INTERNATIONAL STANDARD

ISO/IEC 21122-2

First edition
2019-07

Information technology — JPEG XS low-latency lightweight image coding system —

Part 2:
Profiles and buffer models

Reference number
ISO/IEC 21122-2:2019(E)

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Fax: +41 22 749 09 47
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Contents

Page

Foreword

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

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

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

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

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

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

A list of all parts in the ISO/IEC 21122 series can be found on the ISO website.

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

Introduction

ISO/IEC 21122-1 (JPEG XS) specifies a single syntax designed to serve a wide range of applications, bit rates, resolutions, qualities, and services. Its main target applications are video transport over video links and IP networks, real-time video storage, video memory buffer, omni-directional video capture system, head-mounted displays for virtual or augmented reality and sensor compression for the automotive industry. These applications have different requirements in terms of complexity, latency and compression efficiency. Even within a given application field, different requirements are usually identified depending on the targeted use case.

Considering the impracticality of implementing the full syntax of ISO/IEC 21122-1, and in order to meet the requirements of the different target applications while safeguarding as much as possible the interoperability enabled by the common syntax defined in ISO/IEC 21122-1, a limited number of subsets of this syntax are stipulated by means of "profiles", "levels", and "sublevels".

The coding tools specified in ISO/IEC 21122-1 allow encoder and decoder implementations to limit the end-to-end latency to a fraction of the frame size. To ensure this property, this document specifies a buffer model, consisting of a decoder model and a transmission channel model.

Information technology — JPEG XS low-latency lightweight image coding system —

Part 2:
Profiles and buffer models

1 Scope

This document defines a limited number of subsets of the syntax specified in ISO/IEC 21122-1 and a buffer model to ensure interoperability between implementations in the presence of a latency constraint.

2 Normative references

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

ISO/IEC 21122-1, *JPEG XS low-latency lightweight image coding system — Part 1: Core coding system*

3 Terms, definitions, symbols and abbreviated terms

3.1 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 21122-1 and the following apply.

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

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

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

3.1.1
blanking codestream fragment
placeholder *codestream fragment* (3.1.8) representing blanking periods

3.1.2
horizontal blanking period
timespan expressed in units of the grid point sampling rate between the last *pixel* (3.1.22) of an image line — not being the last line of an image — and the first pixel of the next image line

3.1.3
vertical blanking period
timespan in units of the grid point sampling rate between the last line of an image [including the *horizontal blanking periods* (3.1.2)] and the first line of the next image

3.1.4
buffer model
combination of a *decoder model* (3.1.12) and a *channel model* (3.1.6) whose behaviour can be defined by a set of parameters

3.1.5
buffer model instance
specific configuration of a *buffer model* (3.1.4) specified by the assignment of well-defined values to the buffer model parameters

3.1.6
channel model
model describing the temporal behaviour of the *transmission channel* (3.1.29) connecting an encoder and a decoder

3.1.7
coded codestream fragment
continuous sequence of bits in the codestream containing exactly one packet body and a well-defined number of packet headers, markers and marker segments

3.1.8
codestream fragment
either *coded codestream fragment* (3.1.7), or *blanking codestream fragment* (3.1.1)

3.1.9
code group
group of quantization indices in sign-magnitude representation representing a quantized *coefficient group* (3.1.10)

3.1.10
coefficient group
number of horizontally adjacent wavelet coefficients from the same band and precinct

3.1.11
cycle
single clock period of an encoder or decoder clocked implementation

3.1.12
decoder model
combination of a *decoder unit* (3.1.14) and a *decoder smoothing buffer* (3.1.13)

3.1.13
decoder smoothing buffer
memory buffer that is used to level out changes in the number of bits read by a *decoder unit* (3.1.14) per time unit

3.1.14
decoder unit
module reading a variable number of bits per time unit to generate decoded output *pixels* (3.1.22) with a fixed rate

3.1.15
decomposition level
set of wavelet coefficients resulting from a particular *level* (3.1.21) of recursive application of a wavelet transform

3.1.16
encoder model
combination of an *encoder unit* (3.1.18) and an *encoder smoothing buffer* (3.1.17)

3.1.17
encoder smoothing buffer
memory buffer that is used to level out changes in the number of bits generated by an *encoder unit* (3.1.18) per time unit

3.1.18
encoder unit
module transforming a sequence of input *pixels* (3.1.22) with constant rate into a conforming codestream, producing a bit sequence with variable number of bits generated per time unit

3.1.19
fill level
number of bits stored in the encoder or *decoder smoothing buffer* (3.1.13)

3.1.20
nominal bits per pixel value
mean number of bits allocated per encoded *pixel* (3.1.22) which is used to derive the *sublevel* (3.1.28) constraints by assuming an image with well-defined dimensions and frame rate derived from the *level* (3.1.21)

3.1.21
level
defined set of constraints on the amount of decoded *sampling grid points* (3.1.25) to be processed by an encoder or decoder, both in the spatial and time dimensions

Note 1 to entry: The same set of levels is defined for all profiles. Individual implementations may, within the specified constraints, support a different level for each supported profile.

3.1.22
pixel
position in the *sample grid* (3.1.24) that is populated by a sample value of at least one component

3.1.23
profile
specified subset of the codestream syntax together with admissible parameter values

3.1.24
sample grid
abstract coordinate system on which image sample values are positioned

3.1.25
sampling grid point
position on the *sample grid* (3.1.24), specified by integer horizontal and vertical offset relative to the origin of the sample grid

3.1.26
smoothing buffer unit
level (3.1.21) and *sublevel* (3.1.28) dependent number of bits by which the smoothing buffer size of the *decoder model* (3.1.12) is specified

3.1.27
start of transmission
SoT
time at which the *transmission channel* (3.1.29) starts transmission relative to the start of encoding of the first *codestream fragment* (3.1.8) of a codestream

3.1.28
sublevel
defined set of constraints on the amount of codestream bits to be processed by an encoder or decoder, per unit of time, per column, and per image

Note 1 to entry: The same set of sublevels is defined for all profiles. Individual implementations may, within the specified constraints, support a different sublevel for each supported profile.

3.1.29
transmission channel
facility transferring bits from a source entity to a target entity

3.1.30
transmission channel capacity
maximum number of bits per time unit that a *transmission channel* (3.1.29) can transfer from a source entity to a target entityConventions

3.2 Conformance language

ISO/IEC's use of verbal forms is detailed at:

https://www.iso.org/foreword-supplementary-information.html

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

3.3 Operators

NOTE Many of the operators used in document are similar to those used in the C programming language.

3.3.1 Arithmetic operators

+ addition

− subtraction (as a binary operator) or negation (as a unary prefix operator)

× multiplication

/ division without truncation or rounding

3.3.2 Logical operators

|| logical OR

&& logical AND

! logical NOT

3.3.3 Relational operators

> greater than

≥ greater than or equal to

< less than

≤ less than or equal to

== equal to

!= not equal to

3.3.4 Precedence order of operators

Operators are listed in descending order of precedence. If several operators appear in the same line, they have equal precedence. When several operators of equal precedence appear at the same level in an expression, evaluation proceeds according to the associativity of the operator either from right to left or from left to right.

Operators	Type of operation	Associativity
()	expression	left to right
[]	indexing of arrays	left to right
–	unary negation	
×, /	multiplication, division	left to right
+, –	addition and subtraction	left to right
< , >, ≤, ≥	relational	left to right
&	bitwise AND	left to right
|	bitwise OR	left to right

3.3.5 Mathematical functions

$\lceil x \rceil$	ceil of x: returns the smallest integer that is greater than or equal to x
$\lfloor x \rfloor$	floor of x: returns the largest integer that is less than or equal to x
$\lvert x \rvert$	absolute value of x, $\lvert x \rvert$ equals −x for x < 0, otherwise x
sign(x)	sign of x, 0 if x is 0, +1 if x is positive, −1 if x is negative
$\xi(t)$	step function $\xi(t) = \begin{cases} 1 & t \geq 0 \\ 0 & \text{otherwise} \end{cases}$
$\max_i(x_i)$	maximum of a sequence of numbers $[x_i]$ enumerated by the index i

4 Specifications

4.1 Symbols

$A = [a_1, a_2, \ldots, a_n]$	sequence of elements $a_1 - a_n$
A||B	concatenation of two sequences A and B
C(i)	codestream *i*
D_{c2d}	number of clock cycles between the first bit written into the decoding smoothing buffer and the decoding start of the first codestream fragment of a stream of codestream fragments
$F_{first}(C(i))$	first codestream fragment of codestream C(i)
$F_{last}(C(i))$	last codestream fragment of codestream C(i)
H_f	height of the image in sampling grid points

H_{max} maximum picture height in sampling grid points

L_{max} maximum number of sampling grid points per image

$l_{enc}(t)$ fill level of the encoding smoothing buffer in bits at the end of cycle t

$l_{dec}(t)$ fill level of the decoding smoothing buffer in bits at the end of cycle t

$l_{enc,max}$ capacity in bits of the encoding smoothing buffer

$l_{dec,max}$ capacity in bits of the decoding smoothing buffer

$\tilde{l}_{dec}(t)$ number of bits that can be read from the decoding smoothing buffer in cycle t

$l_{sum}(t)$ sum of encoder and decoder smoothing buffer fill level in bits at cycle t

$\mathbb{N}$ all integer numbers being strictly larger than zero

$\mathbb{N}_0$ all integer numbers being greater than or equal to zero

$N_{b,x}$ size of the horizontal blanking line in sampling grid point clock periods

$N_{b,y}$ size of the vertical blanking period in sampling grid lines

N_{bpp} nominal number of bits allocated per pixel for compression

N_c number of components in an image

$N_{cg}(f)$ number of coefficient groups within codestream fragment f

$N_{cg,hz}$ number of coefficient groups associated to a codestream fragment representing a horizontal blanking period

$N_{cg,vt}$ number of coefficient groups associated to a codestream fragment representing a vertical blanking period

$N_f(i)$ number of codestream fragments within a codestream i

N_g number of coefficients in a code group

$N_{L,y}$ number of vertical decomposition levels

$N_{p,x}$ number of precincts per sampling grid line

$N_{p,y}$ number of precincts per sampling grid column

N_{sbu} number of decoder smoothing buffer units for a given profile

$\mathbb{Q}$ set of rational numbers

$r_{dec}(t)$ number of bits read and removed from the decoder smoothing buffer in clock cycle t

R_{trans} transmission channel capacity, expressed in bits per cycle (having a duration of T)

$R_{t,max}(l_m, l_s)$ maximum admissible encoded throughput in bits per second for a given level

$R_{s,max}$ max grid point sample rate (in samples per second) at decoder output

$S_{bits}(f)$ number of bits forming the codestream fragment f

$S_{c,max}$	targeted maximum number of bytes of an encoded codestream
$S_{sbu}(l_m, l_s)$	size of the smoothing buffer unit in bytes for level l_m and sublevel l_s
$S_{sbo}(p)$	smoothing buffer increment in bits for a profile p
$S_{sl,max}(l_m, l_s)$	maximum size of an encoded codestream in bytes of level l_m and sublevel l_s
$s_x[i]$	subsampling factor of component i in horizontal direction
$s_y[i]$	subsampling factor of component i in vertical direction
T_{enc}	clock period defining the frequency by which code groups are processed by an encoder
T_{dec}	clock period defining the frequency by which code groups are processed by a decoder
$t_{enc,write}(f)$	timestamp in cycles at which the codestream fragment f is written to the encoder smoothing buffer
$t_{dec,start}(f)$	timestamp in cycles at which decoder starts decoding codestream fragment f
$t_{dec,read}(f)$	timestamp in cycles at which codestream fragment f is removed from the decoder smoothing buffer
Tbmd	buffer model type
$W_c[i]$	width of component i in samples
$W_{c,max}$	maximum column width in sampling grid points for a given profile
$w_{dec}(t)$	number of bits written into the decoder smoothing buffer in clock cycle t
W_f	width of the image in sampling grid points
W_{max}	maximum picture width in sampling grid points

4.2 Abbreviated terms

bpp	bits per pixel
DWT	discrete wavelet transform
IDWT	inverse discrete wavelet transform
RCT	reversible colour transform
IRCT	inverse reversible colour transform

4.3 General provisions

For a concrete application, only a subset of the codestream syntax specified in ISO/IEC 21122-1 is needed. Profiles as specified in 3.1.23 define corresponding interoperability points for those applications. In addition to profiles, levels (as specified in 3.1.21) and sublevels (specified in 3.1.28) limit the maximum throughput in the encoded (codestream) and decoded (pixel, spatial) domain. This allows creating cost-efficient implementations serving the needs of the desired applications.

Profiles, levels and sublevels shall be as specified in Annex A.

Keeping the end-to-end latency of an encoding-decoding chain under a given threshold is one of the main goals pursued by the methods defined in ISO/IEC 21122-1. To reach this goal, the definition of buffer models is necessary, consisting of a decoder model and a transmission channel model. The interaction of a hypothetical reference decoder including its decoding smoothing buffer with a constant bitrate channel feeding this buffer shall be as specified in Annexes B and C. The size of this decoding smoothing buffer is specified in Annex A. Codestreams shall be formed such that this decoding smoothing buffer never overflows or underflows.

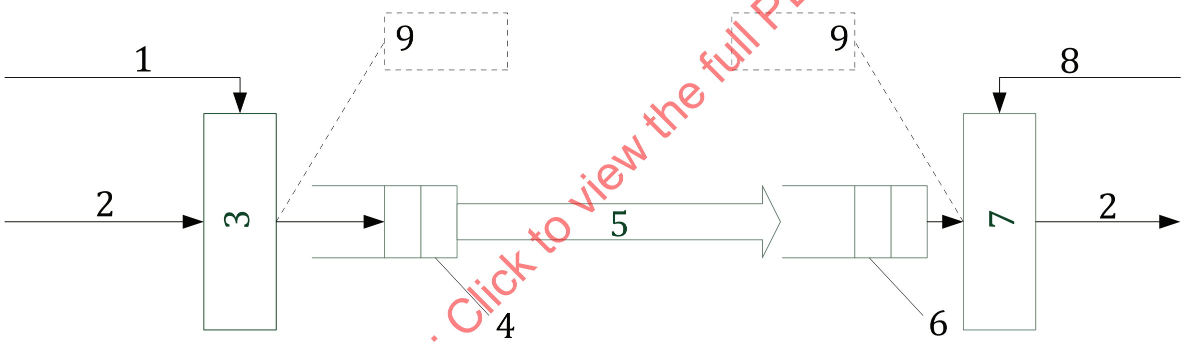
Buffer models are further discussed in Annex D. The buffer model provides encoders with the necessary information to generate codestreams that can be decoded by an arbitrary decoder implementation ensuring system interoperability.

In addition to the size of the decoder smoothing buffer, end-to-end latency also depends on the latency inherent to each processing step of the encoding-decoding chain whose methods are described in ISO/IEC 21122-1. To help implementers estimate the latency of their device, Annex E gives useful information on the minimum latency that can be achieved by the different methods described in ISO/IEC 21122-1.

5 Buffer model

5.1 General system block diagram

The JPEG XS coding system addresses applications where coded images are transferred from a source to a target, as shown in Figure 1. To this end, the encoder is compressing a continuous stream of input pixels into a sequence of bits. These bits are forwarded by means of a transmission channel to the decoder that decompresses the bits to produce a continuous stream of output pixels.



Key

1 encoder clock
2 pixel data
3 encoder unit
4 encoder smoothing buffer
5 transmission channel
6 decoder smoothing buffer
7 decoder unit
8 decoder clock
9 variable bit rate

Figure 1 — General system block diagram

The time instances at which the encoder has to process each pixel are determined by an encoding clock. Similarly, the time instances at which the decoder has to produce each output pixel are determined by a decoding clock. Both clocks are generated by the system.

NOTE In implementations, these clocks can be the same or differ in both frequency and phase. The presented model is independent of whether clocks are synchronized or not.

In accordance with ISO/IEC 21122-1, the pixels of an image are translated into coefficient groups represented as code groups in the codestream. The number of bits necessary to code these code groups may vary from group to group. As a consequence, the encoder writes encoded bits at a variable rate into the encoder smoothing buffer. Similarly, the decoder reads the codestream at a variable rate from the decoder smoothing buffer.

In case the maximum bit rate of the transmission channel is below the peak bit rate generated by the encoder, an encoder smoothing buffer is necessary to decouple generation of bits by the encoder from transmission of bits over the transmission channel. Similarly, a decoder smoothing buffer needs to be provided that decouples the arrival of bits at the rate afforded by the transmission channel and the consumption of bits by the decoder per clock.

Correct operation requires that the decoder buffer never overflows. This is because the decoder is not able to pause the arrival of bits from the transmission channel. Moreover, a buffer underflow in the decoder buffer needs to be avoided. This is because the decoder is required to output pixels in accordance with the timing of its output interface. Hence it needs to be ensured that the bits to be read from the decoding buffer to produce the next pixel in accordance with the decoding clock are available in this decoding buffer.

5.2 Influencing variables on the required buffer sizes

Avoiding any buffer overflow or underflow, as discussed in subclause 5.1, requires properly sizing the decoder smoothing buffer. Moreover, the time at which decoding starts is delayed relative to the starting time of encoding and the start of transmission needs to be carefully set. Those values are influenced by many system parameters, for example:

— The maximum transmission channel bit rate.

— The granularity at which the encoder writes the encoded data and the decoder reads the encoded data.

— The rate control strategy applied by the encoder.

These dependencies cause that encoders and decoders are only interoperable in well-defined conditions. Defining these conditions is the purpose of the buffer model defined in Annex B and Annex C

5.3 Role of the buffer model

The core coding system defined in ISO/IEC 21122-1 can be implemented on a large variety of platforms using many different implementation strategies. Thus, interoperability cannot be achieved by precisely specifying the temporal behaviour of a conforming decoding implementation. Instead, the buffer model defines a simplified decoder model. Interoperability is then achieved by mandating that a conforming decoder shall decode all bit streams being decodable by the simplified decoder model. Similarly, a conforming encoder shall not create bit streams that cannot be decoded by the simplified decoder model.

To this end, Annex B defines a generic JPEG XS decoder model that precisely defines the temporal behaviour of the decoder model assuming a processing granularity of codestream packets. While such a model already defines some fundamental properties of the decodable codestreams, it is still not sufficient to ensure interoperability. The reason is that otherwise codestreams could be constructed that would only be decodable by the decoder model if the transmission channel could transport bits arbitrarily fast. In practice, this is obviously not the case. Consequently, interoperability also requires defining a channel model over which an encoder sends the codestreams to the decoder.

Annex C defines such a channel model assuming a transmission channel with a fixed upper bit rate that is related to the target compression ratio. Together with the decoder model of Annex B, it defines the packet-based constant bit rate buffer model. It describes the conditions for a low latency interoperability between any conforming encoder and any decoder. These conditions are expressed by buffer model parameters that are specified by the profiles and levels defined in Annex A. The properties of such conforming implementations are exemplified in Annex D. Since these properties are direct consequences of Annex B and Annex C, Annex D is informative only.

Annex A
(normative)

Profiles, levels and sublevels

A.1 General

Profiles, levels and sublevels specify restrictions on codestreams and hence limits on the capabilities needed to decode the codestreams. Profiles, levels and sublevels may also be used to indicate interoperability points between individual decoder implementations.

Each profile specifies a subset of algorithmic features and limits on their parameterization that shall be supported by all decoders conforming to that profile. Encoders are not required to make use of all features supported in a profile.

The combination of a level and a sublevel defines a lower bound on the throughput a conforming decoder implementation shall support. To this end, the level gives upper bounds for the image parameters in the decoded domain, namely the maximum image width, the maximum image height and the maximum number of sampling grid points to be processed per second.

The sublevel defines upper bounds in the coded domain, such as the nominal bits per pixel value allocated for an encoded image having maximum width and height. In combination with the constraints set by the levels in the decoded domain, this allows the derivation of upper bounds on the admissible encoded image size and the upper number of bits a decoder is required to decode per second. Moreover, it defines the decoder smoothing buffer unit, whose size is specified in subclause A.4.2.

By these means, the decoding smoothing buffer size can be derived from the profile. In combination with the tool selection performed by a profile, this allows to control the complexity of a decoder implementation.

Figure A.1 depicts the relation between level, sublevel, profile and the corresponding constraints they impose.

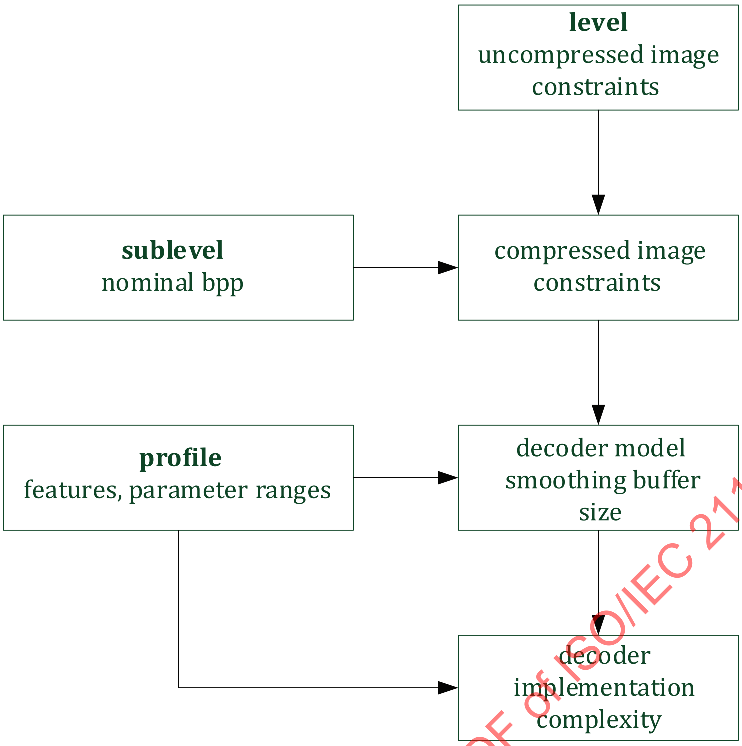


Figure A.1 — Relationship between the different conformance constraints and the impact on the decoder complexity

A.2 Profiles

A.2.1 Definitions of profiles

Profiles specify subsets of coding tools which conforming decoders shall support. Moreover, profiles limit the permitted parameter values. Consequently, profiles are differentiated along the following features:

— decoder smoothing buffer size expressed in smoothing buffer units;

— smoothing buffer offset;

— bit depth;

— chroma sampling formats;

— number of vertical wavelet decompositions;

— number of horizontal wavelet decompositions;

— supported quantizer types;

— maximum column width;

— slice height.

Table A.1, Table A.2 and Table A.3 list the profiles specified in this document.

Table A.1 — JPEG XS Main profiles

Profile	**Main 422.10**	**Main 444.12**	**Main 4444.12**
Number N_{sbu} of smoothing buffer units of the decoder model[a]	16	16	16
Smoothing buffer offset S_{sbo} in bits[b]	1 024	1 024	1 024
Bit depth	8, 10	8, 10, 12	8, 10, 12
Chroma sampling formats	4:0:0, 4:2:2	4:0:0, 4:2:2, 4:4:4	4:0:0, 4:2:2, 4:4:4, 4:2:2:4, 4:4:4:4
Number of vertical decompositions	Up to 1	Up to 1	Up to 1
Number of horizontal decompositions[c]	[1-5]	[1-5]	[1-5]
Quantizer type Qpih	0 (DZQ) 1 (Unform)	0 (DZQ) 1 (Unform)	0 (DZQ) 1 (Uniform)
Column mode (see Cw in ISO/IEC 21122-1)[d]	One column except when the number of vertical decomposition levels is zero[e]	One column except when the number of vertical decomposition levels is zero[e]	One column except when the number of vertical decomposition levels is zero[e]
Slice height in number of image rows	16	16	16
Buffer model Tbmd[f]	1 and 2	1 and 2	1 and 2

[a] The smoothing buffer unit size is determined by the maximum column width in Light-Subline profile and the maximum image width in other profiles. See Formulae (A.1) and (A.2).

[b] The value of 1 024 bits (128 bytes) has been derived from a typical size of the picture header without any extension markers.

[c] As defined in ISO/IEC 21122-2, the number of horizontal wavelet decompositions shall be at least as large as the number of vertical wavelet decompositions.

[d] The column width in sampling grid points is given by $C_s = \left(\begin{cases} 8 \times \mathrm{Cw} \times \max_i (s_x[i]) \times 2^{N_{L,x}} & \text{if Cw} > 0 \\ W_f & \text{otherwise} \end{cases} \right) \leq 2\,048$, where Cw is indicated in the picture header (see ISO/IEC 21122-1), W_f is the image width, $s_x[i]$ is the subsampling factor for component i, and $N_{L,x}$ is the number of horizontal wavelet decompositions.

[e] One column of full width if number of vertical decompositions larger than 0, otherwise any column width conforming with ISO/IEC 21122-1 is allowed.

[f] Decoders shall both support Tbmd=1 and Tbmd=2.

NOTE Tbmd=2 includes Tbmd=1.

Table A.2 — JPEG XS Light profiles

Profile	Light 422.10	Light 444.12	Light-Subline 422.10
Number N_{sbu} of smoothing buffer units of the decoder model[a]	4	4	2
Smoothing buffer offset S_{sbo} in bits[b]	1 024	1 024	1 024
Bit depth	8, 10	8, 10, 12	8, 10
Chroma sampling formats	4:0:0, 4:2:2	4:0:0, 4:2:2, 4:4:4	4:0:0, 4:2:2
Number of vertical decompositions	Up to 1	Up to 1	0
Number of horizontal decompositions[c]	[1-5]	[1-5]	[1-5]
Quantizer type Qpih	0 (DZQ)	0 (DZQ)	0 (DZQ) 1 (Uniform)
Column mode (see Cw in ISO/IEC 21122-1)[d]	Only one column permitted (full width)	Only one column permitted (full width)	Max column width = 2 048
Slice height in number of image rows	16	16	16
Buffer model Tbmd[f]	1 and 2	1 and 2	1 and 2

[a] The smoothing buffer unit size is determined by the maximum column width in Light-Subline profile and the maximum image width in other profiles. See Formulae (A.1) and (A.2).

[b] The value of 1 024 bits (128 bytes) has been derived from a typical size of the picture header without any extension markers.

[c] As defined in ISO/IEC 21122-2, the number of horizontal wavelet decompositions shall be at least as large as the number of vertical wavelet decompositions.

[d] The column width in sampling grid points is given by $C_s = \left(\begin{cases} 8 \times \mathrm{Cw} \times \max_i(s_x[i]) \times 2^{N_{L,x}} & \text{if } \mathrm{Cw} > 0 \\ W_f & \text{otherwise} \end{cases} \right) \leq 2\,048$, where Cw is indicated in the picture header (see ISO/IEC 21122-1), W_f is the image width, s_x[i] is the subsampling factor for component *i*, and $N_{L,x}$ is the number of horizontal wavelet decompositions.

[f] Decoders shall both support Tbmd=1 and Tbmd=2.

NOTE Tbmd=2 includes Tbmd=1.

Table A.3 — JPEG XS High profiles

Profile	High 444.12	High 4444.12
Number N_{sbu} of smoothing buffer units of the decoder model[a]	16	16
Smoothing buffer offset S_{sbo} in bits[b]	1 024	1 024
Bit depth	8, 10, 12	8, 10, 12
Chroma sampling formats	4:0:0, 4:2:2, 4:4:4	4:0:0, 4:2:2, 4:4:4, 4:2:2:4, 4:4:4:4
Number of vertical decompositions	Up to 2	Up to 2
Number of horizontal decompositions[c]	[1-5]	[1-5]
Quantizer type Qpih	0 (DZQ) 1 (Unform)	0 (DZQ) 1 (Unform)
Column mode (see Cw in ISO/IEC 21122-1)[d]	One column except when the number of vertical decom-position levels is zero[e]	One column except when the number of vertical decom-position levels is zero[e]
Slice height in number of image rows	16	16
Buffer model Tbmd[f]	1 and 2	1 and 2

[a] The smoothing buffer unit size is determined by the maximum column width in Light-Subline profile and the maximum image width in other profiles. See Formulae (A.1) and (A.2).

[b] The value of 1 024 bits (128 bytes) has been derived from a typical size of the picture header without any extension markers.

[c] As defined in ISO/IEC 21122-2, the number of horizontal wavelet decompositions shall be at least as large as the number of vertical wavelet decompositions

[d] The column width in sampling grid points is given by $C_s = \left(\begin{cases} 8 \times \text{Cw} \times \max_i(s_x[i]) \times 2^{N_{L,x}} & \text{if Cw} > 0 \\ W_f & \text{otherwise} \end{cases} \right) \leq 2\,048$, where Cw is indicated in the picture header (see ISO/IEC 21122-1), W_f is the image width, s_x[i] is the subsampling factor for component *i*, and $N_{L,x}$ is the number of horizontal wavelet decompositions.

[e] One column of full width if number of vertical decompositions larger than 0, otherwise any column width conforming with ISO/IEC 21122-1 is allowed.

[f] Decoders shall both support Tbmd=1 and Tbmd=2.

NOTE Tbmd=2 includes Tbmd=1.

Figure A.2 represents the relation of the profiles defined in Table A.1, Table A.2 and Table A.3 in terms of inclusivity. The Light422.10 profile is for instance contained in both the Main422.10 profile and the Light444.12 profile. The Main422.10 profile is again included in the Main444.12 profile, etc.

The Light and Light-Subline profiles are independent subsets of the Main profile. They need lower memory and logic resources, and allow for lower latency, at the expense of a lower compression efficiency.

NOTE Figure A.2 does not formulate additional constraints on decoder implementations. The relations presented there are implicit due to the profile specifications in Table A.1, Table A.2 and Table A.3. That is, a decoder conforming to a given profile P in this figure automatically conforms to a profile Q provided there is a path from Q to P in the direction of the arrows.

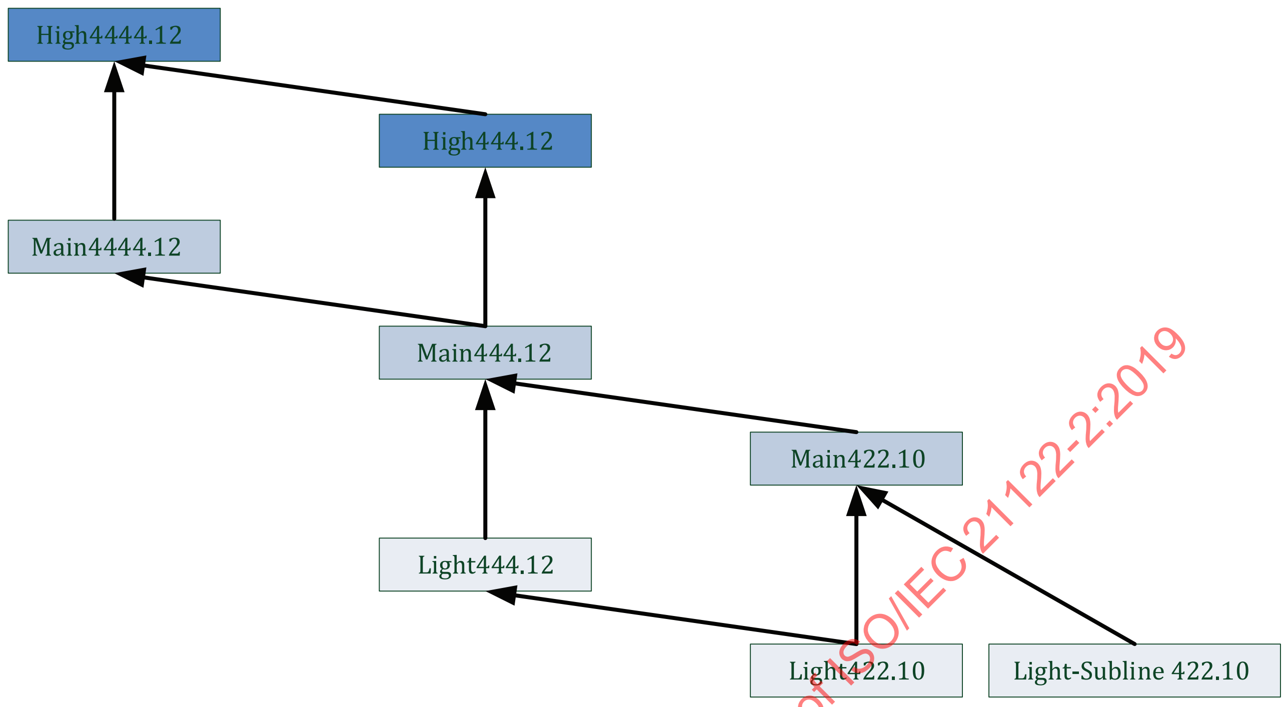


Figure A.2 — Inclusivity relation of the JPEG XS profiles

The resulting maximum bits per pixel (bpp) in the decoded domain for each profile is shown in Table A.4.

Table A.4 — Maximum decoded bpp for each profile defined in this document

Profile	Main 422.10	Main 444.12	Main 4444.12	Light 422.10	Light 444.12	Light-Subline 422.10	High 444.12	High 4444.12
Bit depth	8, 10	8, 10, 12	8, 10, 12	8, 10	8, 10, 12	8, 10	8, 10, 12	8, 10, 12
Chroma sampling formats	4:0:0 4:2:2	4:0:0 4:2:2 4:4:4	4:0:0 4:2:2 4:4:4 4:2:2:4 4:4:4:4	4:0:0 4:2:2	4:0:0 4:2:2 4:4:4	4:0:0 4:2:2	4:0:0 4:2:2 4:4:4	4:0:0 4:2:2 4:4:4 4:2:2:4 4:4:4:4
Max decoded bpp	20	36	48	20	36	20	36	48

A.2.2 Profile signalling in the picture header

The profile of a codestream defines the capabilities of a decoder implementation necessary to decode the image. This profile shall be indicated in the Ppih field in the picture header by the values defined in Table A.5.

The *unrestricted* profile uses the full syntax defined in ISO/IEC 21122-1 without any further constraint. This *unrestricted* profile shall not be considered as a conformance point.

NOTE 1 The *unrestricted* profile is not a conformance point because the syntax defined in ISO/IEC 21122-1 can evolve in the future.

Table A.5 — Mapping of profiles to values of the Ppih field in the picture header

Profile	Ppih	
	Binary	Hex
Unrestricted	0000 0000 0000 0000	0x0000
Light 422.10	0001 0101 0000 0000	0x1500
Light 444.12	0001 1010 0000 0000	0x1A00
Light-Subline 422.10	0010 0101 0000 0000	0x2500
Main 422.10	0011 0101 0100 0000	0x3540
Main 444.12	0011 1010 0100 0000	0x3A40
Main 4444.12	0011 1110 0100 0000	0x3E40
High 444.12	0100 1010 0100 0000	0x4A40
High 4444.12	0100 1110 0100 0000	0x4E40
Reserved for ISO/IEC purposes	all other values	

Conforming encoder and decoder implementations shall only rely on the values given in Table A.5 (and not on the following note).

NOTE 2 The current assignment of Ppih values to profiles is defined such that 4 bits signal the coding tool set, 2 bits signal the chroma format, 2 bits signal the maximum permissible bit depth, 2 bits signal the smoothing buffer unit and 6 bits are currently unused. However, future profiles will possibly deviate from this scheme.

A.3 Levels

Levels define a lower bound on the throughput in the decoded domain that a conforming decoder implementation shall support. Levels are defined along the maximum permissible sampling grid points per line, the maximum number of sampling grid points per column height, the maximum number of sampling grid points per image, and the maximum sampling rate of grid points per second. Table A.6 defines all available levels.

Table A.6 — JPEG XS levels

Level	Maximum picture width W_{max} [sampling grid points]	Max picture height H_{max} [sampling grid points]	Max number of sampling grid points L_{max} per image [sampling grid points]	Max grid point sample rate $R_{s,max}$ [sampling grid points/s]	Examples/ comments
2k-1	2 048	8 192	4 194 304	133 693 440	1280×720@120 1920×1080@60 2048×1536@30 2048×2048@30 2048×1080@60
4k-1	4 096	16 384	8 912 896	267 386 880	1920×1080@120 3840×2160@30 4096×2160@30
4k-2	4 096	16 384	16 777 216	534 773 760	1920×1080@240 3840×2160@60 4096×3072@30 4096×4096@30 4096×2160@60

Table A.6 *(continued)*

Level	Maximum picture width W_{max} [sampling grid points]	Max picture height H_{max} [sampling grid points]	Max number of sampling grid points L_{max} per image [sampling grid points]	Max grid point sample rate $R_{s,max}$ [sampling grid points/s]	Examples/ comments
4k-3	4 096	16 384	16 777 216	1 069 547 520	1920×1080@480 2048×1080@480 4096×3072@60 4096×4096@60 3840×2160@120 4096×2160@120
8k-1	8 192	32 768	35 651 584	1 069 547 520	3840×2160@120 7680×4320@30 8192×4320@30
8k-2	8 192	32 768	67 108 864	2 139 095 040	3840×2160@240 8192×6144@30 8192×8192@30 7680×4320@60 8192×4320@60
8k-3	8 192	32 768	67 108 864	4 278 190 080	3840×2160@480 4096×2160@480 8192×6144@60 8192×8192@60 7680×4320@120 8192×4320@120
10k-1	10 240	40 960	104 857 600	3 342 336 000	10240×7680@30 10240×10240@30 10240×4320@60 10240×5400@60

NOTE 1 Since levels define maximum permissible sample counts and sample rates, a decoder conforming to a specific level is also conforming to all levels that only require a smaller sample count and sample rate than the given level.

NOTE 2 The maximum number of sample grid points is not identical to the product of the maximum picture height and the maximum picture width.

A.4 Sublevels

A.4.1 Definition of sublevels

Sublevels define a lower bound on the throughput in the encoded domain that a conforming decoder implementation shall support. Each sublevel is defined by a nominal bits per pixel (bpp) value N_{bpp} giving the maximum amount of bits per pixel for an encoded image of maximum permissible number of sampling grid points according to the level to which the decoder is conforming. Decoders conforming to a particular level and sublevel shall conform to the following constraints derived from N_{bpp}:

— $S_{sl,max}$: Maximum admissible size of the entire codestream in bytes from SOC to EOC, including all markers. $S_{sl,max}$ is derived from N_{bpp} and the maximum permissible number of sampling grid points L_{max} defined by the level as follows:

$$S_{sl,max} = \left\lfloor \frac{L_{max} \times N_{bpp}}{8} \right\rfloor$$

— $R_{t,max}$: Maximum admissible encoded throughput in bits per second. $R_{t,max}$ is derived from the maximum grid point sample rate $R_{s,max}$ of the level and the nominal bits per pixel value N_{bpp} as follows:

$$R_{t,max} = R_{s,max} \times N_{bpp}$$

Moreover, the size of the smoothing buffer unit S_{sbu} in bits is derived, permitting computation of the overall smoothing buffer of the decoder model defined in Annex B and Annex C.

$$S_{sbu} = \begin{cases} \infty & \text{if level is unrestricted or sublevel is unrestricted} \\ W_{c,max} \times N_{bpp} & \text{otherwise} \end{cases} \quad \text{(A.1)}$$

$W_{c,max}$ is defined to be the maximum column width and depends on the chosen profile as follows:

$$W_{c,max} = \begin{cases} 2\,048 & \text{if profile} = \text{Light-Subline422.10} \\ W_{max} & \text{otherwise} \end{cases} \quad \text{(A.2)}$$

The actual column width is computed by

$$C_s = \begin{cases} 8 \times \text{Cw} \times \max_i(s_x[i]) \times 2^{N_{L,x}} & \text{if Cw} > 0 \\ W_f & \text{otherwise} \end{cases} \quad \text{(A.3)}$$

where

Cw	is indicated in the picture header;
W_f	is the image width;
$s_x[i]$	is the chroma subsampling factor of component i;
$N_{L,x}$	is the number of horizontal wavelet decompositions;
C_s	is the column width in sampling grid points (see ISO/IEC 21122-1).

$W_{c,max}$ is an upper bound for the allowed column width: $C_s \leq W_{c,max}$.

The nominal bpp is not identical to the maximum permissible amount of bits per pixel for an encoded image that does not have maximum width and height. In this case, the amount of bits per pixel for the encoded image may be larger than N_{bpp} as long as the constraints on the codestream defined by the sublevels are followed.

A.4.2 List of sublevels

Table A.7 lists the sublevels defined in this document. The Full sublevel shall only be used if the profile value is not *unrestricted*.

Table A.7 — List of sublevels

Sublevel	Nominal bpp N_{bpp}
Full	Max decoded bpp (See Table A.4).
Sublev12bpp	12
Sublev9bpp	9
Sublev6bpp	6

Table A.7 *(continued)*

Sublevel	Nominal bpp N_{bpp}
Sublev3bpp	3

NOTE 1 By this definition, a decoder conforming to a sublevel defined by a given value of N_{bpp} is also conforming to all sublevels defined by a smaller value of N_{bpp}. That is, sublevels are inclusive.

The resulting constraints on conforming codestreams are listed in Table A.8, Table A.9, Table A.10 and Table A.11.

NOTE 2 Since the constraint on conforming codestreams for the Full sublevel depends on the selected profile, they are not listed in the tables hereunder.

Table A.8 — Codestream constraints for sublevel Sublev3bpp

Level	Size of a smoothing buffer unit S_{sbu} [bits]	Max codestream size $S_{sl,max}$ [bytes]	Max encoded rate $R_{t,max}$ [Mbits/s]
2k-1	6 144	1 572 864	401
4k-1	12 288	3 342 336	802
4k-2	12 288	6 291 456	1 604
4k-3	12 288	6 291 456	3 209
8k-1	24 576	13 369 344	3 209
8k-2	24 576	25 165 824	6 417
8k-3	24 576	25 165 824	12 835
10k-1	30 720	39 321 600	10 027

Table A.9 — Codestream constraints for sublevel Sublev6bpp

Level	Size of a smoothing buffer unit S_{sbu} [bits]	Max codestream size $S_{sl,max}$ [bytes]	Max encoded rate $R_{t,max}$ [Mbits/s]
2k-1	12 288	3 145 728	802
4k-1	24 576	6 684 672	1 604
4k-2	24 576	12 582 912	3 209
4k-3	24 576	12 582 912	6 417
8k-1	49 152	26 738 688	6 417
8k-2	49 152	50 331 648	12 835
8k-3	49 152	50 331 648	25 669
10k-1	61 440	78 643 200	20 054

Table A.10 — Codestream constraints for sublevel Sublev9bpp

Level	Size of a smoothing buffer unit S_{sbu} [bits]	Max codestream size $S_{sl,max}$ [bytes]	Max encoded rate $R_{t,max}$ [Mbits/s]
2k-1	18 432	4 718 592	1 203
4k-1	36 864	10 027 008	2 406
4k-2	36 864	18 874 368	4 812
4k-3	36 864	18 874 368	9 625
8k-1	73 728	40 108 032	9 625
8k-2	73 728	75 497 472	19 251
8k-3	73 728	75 497 472	38 503
10k-1	92 160	117 964 800	30 081

Table A.11 — Codestream constraints for sublevel Sublev12bpp

Level	Size of a smoothing buffer unit S_{sbu} [bits]	Max codestream size $S_{sl,max}$ [bytes]	Max encoded rate $R_{t,max}$ [Mbits/s]
2k-1	24 576	6 291 456	1 604
4k-1	49 152	13 369 344	3 209
4k-2	49 152	25 165 824	6 417
4k-3	49 152	25 165 824	12 835
8k-1	98 304	53 477 376	12 835
8k-2	98 304	100 663 296	25 669
8k-3	98 304	100 663 296	51 338
10k-1	122 880	157 286 400	40 108

A.5 Signalling of levels and sublevels in the Plev field of picture header

The level and sublevel of a codestream define a lower bound on the throughput in both the encoded and decoded domains which a conforming decoder implementation shall support when decoding the codestream into an image. The level and sublevel shall be indicated in the Plev field of the picture header defined in ISO/IEC 21122-1 by the values defined in Table A.12 and Table A.13.

The *unrestricted* level does not impose any constraint on maximum picture width, maximum picture height, maximum number of grid point samples, or maximum grid point sample rate. The *unrestricted* sublevel does not impose any constraint on the nominal bpp.

The *unrestricted* level and sublevel shall not be considered as conformance points.

Table A.12 — Signalling of the levels of a codestream in the Plev field

Level	Binary value of Plev field
Unrestricted	0000 0000 XXXX XXXX
2k-1	0001 0000 XXXX XXXX
4k-1	0010 0000 XXXX XXXX
4k-2	0010 0100 XXXX XXXX
4k-3	0010 1000 XXXX XXXX
8k-1	0011 0000 XXXX XXXX
8k-2	0011 0100 XXXX XXXX
8k-3	0011 1000 XXXX XXXX
10k-1	0100 0000 XXXX XXXX
Reserved for ISO/IEC purposes	all other values
An X indicates either a 0 or a 1.	

Table A.13 — Signalling of the sublevels of a codestream in the Plev field

Sublevel	Binary value of Plev field
Unrestricted	XXXX XXXX 0000 0000
Full	XXXX XXXX 1000 0000
Sublev12bpp	XXXX XXXX 0001 0000
Sublev9bpp	XXXX XXXX 0000 1100
Sublev6bpp	XXXX XXXX 0000 1000
Sublev3bpp	XXXX XXXX 0000 0100
Reserved for ISO/IEC purposes	all other values

Table A.13 *(continued)*

Sublevel	Binary value of Plev field
An X indicates either a 0 or a 1.	

Annex B
(normative)

Packet-based JPEG XS decoder model

B.1 General

This annex defines a generic JPEG XS decoder model that precisely defines the temporal behaviour of a decoder model assuming a processing granularity of codestream packets. The temporal behaviour is necessary to derive the requirements on conforming codestreams for a given transmission channel as defined in Annex C. The profiles and levels define the relevant parameters of the decoder model and transmission channel model to impose requirements on the codestreams and thus ensure interoperability of any decoder implementation.

B.2 Codestream fragments

B.2.1 Coded codestream fragments

ISO/IEC 21122-1 defines the JPEG XS codestream as a sequence of packets, complemented by several marker segments and packet headers located in various positions of the codestream. Each packet consists of a packet header and a packet body. The last packet of a precinct may be followed by fill bytes.

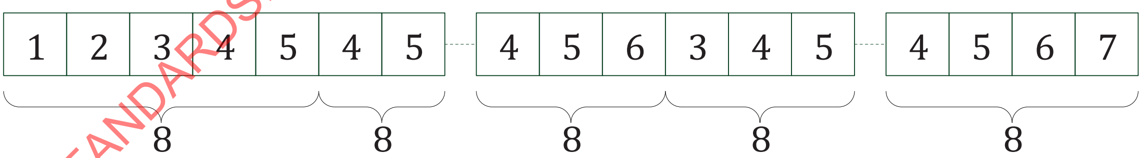
A coded codestream fragment shall be a subset of consecutive bits of the codestream which is built according to the following rules:

— Each coded codestream fragment shall contain exactly one packet, consisting of its packet header and its packet body.

— All headers, marker segments and markers shall be assigned to the subsequent coded codestream fragment of the same codestream. If such a coded codestream fragment does not exist, they shall be assigned to the previous coded codestream fragment.

— The padding bits of a precinct shall be assigned to the previous coded codestream fragment.

Figure B.1 depicts the segmentation of a codestream into fragments.



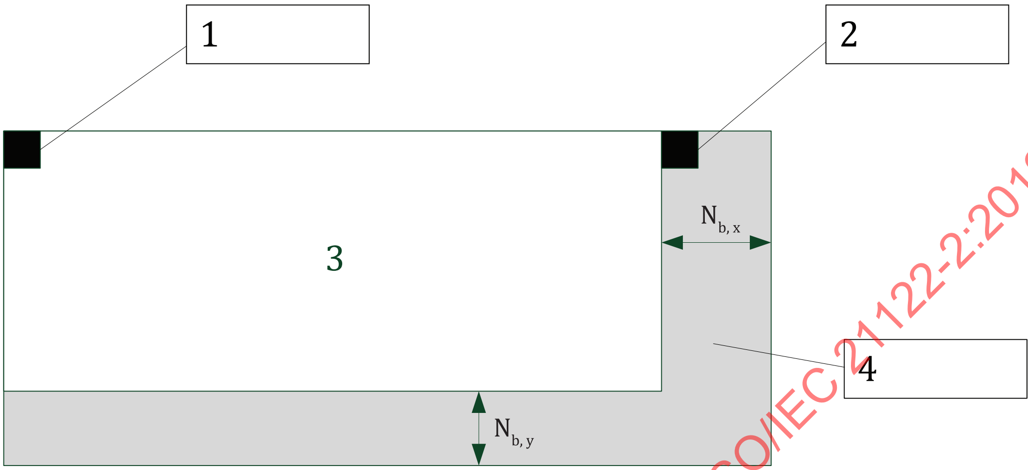
Key

1 file header
2 slice header
3 precinct header
4 packet header
5 packet body
6 fill bytes
7 EOC marker
8 codestream fragment

Figure B.1 — Coded codestream fragment

B.2.2 Blanking codestream fragments

In some systems, images may be embedded into larger pixel containers as illustrated in Figure B.2. The active image pixels are complemented by blanking periods that are not intended for display but that pad the active image to a specified container size.



Key

1 active pixel

2 blanking pixel

3 active image area

4 blanking region

Figure B.2 — Blanking area

When the underlying transport channel allows transmission of bits of encoded data during the blanking periods, improved image quality, in terms of increased target amount of bits per pixel, can be achieved by considering the blanking regions during which transmission occurs in the buffer model. This shall be done by inserting so called blanking codestream fragments into the sequence of coded codestream fragments as depicted in Figure B.3.

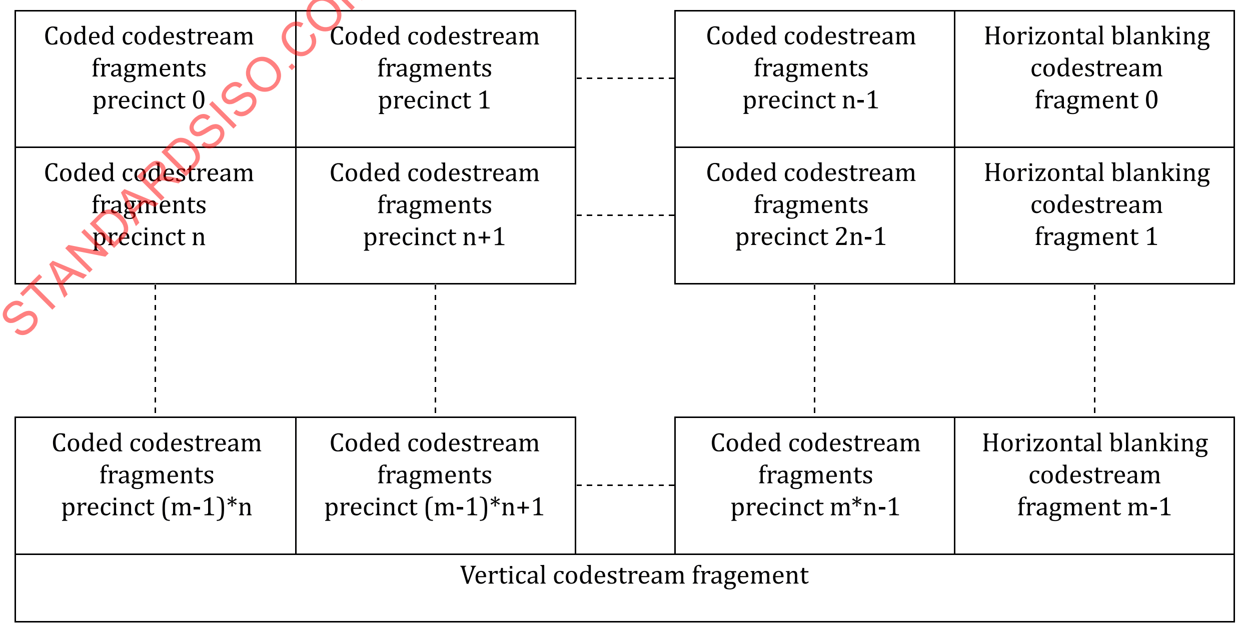


Figure B.3 — Blanking codestream fragments ($n = N_{p,x}$, $m = N_{p,y}$)

According to ISO/IEC 21122-1, the codestream comprises $n = N_{p,x}$ horizontally aligned precincts, where $N_{p,x}$ is the number of columns, and $m = N_{p,y}$ vertically aligned precincts. For consideration of the blanking period in the buffer model, a horizontal blanking codestream fragment f shall be inserted after each coded codestream fragment that terminates a row of horizontally aligned precincts. The size in bits of the horizontal blanking codestream fragment shall equal $S_{bit}(f) = 0$. The duration of the horizontal blanking codestream fragment is expressed in number of coefficient groups $N_{cg,hz}$ associated to the horizontal blanking codestream fragment.

After the last coded codestream fragment of an image, a vertical blanking codestream fragment f is inserted. The size in bits of the vertical blanking codestream fragment shall equal $S_{bit}(f) = 0$. The duration of the vertical blanking codestream fragment is expressed in number of coefficient groups $N_{cg,vt}$ associated to the vertical blanking codestream fragment.

NOTE 1 The time unit for the buffer model is based on coefficients groups. Since the number of all coefficients covered by all coefficient groups in the codestream is possibly larger than the number of image samples due to padding (as explained in ISO/IEC 21122-1), there is no strict integer relation between a pixel clock and a coefficient group clock. In order to avoid using different clocks in the buffer model, the blanking periods are defined as a multiple of coefficient groups instead of pixels.

NOTE 2 The values of $N_{cg,hz}$ and $N_{cg,vt}$ are not signalled in the codestream itself but in the Buffer Model Description box, specified in ISO/IEC 21122-3.

B.2.3 Computation of the number of coefficient groups belonging to a horizontal blanking codestream fragment

The duration of the horizontal blanking period in pixels can be computed by

$$2^{N_{L,y}} \times N_{b,x} \tag{B.1}$$

Since $\sum_{i=1}^{N_c} \frac{1}{s_x[i]}$ coefficient groups represent up to N_g pixels, one coefficient group corresponds to up to:

$$\frac{N_g}{\sum_{i=1}^{N_c} \frac{1}{s_x[i]}} \tag{B.2}$$

pixels. Hence, $N_{cg,hz}$ may be computed as:

$$N_{cg,hz} = \left\lfloor \frac{2^{N_{L,y}} \times N_{b,x} \times \sum_{i=1}^{N_c} \frac{1}{s_x[i]}}{N_g} \right\rfloor \tag{B.3}$$

where

$N_{L,y}$ is the number of vertical decomposition levels;

N_c is the number of components in an image;

$N_{b,x}$ is the number of horizontal blanking periods in the pixel domain during which the transmission channel continues transmission of the codestream;

$s_x[i]$ is the subsampling factor of component i in horizontal direction;

N_g is the number of coefficients in a code group.

This relation is informative only, because the relation between the duration of a pixel and a coefficient group is not prescribed by this document.

B.2.4 Computation of the number of coefficient groups belonging to a vertical blanking codestream fragment

The duration of the vertical blanking period in pixels can be computed by:

$$N_{b,y} \times \left(W_f + N_{b,x}\right)$$

Consequently, $N_{cg,vt}$ may be computed as:

$$N_{cg,vt} = \left\lceil \frac{N_{b,y} \times \left(W_f + N_{b,x}\right) \times \sum_{i=1}^{N_c} \frac{1}{s_x[i]}}{N_g} \right\rceil$$

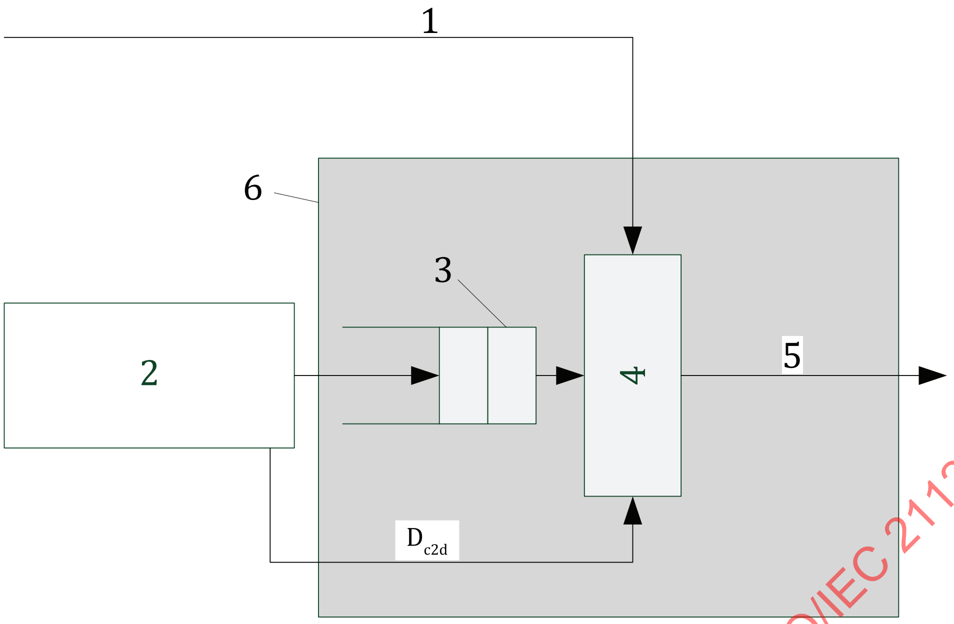
where

- W_f is the width of the image in samples;
- N_c is the number of components in an image;
- $N_{b,y}$ is the number of vertical blanking periods during which the transmission channel continues transmission of the codestream bits;
- $s_x[i]$ is the subsampling factor of component i in horizontal direction;
- N_g is the number of coefficients in a code group.

This relation is informative only, because the relation between the duration of a pixel and a coefficient group is not prescribed by this document.

B.3 Decoder model block diagram

Figure B.4 depicts the decoder block diagram that forms the base of any buffer model.



Key

1 clock
2 transmission channel
3 decoder smoothing buffer
4 decoder unit
5 output pixels
6 decoder model

Figure B.4 — JPEG XS decoder model

The decoder model consists of a decoder unit, clocked by a periodic signal with period *T*. With each new clock cycle, the decoder unit starts decoding a new coefficient group, which — when not being blanking coefficient groups — may eventually be output in form of decoded output pixels. For this purpose, the decoder unit reads with each clock cycle a variable number of bits from a decoder smoothing buffer. This decoder smoothing buffer has well-defined capacity, allowing a maximum amount of bits to be stored. The decoder shall be able to read a variable number of bits per clock cycle, following a first-in-first-out semantic. This means that when a bit v_1 is written into the decoder smoothing buffer before bit v_2, then this bit v_1 shall not be read after bit v_2.

The decoder smoothing buffer is connected to a bit source that writes bits into the decoder smoothing buffer. In Figure B.4, this bit source is represented as a transmission channel, because for typical low latency applications, a decoder is connected to an encoder by means of such a transmission channel. However, in general, any other bit source is valid as well. The value D_{c2d} in Figure B.4 defines the number of clock cycles the start of decoding needs to be delayed relative to the time at which the first bit of the codestream arrives in the decoding smoothing buffer. Different methods are possible to generate this value, and this document does not prescribe any of them.

B.4 Decoder smoothing buffer

Bits written at clock cycle *t* into the decoding smoothing buffer are available to the decoder unit immediately at clock cycle *t*. Let $l_{dec,max} \in \mathbb{N}$ be the maximum number of bits that can be stored in the decoder smoothing buffer. Let $l_{dec}(t)$ be the number of bits stored in the decoder smoothing buffer at the end of clock cycle *t*. Let $w_{dec}(t)$ be the number of bits written into the decoder smoothing buffer in clock cycle t. Let $r_{dec}(t)$ be the number of bits read and removed from the decoder smoothing buffer in clock cycle t. Then the following shall hold for a conforming implementation of the decoder model:

$$\tilde{l}_{dec}(t) = l_{dec}(t-1) + w_{dec}(t) \tag{B.4}$$

$$r_{dec}(t) \le \tilde{l}_{dec}(t) \le l_{dec,max} \tag{B.5}$$

$$l_{dec}(t) = l_{dec}(t-1) + w_{dec}(t) - r_{dec}(t) = \tilde{l}_{dec}(t) - r_{dec}(t) \tag{B.6}$$

B.5 Buffer model types

Table B.1 defines the buffer model types defined in this document.

Table B.1 — Buffer model types

Value of Tbmd	Meaning
0	No upper limit of the decoder buffer assumed
1	Constant bit rate buffer model with limited transmission latency [see Formula (C.5)]
2	Constant bit rate buffer model with full use of decoder smoothing buffer (variable transmission latency) [see Formula (C.5)]
3-255	Reserved for ISO/IEC purposes

Annex C
(normative)

Packet-based constant bit rate buffer model

C.1 General

This annex defines transmission channel models assuming a fixed upper bit rate that is related to the target compression ratio. Together with the decoder model of Annex B, it defines the packet-based constant bit rate buffer model. It describes the conditions for a low latency interoperability of any conforming decoder. These conditions are expressed by buffer model parameters that are specified by the profiles and levels defined in Annex A.

C.2 Decoder unit

In order to decode a codestream fragment $f \geq 1$, the decoder unit reads all bits that have been generated by the encoder for this codestream fragment f. Processing of codestream fragment $f \in \mathbb{N}$ starts at cycle $t_{dec,start}(f)$:

$$t_{dec,start}(f) = t_{channel,start} + D_{c2d} + \sum_{i=1}^{f-1} N_{cg}(i)$$
$$\Leftrightarrow \tag{C.1}$$
$$t_{dec,start}(f) = \begin{cases} t_{channel,start} + D_{c2d} & f = 1 \\ t_{dec,start}(f-1) + N_{cg}(f) & \text{otherwise} \end{cases}$$

$D_{c2d} \in \mathbb{N}$ is the delay of the decoding start relative to the start of the transmission channel. $t_{channel,start} \in \mathbb{Z}$ is the clock cycle in which the transmission channel writes the first bit of the considered codestream into the decoding smoothing buffer. In order to process codestream fragment f, all coded bits of codestream fragment f shall be contained in the smoothing buffer:

$$\tilde{l}_{dec}\left(t_{dec,start}(f)\right) \geq S_{bits}(f) \tag{C.2}$$

The coded bits of each codestream fragment $f \in \mathbb{N}$ are removed from the decoder smoothing buffer at cycle $t_{dec,read}(f) \in \mathbb{N}$. The latter is computed as follows:

$$t_{dec,read}(f) = t_{dec,start}(f) + N_{cg}(f) - 1$$
$$\Leftrightarrow \tag{C.3}$$
$$t_{dec,read}(f) = t_{channel,start} + D_{c2d} + \left(\sum_{i=1}^{f} N_{cg}(i)\right) - 1$$

The number of bits removed from the decoding smoothing buffer is defined as follows:

$$r_{dec}(t) = \begin{cases} S_{bits}(f) & \exists f : t_{dec,read}(f) = t \\ 0 & \text{otherwise} \end{cases}$$

where ∃ stands for "there exists".

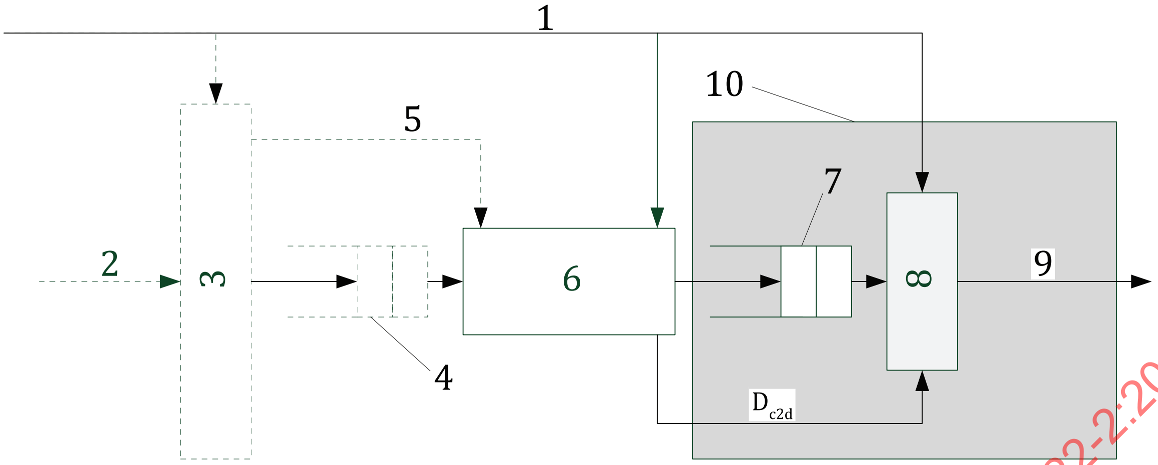
The decoder model is described by pseudo code in Table C.1:

Table C.1 — Decoder model pseudo code

Operation	Notes
`wait`($t_{dec,start}(0)$)	Wait for $t_{dec,start}(0)$ clock cycles
`f=1`	Reset the codestream fragment counter
`while(! End of stream) {`	Repeat until the end of the codestream is reached
	At this time, all bits required to decode codestream fragment f shall be available in the decoding smoothing buffer
`wait`($N_{cg}(f) - 1$)	Wait for $N_{cg}(f) - 1$. clock cycles. $N_{cg}(f)$ is the processing time of codestream fragment f.
`bits= readBits`($S_{bits}(f)$)	Read $S_{bits}(f)$ from the smoothing buffer and remove them from smoothing buffer
`decode_bits`	Decode the bits in the codestream fragment f
`wait(1)`	Wait another cycle to complete codestream fragment *f*
`f=f+1`	Advance to the next codestream fragment
`}`	

C.3 Encoder-decoder system model

The packet-based constant bit rate buffer model is intended for applications where an encoder is connected to a decoder by a transmission channel with a maximum upper bit rate R_{trans} (for VBR applications) or a constant bit rate R_{trans} (for CBR applications) that equals the target compression rate.



Key

1 clock
2 input pixels
3 encoder unit
4 encoder smoothing buffer
5 SoT
6 transmission channel
7 decoder smoothing buffer
8 decoder unit
9 output pixels
10 decoder model

Figure C.1 — System model (dashed elements are informative only)

Figure C.1 illustrates the corresponding block diagram. It extends the decoder model of Figure B.4 by showing the encoder feeding the transmission channel. Both the encoder and the decoder are assumed to be clocked by a common periodic signal with period *T*.

The packet-based constant bit rate buffer model can also be applied in situations where the transmission channel operates in bursts and can hence only be approximated by a constant bit rate channel or a channel with a maximum bit rate. Then, however, means shall be provided to transform the actual transmission behaviour into one corresponding to subclause C.4.

C.4 Transmission channel model

C.4.1 Transmission channel with maximum bit rate

The transmission channel model shall write at most $R_{trans} \in \mathbb{Q}$ bits per clock cycle into the decoder smoothing buffer for all cycles $t \geq t_{channel,start}$.

NOTE 1 This includes the blanking regions defined by $N_{cg,hz}$ and $N_{cg,vt}$.

$$R_{trans} = \frac{S_{c,max} \times 8}{\sum_{f=1}^{N_f} N_{cg}(f)} \tag{C.4}$$

where

$S_{c,max}$ is the maximum size of the entire codestream in bytes from SOC to EOC, including all markers. In case the Lcod field of the picture header is unequal to zero, $S_{c,max}$ shall equal the Lcod value;

N_f is the number of codestream fragments including the blanking codestream fragments;

$N_{cg}(f)$ is the number of coefficient groups in codestream fragment *f*.

In many applications, the compression to apply is defined by a bit per pixel value bpp. In such a case, $S_{c,max}$ can be computed as:

$$S_{c,max} = W_f \times H_f \times bpp$$

The cumulative number of bits written into the decoder smoothing buffer can be computed as:

$$\sum_{\tau=0}^{t} w_{dec}(\tau) \leq \left\lfloor (t - t_{channel,start}) \times R_{trans} \right\rfloor \quad \text{(C.5)}$$

The value of $t_{channel,start}$ depends on the system configuration, and in particular on the encoder. In a full system, $t_{channel,start}$ should be determined by the encoder in such a way that the decoder buffer does not overflow.

C.4.2 Transmission channel with constant bit rate

For transmission channels with constant bit rate, the transmission channel model shall write $R_{trans} \in \mathbb{Q}$ bits per clock cycle into the decoder smoothing buffer for all cycles $t \geq t_{channel,start}$.

NOTE This includes the blanking regions defined by $N_{b,x}$ and $N_{b,y}$.

The value of $t_{channel,start}$ shall be set in such a way that the transmission channel (or any other bit source) is able to generate a continuous stream of bits without any interruptions.

The cumulative number of bits written into the decoder smoothing buffer can be computed as

$$\sum_{\tau=0}^{t} w_{dec}(\tau) = \left\lfloor (t - t_{channel,start}) \times R_{trans} \right\rfloor$$

The value of $t_{channel,start}$ depends on the system configuration, and in particular on the encoder. In a full system, $t_{channel,start}$ should be determined by the encoder.

The transmission channel is therefore described by Table C.2:

Table C.2 — Transmission channel pseudo code

Operation	Notes
`r=0`	Remaining fractional bits
`wait(`$t_{channel,start}$`)`	Wait until transmission channel starts to write bits
`while(!end of stream) {`	Repeat until the source stops sending bits
`writeBits(`$\lfloor R_{Trans} + r \rfloor$`)`	Write $\lfloor R_{Trans} + r \rfloor$ bits to the decoder smoothing buffer
$r = R_{Trans} + r - \lfloor R_{Trans} + r \rfloor$	Update the number of fractional bits
`wait(1)`	Wait for one cycle
`}`	Continue sending data

C.4.3 Relation between the two channel models

In case an encoder generates only codestreams having the maximum size $S_{c,max}$, both channel models defined in subclauses C.4.1 and C.4.2 are effectively the same due to the definition of R_{trans} in Formula (C.4). Consequently, the transmission channel model with constant bit rate can be used to dimension the system. The transmission channel with maximum bit rate then allows interrupting transmission in case the encoder does not pad the codestream to the maximum size.

It is the responsibility of the encoder to manage the transmission in such a way that the decoder buffer does not overflow.

C.5 Decoder smoothing buffer

The decoder smoothing buffer shall behave as described in Clause B.4. Based on the selected buffer model type defined in Clause B.5, the size of the decoding smoothing buffer $l_{dec,max}$ is computed as follows:

$$l^{*}_{dec,max} = S_{sbo} + \begin{cases} \infty & \text{Tbmd} = 0 \\ min\left(l^{cbr}_{dec,max}, \left\lfloor R_{trans} \times \sum_{i=1}^{N_c} \frac{1}{s_x[i]} \times \frac{W_f}{N_g} \times \Delta T_{max,lines} \right\rfloor \right) & \text{Tbmd} = 1 \\ l^{cbr}_{dec,max} & \text{Tbmd} = 2 \end{cases} \tag{C.6}$$

$$\text{where } l_{dec,max} := \left\lfloor \left\lfloor \frac{l^{*}_{dec,max}}{R_{trans}} \right\rfloor \times R_{trans} \right\rfloor \leq l^{*}_{dec,max}$$

NOTE Tbmd = 1 essentially leads to a system where the maximum transmission latency in lines is independent of the image width W_f and the target compression rate R_{trans}. See Clause D.3.

C.6 Buffer model instance

The following buffer model parameters are specified for the profiles, levels or sublevels, referring to the buffer model defined in Clause C.5:

$l^{cbr}_{dec,max} \in \mathbb{N}$ base size of the decoder smoothing buffer in bits due to sublevel and profile, not including the smoothing buffer offset defined in Table C.3.

NOTE The actual number of bits that can be stored in the decoder smoothing buffer is given by Formula (C.5).

$\Delta T_{max,lines}$ upper bound for the transmission latency in lines. Can be infinite.

S_{sbo} offset to the buffer size computation.

The buffer model combined with those values is called a buffer model instance.

C.7 Buffer model instance parameters

Table C.3 specifies how to derive the buffer model instance parameters from the profile, level and sublevel constraints for implementations conforming to the packet-based constant bit rate buffer model defined in subclause C.4.2 and Clause C.5.

Table C.3 — Derivation of the buffer model parameters for the packet-based constant bit rate buffer model

Buffer model instance parameters	Value
$l^{cbr}_{dec,max}$	$N_{sbu} \times S_{sbu}$ (see subclause A.4.2)
$\Delta T_{max,lines}$	N_{sbu} (see Table A.1, Table A.2, Table A.3)
S_{sbo}	(see Table A.1, Table A.2, Table A.3)

C.8 Buffer model conformance

C.8.1 Conformance of a single codestream

A single codestream conforming to the buffer model instance shall also be conforming with ISO/IEC 21122-1. Finally, there shall exist a value $D_{c2d} \in \mathbb{N}$ such that Formulae (B.4) to (B.6) and (C.2) are valid for all cycles t, when the temporal behaviour of the channel and decoder models corresponds to the specifications in Clauses C.2 to C.6.

In case the transmission channel uses variable bit rate as defined in subclause C.4.1, Formula (C.4) only establishes an upper bound on the number of bits $\sum_{\tau=0}^{t} w_{dec}(\tau)$ written to the decoder smoothing buffer. For a conforming codestream, there shall exist a function $w_{dec}(\tau)$ that is consistent with Formula (C.4) and for which Formulae (B.4) to (B.6) and (C.2) are true.

C.8.2 Conformance of a sequence of codestreams

This clause defines the conformance of a sequence $C = [C(1),...,C(n)]$ of codestreams to a buffer model instance. For that, define the symbol $t_{channel,start}(i)$ as the clock cycle where the transmission channel writes the first bit of codestream $C(i)$. Further, define the symbol $D_{c2d}(1)$ to be the delay between $t_{channel,start}(1)$ and the start of decoding of codestream $C(1)$. Moreover, define the symbol $F_{first}(C(i))$ to be the first codestream fragment of codestream $C(i)$ and $F_{last}(C(i))$ to be the last codestream fragment of $C(i)$. Define $S_{c,max}(C(i))$ to be the maximum size in bytes of codestream $C(i)$, and define $N_f(C(i))$ to be the number of codestream fragments for codestream $C(i)$.

Then, for conformance of C to a given buffer model instance, each codestream $C(i)$ shall be conforming to this buffer model instance according to subclause C.8.1, and in addition, the following shall hold:

$$\forall 1 \le i < n : t_{dec,start}\left(F_{first}\left(C\left(i+1\right)\right)\right) = t_{dec,read}\left(F_{last}\left(C\left(i\right)\right)\right) + 1$$

and

$$\frac{S_{c,max}\left(C\left(i\right)\right) \times 8}{\sum_{f=1}^{N_f(C(i))} N_{cg}\left(f\right)} = const = R_{trans}$$

C.8.3 Decoder conformance

A decoder conforming to a buffer model instance shall be able to decode all codestreams that are conforming with the buffer model instance.

C.8.4 Encoder conformance

An encoder is conforming to a buffer model instance if all codestreams it generates are conforming to the buffer model instance. Moreover, the encoder shall provide the codestream fragments at such time instances so that the decoder can receive them through the transmission channel at the time instances assumed by the buffer model.

C.8.5 Decoder implementation deviations

A decoder implementation conforming to a buffer model instance may deviate from the temporal behaviour of the decoder model defined in Clause C.2. However, it is then the responsibility of the implementation to foresee all necessary measures to ensure that this implementation is able to decode all conforming codestreams correctly.

C.8.6 Transmission channel deviations

Decoder implementations conforming to a buffer model instance assume a bit rate transmission behaviour as defined in Clause C.4. In case the actual transmission channel violates these assumptions, means shall be provided to transform the actual transmission behaviour into one corresponding to Clause C.4. These means should be defined in application-specific specifications.

Annex D
(informative)

Encoder model, latency bounds and codestream conformance properties for the packet-based constant bit rate buffer model

D.1 General

Annex B and Annex C define the requirements on conforming decoders and codestreams such that encoder and decoder implementations of different vendors are interoperable. Moreover, they ensure that system implementations with low end-to-end latency are possible independent of the image content contained in the codestream.

Based on the clauses of Annex B and Annex C, it is possible to derive some fundamental properties of conforming codestreams and system implementations, consisting of a conforming encoder and decoder. This includes in particular a latency bound of conforming encoder-decoder systems.

D.2 Encoder model

The encoder unit of the system model described in Clause C.3 generates a sequence of codestreams being a sequence of codestream fragments. Each codestream fragment $f \in \mathbb{N}$ is written to the encoder smoothing buffer at cycle $t_{enc,write}(f) \in \mathbb{N}_0$. The latter depends on the number of coefficient groups $N_{cg}(i)$ in codestream fragment i:

$$t_{enc,write}(f) = \left(\sum_{i=1}^{f} N_{cg}(i) \right) - 1 = N_{cg}(f) + \begin{cases} -1 & f = 1 \\ t_{enc}(f-1) & \text{otherwise} \end{cases} \quad \text{(D.1)}$$

Let $S_{bits}(f)$ be the number of coded bits for codestream fragment f. Then data generation of the encoder model is described by the pseudo code in Table D.1:

Table D.1 — Encoder pseudo code

Operation	Notes
f=1	Reset codestream fragment counter
while(!end of stream) {	Repeat until the stream is interrupted
wait($N_{cg}(f)$ – 1)	Wait $N_{cg}(f)$ – 1 clock cycles
writeBits(bits, $S_{bits}(f)$)	Write $S_{bits}(f)$ bits to smoothing buffer
wait(1)	Go to next cycle
f = f+1	Go to next fragment
}	

D.3 Buffer relations

For the constant bit rate transmission channel model defined in subclause C.4.2, the fill level of the encoder smoothing buffer at the end of cycle $t \in \mathbb{N}_0$ is computed by:

$$l_{enc}(t) = \sum_{f=1}^{\infty} S_{bits}(f) \times \xi(t - t_{enc,write}(f)) - \lfloor (t + 1 - t_{channel,start}) \times R_{trans} \rfloor \times \xi(t - t_{channel,start})$$

NOTE 1 When $t_{enc,write}(f) = 0$ and $t_{channel,start} = 0$, the encoder produces data in the cycle $t = 0$. and transmission immediately starts at $t = 0$.

From Formulae (C.2) and (D.1), it follows that:

$$t_{dec,read}(f) = \underbrace{t_{channel,start} + D_{c2d}}_{t_{dec,start}(1)} + t_{enc,write}(f)$$

For a constant bit rate transmission channel model of subclause C.4.2, the fill level of the decoder smoothing buffer at the end of cycle $t_{dec,start}(1) + t$ is computed as:

$$\begin{aligned} l_{dec}(t_{dec,start}(1) + t) \\ &= \lfloor (t_{dec,start}(1) + t - t_{channel,start} + 1) \times R_{trans} \rfloor \\ &- \sum_{f=1}^{\infty} S_{bits}(f) \times \xi(t - t_{enc,write}(f)) \end{aligned} \tag{D.2}$$

$t_{dec,start}(1)$ is the cycle where the decoder starts decoding.

From this, it follows that

$$\begin{aligned} l_{enc}(t) + l_{dec}(t_{dec,start}(1) + t) \\ &= \lfloor (t_{dec,start}(1) + t - t_{channel,start} + 1) \times R_{trans} \rfloor \\ &- \lfloor (t + 1 - t_{channel,start}) \times R_{trans} \rfloor \times \xi(t - t_{channel,start}) \end{aligned}$$

For $t \geq t_{channel,start}$, it follows that

$$\begin{aligned} l_{enc}(t) + l_{dec}(t_{dec,start}(1) + t) = l_{sum} + \epsilon_1(t) \in \mathbb{N} \\ |\epsilon_1(t)| < 1 \\ l_{sum} = t_{dec,start}(1) \times R_{trans} = const \\ R_{trans} \in \mathbb{N} \Rightarrow \epsilon_1(t) = 0 \end{aligned} \tag{D.3}$$

This is equivalent to

$$\begin{aligned} l_{enc}(t) + l_{dec}(t_{dec,start}(1) + t) = \lceil l_{sum} \rceil + \epsilon_2(t) \in \mathbb{N} \\ \epsilon_2(t) \in \{0, -1\} \\ R_{trans} \in \mathbb{N} \Rightarrow \epsilon_2(t) = 0 \end{aligned} \tag{D.4}$$

The maximum decoder buffer fill level is given for

$$l_{enc}(t) = 0 \Rightarrow l_{dec}(t_{dec,start}(1) + t) = \lceil l_{sum} \rceil + \epsilon_2(t)$$

Hence,

$$l_{dec,max} = \lceil l_{sum} \rceil = \lceil t_{dec,start}(1) \times R_{trans} \rceil$$

As $t_{dec,start}(1) \in \mathbb{N}$,

$$t_{dec,start}(1) = \left\lfloor \frac{l^*_{dec,max}}{R_{trans}} \right\rfloor = \left\lfloor \frac{l_{dec,max}}{R_{trans}} \right\rfloor \quad \text{(D.5)}$$

Given that the encoder model starts encoding at cycle $t = 0$, $t_{dec,start}(1)$ corresponds to the coding latency in coefficient groups of the system model depicted in Figure C.1. In case the horizontal blanking period is set to zero ($N_{cg,hz} = 0$), the latency in lines is thus given by

$$\frac{t_{dec,start}(1) \times N_g}{W_f \times \sum_{i=1}^{N_c} \frac{1}{s_x[i]}} = \frac{N_g}{W_f \times \sum_{i=1}^{N_c} \frac{1}{s_x[i]}} \times \left\lfloor \frac{l^*_{dec,max}}{R_{trans}} \right\rfloor$$

If the buffer model type is $Tbmd = 1$, and $S_{sbo} = 0$, then it follows that

$$l^*_{dec,max} = \left\lfloor R_{trans} \times \sum_{i=1}^{N_c} \frac{1}{s_x[i]} \times \frac{W_f}{N_g} \times \Delta T_{max,lines} \right\rfloor .$$

From this, it follows that

$$\frac{t_{dec,start}(1) \times N_g}{W_f \times \sum_{i=1}^{N_c} \frac{1}{s_x[i]}} = \frac{N_g}{W_f \times \sum_{i=1}^{N_c} \frac{1}{s_x[i]}} \times \left\lfloor \frac{\left\lfloor R_{trans} \times \sum_{i=1}^{N_c} \frac{1}{s_x[i]} \times \frac{W_f}{N_g} \times \Delta T_{max,lines} \right\rfloor}{R_{trans}} \right\rfloor \approx \Delta T_{max,lines}$$

Therefore, buffer model type $Tbmd = 1$ leads to a system where the end-to-end transmission latency in lines is approximately bounded by the parameter $\Delta T_{max,lines}$.

NOTE 2 When $N_{cg,hz} > 0$, the latency in lines becomes slightly smaller.

NOTE 3 The fact that $S_{sbo} > 0$ causes the latency bound to become slightly larger.

D.4 Minimum decoder delay D_{c2d}

In order to avoid a buffer underflow in the decoder, according to Formulae (B.4), (B.5) and (C.2) the following condition shall hold:

$$l_{dec}\left(t_{dec,start}(f) - 1\right) + R_{trans} \geq S_{bits}(f)$$

For the constant bit rate transmission channel model, the left side of this formula computes to:

$$l_{dec}\left(t_{dec,start}(f) - 1\right) + R_{trans} = \left\lfloor R_{trans} \cdot \left(t_{dec,start}(f) - t_{channel,start}\right) + R_{trans} \right\rfloor - \sum_{i=1}^{f-1} S_{bits}(i) .$$

NOTE In case of $t_{dec,start}(f) = t_{channel,start}$, $l_{dec}(t_{dec,start}(f) - 1) = 0$.