller coded pictures can catch a decoder off guard.

2.4.4.3.3 Test bitstream #3

Specification: A series of consecutive frame P-pictures with all macroblocks using dual-prime prediction. Luminance sample rate and bitrate are the maximum allowed for the profile-and-level combination. Maximize number of half-sample prediction in both the horizontal and vertical directions, for both luminance and chrominance blocks.

Functional stage: prediction bandwidth

Purpose: Check that the decoder handles the worst case of prediction bandwidth. Prediction bandwidth is at a maximum in this mode due to the small block sizes and two prediction sources.

2.4.4.3.4 Test bitstream #4

Specification: A bitstream with all macroblock_type transitions in frame and field pictures.

Functional stage: parser

Purpose: Check that decoder handles all scenarios in parsing tree.

2.4.4.3.5 Test bitstream #5

Specification: A bitstream where every slice contains only one macroblock, and where intra_slice_bit is present in every slice. Luminance sample rate and bitrate are the maximum allowed for the profile-and-level combination.

Functional stage: VLD and parser

Purpose: Check that decoder handles bitstreams with very short slices. CPU-oriented designs have large overhead for each slice and macroblock header.

2.4.4.3.6 Test bitstream #6

Specification: A bitstream with many different combinations of values for top_field_first, repeat_first_field, alternate_scan, intra_vlc_format, picture_structure, concealment_motion_vectors, intra_dc_precision, f_codes, q_scale_type, progressive_frame, frame_pred_frame_dct, variable numbers of consecutive coded B-frames, coded P-frames and coded I-frames, with some coded I-frames in the form of “I-P field-pictures”, with downloaded quantization weighting matrices. Ideally the bitstream should contain all possible legal combinations. Various syntax switches are toggled from picture-to-picture.

Functional stage: parser and control

Purpose: Check that decoder handle all scenarios.

2.4.4.3.7 Test bitstream #7

Specification: A bitstream with simultaneous burst of coded bits and maximum bandwidth dual-prime MC, followed by remaining macroblocks outside the burst with Dual Prime MC. Luminance sample rate and bitrate are the maximum allowed for the profile-and-level combination. Maximize number of half-sample predictions in both the horizontal and vertical directions, luminance and chrominance blocks.

Functional stage: VLD and prediction bandwidth

Purpose: DRAM is shared by VLD, MCP, and Display functions. This combination presents the longest sustainable period (whole picture) for DRAM bandwidth

2.4.4.3.8 Test bitstream #8

Specification: All possible VLC’s symbols and IDCT mismatch. Mismatch and saturation.

Functional stage: parser ; IDCT accuracy

Purpose: Test that decoders has included the complete VLC tables and implements mismatch control.

2.4.4.3.9 Test bitstream #9

This test has been removed from the test suite specification.

2.4.4.3.10 Test bitstream #10

Specification: Bitstream with only intra macroblocks using only the DC coefficient and predicted macroblocks having no DCT coefficients. Reconstructed motion vectors used for predicting both luminance and chrominance have all possible combinations of half-sample and full-sample values, both for the horizontal and the vertical coordinates, and all those combinations are used for each prediction

mode in both frame and field pictures, and with both interlaced and progressive chroma format in the case of 4:2:0 frame pictures.

Functional stage: MCP

Purpose: Check that decoder implements motion compensation stages with full accuracy in all cases. Except for reconstruction of Intra DC blocks, the test does not involve other decoder functions such as IDCT, inverse quantization and mismatch control. When a static decoder test is performed using the static test technique described in this document, the decoder under test shall reconstruct samples identical to those reconstructed by a reference decoder for all predicted macroblocks.

2.4.4.3.11 Test bitstream #11

Specification: Flat distribution of VLC events (worst case for constant rate symbolic VLD's) on B and P pictures. Luminance sample rate and bitrate are the maximum allowed for the profile-and-level combination.

Functional stage: VLD

Purpose: Check that decoder does not rely on statistically low count of symbols over global areas to meet real-time constraints.

2.4.4.3.12 Test bitstream #12

Specification: Bursty case for number of bits per macroblock with different burst location within picture (top, bottom), followed Bi-directional macroblocks. All motion vectors with half-sample components. Macroblocks outside the burst concentration have all bi-directional prediction. Luminance sample rate and bitrate are the maximum allowed for the profile-and-level combination. Half-sample in both the horizontal and vertical directions, luminance and chrominance blocks. Maximize number of prediction blocks required to reconstruct a macroblock.

Functional stage: VLD and prediction bandwidth

Purpose: Check that decoder does not rely upon statistically small number of coded bits over local areas.

2.4.4.3.13 Test bitstream #13

Specification: A series of consecutive Field-coded P-pictures, all macroblocks using Dual Prime prediction. As many half-sample components as possible in both the horizontal and vertical directions, luminance and chrominance blocks. Luminance sample rate and bitrate are the maximum allowed for the profile-and-level combination. Maximize number of prediction blocks required to reconstruct a macroblock.

Functional stage: prediction bandwidth

Purpose: Check that decoder handles largest prediction bandwidth with field-coded P-pictures. This test is somehow similar to Test bitstream #3, except that it uses field-pictures with Dual Prime.

2.4.4.3.14 Test bitstream #14

Specification: A bitstream with a series of consecutive Field coded B-pictures with 16x8 bi-directional macroblock motion compensation. Sequence contains many consecutive B pictures. Luminance sample rate and bitrate are the maximum allowed for the profile-and-level combination. Use half-sample prediction in both the horizontal and vertical directions, for all luminance and chrominance blocks. Maximize number of prediction blocks required to reconstruct a macroblock.

Functional stage: prediction bandwidth

Purpose: Check that decoder can cope with this case of worst case bandwidth. This test is somehow similar to Test bitstream #1, except that it uses field-pictures.

2.4.4.3.15 Test bitstream #15

Specification: Bitstream with R/P bits worth of extra_bit_slice in picture. Luminance sample rate and bitrate are the maximum allowed for the profile-and-level combination.

Functional stage: Parser

Purpose: Check that decoder is capable of handling a large number of bits concentrated in the extra bit slice loop.

2.4.4.3.16 Test bitstream #16

Specification: ISO/IEC 11172-2 (MPEG-1) constrained parameter bitstream. Luminance sample rate and bitrate are the maximum allowed for MPEG-1 constrained parameter bitstream.

Functional stage: overall

Purpose: Check that decoder can decode MPEG-1 constrained bitstreams.

2.4.4.3.17 Test bitstream #17

This test has been removed from the test suite specification.

2.4.4.3.18 Test bitstream #18

Specification: Low delay sequence with skipped pictures. Luminance sample rate and bitrate are the maximum allowed for the profile-and-level combination.

Functional stage: controller

Purpose: Check that decoder is capable of decoding low delay mode and knows how to recognize and deal with skipped pictures and buffer underflows in the VBV model.

2.4.4.3.19 Test bitstream #19

Specification: A bitstream implementing a test close to the IEEE 1180 IDCT mismatch test, to test the decoder's IDCT statistical accuracy. Can be done using P-pictures with a flat custom quantization matrix with all 16, and a quantizer stepsize of 0.5. Use whatever number of frames are required to satisfy statistic count. Note that because of saturation in [0, 255], the test cannot emulate exactly the IEEE 1180 IDCT test.

Functional stage: IDCT

Purpose: Check IDCT decoder accuracy. This is not a drift test since all macroblocks are of type Intra.

2.4.4.3.20 Test bitstream #20

Restriction: Only for profile-and-level combinations supporting SNR scalability:

Specification: Maximum VLD bandwidth on both layers (base and enhancement) with burst of escape codes, bursts of short VLC’s and maximum buffer size on both layers Luminance sample rate and bitrate are the maximum allowed for the profile-and-level combination.

Functional stage: test of parser(s)

Purpose: Test of the maximum VLD bandwidth on both layers (base and enhancement) with burst of escape codes, bursts of short VLC’s and maximum buffer size on both layers. Some designs may not be able to handle both layers

2.4.4.3.21 Test bitstream #21

Restriction: Only for profile-and-level combinations supporting SNR scalability:

Specification: Skipped macroblocks on base layer, on enhancement layer, and on both layers together. Test of the DCT type in the enhancement layer while macroblocks are skipped in the base layer. Luminance sample rate and bitrate are the maximum allowed for the profile-and-level combination.

Functional stage: test of parser

Purpose: Test of skipped MB's on base layer, on enhancement layer, and on both layers together. Test of the DCT type in the enhancement layer while macroblocks are skipped in the base layer. Sloppy decoders may not be able to handle skipped macroblocks in one of the layers.

2.4.4.3.22 Test bitstream #22

Restriction: Only for profile-and-level combinations supporting SNR scalability:

Specification: Different weighting matrices, different scanning on the two layers. Luminance sample rate and bitrate are the maximum allowed for the profile-and-level combination.

Functional stage: test of decoder

Purpose: Test of different weighting matrices, different scanning on the two layers. Sloppy decoders may not be able to handle different weighting matrices or scanning order.

2.4.4.3.23 Test bitstream #23

Restriction: Only for profile-and-level combinations supporting Spatial scalability:

Specification: All macroblock transitions in enhancement layer, all possible VLC symbols in enhancement layer, and all cases of motion vector updating, 3:1 horizontal and 2:1 vertical up-sampling, panning, all cases of up-conversion (interlace to interlace, interlace to progressive, progressive to interlace, etc.), all weight code tables, regions with spatial prediction only.

Functional stage: static test of spatially scalable decoder

Purpose: Test of all macroblock transitions in enhancement layer, all possible VLC symbols in enhancement layer, and all cases of motion vector updating. Sloppy decoders may not be able to handle all possible cases.

2.4.4.3.24 Test bitstream #24

Restriction: Only for profile-and-level combinations supporting Spatial scalability:

Specification: Different numbers of consecutive I, P and B frames in base and enhancement layer, spatial prediction based on the second most recently decoded base layer picture, mixing frame and field pictures, 2:1 upsampling in both directions.

Functional stage: static test of spatially scalable decoder

Purpose: Test that decoder can cope with different numbers of consecutive I, P and B frames in base and enhancement layer, test that decoder handles properly spatial predictions based on the second most recently decoded base layer picture, test that decoder handles properly mixed frame and field pictures.

2.4.4.3.25 Test bitstream #25

Specification: Bitstream causing maximum saturation of the inverse quantization by creating the greatest amplitude combinations of macroblock quantization (code 31), visual weighting matrix (value 255), and DCT coefficient (value -2047 or 2047).

Functional stage: inverse quantization

Purpose: Test that decoder implements properly the saturation of the inverse quantization (before the mismatch control).

2.4.4.3.26 Test bitstream #26

Specification: Bitstream causing large positive sample domain coefficients f[y][x] (e.g., 255) added to large predicted values p[y][x] (e.g., 255), or large negative sample domain coefficients f[y][x] (e.g., -256) added to small predicted values p[y][x] (e.g., 0).

Functional stage: addition of the output of IDCT f[y][x] to the predicted values p[y][x] and saturation of the result to the range [0, 255].

Purpose: Test that decoder implements properly the addition of the output of IDCT f[y][x] to the predicted values p[y][x] and saturation of the result to the range [0, 255].

2.4.4.3.27 Test bitstream #27

Specification: A bitstream with 16 bytes "extra_information_slice" in slice headers, and groups of 4096 bit of reserved and compatible extensions.

Functional stage: parser (discarding of reserved data).

Purpose: Test that decoder implements correctly parsing and discarding of certain types of reserved data (to ensure forward compatibility with future extensions of the standard), at least when a reasonable amount of those reserved data are present.

2.4.4.3.28 Test bitstream #28

Specification: A bitstream with zero byte stuffing :

• In the first half of the bitstream: at one of the legal positions in the bitstream, there will be at least 0,9*VBV_buffer_size worth of zero bit stuffing.

• In the second half of the bitstream, there will be in each picture, at a legal position, between R/P and 0,9*R/P zero bit stuffing (R=maximum bit rate of the bitstream ; 1/P= time between two consecutive pictures).

Functional stage: parser (discarding of stuffing).

Purpose: Test that decoder is capable of discarding stuffing in the worst case (almost a full VBV worth of stuffing).

2.4.4.3.29 Test bitstream #29

Specification: A bitstream with frame pictures, with motion vectors that are as large as permitted by the profile-and-level combination.

Functional stage: reconstruction of motion vectors, MCP, control

Purpose: Check that decoder implements motion compensation properly when motion vectors are very large.

2.4.4.3.30 Test bitstream #30

Specification: A bitstream with quantizer matrices (intra and non-intra, and if permitted, chroma matrices too). Matrices are not symmetrical (e.g., matrix coefficients are random numbers in the range [1, 255]). If permitted, use of both scanning orders.

Functional stage: quantizer matrix download, matrix scanning.

Purpose: Check that decoder can download properly quantizer matrices and that it uses of correct scanning of the matrices (i.e. not transposed).

2.4.4.3.31 Test bitstream #31

Specification: An ISO/IEC 11172-2 bitstream with D-pictures only, with constrained_parameter_flag = 0, frame size, bitrate and vbv_buffer_size set to the maximum allowed by the profile-and-level combination of the decoder.

Functional stage: overall

Purpose: Check that decoder can decode ISO/IEC 11172-2 bitstreams with D-picture only, with parameters in the range supported by the profile-and-level combination of the decoder.

2.4.4.3.32 Test bitstream #32

Specification: An ISO/IEC 11172-2 bitstream with constrained_parameter_flag = 1 and horizontal_size = 768.

Functional stage: overall

Purpose: Check that decoder can decode ISO/IEC 11172-2 constrained parameter bitstreams with the maximum .horizontal_size allowed when constrained_parameter_flag = 1.

2.4.4.3.33 Test bitstream #33

Specification: An ISO/IEC 11172-2 bitstream with constrained_parameter_flag = 1, vertical_size > 480 lines and frame_rate different from ‘25Hz’.

Functional stage: overall

Purpose: Check that decoder can decode ISO/IEC 11172-2 constrained parameter bitstreams vertical_size > 480 lines and frame_rate different from ‘25Hz’ (this combination is not allowed in some profile-and-level combinations, but is allowed for ISO/IEC 11172-2 constrained parameter bitstreams, as long as horizontal_size is small enough).

2.4.4.3.34 Test bitstream #34

Specification: A bitstream in which the output of the non-saturated integer number mathematical IDCT f ’ (x, y), as defined in Annex A of ISO/IEC 13818-2, has large absolute values but values within the range [-384, 383] for each coded block. If decoder under test uses the same IDCT for decoding ISO/IEC 11172-2 and ISO/IEC 13818-2 bitstreams, then this test bitstream can be implemented as an ISO/IEC 11172-2 constrained parameter bitstream.

Functional stage: IDCT

Purpose: Check that IDCT decoder accuracy meets the requirements defined in Annex A of ISO/IEC 13818-2. The peak error for a compliant decoder shall be less or equal to than 2 when decoding this bitstream. Note that for blocks where f ’ (x, y) has values within the range [-300, 300], decoders that have a peak error larger than 1 may not be compliant with the IEEE 1180 IDCT specification.

2.4.4.4 Implementation of the static test

For each bitstream of the test suite, the following operations are performed.

The bitstream is decoded by the decoder under test. All the samples reconstructed by the decoder under test are captured and stored for future use.

The bitstream is then decoded by the reference decoder as follows:

Before decoding each P- or B-picture, the frame buffers of the reference decoder are initialized with the reconstructed samples captured from the decoder under test that correspond to those reference frames.

This method called “frame buffer intercept method” guarantees that the decoder under test and the reference decoder use the same reference frames, and therefore that mismatch does not accumulate. See Figure 1.

Then the samples reconstructed by the reference decoder are captured for each reconstructed picture, and compared to those reconstructed by the decoder under test (previously captured) for the same picture.

This methodology guarantees that there cannot be accumulations of errors, and that the difference observed for each sample only involves one IDCT process.

Figure 1. Frame buffer intercept method

```
[B]---->[S]---> (+) --[C]-------->  [O]
    |            ^
    |            |             Decoder under test
    |            |
    |          [MCP]<--[R]
    |                  [e]
    |                  [f]
    |                  [e]
    |                  [r]
    |                  [e]
    |                  [n]
    |                  [c]
    |          [MCP]<--[e]     Reference Decoder
    |            |
    |            |
    --->[S]---->(+) ---[C]--------> [O]
```

```
B:   test bitstream
S:   decoding processing units ISO/IEC 13818-2 clauses 7.2 to 7.5
MCP: motion compensation unit (ISO/IEC 13818-2 clause 7.6)
R:   reference frame
O:   output of decoder (reconstructed samples)
C:   clipping stage    [0,+255]
U:   current frame
```

Note: R is kept identical in both the Reference and Test Decoders.

2.4.4.5 Implementation of the dynamic test

The dynamic test is often easier to perform on the complete decoder system, which includes a systems decoder, a video decoder and a display process. It is possible to record the output of the display process and to check that display order and timing of fields or frames are correct. However, since the display process is not within the normative scope of ISO/IEC 13818-2, there may be cases where the output of the display process is wrong even though the video decoder is compliant. In this case, the output of the video decoder itself (before the display process) must be captured in order to perform the dynamic tests on the video decoder.

The test includes verifying that the output of the decoding process matches exactly the specification of subclause 7.12 of ISO/IEC 13818-2, both in terms of sequence of events and in terms of timing between events, the evens considered being the output of a reconstructed field or frame by the decoder to the display process.

In particular the field or frame order and timing shall be correct, field parity must be accurate (e.g. the first output field of interlaced frame with top_field_first equals to zero must be the bottom field), and that fields or frames that are coded as being repeated are indeed repeated at the output of the decoding process.

2.4.4.6 Decoder conformance

In order for a decoder of a particular profile-and-level to claim compliance to the standard described by this document, the decoder shall pass successfully both the static test defined in 2.4.5.1 and the dynamic test defined in 2.4.5.2 with all the bitstreams of the normative test suite specified for testing decoders of this particular profile-and-level.

The normative test suites for each profile-and-level combination are defined by Table 2-1. 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. In Table 2-1, “test bitstream directory” is the name of the directory that contains the test bitstream (and associated data) in the electronic annex. Bitstream specification # indicates the test bitstream specification used for each bitstream. Note that at the time of publication of this specification, the test suites of the electronic annex do not contain any bitstream corresponding to specifications #15, #19, #25, #26 and #33.

When the test suite for a profile-and-level combination does not include any bitstream of this same profile-and-level, it is not possible to test adequately compliance to the standard for decoders of that profile-and-level. At this time this standard provides no adequate tests to verify compliance of decoders of the following profile-and-level: HP@HL, HP@H-14, HP@ML, SNR@LL, MP@HL, MP@H-14, MP@LL.

Table 2-1

Bitstream specification #	test bitstream directory	Profile-and-level of the bitstream	Normative test suite for 4:2:2@ML	Normative test suite for SP@ML	Normative test suite for MP@LL	Normative test suite for MP@ML	Normative test suite for MP@H-14	Normative test suite for MP@HL	Normative test suite for SNR@LL	Normative test suite for SNR@ML	Normative test suite for Spt@H-14	Normative test suite for HP@ML	Normative test suite for HP@H-14	Normative test suite for HP@HL
30	tcela/tcela-16-matrices	11172-2	x	x	x	x	x	x	x	x	x	x	x	x
31	tcela/tcela-18-d-pict	11172-2	x	x	x	x	x	x	x	x	x	x	x	x
34	compcore/ccm1	11172-2	x	x	x	x	x	x	x	x	x	x	x	x
32	tcela/tcela-19-wide	11172-2	x	x	x	x	x	x	x	x	x	x	x	x
3	toshiba/toshiba_DPall-0	SP@ML	x	x		x	x	x	x	x	x	x	x	x
3	nokia/nokia6	SP@ML	x	x		x	x	x	x	x	x	x	x	x
3	nokia/nokia_7	SP@ML	x	x		x	x	x	x	x	x	x	x	x
3	tcela/tcela-14-bff-dp	SP@ML	x	x		x	x	x	x	x	x	x	x	x
7	ibm/ibm-bw-v3	SP@ML	x	x		x	x	x	x	x	x	x	x	x
13	tcela/tcela-8-fp-dp	SP@ML	x	x		x	x	x	x	x	x	x	x	x
13	tcela/tcela-9-fp-dp	SP@ML	x	x		x	x	x	x	x	x	x	x	x
16	mei/MEI.stream16v2	SP@ML	x	x		x	x	x	x	x	x	x	x	x
16	mei/MEI.stream16.long	SP@ML	x	x		x	x	x	x	x	x	x	x	x
18	ntr/ntr_skipped_v3	SP@ML	x	x		x	x	x	x	x	x	x	x	x
27	teracom/teracom_vlc4	SP@ML	x	x		x	x	x	x	x	x	x	x	x
28	tcela/tcela-15-stuffing	SP@ML	x	x		x	x	x	x	x	x	x	x	x
29	tcela/tcela-17-dots	SP@ML	x	x		x	x	x	x	x	x	x	x	x
1	gi/gi4	MP@ML	x			x	x	x	x	x	x	x	x	x
1	gi/gi6	MP@ML	x			x	x	x	x	x	x	x	x	x
1	gi/gi_from_tape	MP@ML	x			x	x	x	x	x	x	x	x	x
1	gi/gi7	MP@ML	x			x	x	x	x	x	x	x	x	x
1	gi/gi_9	MP@ML	x			x	x	x	x	x	x	x	x	x
1	ti/TI_c1_2	MP@ML	x			x	x	x	x	x	x	x	x	x
2	tceh/tceh_conf2	MP@ML	x			x	x	x	x	x	x	x	x	x
2	mei/mei.2conftest.4f	MP@ML	x			x	x	x	x	x	x	x	x	x
2	mei/mei.2conftest.60f.new	MP@ML	x			x	x	x	x	x	x	x	x	x
4	tek/Tek-5.2	MP@ML	x			x	x	x	x	x	x	x	x	x

4	tek/Tek-5-long	MP@ML	x			x	x	x	x	x	x	x	x	x
5	tcela/tcela-6-slices	MP@ML	x			x	x	x	x	x	x	x	x	x
5	tcela/tcela-7-slices	MP@ML	x			x	x	x	x	x	x	x	x	x
6	sony/sony-ct1	MP@ML	x			x	x	x	x	x	x	x	x	x
6	sony/sony-ct2	MP@ML	x			x	x	x	x	x	x	x	x	x
6	sony/sony-ct3	MP@ML	x			x	x	x	x	x	x	x	x	x
6	sony/sony-ct4	MP@ML	x			x	x	x	x	x	x	x	x	x
8	att/att_mismatch	MP@ML	x			x	x	x	x	x	x	x	x	x
8	teracom/teracom_vlc4	MP@ML	x			x	x	x	x	x	x	x	x	x
10	ccett/mcp10ccett	MP@ML	x			x	x	x	x	x	x	x	x	x
11	lep/bits_conf_lep_11	MP@ML	x			x	x	x	x	x	x	x	x	x
12	hhi/hhi_burst_short	MP@ML	x			x	x	x	x	x	x	x	x	x
12	hhi/hhi_burst_long	MP@ML	x			x	x	x	x	x	x	x	x	x
14	tcela/tcela-10-killer	MP@ML	x			x	x	x	x	x	x	x	x	x
23	tceh/tceh_conf23.v2	Spt@H-14									x		x	x
24	hhi/hhi_spat23	Spt@H-14									x		x	x
20	tceh/tceh25_conf	SNR@ML								x	x	x	x	x
22	hhi/hhi22_snr	SNR@ML								x	x	x	x	x
21	ti/ti_21	SNR@ML								x	x	x	x	x
2	Tek6-422-bigBpic	4:2:2@ML	x											
2	Tek6-422-bigBpic-long	4:2:2@ML	x											
5	Tek7-422-smallSlices	4:2:2@ML	x											
5	Tek7-422-smallSlices-long	4:2:2@ML	x											
14	Tek8-422-16x8inBpics	4:2:2@ML	x											
14	Tek8-422-16x8inBpics-long	4:2:2@ML	x											
11	Tek9-422-uniformVLC	4:2:2@ML	x											
11	Tek9-422-uniformVLC-long	4:2:2@ML	x											
7	ibm_dp_intra_422	4:2:2@ML	x											
1	sony_422_id01-1	4:2:2@ML	x											
3	sony_422_id03-1	4:2:2@ML	x											
13	sony_422_id13-1	4:2:2@ML	x											
12	hhi_burst_422_short	4:2:2@ML	x											
12	hhi_burst_422_long	4:2:2@ML	x											

Successfully passing the conformance tests defined in this document only provides a strong presumption that the decoder under test is compliant, i.e. that it does indeed meet all the requirements specified in ISO/IEC 13818-2.

Additional tests may be necessary to check more thoroughly that a decoder implements properly all the requirements specified in ISO/IEC 13818-2.

2.4.5 Conformance of scalable bitstreams and decoders

This subclause contains additional information to clarify the compliance assessment procedure for profile-and-level combinations that include any of the scalable video coding methods that are defined in chapters 7.7 to 7.11 of ISO/IEC 13818-2. It is to be seen as a supplement to the preceding part of chapter 2.4 of this specification.

Scalable video coding involves a plurality of video bitstreams forming a scalable hierarchy of bitstreams and the appropriate encoders and decoders to generate and decode them, respectively.

Some profiles-and-level combinations in Chapter 8 of ISO/IEC 13818-2 define requirements for such scalable hierarchies of bitstreams and the associated decoders.

The assessment of the compliance of scalable video bitstream hierarchies and corresponding decoders is generally done as detailed in the preceding part of this chapter 2.4. However a couple of differences, related to the presence of a plurality of bitstreams in the scalable hierarchy must be noted:

The term 'bitstream' now refers to one out of a set of ISO/IEC 13818 video bitstreams forming a scalable hierarchy of bitstreams.

The term 'encoder' now refers to a ISO/IEC 13818 video encoder defined as a process that generates a scalable hierarchy of ISO/IEC 13818 video bitstreams.

The term 'decoder' now refers to a ISO/IEC 13818 video decoder, i.e., an embodiment of the decoding process for decoding a scalable hierarchy of bitstreams as specified by ISO/IEC 13818-2.

2.4.5.1 Definition of scalable video bitstream hierarchy compliance

In a compliant scalable hierarchy of bitstreams each individual bitstream shall be compliant (as defined in 2.4.2) to its profile-and-level as specified in the sequence_extension() of the bitstream. Furthermore, the individual bitstreams of a scalable hierarchy shall meet additional (stricter) constraints defined in Chapter 8 of ISO/IEC 13818-2 for scalable profile-and-level combinations.

2.4.5.1.1 Requirements and restrictions related to profile-and-level

A compliant bitstream with a profile-and-level indication as specified in its sequence_extension() in conjunction with the associated (compliant) lower layer bitstream(s) of this scalable hierarchy shall be decodable by any compliant ISO/IEC 13818 video decoder that supports this profile-and-level combination.

2.4.5.1.2 Encoder requirements and recommendations

2.4.5.1.2.1 Encoder requirements

The requirements detailed in 2.4.2.3 must be met for each individual bitstreams of the scalable hierarchy generated by an encoder.

2.4.5.1.2.2 Encoder recommendations

It is strongly recommended that scalable video encoders capable of producing P-pictures implement Note 2 of subclause 7.4.4 of ISO/IEC 13818-2 in each layer of the ordered set of bitstreams.

It is also strongly recommended that the temporal interval between frames using the same value for temporal_reference be as large as possible, so that ambiguities in the synchronisation of the bitstreams of a scalable hierarchy is unlikely when the hierarchy is not embedded in a systems multiplex according to ISO/IEC 13818-1 (MPEG-2 Systems).

2.4.5.2 Procedure for testing bitstream compliance

When testing the compliance of a bitstream that is member of a scalable hierarchy, the conformance test shall verify that the decoder does not violate the following two sets of constraints.

- Constraints corresponding to the profile-and-level as specified in the sequence_extension() of the bitstream under test.
- Additional constraints for the bitstream under test as given in the definition of the profile-and-level as specified in the sequence_extension() of the other layer bitstreams of the scalable hierarchy.

2.4.5.3 Definition of video decoder compliance

When a decoder claims to be compliant with a given scalable profile-and-level, the embedded decoder(s) that decode the associated lower layer bitstream(s) of a scalable bitstream hierarchy shall pass the conformance test corresponding to its (their) respective profile-and-level combination(s).

Any profile-and-level combination, options and parameter values that are allowed for the lower layer bitstream(s) according to the definition of the given profile-and-level of the decoder under test must be supported.

2.4.5.4 Procedure to test decoder compliance

Tests of the scalable functionalities of a decoder always involve the decoding of a bitstream from a scalable hierarchy including its lower layer bitstream(s), unless the base layer bitstream or any applicable non-scalable test bitstream is the only decoder input for a specific test.

Note that the test of a scalable decoder not only includes decoding of scalable hierarchies of bitstreams but also of non-scalable bitstreams conforming to those profile-and-level indications that must also be decodable by the scalable decoder under test according to Chapter 8.

2.4.5.4.1 Dynamic tests

Dynamic tests of a scalable decoder shall also check that the timing relation between the bitstreams of a scalable hierarchy is correct.

2.4.5.4.2 Specification of the test bitstreams

To test the compliance of a decoder of a profile-and-level allowing scalable coding, it is necessary to generate scalable hierarchies of bitstreams. The test bitstreams of the normative test suites provided for testing scalable profiles are individual video bitstreams. However it is necessary to multiplex these bitstreams according to ISO/IEC 13818-1 (MPEG-2 Systems) to unambiguously convey the necessary timing information to the decoder. This simple exercise is left to the reader who is advised to study carefully ISO/IEC 13818-1 before generating a multiplexed bitstream.

2.4.5.4.3 Implementation of the static test for SNR scalability

In the case of SNR scalability, the frame buffer intercept method as detailed in 2.4.5.4 shall be applied with a base layer and an enhancement layer bitstream instead of one input bitstream B. The decoding process S of the figure in 2.4.5.4 decodes the two bitstreams according to ISO/IEC 13818-2 subclauses 7.2 to 7.5 and 7.8.

2.4.5.4.4 Implementation of the static test for spatial scalability

In case of spatial scalability, the frame buffer intercept method as detailed in 2.4.5.4 shall be applied also to the spatial reference frames (i.e. the output frames of the lower layer decoder).

More precisely this means:

All bitstreams of the scalable hierarchy are decoded by the decoder under test. All the samples reconstructed by the decoder under test, including the samples reconstructed from decoding the lower layer bitstream(s), that are used for spatial prediction, are captured and stored for future use.

The compliance of the embedded lower layer decoder must be tested first, as described in chapter 2.4.5. Assuming a compliant embedded lower layer decoder within the decoder under test, now the upper layer bitstream is decoded by the reference decoder as follows:

Before decoding each P- or B-picture, the frame buffers for temporal prediction reference frames in the reference decoder are initialized with the reconstructed samples captured from the decoder under test that correspond to those reference frames.

Additionally the frame buffer for the spatial prediction reference frame of the reference decoder is initialized from the reconstructed samples corresponding to this reference frame and captured from the embedded lower layer decoder within the decoder under test.

2.4.5.4.5 Implementation of the dynamic test

A dynamic test of a scalable decoder should be done using a complete decoder system, which includes a systems decoder, a video decoder and a display process. This is to assure a proper timing relation between the bitstreams of a scalable hierarchy to be decoded.

2.5 Audio

This subclause describes tests to verify audio bitstream conformance. These tests are applied to audio bitstreams that are known to be free of errors caused by transmission. For each test the condition or conditions that must be satisfied are given, as well as the prerequisites or conditions in which the test can be applied. Note that the application of these tests requires parsing of the bitstream to the appropriate levels.

2.5.1 Audio bitstreams

Audio encoders may apply restrictions to the following parameters of audio bitstreams (see ISO/IEC 13818-3):

2.5.1.1 Extension of ISO/IEC 11172-3 audio coding to lower sampling frequencies

a) layer
b) bitrate_index
c) sampling_frequency
d) mode
e) mode_extension
f) emphasis
g) generation of crc_check
h) value of fixed bitrate when coding in free format mode.
i) generation of ancillary data

2.5.1.2 Low bit rate coding of Multichannel Audio

a) layer
b) bitrate_index
c) sampling_frequency
d) mode in MPEG-1 header
e) center, surround and LFE in MC-header
f) mode extension in MPEG-1 header
g) emphasis
h) generation of mpeg1_error_check
i) value of fixed bitrate when coding in free format mode
j) generation of mpeg1_ancillary data and ext_ancillary_data
k) use and length of extension stream
l) dematrix procedure
m) no_of_multi_lingual_ch
n) multi_lingual_fs
o) multi_lingual_layer
p) n_ad_bytes
q) mc_prediction_on
r) delay_comp
s) prediction order (predsi)

for Layer I, II only

t) tc_sbgr_select
u) dyn_cross_on

for Layer III only

v) seg_list_present

w) dematrix_length, dematrix_select
if seg_list_present, dematrix_length and dematrix_select and the related procedures are not supported the Layer 3 encoder will only be able to produce simulcast MC-bitstreams (tc_present == 1 for all audio channels)

The use of higher order prediction (more than zero-order), and/or delay compensation will limit the editability of the coded bitstreams.

2.5.2 Audio bitstream tests

2.5.2.1 Extension of ISO/IEC 11172-3 audio coding to lower sampling frequencies

2.5.2.1.1 General tests (for all layers)

layer: the Layer field shall not be encoded with the binary value 00.

bitrate: the bitrate field shall not be encoded with the binary value 1111.

sampling_frequency: the sampling frequency field shall not be encoded with the binary value 11.

padding: padding shall be applied such that the accumulated length of the coded audio frames, after a certain number of audio frames, shall not deviate more than (+0,-1) slot from the value specified in 2.4.2.3 of ISO/IEC 11172-3. This shall apply only if the layer, the bitrate and the sampling frequency do not change in the course of the considered audio frames.

emphasis: the emphasis field shall not be encoded with the binary value 10.

protection: if the protection bit is set to ‘0’, then the correct CRC16 value shall be in the crc_check field

ID: the ID flag shall be set to ‘0’.

2.5.2.1.2 Tests on Layer I

allocation: the allocation[sb] or allocation[ch][sb] field shall not be encoded with the binary value 1111.

scalefactor: the scalefactor[sb] or scalefactor[ch][sb] field shall not refer to index 63.

samples: for the coded representation of subband samples the valid range is from zero up to (nlevels - 2), where nlevels equals the number of levels used for quantization of that sample, that is the coded representation of a sample shall not consist of a bitstring with only ‘1’s.

frame length (1): the bit allocation shall be such that the total number of bits for a frame does not exceed the frame length for Layer I.

frame length (2): for Layer 1 the frame length shall equal the number of slots times the slot size for Layer I.

2.5.2.1.3 Tests on Layer II

scalefactor: the scalefactor[sb][p] or scalefactor[ch][sb][p] field shall not refer to index 63

samples: for un-grouped samples the coded representation of subband samples the valid range is from zero up to (nlevels -2), where nlevels equals the number of levels used for quantization of that sample, that is the coded representation of a sample shall not consist of a bitstring with only ‘1’s. For grouped samples the range shall be from zero up to 26 if nlevels equals 3, from zero up to 124 if nlevels equals 5, and from zero up to 728 if nlevels equals 9.

frame length (1): the bit allocation and the scalefactor select information shall be such that the total number of bits for a frame does not exceed the frame length for Layer II.

frame length (2): for Layer II the frame length shall equal the number of slots times the slot size for Layer II.

2.5.2.1.4 Tests on Layer III

part2_3_length: the value encoded in the part2_3_length[gr] or part2_3_length[gr][ch] field shall correspond to the total length of scalefactors and Huffman encoded data.

table_select: the table_select[region][gr] or table_select[region][gr][ch] fields shall be encoded correctly.

frame_length (1): the Huffman code data shall be such that the total number of bits for a frame does not exceed the frame length for Layer III.

frame length (2): for Layer III the frame length shall equal the number of slots times the slot size for Layer III.

buffer control: the value of main_data_begin shall comply with the buffer considerations specified in 2.4.3.4 of ISO/IEC 11172-3.

2.5.2.2 Low bit rate coding of Multichannel Audio

Due to the compatibility of ISO/IEC 13818-3 multichannel audio coding with ISO/IEC 11172-3, ISO/IEC 11172-4 applies to the MPEG-1 part of the bitstream. Furthermore, the following tests apply to an ISO/IEC 13818-3 bitstream.

2.5.2.2.1 General Tests (for all layers)

ID: the ID bit shall be set to ‘1’.

ext_ID_bit: the ext_ID_bit shall be set to ‘0’

Table 2-2 gives an overview of the allowed combinations of mode in the MPEG-1 header and the multichannel options in the mc_header.

Table 2-2

MPEG-1 mode	MPEG-2 multichannel option				
	Center	mono/stereo surround	2nd stereo programme	LFE	multilingual
mono/dual	no	no	yes	no	yes
stereo/joint stereo	yes	yes	yes	yes	yes

dematrix_procedure: The dematrix_procedure ‘10’ may only occur in 3/1 or in 3/2 configuration.

2.5.2.2.2 Tests on Layer I and Layer II

tc_allocation: The following combinations of configuration and tc_allocation are not allowed.

configuration	forbidden tc_allocation
3/1	5, 6, 7
3/0 or 3/0+2/0	3
2/1	3

If phantom coding is used, the combinations of configuration and tc_allocation in the subband groups 10 and 11 are further restricted:

configuration	forbidden tc_allocation
3/2	1, 2, 6, 7
3/1	1, 2, 5, 6, 7
3/0 or 3/0+2/0	1, 2, 3
2/2 and 2/1	not applicable with phantom coding

dyn_cross_mode: The following combinations of configuration and dyn_cross_mode are not allowed.

configuration	forbidden dyn_cross_mode
3/2	15
3/1	5, 6, 7
2/2	5, 6, 7

lf_scalefactor, scalefactor[mch][sb][p],scalefactor[mlch][sb][p]: the lf_scalefactor and scalefactor[mch][sb][p] and scalefactor[mlch][sb][p] fields shall not refer to index 63.

lf_sample[gr], sample[mch,sb,s], sample[mlch,sb,s], samplecode[mch,sb,gr], samplecode[mlch,sb,gr]: for un-grouped samples the coded representation of subband samples the valid range is from zero up to (nlevels -2), where nlevels equals the number of levels used for quantization of that sample, that is the coded representation of a sample shall not consist of a bitstring with only ‘1’s. For grouped samples the range shall be from zero up to 26 if nlevels equals 3, from zero up to 124 if nlevels equals 5, and from zero up to 728 if nlevels equals 9. The LFE samples (lf_sample[gr]) shall not be grouped.

frame length (1): the bit allocation and, in Layer II, the scalefactor select information of the MPEG-1 defined part, and the bit allocation, scalefactor select information, and composite status info of the multichannel extension, and the bit allocation and the scalefactor select information of the multilingual extension, shall be such that the total number of bits for a frame does not exceed the frame length plus the length of the extension frame.

frame length (2): the frame length shall equal the number of slots times the slot size.

frame length (3): the number of bits in the MPEG-1 defined part, plus the number of bits in the MPEG-2 header (‘mc_header()’ field), plus the number of bits in the ‘mpeg1_ancillary_data()’ field, if present, shall not exceed the number of bits in the base frame.

2.5.2.2.3 Tests on Layer III

part2_3_length: the value encoded in the part2_3_length[gr] or part2_3_length[gr][ch] field shall correspond to the total length of scalefactors and Huffman encoded data.

table_select: the table_select[region][gr] or table_select[region][gr][ch] fields shall be encoded correctly.

frame_length (1): the Huffman code data shall be such that the total number of bits for a frame does not exceed the frame length for Layer III.

frame length (2): for Layer III the frame length shall equal the number of slots times the slot size for Layer III.

frame length (3): the number of bits in the MPEG-1 defined part, plus the number of bits in the MPEG-2 header (‘mc_header()’ field), plus the number of bits in the ‘mpeg1_ancillary_data()’ field, if present, shall not exceed the number of bits in the base frame.

buffer control: the value of main_data_begin shall comply with the buffer considerations specified in 2.4.3.4 of ISO/IEC 11172-3.

2.5.3 Audio decoder characteristics

2.5.3.1 Extension of ISO/IEC 11172-3 audio coding to lower sampling frequencies

An ISO/IEC 13818-3 Low Sampling Frequency audio decoder may support only specific values, or a specific range, or a specific combination of values or ranges of the following parameters in audio bitstreams. These parameters are encoded directly or indirectly in the bitstream.

a) layer
b) bitrate_index
c) sampling_frequency
d) mode
e) mode_extension
f) emphasis

Furthermore, an ISO/IEC 13818-3 Low Sampling Frequency audio decoder may constrain the support of free format mode. For an ISO/IEC 13818-3 Low Sampling Frequency audio decoder the handling of ancillary data and error protection (crc_check) shall be specified, as well as the single channel performance (single channel output at one or at both output channels).

Conformance of an audio decoder to ISO/IEC 13818-3 Low Sampling Frequency requires that the output signal of the decoder is reconstructed accurately. For actual tests see 2.5.4.

An ISO/IEC 13818-3 Low Sampling Frequency compliant audio decoder that is able to support, for a certain layer N, where N indicates I, II, or III, at least one but not all combinations of the options defined in 2.5.1.1 such as bit rates, sampling rates and modes, will be designated as an ISO/IEC 13818-3 Low Sampling Frequency Layer N audio decoder. Decoders that support all combinations are designated as Full ISO/IEC 13818-3 Low Sampling Frequency Layer N audio decoders.

2.5.3.2 Low bit rate coding of Multichannel Audio

An ISO/IEC 13818-3 Multichannel audio decoder may support only specific values, or a specific range, or a specific combination of values or ranges of the following parameters in audio bitstreams. These parameters are encoded directly or indirectly in the bitstream.

a) layer
b) bitrate_index
c) sampling_frequency
d) mode in MPEG-1 header
e) center, surround and LFE in MC-header
f) mode extension in MPEG-1 header
g) emphasis
h) generation of mpeg1_ancillary_data and ext_ancillary_data
i) use of extension stream
j) length of extension stream (ext_length)
k) dematrix procedure
l) no_of_multi_lingual_ch

m) multi_lingual_fs
n) multi_lingual_layer
o) mc_prediction_on
p) delay_comp
q) prediction order (predsi)

for Layer I, II only:
r) tc_sbgr_select
s) dyn_cross_on

for Layer III only:
t) seg_list_present
u) dematrix_length, dematrix_select
if seg_list_present, dematrix_length and dematrix_select and the related procedures are not supported the Layer 3 decoder will only be able to decode simulcast MC-bitstreams (tc_present == 1 for all audio channels)

Furthermore, an ISO/IEC 13818-3 Multichannel audio decoder may constrain the support of free format mode. For an ISO/IEC 13818-3 Multichannel audio decoder the handling of ancillary data and error protection (crc_check) shall be specified, as well as the single channel performance (single channel output at one or more output channels).

Conformance of an audio decoder to ISO/IEC 13818-3 Multichannel Audio requires that the output signal of the decoder is reconstructed accurately. For actual tests see 2.5.4.

An ISO/IEC 13818-3 Multichannel compliant audio decoder that is able to support at least one but not all combinations of the options defined in 2.5.1.2 such as bit rates, sampling rates and modes, will be designated as an ISO/IEC 13818-3 Multichannel Layer N audio decoder.

An ISO/IEC 13818-3 Multichannel compliant audio decoder that is able to support at least the features according to table 2-3 will be designated as a Core ISO/IEC 13818-3 Multichannel Layer N decoder or as a Full ISO/IEC 13818-3 Multichannel Layer N decoder respectively.

Table 2-3

	Layer I		Layer II		Layer III	
bitstream characteristic	Core	Full	Core	Full	Core	Full
MPEG-1 Layer I	Y	Y	N	Y	N	Y
MPEG-1 Layer II	N	N	Y	Y	N	Y
MPEG-1 Layer III	N	N	N	N	Y	Y
Configuration 3/2	Y	Y	Y	Y	Y	Y
Configurations 3/1, 3/0, 2/2, 2/1	N	Y	N	Y	N	Y
Configuration 2/0, 1/0	Y	Y	Y	Y	Y	Y
2nd stereo	N	Y	N	Y	N	Y
LFE	N	Y	N	Y	N	Y
free format	N	Y	N	Y	N	Y
extension bitstream	N	Y	N	Y	N	Y
channel switching	Y	Y	Y	Y	Y	Y
dynamic cross-talk/joint coding	Y	Y	Y	Y	Y	Y
prediction zero order	N	Y	N	Y	N	Y
prediction higher order, delay	N	Y	N	Y	-	-
dematrix procedure ‘0’,’1’,’3’	Y	Y	Y	Y	Y	Y
dematrix procedure ‘2’	N	Y	N	Y	N	Y
multilingual Layer II	N	Y	N	Y	N	Y
multilingual Layer II half Fs	N	Y	N	Y	N	Y
multilingual Layer III	N	N	N	N	N	Y
multilingual Layer III half Fs	N	N	N	N	N	Y

2.5.4 Audio decoder tests

To test audio decoders, ISO/IEC JTC 1/SC 29/WG 11 supplies a number of test sequences. Supplied sequences cover Full Layer N Decoders and Core Layer N decoders. For a supplied test sequence, testing can be done by comparing the output of a decoder under test with a reference output also supplied by ISO/IEC JTC 1/SC 29/WG 11. Measurements are carried out relative to full scale where the output signals of the decoders are normalised to be in the range between -1 and +1.

To be called an ISO/IEC 13818-3 audio decoder, the decoder shall provide an output such that the RMS level of the difference signal between the output of the decoder under test and the supplied reference output is less than 2^{-15}/sqrt(12) for the supplied sine sweep (20Hz-10kHz) with an amplitude of -20dB relative to full scale. In addition, the difference signal shall have a maximum absolute value of at most 2^{-14} relative to full-scale.

To be called a limited accuracy ISO/IEC 13818-3 audio decoder, the decoder shall provide an output for a provided test sequence such that the RMS level of the difference signal between the output of the decoder under test and the supplied reference output is less than 2^{-11}/(sqrt(12) for the supplied sine sweep (20Hz-10kHz) with an amplitude of -20dB relative to full scale.

The above two tests only verify the computational accuracy of an implementation.

2.5.4.1 Calculation for RMS

All measurements are carried out relative to full scale where the output signals of the decoder and supplied test sequences are normalised to be in the range between -1,0 and +1,0. The supplied sine sweep with an amplitude of -20dB relative to full scale has an absolute amplitude of ±0,1. This test sequence has a precision (P) of 24 bit. i.e. the MSB represents the value of -1, the MSB-1 bit represents the value of +1/2, etc.

MSB	-1/2^0	= -1
MSB-1	1/2^1	= ½
MSB-2	1/2^2	= 1/4
MSB-23	1/2^23	= 1/8 388 608.

The output signal of the decoder under test requires to be in the same format. In the case that the output of the decoder has a precision of P' bits and if P' is smaller than 24, then the values for the bits between the positions P'-1 and 24 shall be set to zero.

MSB	-1/2^0	= -1
MSB-1	1/2^1	= 1/2
...		
MSB-(P'-1)	1/2^P'	= 1/2^P'
MSB-P'	0	= 0
...		
MSB-23	0	= 0

In the next step the difference (diff) of the samples of these signals has to be calculated. If two channels are present (in case of stereo, joint-stereo and dual-channel) both channels shall be tested. The total number of samples for each channel is N.

diff(n) = 'output signal of decoder under test(n)' - 'supplied sine sweep(n)'

for n = 1 to N

The values of all difference samples shall be squared, summed, divided by N and then the square-root shall be calculated. This calculation finally gives the RMS level.

RMS = sqrt(1/N*sum(diff^2))

The decoder under test may be called an ISO/IEC 13818-3 audio decoder, if the RMS is less than 1/(2^15 * 12^0,5) and if the maximum absolute value is less than or equal to 1/2^14.

The decoder under test may be called a limited accuracy ISO/IEC 13818-3 audio decoder, if the RMS is less than 1/(2^11 * 12^0,5).

2.5.4.2 Descriptions of the audio test bitstreams

Compressed bitstreams are provided, according to ISO/IEC 13818-3. Detailed descriptions of the bitstreams are furnished below. The following file name extensions are used to identify different parts

testXX.mpg: MPEG-2 audio bitstream
testXX.ext: MPEG-2 audio extension bitstream (optional)
testXX.txt: description file

Where XX stands for the bitstream number as indicated in tables 2-4 through 2-7.

Table 2-4

feature	Bitstream number									
	01	02	03	04	05	06	07	08	09	10
Layer	II	II	II	II	II	II	II	II	II	II
matrix	3	0	1	0	3	3	0	0	0	0
extension	N	N	N	N	N	N	Y	Y	N	Y
transmission channel allocation	-	Y	Y	Y	-	-	Y	Y	Y	Y
phantom centre	N	N	N	Y	N	N	N	N	N	N
prediction	N	N	N	N	N	N	N	N	Y (no delay)	N
Dynamic crosstalk	N	N	N	N	N	N	N	Y	N	N
sampling frequency	44.1	48	32	48	44.1	48	48	48	48	48
LFE	Y	N	N	N	N	N	N	N	N	N
multilingual	N	N	N	N	N	N	N	N	N	7
multilingual sampling frequency	-	-	-	-	-	-	-	-	-	48
ancillary data	N	N	N	N	Y	N	N	N	N	N
bitrate base stream	384	384	320	256	256	384	384	256	384	384
bitrate extension stream	-	-	-	-	-	-	128	256	-	MAX
configuration	3/2	3/2	3/1	3/0	2/1	2/0 +2/0	3/0 +2/0	3/2	3/2	3/2

Table 2-5

Feature	bitstream number									
	12	13	14	15	16	17	18	19	20	21
Layer	II	II	II	II	II	II	II	II	II	II
matrix	0	0	3	3	3	1	0	0	0	0
extension	Y	Y	N	N	N	Y	N	N	Y	Y
transmission channel allocation	Y	Y	-	-	-	Y	Y	Y	Y	Y
phantom centre	N	Y	N	N	N	N	N	N	N	N
prediction	N	Y (no delay)	N	Y	Y	N	N	N	N	N
Dynamic crosstalk	N	Y	N	N	N	N	Y	Y	Y	Y
sampling frequency	48	48	44.1	48	48	48	48	48	44.1	44.1
LFE	N	Y	N	N	N	Y	N	N	N	N
multilingual	7	N	N	N	N	N	N	N	N	N
multilingual sampling frequency	24	-	-	-	-	-	-	-	-	-
ancillary data	N	Y	Y	N	N	N	N	N	N	N
bitrate base stream	384	384	384	384	384	384	384	384	VBR	VBR
bitrate extension stream	384	MAX	-	-	-	256	-	-	VBR	VBR
configuration	3/2	3/2	3/2	3/2	3/2	3/2	3/1	2/1	3/2	3/2

Note for bitstreams 20:

Variable bit rate. Note that not all decoders are required to support this feature.

Note for bitstreams 35:

Bitstream intended for accuracy test. Output represented with 24 bit resolution.

Table 2-6

Feature	bitstream number		
	22	23	35
Layer	I	III	II
matrix	0	0	3
extension	Y	Y	N
transmission channel allocation	Y	N/A	-
phantom centre	N	Y	N
prediction	N	N	N
Dynamic crosstalk	N	N/A	N
sampling frequency	44.1	48	44.1
LFE	N	N	N
multilingual	N	N	N
multilingual sampling frequency	-	-	-
ancillary data	N	N	N
bitrate base stream	384	320	384
bitrate extension stream	384	320	-
configuration	3/2	3/2	3/2
dematrixing	N/A	Y	N/A
segment lists	N/A	Y	N/A
matrix attenuation	N/A	N	N/A
scale factor select info	N/A	N	N/A

Note for bitstreams 35:

Bitstream intended for accuracy test. Output represented with 24 bit resolution.

Table 2-7

Feature	Bitstream number				
	24-26	27 - 29	30 - 31	33	34
Layer	II	II	II	I	II
sampling frequency	16, 22,05, 24	24	24	22,05	24
bitrate	96	16, 96, 160	96	128	160
mode	Joint Stereo	Joint Stereo	Single Channel, Dual Channel	Joint Stereo	Stereo
ancillary data	N	N	N	N	N

Note for bitstreams 34:

Bitstream intended for accuracy test. Output represented with 24 bit resolution.

Annex A
(informative)
Patent statements

The user's attention is called to the possibility that, for some of the processes specified in this part of ISO/IEC 13818, conformance with this specification may require use of an invention covered by patent rights.

By publication of this part of ISO/IEC 13818, no position is taken with respect to the validity of this claim or of any patent rights in connection therewith. However, each company listed in this Annex has undertaken to file with the Information Technology Task Force (ITTF) a statement of willingness to grant a license under such rights that they hold on reasonable and non-discriminatory terms and conditions to applicants desiring to obtain such a license.

Information regarding such patents can be obtained from the following organisations.

The table summarises the formal patent statements received and indicates the parts of the standard to which the statement applies. Three "N"s in the row corresponding to a company mean that the statement from the company did not mention any part. The list includes all organisations that have submitted informal patent statements. However, if no "X" is present, no formal patent statement has yet been received from that organisation.

Table A-1

Company	**ISO/IEC 13818-2**	**ISO/IEC 13818-3**	**ISO/IEC 13818-1**
AT&T	X	X	X
BBC Research Department	X		
Belgian Science Policy Office	N	N	N
Bellcore	X		
BOSCH	X	X	X
CCETT			
Columbia University in the City of New York	N	N	N
Compression Labs, Inc.	N	N	N
CSELT	X		
David Sarnoff Research Center	X	X	X
Deutsche Thomson-Brandt GmbH	N	N	N
France Telecom CNET	N	N	N
Fraunhofer Gesellschaft	N	N	N
Fujitsu Limited	X		
GC Technology Corporation	N	N	N
General Instruments	N	N	N
Goldstar Co., Ltd.	N	N	N
Hitachi, Ltd.	N	N	N
International Business Machines Corporation	X	X	X
IRT		X	
KDD Co., Ltd.	X		
Massachusetts Institute of Technology	X	X	X
Matsushita Electric Industrial Co., Ltd.	N	N	N
Mitsubishi Electric Corporation	N	N	N
National Transcommunications Limited	N	N	N
NEC Corporation	N	N	N
Nippon Hoso Kyokai	X		
Nippon Telegraph and Telephone Corporation	N	N	N
Nokia Corporation	X		
Norwegian Telecom	X		
OKI Electric Industry Co., Ltd.	N	N	N
Philips Electronics N.V.	N	N	N
Qualcomm Incorporated	X		
Royal PTT Nederland N.V., PTT Research (NL)			
Samsung Electronics Co., Ltd.	X	X	X
Scientific-Atlanta, Inc.	X	X	X
Sharp Corporation	N	N	N
Siemens AG	N	N	N
Sony Corporation	N	N	N
Texas Instruments Incorporated	N	N	N
Thomson Consumer Electronics	N	N	N
Toshiba Corporation	X		
TV/COM International	X	X	X
Victor Company of Japan Limited	X	X	X

Annex B
(informative)
Bibliography

1 Arun N. Netravali & Barry G. Haskell "Digital Pictures, representation and compression" Plenum Press, 1988

2 Didier Le Gall "MPEG: A Video Compression Standard for Multimedia Applications" Trans. ACM, April 1991

3 C Loeffler, A Ligtenberg, G S Moschytz "Practical fast 1-D DCT algorithms with 11 multiplications" Proceedings IEEE ICASSP-89, Vol. 2, pp 988-991, Feb. 1989

4 See the Normative Reference for ITU-R Rec 601 (formerly CCIR Rec 601)

5 See the Normative Reference for IEC Standard Publication 461

6 See the Normative Reference for ITU-T Rec. H.261

7 See the Normative reference for IEEE Standard Specification P1180-1990

8 ISO/IEC 10918-1 | ITU-T T.81 (JPEG)

9 E Viscito and C Gonzales "A Video Compression Algorithm with Adaptive Bit Allocation and Quantization", Proc SPIE Visual Communications and Image Proc '91 Boston MA November 10-15 Vol. 1605 205, 1991

10 A Puri and R Aravind "Motion Compensated Video Coding with Adaptive Perceptual Quantization", IEEE Trans. on Circuits and Systems for Video Technology, Vol. 1 pp 351 Dec. 1991.

11 C. Gonzales and E. Viscito, "Flexibly scalable digital video coding". Image Communications, Vol. 5, Nos. 1-2, February 1993

12 A.W.Johnson, T.Sikora and T.K. Tan, "Filters for Drift Reduction in Frequency Scalable Video Coding Schemes" <*Transmitted for publication to Electronic Letters.*>

13 R.Mokry and D.Anastassiou, "Minimal Error Drift in Frequency Scalability for Motion-Compensated DCT Coding". IEEE Transactions on Circuits and Systems for Video Technology, <*accepted for publication*>

14 K.N. Ngan, J. Arnold, T. Sikora, T.K. Tan and A.W. Johnson. "Frequency Scalability Experiments for MPEG-2 Standard". Asia-Pacific Conference on Communications, Korea, August 1993.

15 T. Sikora, T.K. Tan and K.N. Ngan, "A Performance Comparison of Frequency Domain Pyramid Scalable Coding Schemes Within the MPEG Framework". Proc. PCS, Picture Coding Symposium, Lausanne, pp. 16.1 - 16.2, Switzerland March 1993.

16 Masahiro Iwahashi, “Motion Compensation Technique for 2:1 Scaled-down Moving Pictures”. 8-14, Picture Coding Symposium '93.

17 Sikora, T. and Pang, K., "Experiments with Optimal Block-Overlapping Filters for Cell Loss Concealment in Packet Video", Proc. IEEE Visual Signal Processing and Communications Workshop, Melbourne, 21-22 Sept. 1993, pp. 247-250.

18 A. Puri "Video Coding Using the MPEG-2 Compression Standard", <to appear> Proc SPIE Visual Communications and Image Proc '93 Boston MA November, 1993.

19 A. Puri and A. Wong "Spatial Domain Resolution Scalable Video Coding", <to appear> Proc SPIE Visual Communications and Image Proc '93 Boston MA November, 1993.

Annex C
(informative)
Systems test bitstreams

This annex lists the bitstream descriptions and bitstream suites that can be used to test decoder compliance.

Not all the tests listed in subclauses 2.3.2 and 2.3.3 are covered by these bitstreams, however, some of the fundamental decoder requirements are believed to be covered by these test suites. A compliant decoder is expected to maintain correct audio/visual synchronization while decoding these bitstreams. The bitstreams listed in Table C-1 are provided in an electronic annex that forms an integral part of this specification.

Table C-1

Test bitstream description	**Test Bitstream Directory**
decoding discontinuities (subclause 2.3.3.1.1)	philips/syncho.bin
time base discontinuities (subclause 2.3.3.1.2)	gi/pcr_discon_r1.tp sa/sa.m2t.1 philips/syncho.bin
continuity counter discontinuities (subclause 2.3.3.1.1)	sa/sa.m2t.1 sa/sa.m2t.2
change in picture size / resolution (subclause 2.3.3.1.1)	philips/syncho.bin
change in audio bitrate (subclause 2.3.3.1.1	philips/syncho.bin
duplicate packets & transport_error_indicator = 1 (subclause D.2)	gi/dup_err_pkt.tp
TBn nearly overflow (subclause 2.3.2.4)	
Bn nearly overflow / underflow (subclause 2.3.2.4)	philips/syncho.bin
MBn, EBn nearly overflow / underflow (subclause 2.3.2.4)	divicom/overflow_EB.m2t divicom/overflow_MB.m2t philips/syncho.bin divicom/underflow_EB.m2t
Use of all private options (subclause 2.3.3.6)	gi/priv_data.tp
Trick modes (subclause 2.3.3.7)	philips/sm4.ts
Change in program definition (subclause 2.3.3.5.1)	gi/psi_change_r1.tp
Transport stream with scalable video	hhi/TPhhi_bits.conf
Use of optional fields	
Data of (nearly) 3 time bases in STD	
Reserved fields containing 0’s	

Annex D
(informative)
Systems decoder characteristics beyond conformance

D.1 Number of PIDs that can be processed

A compliant decoder must be able to demultiplex and process all PIDs carrying information related to the program it is receiving. These include PID 0 (which carries the PAT), the PMT_PID, and at least one PID for the service data. The PCR does not necessarily have a different PID, but if it does then the PCR_PID must also be demultiplexed. If a program with video and audio elementary streams is being decoded, at least four PIDs must be demultiplexed. The following table shows the number of PIDs which it may be necessary for a decoder to demultiplex in order to receive a single program. For some applications it may not be necessary for the decoder to process all these PIDs simultaneously.

DATA IN PID	**NUMBER OF PIDS**
PAT (PID 0)	1
PMT_PID	1
PCR_PID	0 - 1
Video	0 - V
Audio	0 - A
CA (PID 1)	0 - 1
EMM	0 - 1
ECM	0 - C
NIT	0 - 1
Other	0 - N

The values V, A, C, and N are the maximum number of PIDs carrying the corresponding data in a single program.

D.2 Error handling

For error conditions that do not result in unacceptable decoding artifacts, a decoder should:

- maintain the system time base in the event of errored or missing packets that carry PCRs
- maintain its program information database in the event of receiving PSI sections with errors that are detectable using the CRC-32 fields.

A decoder should detect and process error conditions which are indicated by one or more of the following:

- transport_error_indicator = 1
- continuity_counter errors
- CRC-32 errors in PSI sections.

D.3 Program acquisition

A decoder manufacturer should provide the following information on program acquisition:

- The number of programs directly accessible. The size of the memory dedicated to the storage of the PMT and PAT tables should also be provided.
- The maximum time during which perceivable effects due to decoder STC pull-in to the received PCRs may occur. Ideally, there should be no perceivable effects.

D.4 Input processing capabilities

An ISO/IEC 13818-1 Transport Stream or Program Stream decoder may support specific values only, or a specific range of input rates. The STD buffer sizes will be constrained and in case of a Program Stream, there will be a constraint on the PES Packet rate.

A Program Stream decoder may constrain the support of fixed and/or variable bitrate operation (see definition of the fixed_flag field in subclause 2.5.3.6 of ISO/IEC 13818-1), and may require locking between the 27 MHz system clock and the audio sampling frequency and/or the video picture rate (see the system_audio_lock_flag and the system_video_lock_flag fields in subclause 2.5.3.6 of ISO/IEC 13818-1).

D.5 Presentation Timing

The delay due to the implementation of the decoder must be accounted for when calculating the actual presentation time from the PTS. The maximum should not exceed 100 ms.