International Standard

ISO/IEC 21122-2

Third edition
2024-08

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

Part 2: Profiles and buffer models

Technologies de l'information — Système de codage d'images léger à faible latence JPEG XS —

Partie 2: Profils et modèles tampons

Reference number
ISO/IEC 21122-2:2024(en)

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

Published in Switzerland

Contents

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 or www.iec.ch/members_experts/refdocs).

ISO and IEC draw attention to the possibility that the implementation of this document may involve the use of (a) patent(s). ISO and IEC take no position concerning the evidence, validity or applicability of any claimed patent rights in respect thereof. As of the date of publication of this document, ISO and IEC had not received notice of (a) patent(s) which may be required to implement this document. However, implementers are cautioned that this may not represent the latest information, which may be obtained from the patent database available at www.iso.org/patents and https://patents.iec.ch. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

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. In the IEC, see www.iec.ch/understanding-standards.

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*.

This third edition cancels and replaces the second edition (ISO 21122-2:2022), which has been technically revised. It also incorporates the Amendment ISO/IEC 21122-2:2022/Amd 1:2022.

The main changes compared to the previous edition are:

— addition of conformance points for new profiles;

— addition of the TDC 444.12 and TDC MLS 444.12 profiles for compression of image sequences;

— addition of the CHigh 444.12 profile;

— addition of the MLS.16 profile;

— addition of the frame buffer bandwidth levels and model.

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

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 https://www.iso.org/members.html and https://www.iec.ch/national-committees.

Introduction

This document is part of a series of standards for a low-latency lightweight image coding system, denoted as JPEG XS. While ISO/IEC 21122-1 specifies a full set of compression coding tools needed to satisfy all the requirements of JPEG XS, a targeted application can often work with a simpler and reduced set of coding tools, and with or without tighter constraints, to meet its targeted goals. For this reason, profiles, levels, and sublevels are defined in this document. These three concepts facilitate partial and reduced complexity implementations of ISO/IEC 21122-1 depending on specific application use cases and requirements, while also safeguarding interoperability.

This document specifies a limited number of profiles to represent interoperability subsets of the codestream syntax specified in ISO/IEC 21122-1 with each profile serving specific application use cases. In other word, profiles select a subset of the available coding tools. In addition, levels and sublevels provide limits to the maximum throughput in respectively the decoded (spatial/pixel) and the encoded (codestream) domains. In this way, profiles, levels and sublevels allow designing cost-efficient implementations that serve the needs of the desired applications.

A major requirement of JPEG XS is to allow low end-to-end latency, limited to a fraction of the frame size. To ensure this low-latency property, this document also specifies a buffer model, consisting of a decoder model and a transmission channel model. The models show the interaction of a hypothetical reference decoder, including its smoothing buffer with a constant bitrate channel feeding this buffer. The size of the decoder smoothing buffer is computed from the profile, level, and sublevel. Codestreams are formed such that the buffer of a decoder, operating according to this buffer model, never overflows or underflows. In effect, 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, this document gives extra information on the minimum latency that can be achieved by the different methods described in ISO/IEC 21122-1.

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

Part 2:
Profiles and buffer models

1 Scope

This document defines several subsets of the syntax specified in ISO/IEC 21122-1 as profiles. It also defines lower bounds on the throughput in the decoded domain via levels and the encoded domain via sublevels that a conforming decoder implementation shall support. Furthermore, it defines 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, *Information technology — JPEG XS low-latency lightweight image coding system — Part 1: Core coding system*

3 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 terminology databases for use in standardization at the following addresses:

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

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

3.1
blanking codestream fragment
placeholder codestream fragment representing blanking periods

3.2
buffer model
combination of a *decoder model* (3.8) and a *channel model* (3.4) whose behaviour can be defined by a set of parameters

3.3
buffer model instance
specific configuration of a *buffer model* (3.2) specified by the assignment of well-defined values to the *buffer model* parameters

3.4
channel model
model describing the temporal behaviour of the *transmission channel* (3.26) connecting an encoder and a decoder

3.5
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

3.6
codestream fragment
either *coded codestream fragment* (3.5) or *blanking codestream fragment* (3.1)

3.7
cycle
clock cycle
single clock period of an encoder or decoder clocked implementation

3.8
decoder model
combination of a *decoder unit* (3.10) and a *decoder smoothing buffer* (3.9)

3.9
decoder smoothing buffer
memory buffer that is used to level out changes in the number of bits read by a *decoder unit* (3.10) per time unit

3.10
decoder unit
module reading a variable number of bits (from the smoothing buffer) per time unit to generate decoded output pixels at a fixed output rate

3.11
encoder model
combination of an *encoder unit* and an *encoder smoothing buffer* (3.12)

3.12
encoder smoothing buffer
memory buffer that is used to level out changes in the number of bits generated by an *encoder unit* (3.13) per time unit

3.13
encoder unit
module transforming a sequence of input pixels with constant rate into a conforming codestream, producing a bit sequence with variable number of bits generated per time unit

3.14
fill level
number of bits stored in the encoder or *decoder smoothing buffer* (3.9)

3.15
horizontal blanking period
timespan expressed in units of the grid point sampling rate between the last pixel of an image line — not being the last line of an image — and the first pixel of the next image line

3.16
level
defined set of constraints on the number of decoded samples to be processed by an encoder or decoder, both in the spatial and temporal 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* (3.19)

3.17
nominal bits per pixel value
mean number of bits allocated per encoded pixel which is used to derive the *sublevel* constraints by assuming an image with well-defined dimensions and frame rate derived from the *level*

3.18
pixel
samples of all components at a single *sampling grid point* (3.20)

3.19
profile
specified subset of the codestream syntax together with admissible parameter values

3.20
sampling grid point
position on the sample grid, specified by integer horizontal and vertical offset relative to the origin of the sample grid

3.21
smoothing buffer unit
level- and *sublevel*-dependent number of bits by which the smoothing buffer size of the *decoder model* is specified

3.22
start of transmission
SoT
time at which the *transmission channel* starts transmission relative to the start of encoding of the first *codestream fragment* of a codestream

3.23
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

3.24
TDC disabled codestream
codestream that contains zero `SLI` markers

Note 1 to entry: See ISO/IEC 21122-1.

3.25
TDC enabled codestream
codestream that contains one or more `SLI` markers

Note 1 to entry: See ISO/IEC 21122-1.

3.26
transmission channel
facility transferring bits from a source entity to a target entity

3.27
transmission channel capacity
maximum number of bits per time unit that a *transmission channel* (3.26) can transfer from a source entity to a target entity

3.28
vertical blanking period
timespan expressed in units of the grid point sampling rate between the last line of an image — including the *horizontal blanking periods (3.15)* — and the first line of the next image

4 Abbreviated terms

bpp	bits per pixel
CBR	constant bit rate
CFA	colour filter array
DWT	discrete wavelet transform
FBB	frame buffer bandwidth
IDWT	inverse discrete wavelet transform
IRCT	inverse reversible colour transform
MLS	mathematically lossless
RCT	reversible colour transform
RGB	red green blue
TDC	temporal differential coding (see ISO/IEC 21122-1)
VBR	variable bit rate

5 Symbols

B_r	number of bits required to encode a bitplane count in raw
B_w	nominal overall bit precision of the wavelet coefficients
$B[i]$	precision in bits of component i
C_{pih}	colour transformation type
$c'[p,\lambda,b,x]$	wavelet coefficient in precinct p, line λ, band b and position x
C_s	width of precincts other than the rightmost precinct in sample grid points
C_w	width of precincts in multiples of 8 LL subsampled band sample grid points
$C(i)$	the i -th codestream in a sequence of codestreams
D_{c2d}	number of clock cycles between the first bit written into the decoder smoothing buffer and the decoding start of the first fragment of the stream of codestream fragments
$F_{first}(i)$	first fragment of codestream $C(i)$
$F_{last}(i)$	last fragment of codestream $C(i)$
F_q	number of fractional bits in the representation of wavelet coefficients
H_f	height of the image in sampling grid points
H_p	height of a precinct in lines
H_{max}	maximum image height in sampling grid points
L_{cod}	field in the picture header indicating the codestream size in bytes (see ISO/IEC 21122-1)

L_h long header flag in the picture header

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

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

$l_{dec,avail}(t)$ number of bits that can be read from the decoding smoother buffer in clock cycle t

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

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

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

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

$M_f[p,\lambda,b,g_f]$ frame buffer bitplane count for a group g_f in precinct p at line λ of band b

N_c number of components in an image

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

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

$N_{cg}(f)$ 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_{bpp} nominal number of bits allocated per pixel for compression

$N_{bpp,max}$ maximum number of decoded bits per pixel

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

N_g number of coefficients in a code group

$N_{L,x}$ maximum number of horizontal decomposition levels of all components

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

$N_{p,cg}$ number of pixels in one coefficient group

$N_{p,x}$ 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

N all integer numbers being strictly larger than zero

N_0 all integer numbers being greater than or equal to zero

N_{fg} number of frame buffer wavelet coefficients within one frame buffer group

P_{lev} level and sublevel indication of a codestream

$Q_f[p]$ quantization parameter to which precinct p is quantized for storage in the frame buffer

Q_{pih}	quantization type
$\mathbb{Q}$	set of rational numbers
$R_{\mathrm{f}}[p]$	refinement parameter of the quantization to which precinct p is quantized for storage in the frame buffer
$R_{\mathrm{s,max}}$	maximum grid point sample rate (in samples per second) at decoder output
$R_{\mathrm{t,fb,max}}$	maximum bi-directional frame buffer bandwidth
$R_{\mathrm{t,max}}(l_{\mathrm{m}}, l_{\mathrm{s}})$	maximum admissible encoded throughput in bits per second for a given level
R_{trans}	transmission channel capacity, expressed in bits per clock cycle
$r_{\mathrm{dec}}(t)$	number of bits read and removed from the decoder smoothing buffer in clock cycle t
$S_{\mathrm{c,max}}$	targeted maximum number of bytes of a codestream
$S_{\mathrm{bits}}(f)$	number of bits forming the codestream fragment f
S_{d}	number of components for which the wavelet decomposition is suppressed
$S_{\mathrm{sbo}}(p)$	smoothing buffer offset in bits for a profile p
$S_{\mathrm{sbu}}(l_{\mathrm{m}}, l_{\mathrm{s}})$	size of the smoothing buffer unit in bytes for level l_{m} and sublevel l_{s}
$S_{\mathrm{sl,max}}(l_{\mathrm{m}}, l_{\mathrm{s}})$	maximum size of an encoded codestream in bytes of level l_{m} and sublevel l_{s}
$s_{\mathrm{x}}[i]$	sampling factor of component i in horizontal direction
$s_{\mathrm{y}}[i]$	sampling factor of component i in vertical direction
T_{bmd}	buffer model type
T_{dec}	clock period defining the frequency by which code groups are processed by a decoder
T_{enc}	clock period defining the frequency by which code groups are processed by an encoder
$t_{\mathrm{dec,read}}(f)$	timestamp in cycles at which codestream fragment f is removed from the decoder smoothing buffer
$t_{\mathrm{dec,start}}(f)$	timestamp in cycles at which decoder starts decoding codestream fragment f
$t_{\mathrm{enc,write}}(f)$	timestamp in cycles at which the codestream fragment f is written to the encoder smoothing buffer
$T_{f}[p,b]$	frame buffer truncation point for band b in precinct p
$W_{\mathrm{c}}[i]$	width of component i in samples
$W_{\mathrm{c,max}}$	maximum column width in sampling grid points for a given profile
W_{f}	width of the image in sampling grid points
W_{max}	maximum image width in sampling grid points
$w_{\mathrm{dec}}(t)$	number of bits written into the decoder smoothing buffer in clock cycle t
$W_{\mathrm{pb}}[p,b]$	width of band b of precinct p in coefficients

$W_p[p]$ width of the precinct p in sampling grid points

Z set of all integer numbers

6 Conventions

6.1 Conformance language

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.

6.2 Operators

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

6.2.1 Arithmetic operators

+	addition
−	subtraction (as a binary operator) or negation (as a unary prefix operator)
×	multiplication
/	division without truncation or rounding

6.2.2 Logical operators

||	logical OR
&&	logical AND
!	logical NOT

6.2.3 Relational operators

>	greater than
≥	greater than or equal to
<	less than
≤	less than or equal to
==	equal to
!=	not equal to

6.2.4 Other operators

( )	expression
[ ]	indexing of arrays

6.2.5 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 left-associativity — thus, evaluate from left to right.

( )	expression
[ ]	indexing of arrays
−	unary negation
×, /	multiplication, division
+, −	addition, subtraction
<, >, ≤, ≥, ==, !=	relational comparison

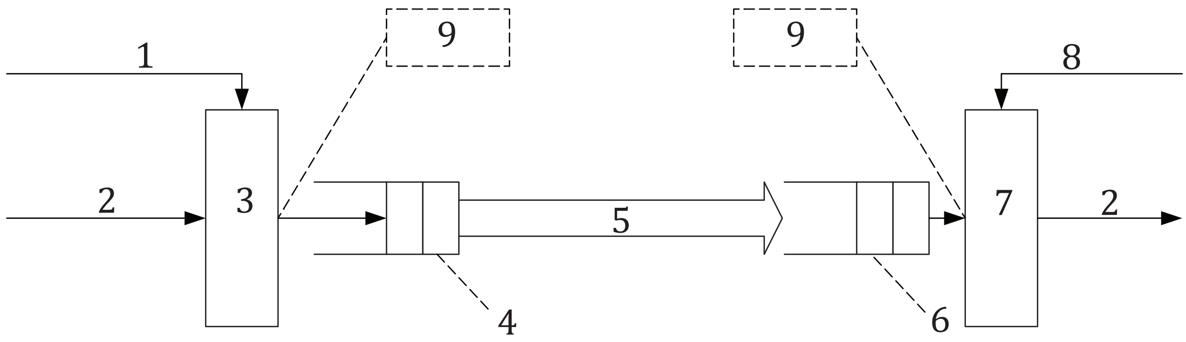
6.2.6 Mathematical functions

$\lceil x \rceil$	ceil of x, equals the smallest integer that is greater than or equal to x
$\lfloor x \rfloor$	floor of x, equals the largest integer that is less than or equal to x
$\lvert x \rvert$	absolute value of x, $\lvert x \rvert = \begin{cases} -x & \text{if } x < 0 \\ x & \text{if } x \geq 0 \end{cases}$
$\text{sign}(x)$	sign of x, $\text{sign}(x) = \begin{cases} -1 & \text{if } x < 0 \\ 0 & \text{if } x = 0 \\ 1 & \text{if } x > 0 \end{cases}$
$\xi(t)$	step function, $\xi(t) = \begin{cases} 0 & \text{if } t < 0 \\ 1 & \text{if } t \geq 0 \end{cases}$
$\max_i(x_i)$	maximum of a sequence of numbers $[x_i]$ enumerated by the index i
∃	the mathematical symbol to represent *there exists*
∀	The mathematical symbol to represent *for all*

7 Buffer model

7.1 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 processes each pixel are determined by an encoding clock. Similarly, the time instances at which the decoder produces 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. Consequently, 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 cycle.

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.

7.2 Influencing variables on the required buffer sizes

Avoiding any buffer overflow or underflow, as discussed in 7.1, requires sizing the decoder smoothing buffer properly. 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 at which 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 provided by means of the buffer model as defined in Annex B and Annex C.

7.3 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 conforming decoder. These conditions are expressed by buffer model parameters that are specified by the profiles and levels defined normatively in Annex A. The properties of such conforming implementations are exemplified in Annex D. Since these properties are direct consequences of normative Annex B and normative Annex C, Annex D is informative only. In addition, an informative latency analysis of each block of the JPEG XS core coding system, as defined in ISO/IEC 21122-1, is provided in Annex E.

8 Interpretation of Bayer data

ISO/IEC 21122-1 defines coding tools and signalling for compression of Bayer-type CFA image data. According to this specification, each sampling grid point represents a super-pixel of four sensor elements containing at least one sample of each component. Thus, Bayer data is interpreted as an image having four components, where each sampling grid point describes four spatially disjoint sensor elements (one element per Bayer channel).

Moreover, regardless of the Bayer sensor spatial subpixel arrangement, the Star-Tetrix colour transform of ISO/IEC 21122-1 defines a strict order on the components assigning the red channel to component 0, the green channels to components 1 and 2, and the blue channel to component 3. The spatial subpixel arrangement is signalled by the CRG marker. Figure 2 shows only one of the four potential subpixel arrangements of a Bayer-type CFA. In this figure, squares represent individual sensor elements and circles represent sampling grid points. Groups of four sensor elements overlapping with the same sampling grid point form one super-pixel.

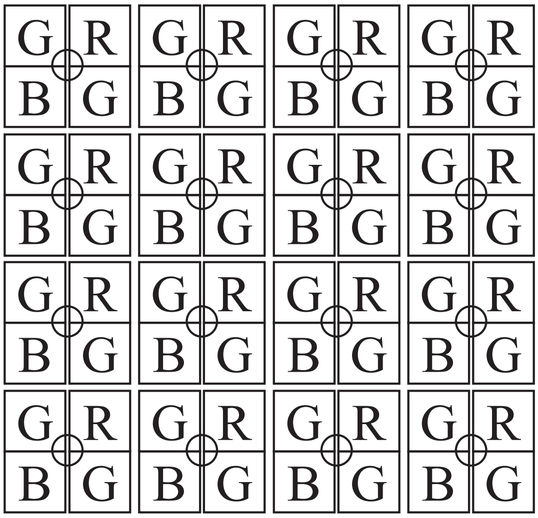


Figure 2 — Example of the interpretation of a GRBG Bayer-type CFA image

9 Conformance

The syntax of JPEG XS codestreams specified in ISO/IEC 21122-1 comes with a set of compression coding tools, capabilities, parameters, and configuration possibilities, for which Annex A provides the normative constraints. As such, compliant JPEG XS codestreams and decoder implementations shall be in accordance with the provisions given in Annex A. In addition, compliant JPEG XS codestreams and decoder implementations shall also be in accordance with the packet-based JPEG XS decoder model as described in Annex B and the packet-based constant bit rate buffer model as described in Annex C.

Annex A
(normative)

Profiles, levels, sublevels and frame buffer bandwidth levels

A.1 General

Profiles, levels, sublevels and frame buffer bandwidth (FBB) levels specify restrictions on codestreams and hence limits on the capabilities needed to decode the codestreams. Profiles, levels, sublevels and FBB levels may also be used to indicate interoperability points between individual decoder implementations. Additionally, A.8 defines conformance points that further constrain the allowed combinations of levels, sublevels and FBB levels for specific profiles.

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 encoded 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 A.4.

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.

In addition, codestreams that use the `SLI` marker of ISO/IEC 21122-1 need the support of a frame buffer to be decoded. In this context, a codestream is *TDC enabled* when it contains at least one `SLI` marker (see ISO/IEC 21122-1), and *TDC disabled* otherwise. The FBB levels provide a lower bound on the throughput that a conforming decoder implementation shall support for accessing its frame buffer. From the codestream perspective, FBB levels define an upper bound on the required bandwidth to access the frame buffer. By means of the FBB level, this annex specifies the FBB model that places constraints on the usage of the frame buffer by the codestream.

NOTE Even though the definition for *TDC enabled* codestreams requires *at least one* `SLI` marker, currently none of the profiles allow mixing `SLH` and `SLI` slice headers in a codestream.

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

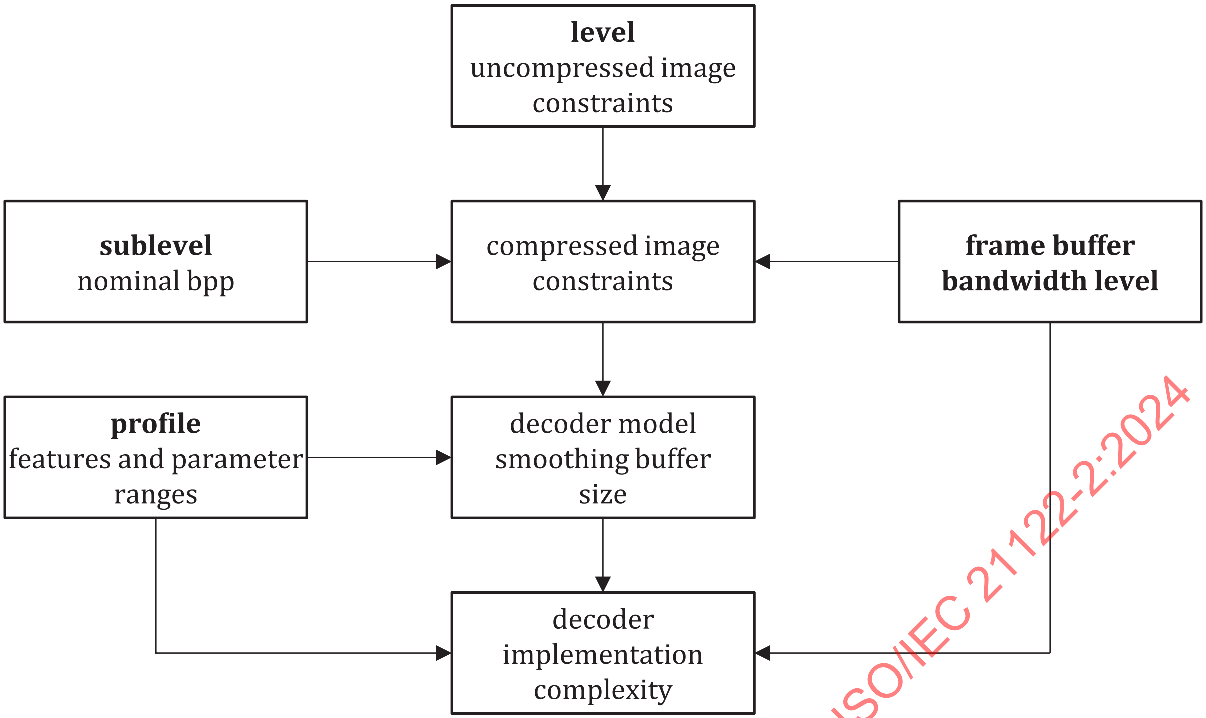


Figure A.1 — Relationship between the different conformance constraints and the impact on the decoder complexity

A.2 Profiles

A.2.1 Definition of profiles

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

— component bit precision ($B[i]$);

— maximum bits per pixel (bpp) in the decoded domain ($N_{bpp,max}$);

— internal precision (B_w , see ISO/IEC 21122-1);

— number of bits to encode a bitplane count in raw (B_r , see ISO/IEC 21122-1);

— number of fractional bits for DWT coefficients (F_q , see ISO/IEC 21122-1);

— non-linear transform;

— raw-mode selection per packet flag (R_l , see ISO/IEC 21122-1);

— chroma sampling formats;

— colour transformation (C_{pih} , see ISO/IEC 21122-1);

— size and extent of the colour transformation (C_f , see ISO/IEC 21122-1 and NOTE 1);

NOTE 1 The size and extent of the colour transformation is signalled in the CTS marker when C_{pih} is set to 3 (see ISO/IEC 21122-1). For other values of C_{pih} , the CTS marker is not present.

— number of vertical wavelet decompositions ($N_{L,y}$, see ISO/IEC 21122-1);

— number of horizontal wavelet decompositions ($N_{L,x}$, see ISO/IEC 21122-1 and NOTE 2);

NOTE 2 As defined in ISO/IEC 21122-1, the number of vertical wavelet decompositions is always lower than or equal to the number of horizontal wavelet decompositions.

— number of components for which to suppress the wavelet decomposition (S_d , see ISO/IEC 21122-1);

— supported quantizer types (Q_{pih} , see ISO/IEC 21122-1);

— maximum column width (C_w , see ISO/IEC 21122-1 and NOTE 3);

NOTE 3 The column width in sampling grid points is given by $\begin{cases} 8 \times C_w \times \max_i (s_x[i]) \times 2^{N_{L,x}} & \text{if } C_w > 0 \\ W_f & \text{otherwise} \end{cases}$ where C_w is indicated in the picture header (see ISO/IEC 21122-1), W_f is the image width, $s_x[i]$ is the horizontal sampling factor for component i , and $N_{L,x}$ is the number of horizontal wavelet decompositions.

— temporal prediction control marker usage (TPC, see ISO/IEC 21122-1);

— slice headers (SLH and SLI usage, see ISO/IEC 21122-1);

— slice height;

— frame buffer bandwidth model group size (N_{fg} , see A.7.2);

— decoder smoothing buffer size expressed in smoothing buffer units (N_{sbu} , see NOTE 4);

NOTE 4 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 A.5, Formula (A.3) and A.5, Formula (A.4).

— smoothing buffer offset (S_{sbo} , see NOTE 5);

NOTE 5 The commonly used value of 1 024 bits (128 bytes) has been derived from a typical size of the picture header without any extension markers.

— buffer model (T_{bmd} , see NOTE 6);

NOTE 6 A decoder that supports $T_{bmd} = 2$ automatically also supports $T_{bmd} = 1$.

— long header enforcement flag (L , see ISO/IEC 21122-1).

For all profiles, the bit precision $B[i]$ ($0 \le i < N_c$) of all components shall be identical.

Profile parameter values that allow the choice between more than one value shall always be selected in accordance with ISO/IEC 21122-1.

Table A.1, Table A.2, Table A.3, Table A.4, Table A.5, Table A.6, and Table A.7 list all the profiles specified in this document.

Table A.1 — JPEG XS Main profiles

Profile	Main 420.12	Main 422.10	Main 444.12	Main 4444.12
P_{pih} field	0x3240	0x3540	0x3A40	0x3E40
Component bit precision $B[i]$	8, 10, 12	8, 10	8, 10, 12	8, 10, 12
Maximum decoded bpp $N_{bpp,max}$	18	20	36	48
Internal precision B_w	20	20	20	20
# bits to encode a bitplane count in raw B_r	4	4	4	4
# fractional bits for DWT coefficients F_q	8	8	8	8
Non-linear transform	Disallowed	Disallowed	Disallowed	Disallowed
Raw-mode selection per packet flag R_l	0	0	0	0
Chroma sampling formats	4:2:0	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
Colour transformation C_{pih}	0 (None)	0 (None)	0 (None) for any sampling format, or optionally 1 (RCT) for 4:4:4	0 (None) for any sampling format, or optionally 1 (RCT) for 4:4:4 and 4:4:4:4
Number of vertical decompositions $N_{L,y}$	1	0, 1	0, 1	0, 1
Number of horizontal decompositions $N_{L,x}$	1-5	1-5	1-5	1-5
Number components with suppressed decomposition S_d	0	0	0	0
Quantizer type Q_{pih}	0 (DZQ), 1 (Uniform)	0 (DZQ), 1 (Uniform)	0 (DZQ), 1 (Uniform)	0 (DZQ), 1 (Uniform)
Column mode C_w	One column of full image width	One column except when the number of vertical decomposition levels is zero[a]	One column except when the number of vertical decomposition levels is zero[a]	One column except when the number of vertical decomposition levels is zero[a]
TPC marker	Disallowed	Disallowed	Disallowed	Disallowed
Slice headers	SLH only	SLH only	SLH only	SLH only
Slice height in number of image rows	16	16	16	16
Number of smoothing buffer units, N_{sbu}, of the decoder model	16	16	16	16
Smoothing buffer offset S_{sbo} in bits	1 024	1 024	1 024	1 024
Buffer model T_{bmd}	1, 2	1, 2	1, 2	1, 2
Long header enforcement flag L_h	0	0	0	0

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

Table A.2 — JPEG XS Light profiles

Profile	Light 422.10	Light 444.12	Light-Subline 422.10
P_{pih} field	0x1500	0x1A00	0x2500
Component bit precision B[i]	8, 10	8, 10, 12	8, 10
Maximum decoded bpp $N_{bpp,max}$	20	36	20
Internal precision B_w	20	20	20
# bits to encode a bitplane count in raw B_r	4	4	4
# fractional bits for DWT coefficients F_q	8	8	8
Non-linear transform	Disallowed	Disallowed	Disallowed
Long header enforcement flag L_h	0	0	0
Raw-mode selection per packet flag R_l	0	0	0
Chroma sampling formats	4:0:0, 4:2:2	4:0:0, 4:2:2, 4:4:4	4:0:0, 4:2:2
Colour transformation C_{pih}	0 (None)	0 (None) for any sampling format, or optionally 1 (RCT) for 4:4:4	0 (None)
Number of vertical decompositions $N_{L,y}$	0, 1	0, 1	0
Number of horizontal decompositions $N_{L,x}$	1-5	1-5	1-5
Number components with suppressed decomposition S_d	0	0	0
Quantizer type Q_{pih}	0 (DZQ)	0 (DZQ)	0 (DZQ), 1 (Uniform)
Column mode C_w	One column of full image width	One column of full image width	Maximum column width of 2 048 grid points
TPC marker	Disallowed	Disallowed	Disallowed
Slice headers	SLH only	SLH only	SLH only
Slice height in number of image rows	16	16	16
Number of smoothing buffer units, N_{sbu} , of the decoder model	4	4	2
Smoothing buffer offset S_{sbo} in bits	1 024	1 024	1 024
Buffer model T_{bmd}	1, 2	1, 2	1, 2

Table A.3 — JPEG XS High profiles

Profile	High 420.12	High 444.12	High 4444.12
P_{pih} field	0x4240	0x4A40	0x4E40
Component bit precision B[i]	8, 10, 12	8, 10, 12	8, 10, 12
Maximum decoded bpp $N_{bpp,max}$	18	36	48
Internal precision B_w	20	20	20
# bits to encode a bitplane count in raw B_r	4	4	4
# fractional bits for DWT coefficients F_q	8	8	8

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

Table A.3 *(continued)*

Profile	High 420.12	High 444.12	High 4444.12
Non-linear transform	Disallowed	Disallowed	Disallowed
Long header enforcement flag L_h	0	0	0
Raw-mode selection per packet flag R_l	0	0	0
Chroma sampling formats	4:2:0	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
Colour transformation C_{pih}	0 (None)	0 (None) for any sampling format, or optionally 1 (RCT) for 4:4:4	0 (None) for any sampling format, or optionally 1 (RCT) for 4:4:4 and 4:4:4:4
Number of vertical decompositions $N_{L,y}$	1, 2	0, 1, 2	0, 1, 2
Number of horizontal decompositions $N_{L,x}$	1-5	1-5	1-5
Number components with suppressed decomposition S_d	0	0	0
Quantizer type Q_{pih}	0 (DZQ), 1 (Uniform)	0 (DZQ), 1 (Uniform)	0 (DZQ), 1 (Uniform)
Column mode C_w	One column of full image width	One column except when the number of vertical decomposition levels is zero[a]	One column except when the number of vertical decomposition levels is zero[a]
`TPC` marker	Disallowed	Disallowed	Disallowed
Slice headers	`SLH` only	`SLH` only	`SLH` only
Slice height in number of image rows	16	16	16
Number of smoothing buffer units, N_{sbu} , of the decoder model	16	16	16
Smoothing buffer offset S_{sbo} in bits	1 024	1 024	1 024
Buffer model T_{bmd}	1, 2	1, 2	1, 2

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

Table A.4 — JPEG XS Additional High profiles

Profile	CHigh 444.12[a]
P_{pih} field	`0x4A44`
Component bit precision B[i]	8, 10, 12
Maximum decoded bpp $N_{bpp,max}$	36
Internal precision B_w	20
# bits to encode a bitplane count in raw B_r	4
# fractional bits for DWT coefficients F_q	8
Non-linear transform	Disallowed
Long header enforcement flag L_h	0

[a] Additional restrictions, specified in A.8, apply when using this profile.

[b] Conforming to ISO/IEC 21122-1, zero (0) vertical decompositions cannot be used in combination with 4:2:0 chroma sampling.

Table A.4 *(continued)*

Profile	CHigh 444.12[a]
Raw-mode selection per packet flag R_l	1
Chroma sampling formats	4:0:0, 4:2:0, 4:2:2, 4:4:4
Colour transformation C_{pih}	0 (None) for any sampling format, or optionally 1 (RCT) for 4:4:4
Wavelet decompositions ($N_{L,x}$, $N_{L,y}$)	(3, 0) if not 4:2:0[b], (4, 0) if not 4:2:0[b], (4, 1), (5, 1), (5, 2)
Number components with suppressed decomposition S_d	0
Quantizer type Q_{pih}	0 (DZQ), 1 (Uniform)
Column mode C_w	0 (disallowed)
`TPC` marker	Disallowed
Slice headers	`SLH` only
Slice height in number of image rows	16
Number of smoothing buffer units, N_{sbu} , of the decoder model	16
Smoothing buffer offset S_{sbo} in bits	1 024
Buffer model T_{bmd}	1, 2

[a] Additional restrictions, specified in A.8, apply when using this profile.

[b] Conforming to ISO/IEC 21122-1, zero (0) vertical decompositions cannot be used in combination with 4:2:0 chroma sampling.

Table A.5 — JPEG XS TDC profiles

Profile	TDC 444.12[a]	TDC MLS 444.12[a]
P_{pih} field	`0x4A45`	`0x6A45`
Component bit precision B[i]	8, 10, 12	8, 10, 12
Maximum decoded bpp $N_{bpp,max}$	36	36
Internal precision B_w	20	Component precision
# bits to encode a bitplane count in raw B_r	4	4
# fractional bits for DWT coefficients F_q	8	0
Non-linear transform	Disallowed	Disallowed
Long header enforcement flag L_h	0	0
Raw-mode selection per packet flag R_l	1	1
Chroma sampling formats	4:0:0, 4:2:0, 4:2:2, 4:4:4	4:0:0, 4:2:0, 4:2:2, 4:4:4

[a] Additional restrictions, specified in A.8, apply when using this profile.

[b] Conforming to ISO/IEC 21122-1, zero (0) vertical decompositions cannot be used in combination with 4:2:0 chroma sampling.

Table A.5 *(continued)*

Profile	TDC 444.12[a]	TDC MLS 444.12[a]
Colour transformation C_{pih}	0 (None) for any sampling format, or optionally 1 (RCT) for 4:4:4	0 (None) for any sampling format, or optionally 1 (RCT) for 4:4:4
Wavelet decompositions ($N_{L,x}$, $N_{L,y}$)	(3, 0) if not 4:2:0[b], (4, 0) if not 4:2:0[b], (4, 1), (5, 1), (5, 2)	(3, 0) if not 4:2:0[b], (4, 0) if not 4:2:0[b], (4, 1), (5, 1), (5, 2)
Number components with suppressed decomposition S_d	0	0
Quantizer type Q_{pih}	0 (DZQ), 1 (Uniform)	0 (DZQ), 1 (Uniform)
Column mode C_w	0 (disallowed)	0 (disallowed)
TPC marker	Optional (allowed)	Optional (allowed)
Slice headers	SLI only	SLI only
Slice height in number of image rows	16	16
Frame buffer bandwidth model group size N_{fg}	8	8
Number of smoothing buffer units, N_{sbu} , of the decoder model	16	16
Smoothing buffer offset S_{sbo} in bits	1 024	1 024
Buffer model T_{bmd}	1, 2	1, 2

[a] Additional restrictions, specified in A.8, apply when using this profile.

[b] Conforming to ISO/IEC 21122-1, zero (0) vertical decompositions cannot be used in combination with 4:2:0 chroma sampling.

Table A.6 — JPEG XS MLS profiles

Profile	MLS.12	MLS.16
P_{pih} field	0x6EC0	0x6ED0
Component bit precision B[i]	8, 10, 12	8, 10, 12, 14, 16
Maximum decoded bpp $N_{bpp,max}$	48	64
Internal precision B_w	Component precision	Component precision
# bits to encode a bitplane count in raw B_r	4	5
# fractional bits for DWT coefficients F_q	0	0
Non-linear transform	Disallowed	Disallowed
Long header enforcement flag L_h	0	0
Raw-mode selection per packet flag R_l	0	0
Chroma sampling formats	4:0:0, 4:2:0, 4:2:2, 4:4:4, 4:2:2:4, 4:4:4:4	4:0:0, 4:2:0, 4:2:2, 4:4:4, 4:2:2:4, 4:4:4:4

[a] Conforming with ISO/IEC 21122-1, zero (0) vertical decompositions cannot be used in combination with 4:2:0 chroma sampling.

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

[c] When T_{bmd} is set to 0 (Unconstrained), the S_{sbo} value becomes irrelevant because the buffer model assumes an infinite large buffer exists (see C.5, Formula (C.9)).

Table A.6 *(continued)*

Profile	MLS.12	MLS.16
Colour transformation C_{pih}	0 (None) for any sampling format, or optionally 1 (RCT) for 4:4:4 and 4:4:4:4	0 (None) for any sampling format, or optionally 1 (RCT) for 4:4:4 and 4:4:4:4
Number of vertical decompositions $N_{L,y}$	0 if not 4:2:0[a], 1, 2	0 if not 4:2:0[a], 1, 2
Number of horizontal decompositions $N_{L,x}$	1-5	1-5
Number components with suppressed decomposition S_d	0	0
Quantizer type Q_{pih}	0 (DZQ), 1 (Uniform)	0 (DZQ), 1 (Uniform)
Column mode C_w	One column except when the number of vertical decomposition levels is zero[b]	One column except when the number of vertical decomposition levels is zero[b]
TPC marker	Disallowed	Disallowed
Slice headers	SLH only	SLH only
Slice height in number of image rows	16	16
Number of smoothing buffer units, N_{sbu}, of the decoder model	Unconstrained	Unconstrained
Smoothing buffer offset S_{sbo} in bits	0[c]	0[c]
Buffer model T_{bmd}	0 (Unconstrained)	0 (Unconstrained)

[a] Conforming with ISO/IEC 21122-1, zero (0) vertical decompositions cannot be used in combination with 4:2:0 chroma sampling.

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

[c] When T_{bmd} is set to 0 (Unconstrained), the S_{sbo} value becomes irrelevant because the buffer model assumes an infinite large buffer exists (see C.5, Formula (C.9)).

Table A.7 — JPEG XS Bayer profiles

Profile	LightBayer	MainBayer	HighBayer
P_{pih} field	0x9300	0xB340	0xC340
Component bit precision B[i]	10, 12, 14, 16	10, 12, 14, 16	10, 12, 14, 16
Maximum decoded bpp $N_{bpp,max}$	64	64	64
Internal precision B_w	18 if NLT is used, 20 if NLT is not used[a]	18 if NLT is used, 20 if NLT is not used[a]	18 if NLT is used, 20 if NLT is not used[a]
# bits to encode a bitplane count in raw B_r	4	4	4
# fractional bits for DWT coefficients F_q	6 if $B_w = 18$, 8 if $B_w = 20$ [b]	6 if $B_w = 18$, 8 if $B_w = 20$ [b]	6 if $B_w = 18$, 8 if $B_w = 20$ [b]
Non-linear transform	None, Quadratic, Extended	None, Quadratic, Extended	None, Quadratic, Extended
Long header enforcement flag L_h	0, 1	0, 1	0, 1

[a] The internal precision, B_w, is selected in conformance with ISO/IEC 21122-1. The value 20 is used when the non-linear transform is not used (i.e. no NLT marker is present), otherwise the value 18 is used (i.e. the NLT marker is present).

[b] The number of fractional bits for DWT coefficients, F_q, is selected as specified in ISO/IEC 21122-1. The value 6 is used when B_w is set to 18, while the value 8 is used when B_w is set to 20.

[c] In this profile, with zero (0) vertical decompositions, the raw-mode selection per packet flag will not influence the resulting codestream. However, it is set to 1 to match with the MainBayer and HighBayer profiles.

Table A.7 *(continued)*

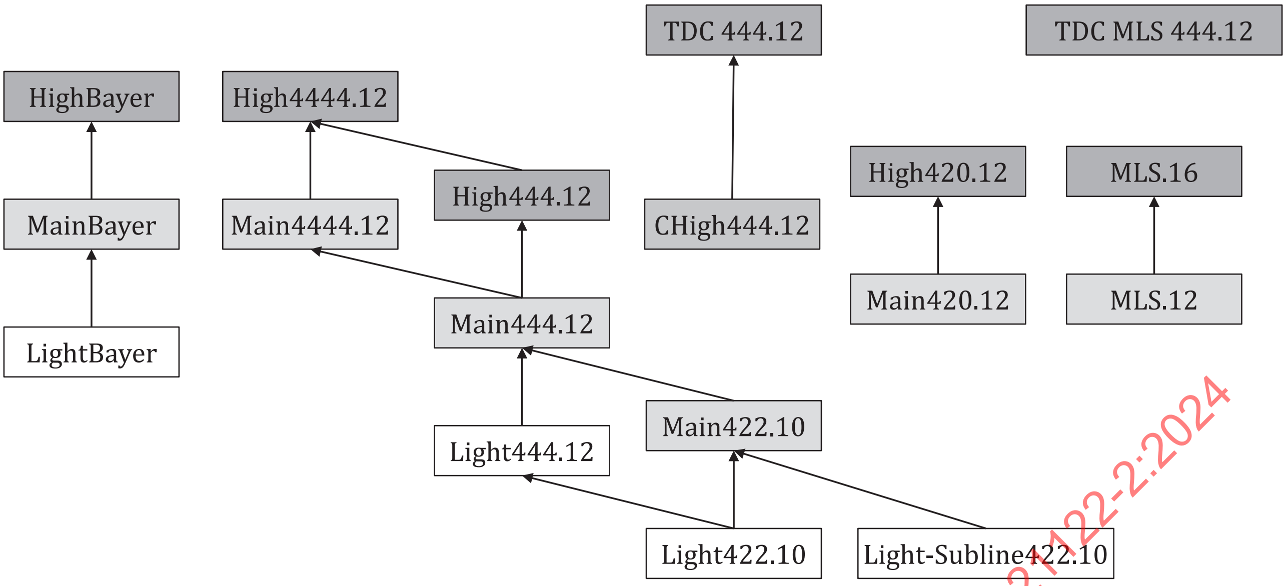
Profile	LightBayer	MainBayer	HighBayer
Raw-mode selection per packet flag R_l	1[c]	1	1
Chroma sampling formats	Bayer pattern interpreted as 4-dimensional vectors	Bayer pattern interpreted as 4-dimensional vectors	Bayer pattern interpreted as 4-dimensional vectors
Colour transformation C_{pih}	3 (Star-Tetrix)	3 (Star-Tetrix)	3 (Star-Tetrix)
Size and extent of the colour transformation C_f	3 (Inline)	0 (Full), 3 (Inline)	0 (Full), 3 (Inline)
Number of vertical decompositions $N_{L,y}$	0	0, 1	0, 1, 2
Number of horizontal decompositions $N_{L,x}$	1-5	1-5	1-5
Number components with suppressed decomposition S_d	1	1	1
Quantizer type Q_{pih}	0 (DZQ), 1 (Uniform)	0 (DZQ), 1 (Uniform)	0 (DZQ), 1 (Uniform)
Column mode C_w	0 (disallowed)	0 (disallowed)	0 (disallowed)
TPC marker	Disallowed	Disallowed	Disallowed
Slice headers	SLH only	SLH only	SLH only
Slice height in number of image rows	16	16	16
Number of smoothing buffer units, N_{sbu}, of the decoder model	4	8	16
Smoothing buffer offset S_{sbo} in bits	1 024	1 024	1 024
Buffer model T_{bmd}	1, 2	1, 2	1, 2

[a] The internal precision, B_w, is selected in conformance with ISO/IEC 21122-1. The value 20 is used when the non-linear transform is not used (i.e. no NLT marker is present), otherwise the value 18 is used (i.e. the NLT marker is present).

[b] The number of fractional bits for DWT coefficients, F_q, is selected as specified in ISO/IEC 21122-1. The value 6 is used when B_w is set to 18, while the value 8 is used when B_w is set to 20.

[c] In this profile, with zero (0) vertical decompositions, the raw-mode selection per packet flag will not influence the resulting codestream. However, it is set to 1 to match with the MainBayer and HighBayer profiles.

Figure A.2 represents the relation of the profiles defined in Table A.1, Table A.2, Table A.3, Table A.4, Table A.5, Table A.6, and Table A.7 in terms of inclusivity. An arrow in the figure represents an implicit *... is included in ...* relation due to the profile specifications in Table A.1, Table A.2, Table A.3, Table A.4, Table A.5, Table A.6, and Table A.7.



NOTE This figure does not formulate any additional constraints on decoder implementations. The relations presented there are implicit due to the profile specifications in Table A.1, Table A.2, Table A.3, Table A.4, Table A.5, Table A.6, and Table A.7. That is, a codestream conforming to a given profile *P* in this figure automatically conforms to a profile *Q* provided there is a path from *P* to *Q* in the direction of the arrows.

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

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 P_{pih} field of the picture header (see ISO/IEC 21122-1) by the respective values as defined in Table A.1, Table A.2, Table A.3, Table A.4, Table A.5, Table A.6, and Table A.7.

The `Unrestricted` profile, indicated by having P_{pih} equal to `0x0000`, 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.

All other values for P_{pih} are reserved for ISO/IEC purposes.

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

A.3 Levels

A.3.1 Definition of 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 allowed 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. These levels apply to all the profiles defined. Table A.8 de fines all available levels.

Additionally, an `Unrestricted` level exists that does not impose any constraint on maximum image width, maximum image height, maximum number of grid point samples, or maximum grid point sample rate. The `Unrestricted` level shall not be considered as conformance point.

Table A.8 — JPEG XS levels

<table>
<tr><th>Level[a]</th><th>Maximum picture width W_{max} (sampling grid points)[b]</th><th>Maximum image height H_{max} (sampling grid points)[b]</th><th>Maximum number of sampling grid points L_{max} per image (sampling grid points)[b]</th><th>Maximum grid point sample rate $R_{s,max}$ (sampling grid points/s)[bc]</th><th>Example resolutions (in sampling grid points)[d]</th></tr>
<tr><td><code>1k-1</code></td><td rowspan="2">1 280</td><td rowspan="2">5 120</td><td rowspan="2">2 621 440</td><td rowspan="2">83 558 400</td><td>1 280×720@60</td></tr>
<tr><td><code>Bayer2k-1</code></td><td>2 048×1 080@60
2 560×1 440@60</td></tr>
<tr><td><code>2k-1</code></td><td rowspan="2">2 048</td><td rowspan="2">8 192</td><td rowspan="2">4 194 304</td><td rowspan="2">133 693 440</td><td>1 280×720@120
1 920×1 080@30
1 920×1 080@60
2 048×1 536@30
2 048×2 048@30
2 048×1 080@60</td></tr>
<tr><td><code>Bayer4k-1</code></td><td>2 560×1 440@120
3 840×2 160@30
3 840×2 160@60
4 096×3 072@30
4 096×4 096@30
4 096×2 160@60</td></tr>
<tr><td><code>4k-1</code></td><td rowspan="2">4 096</td><td rowspan="2">16 384</td><td rowspan="2">8 912 896</td><td rowspan="2">267 386 880</td><td>1 920×1 080@120
3 840×2 160@30
4 096×2 160@30</td></tr>
<tr><td><code>Bayer8k-1</code></td><td>3 840×2 160@120
7 680×4 320@30
8 192×4 320@30</td></tr>
<tr><td><code>4k-2</code></td><td rowspan="2">4 096</td><td rowspan="2">16 384</td><td rowspan="2">16 777 216</td><td rowspan="2">534 773 760</td><td>1 920×1 080@240
3 840×2 160@60
4 096×3 072@30
4 096×4 096@30
4 096×2 160@60</td></tr>
<tr><td><code>Bayer8k-2</code></td><td>3840×2160@240
7 680×4 320@60
8 192×6 144@30
8 192×8 192@30
8 192×4 320@60</td></tr>
</table>

[a] Levels have double names to allow logical association with their supported resolutions. In the case of Bayer data, each sampling grid represents a super-pixel. This means that the total number of sampling grid points required to represent a Bayer image is four times smaller (i.e. half width and half height) than the total number of sensor elements.

[b] 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.

[c] The maximum number of sampling grid points is not identical to the product of the maximum image height and the maximum image width.

[d] In case of non-Bayer pattern data, each sampling grid point represents a set of samples over the components. In the case of Bayer pattern data, each sampling grid point represents a super-pixel, i.e. a 2×2 arrangement of sensor elements.

Table A.8 *(continued)*

Level[a]	Maximum picture width W_{max} (sampling grid points)[b]	Maximum image height H_{max} (sampling grid points)[b]	Maximum number of sampling grid points L_{max} per image (sampling grid points)[b]	Maximum grid point sample rate $R_{s,max}$ (sampling grid points/s)[bc]	Example resolutions (in sampling grid points)[d]
4k-3	4 096	16 384	16 777 216	1 069 547 520	1 920×1 080@480 2 048x1 080@480 4 096×3 072@60 4 096×4 096@60 3 840×2 160@120 4 096×2 160@120
Bayer8k-3					3 840×2 160@480 4 096×2 160@480 8 192×6 144@60 8 192×8 192@60 7 680×4 320@120 8 192×4 320@120
8k-1	8 192	32 768	35 651 584	1 069 547 520	3 840×2 160@120 7 680×4 320@30 8 192×4 320@30
Bayer16k-1					7 680×4 320@120 15 360×8 640@30 16 384×8 640@30
8k-2	8 192	32 768	67 108 864	2 139 095 040	3 840×2 160@240 8 192×6 144@30 8 192×8 192@30 7 680×4 320@60 8 192×4 320@60
Bayer16k-2					7 680×4 320@240 16 384×12 288@30 16 384×16 384@30 15 360×8 640@60 16 384×8 640@60
8k-3	8 192	32 768	67 108 864	4 278 190 080	3 840×2 160@480 4 096×2 160@480 8 192×6 144@60 8 192×8 192@60 7 680×4 320@120 8 192×4 320@120
Bayer16k-3					7 680×4 320@480 8 192×4 320@480 16 384×12 288@60 16 384×16 384@60 15 360×8 640@120 16 384×8 640@120

[a] Levels have double names to allow logical association with their supported resolutions. In the case of Bayer data, each sampling grid represents a super-pixel. This means that the total number of sampling grid points required to represent a Bayer image is four times smaller (i.e. half width and half height) than the total number of sensor elements.

[b] 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.

[c] The maximum number of sampling grid points is not identical to the product of the maximum image height and the maximum image width.

[d] In case of non-Bayer pattern data, each sampling grid point represents a set of samples over the components. In the case of Bayer pattern data, each sampling grid point represents a super-pixel, i.e. a 2×2 arrangement of sensor elements.

Table A.8 *(continued)*

Level[a]	Maximum picture width W_{max} (sampling grid points)[b]	Maximum image height H_{max} (sampling grid points)[b]	Maximum number of sampling grid points L_{max} per image (sampling grid points)[b]	Maximum grid point sample rate $R_{s,max}$ (sampling grid points/s)[bc]	Example resolutions (in sampling grid points)[d]
10k-1	10 240	40 960	104 857 600	3 342 336 000	10 240×7 680@30 10 240×10 240@30 10 240×4 320@60 10 240×5 400@60
Bayer20k-1					20 480×15 360@30 20 480×20 480@30 20 480×8 640@60 20 480×10 800@60

[a] Levels have double names to allow logical association with their supported resolutions. In the case of Bayer data, each sampling grid represents a super-pixel. This means that the total number of sampling grid points required to represent a Bayer image is four times smaller (i.e. half width and half height) than the total number of sensor elements.

[b] 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.

[c] The maximum number of sampling grid points is not identical to the product of the maximum image height and the maximum image width.

[d] In case of non-Bayer pattern data, each sampling grid point represents a set of samples over the components. In the case of Bayer pattern data, each sampling grid point represents a super-pixel, i.e. a 2×2 arrangement of sensor elements.

A.3.2 Level signalling in the picture header

The level shall be indicated in the P_{lev} field of the picture header defined in ISO/IEC 21122-1 by the values defined in Table A.9.

Table A.9 — Signalling of the levels of a codestream in the P_{lev} field

Level	Binary value of P_{lev} field[a]
Unrestricted	0000 00XX XXXX XXXX
1k-1	0000 01XX XXXX XXXX
Bayer2k-1	
2k-1	0001 00XX XXXX XXXX
Bayer4k-1	
4k-1	0010 00XX XXXX XXXX
Bayer8k-1	
4k-2	0010 01XX XXXX XXXX
Bayer8k-2	
4k-3	0010 10XX XXXX XXXX
Bayer8k-3	
8k-1	0011 00XX XXXX XXXX
Bayer16k1	
8k-2	0011 01XX XXXX XXXX
Bayer16k-2	
8k-3	0011 10XX XXXX XXXX
Bayer16k-3	
10k-1	0100 00XX XXXX XXXX
Bayer20k-1	
Reserved for ISO/IEC purposes	all other values

[a] An X indicates either a 0 or a 1, as defined in Table A.11 and Table A.19.

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) (see A.5, NOTE 1) value N_{bpp} giving the maximum number of bits per pixel for an encoded image of maximum permissible number of sampling grid points according to the profile and level to which the decoder is conforming (see A.2 and A.3).

Table A.10 lists the sublevels defined in this standard. The `Full` sublevel shall only be used if the profile value (P_{pih}) is not `Unrestricted`.

Additionally, an `Unrestricted` sublevel exists that does not impose any constraint on the nominal bits per pixel. The `Unrestricted` sublevel shall not be considered as conformance point.

Table A.10 — JPEG XS sublevels

Sublevel	Nominal bpp, N_{bpp}
`Full`	$N_{bpp,max}$
`Sublev12bpp`	12
`Sublev9bpp`	9
`Sublev6bpp`	6
`Sublev4bpp`	4
`Sublev3bpp`	3
`Sublev2bpp`	2

A.4.2 Sublevel signalling in the picture header

The sublevel shall be indicated in the P_{lev} field of the picture header defined in ISO/IEC 21122-1 by the values defined in Table A.11.

Table A.11 — Signalling of the sublevels of a codestream in the P_{lev} field

Sublevel	Binary value of P_{lev} field[a]
`Unrestricted`	`XXXX XXXX 0XX0 0000`
`Full`	`XXXX XXXX 1XX0 0000`
`Sublev12bpp`	`XXXX XXXX 0XX1 0000`
`Sublev9bpp`	`XXXX XXXX 0XX0 1100`
`Sublev6bpp`	`XXXX XXXX 0XX0 1000`
`Sublev4bpp`	`XXXX XXXX 0XX0 0110`
`Sublev3bpp`	`XXXX XXXX 0XX0 0100`
`Sublev2bpp`	`XXXX XXXX 0XX0 0011`
Reserved for ISO/IEC purposes	all other values

[a] An X indicates either a 0 or a 1, as defined in Table A.9 and Table A.19.

A.5 Level and sublevel conformance

Decoders conforming to a particular level and sublevel shall comply to the following constraints derived from the level and sublevel:

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

$$S_{\text{sl,max}} = \left\lfloor \frac{L_{\text{max}} \times N_{\text{bpp}}}{8} \right\rfloor \tag{A.1}$$

NOTE 1 The bits per pixel is here referred to as the total amount of bits required across all the component sample values at a given sampling grid point. For Bayer images this represents the total amount of bits required to represent a super-pixel (see Clause 8).

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

$$R_{\text{t,max}} = R_{\text{s,max}} \times N_{\text{bpp}} \tag{A.2}$$

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.

$$S_{\text{sbu}} = \begin{cases} \infty & \text{if either level or sublevel is Unrestricted} \\ W_{\text{c,max}} \times N_{\text{bpp}} & \text{otherwise} \end{cases} \tag{A.3}$$

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

$$W_{\text{c,max}} = \begin{cases} 2{,}048 & \text{if profile is Light-Subline 422.10} \\ W_{\text{max}} & \text{otherwise} \end{cases} \tag{A.4}$$

The actual column width is computed by

$$C_{\text{s}} = \begin{cases} 8 \times C_{\text{w}} \times \max_i (s_x[i]) \times 2^{N_{L,x}} & \text{if } C_{\text{w}} > 0 \\ W_{\text{f}} & \text{otherwise} \end{cases} \tag{A.5}$$

$W_{\text{c,max}}$ is an upper bound for the allowed column width, $C_{\text{s}} \leq W_{\text{c,max}}$.

The nominal bits per pixel, N_{bpp}, is not identical to the maximum permissible number of bits per pixel for an encoded image that does not have maximum width and height. In this case, the number 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 decoder conforming to a specific level shall also fully decode codestreams signalling lower levels. Similarly, a decoder conforming to a specific sublevel shall also fully decode codestreams signalling lower sublevels.

NOTE 2 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.

NOTE 3 Specifying the `Full` sublevel is different from specifying the `Unrestricted` sublevel, as the `Full` sublevel still restricts the maximum decoded bits per pixel.

NOTE 4 Compressing data as mathematically lossless in the `MLS.12` or `MLS.16` profiles can require setting the sublevel to `Unrestricted`.

NOTE 5 In the case of Bayer pattern data, the nominal bits per pixel, N_{bpp}, represents the number of bits per super-pixel.

NOTE 6 Since the constraint on conforming codestreams for the `Full` sublevel depends on the selected profile, they are not listed in the tables hereunder.

NOTE 7 The resulting constraints on conforming codestreams as specified in clause A.4 are listed in Table A.12, Table A.13, Table A.14, Table A.15, Table A.16, and Table A.17.

Table A.12 — Codestream constraints for sublevel `Sublev2bpp` (informative)

Level	Size of a smoothing buffer unit S_{sbu} (bits)	Maximum codestream size $S_{\text{sl,max}}$ (bytes)	Maximum encoded rate $R_{\text{t,max}}$ (Mbits/s)
`1k-1`	2 560	655 360	167
`2k-1`	4 096	1 048 576	267
`4k-1`	8 192	2 228 224	534
`4k-2`	8 192	4 194 304	1 069
`4k-3`	8 192	4 194 304	2 139
`8k-1`	16 384	8 912 896	2 139
`8k-2`	16 384	16 777 216	4 278
`8k-3`	16 384	16 777 216	8 556
`10k-1`	20 480	26 214 400	6 684

Table A.13 — Codestream constraints for sublevel `Sublev3bpp` (informative)

Level	Size of a smoothing buffer unit S_{sbu} (bits)	Maximum codestream size $S_{\text{sl,max}}$ (bytes)	Maximum encoded rate $R_{\text{t,max}}$ (Mbits/s)
`1k-1`	3 840	983 040	250
`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.14 — Codestream constraints for sublevel `Sublev4bpp` (informative)

Level	Size of a smoothing buffer unit S_{sbu} (bits)	Maximum codestream size $S_{\text{sl,max}}$ (bytes)	Maximum encoded rate $R_{\text{t,max}}$ (Mbits/s)
`1k-1`	5 120	1 310 720	334
`2k-1`	8 192	2 097 152	534
`4k-1`	16 384	4 456 448	1 069
`4k-2`	16 384	8 388 608	2 139
`4k-3`	16 384	8 388 608	4 278
`8k-1`	32 768	17 825 792	4 278
`8k-2`	32 768	33 554 432	8 556
`8k-3`	32 768	33 554 432	17 112
`10k-1`	40 960	52 428 800	13 369

Table A.15 — Codestream constraints for sublevel `Sublev6bpp` (informative)

Level	Size of a smoothing buffer unit S_{sbu} (bits)	Maximum codestream size $S_{sl,max}$ (bytes)	Maximum encoded rate $R_{t,max}$ (Mbits/s)
`1k-1`	7 680	1 966 080	501
`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.16 — Codestream constraints for sublevel `Sublev9bpp` (informative)

Level	Size of a smoothing buffer unit S_{sbu} (bits)	Maximum codestream size $S_{sl,max}$ (bytes)	Maximum encoded rate $R_{t,max}$ (Mbits/s)
`1k-1`	11 520	2 949 120	752
`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.17 — Codestream constraints for sublevel `Sublev12bpp` (informative)

Level	Size of a smoothing buffer unit S_{sbu} (bits)	Maximum codestream size $S_{sl,max}$ (bytes)	Maximum encoded rate $R_{t,max}$ (Mbits/s)
`1k-1`	15 360	3 932 160	1 002
`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.6 Frame buffer bandwidth levels

A.6.1 Definition of frame buffer bandwidth levels

For a *TDC enabled* codestream, the frame buffer bandwidth (FBB) levels define a constraint to provide a lower bound on the throughput that a conforming decoder implementation shall support for accessing its frame buffer. The throughput, given by B_f, and the constraint are specified by the FBB model as described in A.7.

A *TDC disabled* codestream does not depend on a frame buffer, and hence, the FBB level constraint is not applicable.

Table A.18 lists the FBB levels defined in this standard.

Additionally, an `Unrestricted` FBB level does not impose any constraint on the frame buffer bandwidth. The `Unrestricted` FBB level shall not be considered as conformance point.

Table A.18 — JPEG XS FBB levels

FBB level	B_f
`FbblevFull`	∞
`Fbblev12bpp`	12
`Fbblev4.5bpp`	4.5
`Fbblev3bpp`	3

A.6.2 Frame buffer bandwidth level signalling in the picture header

The FBB shall be indicated in the P_{lev} field of the picture header defined in ISO/IEC 21122-1 by the values defined in Table A.19. A *TDC disabled* codestream shall set the FBB level to Unrestricted.

Table A.19 — Signalling of the FBB level of a codestream in the P_{lev} field

FBB level	Binary value of P_{lev} field[a]
`Unrestricted`	`XXXX XX00 X00X XXXX`
`FbblevFull`	`XXXX XX11 X11X XXXX`
`Fbblev12bpp`	`XXXX XX11 X00X XXXX`
`Fbblev4.5bpp`	`XXXX XX01 X00X XXXX`
`Fbblev3bpp`	`XXXX XX00 X11X XXXX`
Reserved for ISO/IEC purposes	all other values

[a] An X indicates either a 0 or a 1, as defined in Table A.9 and Table A.11.

A.7 Frame buffer bandwidth level conformance

Decoders conforming to a particular FBB level shall support a minimal throughput bound that is calculated by the FBB model. This conformance only applies when decoding *TDC enabled* codestreams.

A.7.1 Frame buffer quantization

ISO/IEC 21122-1 refers to an implementation-specific `compute_fb_qr(p)` method to compute for each precinct p the corresponding $Q_f[p]$ and $R_f[p]$ quantization parameter values. These quantization parameter values are explicitly signalled in the codestream and allow calculation of the truncation values $T_f[p,b]$, one for each band b of each precinct p, as specified by the `compute_truncation` function. As described in ISO/IEC 21122-1, wavelet prediction coefficients of each band b in each precinct p in the frame buffer are quantized using dead-zone quantization with a reconstruction point $r=0$, by shifting with their corresponding truncation value $T_f[p,b]$.

A.7.2 Frame buffer bandwidth model

The FBB model provides the constraints for the signalled $Q_f[p]$ and $R_f[p]$ values — one pair per precinct — that any compliant *TDC enabled* codestream shall conform to. Doing so provides to implementations an upper limit for the frame buffer access bandwidth required to decode the codestream. Any selection of $Q_f[p]$ and $R_f[p]$ for a precinct p in a codestream that complies to this model is considered valid.

As specified in ISO/IEC 21122-1, given $Q_f[p]$ and $R_f[p]$ for precinct p, a truncation point $T_f[p,b]$ for band b is computed by:

$$T_f[p,b]=\text{compute\_truncation}(b,Q_f[p],R_f[p],0) \quad \text{(A.6)}$$

Each wavelet coefficient $c'[p,\lambda,b,x]$ in precinct p, line λ, band b and position x is quantized by its respective $T_f[p,b]$ to calculate the corresponding wavelet coefficient $\tilde{f}[p,\lambda,b,x]$ in the frame buffer as:

$$\tilde{f}[p,\lambda,b,x]=\begin{cases}\text{sign}(c'[p,\lambda,b,x])\times\left\lfloor |c'[p,\lambda,b,x]| / 2^{T_f[p,b]}\right\rfloor & \text{if } x<W_{pb}[p,b]\\ 0 & \text{otherwise}\end{cases} \quad \text{(A.7)}$$

NOTE 1 $\tilde{f}[p,\lambda,b,x]$ is identical to $f[p,\lambda,b,x]$ as defined in ISO/IEC 21122-1 when $x<W_{pb}[p,b]$.

Consecutive wavelet coefficients $\tilde{f}[p,\lambda,b,x]$ in the frame buffer are grouped into frame buffer groups for the purpose of modeling the frame buffer bandwidth. The number of frame buffer wavelet coefficients within one frame buffer group is denoted by N_{fg} and is constant throughout all bands and precincts. Permissible values of N_{fg} are defined by the profile.

With this frame buffer wavelet coefficient grouping, the contents of the group g_f consists of the wavelet coefficients $\tilde{f}[p,\lambda,b,x]$ with $x=g_f\times N_{fg}+i$ and $0\le i<N_{fg}$ of the frame buffer.

Then, let $M_f[p,\lambda,b,g_f]$ represent the frame buffer bitplane count for a group g_f in precinct p at line λ of band b, calculated as:

$$M_f[p,\lambda,b,g_f]=\max_{0\le i<N_{fg}}\left(\left\lceil \log_2\left(\left|\tilde{f}\left[p,\lambda,b,g_f\times N_{fg}+i\right]\right|+1\right)\right\rceil\right) \quad \text{(A.8)}$$

NOTE 2 The bitplane count values $M_f[p,\lambda,b,g_f]$ are based on coefficients quantized by $Q_f[p]$ and $R_f[p]$, as opposed to the bitplane count values $M[p,\lambda,b,g]$ as defined in ISO/IEC 21122-1.

Using the frame buffer bitplane count values, a frame buffer storage cost $B_{f,bpp}[p]$ for each precinct p is calculated as:

$$B_{f,bpp}[p]=\frac{\sum_{\forall\lambda,b,g_f}\left(B_r+(M_f[p,\lambda,b,g_f]+\text{sign}(M_f[p,\lambda,b,g_f]))\times N_{fg}\right)}{W_p[p]\times H_p} \quad \text{(A.9)}$$

NOTE 3 The $\text{sign}(x)$ function used in A.7.2, Formula (A.9) generates a value of either 0 or 1, depending on whether $M_f[p,\lambda,b,g_f]$ is zero or not (it cannot be negative). That is, sign bits are always included unless the bitplane count $M_f[p,\lambda,b,g_f]$ of group g_f is zero.

Finally, the FBB model places the following constraint for storing framebuffer coefficients of each individual precinct p:

$$\forall p: B_{f,bpp}[p]\le B_f \quad \text{(A.10)}$$

where B_f is specified by the FBB level as described in A.6.1.

A decoder conforming to a specific FBB level shall also fully decode codestreams signalling lower FBB levels.

NOTE 4 By this definition, a decoder conforming to a FBB level defined by a given value of B_f is also conforming to all FBB levels defined by a smaller value of B_f. That is, FBB levels are inclusive.

A.7.3 Model explanation

This clause provides a high-level and informative explanation of the frame buffer bandwidth model as given by clause A.7.2.

Equation A.7.2, Formula (A.9) allows to assign a cost (expressed in number of bits per pixel) to store values of a precinct p in the frame buffer. The cost calculation is based on the sum of:

— The cost to signal one bitplane count per group of N_{fg} values using B_r bits.

— The cost to signal N_{fg} sign bits, if the bitplane count is non-zero.

— The cost of the bitplane data bits for N_{fg} values.

This sum is divided by the total number of values in the precinct, given by $W_p[p]\times H_p$. From A.7.2, Formula (A.6) and A.7.2, Formula (A.7) it follows that the cost $B_{f,bpp}[p]$ is directly controllable by the selection of $Q_f[p]$ and $R_f[p]$ for precinct p. As such, the model constrains for each individual precinct p the selection of $Q_f[p]$ and $R_f[p]$ via A.7.2, Formula (A.10). The values of B_r and N_{fg} are specified in the profiles as specified in clause A.2, while B_f is given by the frame buffer bandwidth level as specified in clause A.6.1.

NOTE The resulting frame buffer bandwidth constraints (in Mbit/s) on conforming codestreams as specified in clause A.7 are listed in Table A.20, Table A.21, and Table A.22. The bi-directional bandwidth $R_{t,fb,max}$ in Mbit/s is calculated by $R_{t,fb,max}=2\times R_{s,max}\times B_f$.

Table A.20 — Frame buffer bandwidth constraints for FBB level `Fbblev3bpp` (informative)

Level	Maximum frame buffer access rate (Mbits/s)
`1k-1`	501
`2k-1`	802
`4k-1`	1 604
`4k-2`	3 209
`4k-3`	6 417
`8k-1`	6 417
`8k-2`	12 835
`8k-3`	25 669
`10k-1`	20 054

Table A.21 — Frame buffer bandwidth constraints for FBB level `Fbblev4.5bpp` (informative)

Level	Maximum frame buffer access rate (Mbits/s)
`1k-1`	752
`2k-1`	1 203
`4k-1`	2 406
`4k-2`	4 813
`4k-3`	9 626
`8k-1`	9 626
`8k-2`	19 252
`8k-3`	38 504
`10k-1`	30 081

Table A.22 — Frame buffer bandwidth constraints for FBB level `Fbblev12bpp` (informative)

Level	Maximum frame buffer access rate (Mbits/s)
1k-1	2 005
2k-1	3 209
4k-1	6 417
4k-2	12 835
4k-3	25 669
8k-1	25 669
8k-2	51 338
8k-3	102 677
10k-1	80 216

A.8 Conformance points

For the `CHigh 444.12` profile as given in Table A.4 at least one of the conformance points as given in Table A.23 shall apply.

For the `TDC 444.12` and `TDC MLS 444.12` profiles as given in Table A.5 at least one of the conformance points as given in Table A.24 shall apply.

Signalling lower levels, sublevels, or FBB levels within one selected conformance point is allowed.

Table A.23 — High profile family conformance points

Name	Profile	Maximum Level	Maximum Sublevel	Allowed chroma subsampling formats
chigh-2k-422-4	CHigh 444.12	2k-1	Sublev4bpp	4:0:0, 4:2:2
chigh-2k-444-4	CHigh 444.12	2k-1	Sublev4bpp	4:0:0, 4:2:0, 4:2:2, 4:4:4
chigh-2k-444-12	CHigh 444.12	2k-1	Sublev12bpp	4:0:0, 4:2:0, 4:2:2, 4:4:4
chigh-4k-422-4	CHigh 444.12	4k-2	Sublev4bpp	4:0:0, 4:2:2
chigh-4k-444-4	CHigh 444.12	4k-2	Sublev4bpp	4:0:0, 4:2:0, 4:2:2, 4:4:4
chigh-4k-444-12	CHigh 444.12	4k-2	Sublev12bpp	4:0:0, 4:2:0, 4:2:2, 4:4:4
chigh-8k-422-4	CHigh 444.12	8k-2	Sublev4bpp	4:0:0, 4:2:2
chigh-8k-444-4	CHigh 444.12	8k-2	Sublev4bpp	4:0:0, 4:2:0, 4:2:2, 4:4:4
chigh-8k-444-12	CHigh 444.12	8k-2	Sublev12bpp	4:0:0, 4:2:0, 4:2:2, 4:4:4

Table A.24 — TDC profile family conformance points

Name	Profile	Maximum Level	Maximum Sub-level	Maximum FBB level	Maximum number of sampling grid points L_{max} per image (sampling grid points)
tdc-2k-fb4.5	TDC 444.12	2k-1	Sublev4bpp	Fbblev4.5bpp	2 211 840
tdc-2k-fb12	TDC 444.12	2k-1	Sublev4bpp	Fbblev12bpp	2 211 840
tdc-mls-2k	TDC MLS 444.12	2k-1	Sublev4bpp	FbblevFull	2 211 840
tdc-4k-fb12	TDC 444.12	4k-2	Sublev4bpp	Fbblev12bpp	8 847 360
tdc-mls-4k	TDC MLS 444.12	4k-2	Sublev4bpp	FbblevFull	8 847 360
tdc-8k-fb3	TDC 444.12	8k-2	Sublev4bpp	Fbblev3bpp	35 389 440
tdc-8k-fb12	TDC 444.12	8k-2	Sublev4bpp	Fbblev12bpp	35 389 440
tdc-mls-8k	TDC MLS 444.12	8k-2	Sublev4bpp	FbblevFull	35 389 440
NOTE The TDC family of conformance points in this table all reduce the maximum number of sampling grid points Lmax per image from what the respective signalled level would otherwise allow. This is done to effectively constrain the maximum size of the frame buffer in the decoder.					

Table A.25 provides as guidance a set of relevant design parameters that directly follow from the conformance points given in Table A.5.

Table A.25 — Design parameters for the TDC conformance points (informative)

Name	Maximum encoded rate $R_{t,max}$ (Mbits/s)	Maximum frame buffer access rate (Mbits/s)	Maximum frame buffer size (kB, or 10^3 bytes)
tdc-2k-fb4.5	534	1 204	1 180
tdc-2k-fb12	534	3 208	3 146
tdc-mls-2k	4 813	9 626	9 438
tdc-4k-fb12	2 139	12 834	13 272
tdc-mls-4k	19 252	38 504	39 814
tdc-8k-fb3	8 556	12 834	13 272
tdc-8k-fb12	8 556	51 338	53 085
tdc-mls-8k	77 007	154 014	159 253

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 filler/padding bytes.

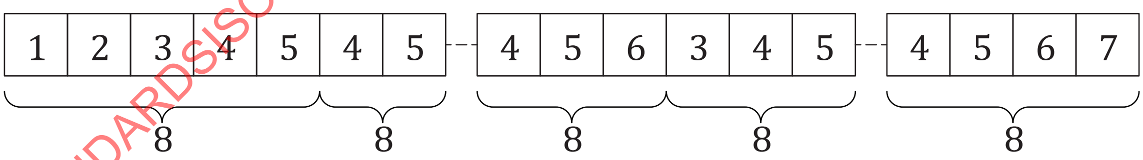
A coded codestream fragment shall be a subset of consecutive bits of the codestream which is built according to the following rules:

— 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 boxes (see B.2.1, NOTE) + codestream header
2 slice header
3 precinct header
4 packet header
5 packet body

6 filler/padding bytes

7 EOC marker

8 codestream fragment

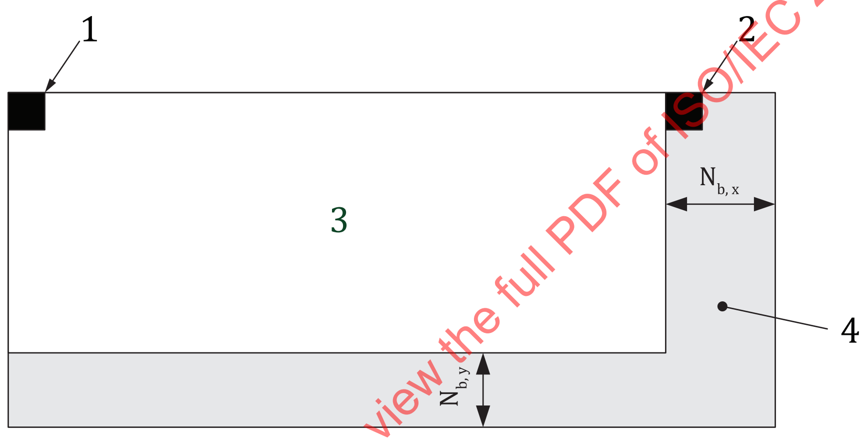
NOTE See ISO/IEC 21122-3 for the definition of boxes that can precede the codestream header.

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. $N_{b,x}$ and $N_{b,y}$ represent respectively the horizontal and vertical blanking periods, expressed in number of sampling grid points.

NOTE 1 In the case of Bayer pattern data, each sampling grid point represents a 2×2 arrangement of sensor elements.



Key

1 active pixel

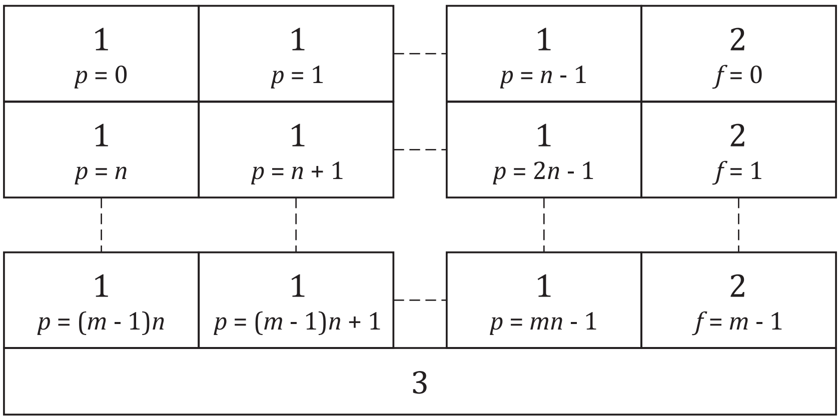
2 blanking pixel

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 number 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.



Key

1 coded codestream fragments for precinct p

2 horizontal blanking codestream fragment f

3 vertical blanking codestream fragment

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 $S_{bits}(f) = 0$. 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.

A vertical blanking codestream fragment f shall be inserted either after the last coded codestream fragment of the image in the case that horizontal blanking is not used (i.e. when $N_{cg,hz} = 0$), or alternatively after the last horizontal blanking codestream fragment of the image in the case that horizontal blanking is enabled (i.e. when $N_{cg,hz} > 0$). The size in bits of the vertical blanking codestream fragment shall equal $S_{bits}(f) = 0$. 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 2 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. To avoid using different clocks in the buffer model, the blanking periods are defined as a multiple of coefficient groups instead of pixels.

NOTE 3 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

$$2^{N_{L,y}} \times N_{b,x} \qquad \text{(B.1)}$$

Given that $\sum_{i=0}^{N_c-1} \frac{1}{s_x[i] \times s_y[i]}$ coefficient groups represent up to N_g pixels, one coefficient group corresponds to up to $N_{p,cg}$ pixels, where:

$$N_{p,cg} = N_g / \left(\sum_{i=0}^{N_c-1} \frac{1}{s_x[i] \times s_y[i]} \right) \qquad \text{(B.2)}$$

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

$$N_{cg,hz} = \left\lceil \frac{2^{N_{L,y}} \times N_{b,x}}{N_{p,cg}} \right\rceil \qquad \text{(B.3)}$$

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:

$$N_{b,y} \times (W_f + N_{b,x}) \qquad \text{(B.4)}$$

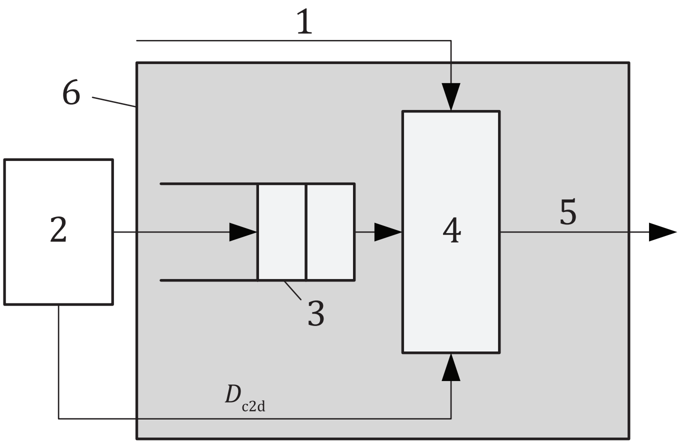
Consequently, $N_{cg,vt}$ may be computed as:

$$N_{cg,vt} = \left\lceil \frac{N_{b,y} \times (W_f + N_{b,x})}{N_{p,cg}} \right\rceil \qquad \text{(B.5)}$$

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 number 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 decoder 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:

$$l_{dec,avail}(t) = l_{dec}(t-1) + w_{dec}(t) \tag{B.6}$$

$$r_{dec}(t) \le l_{dec,avail}(t) \le l_{dec,max} \tag{B.7}$$

$$l_{dec}(t)=l_{dec}(t-1)+w_{dec}(t)-r_{dec}(t)=l_{dec,avail}(t)-r_{dec}(t) \quad \text{(B.8)}$$

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 T_{bmd}	Meaning
0	No upper limit of the decoder buffer assumed
1	Constant bit rate buffer model with limited transmission latency, see C.4.1, Formula (C.7)
2	Constant bit rate buffer model with full use of decoder smoothing buffer (variable transmission latency), see C.4.1, Formula (C.7)
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 $f \in \mathrm{N}$ starts at cycle $t_{\mathrm{dec,start}}(f)$:

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

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

$$l_{\mathrm{dec,avail}}\left(t_{\mathrm{dec,start}}(f)\right) \geq S_{\mathrm{bits}}(f) \quad \text{(C.2)}$$

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

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

The number of bits removed from the decoding smoothing buffer at clock cycle t is defined as follows:

$$r_{\mathrm{dec}}(t) = \begin{cases} S_{\mathrm{bits}}(f) & \exists f \in \mathrm{N} : t_{\mathrm{dec,read}}(f) = t \\ 0 & \text{otherwise} \end{cases} \quad \text{(C.4)}$$

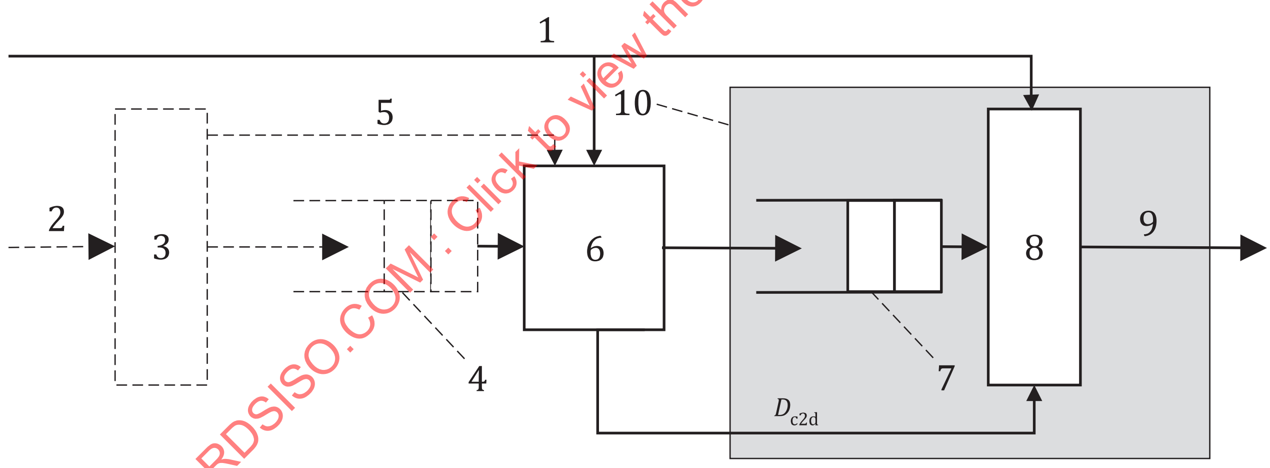
The decoder model is described by pseudo code in Table C.1.

Table C.1 — Decoder model pseudo code

Operation	Notes
`wait` ($t_{dec,start}(1)$)	Wait for $t_{dec,start}(1)$ 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 = read_bits` ($S_{bits}(f)$)	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 clock cycle to complete codestream fragment f
`f = f + 1`	Advance to the next codestream fragment
`}`	End of while

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 start of transmission (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 $T = T_{enc} = T_{dec}$.

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 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 clock cycles $t \geq t_{channel,start}$.

NOTE The clock cycles also include those for the blanking regions defined by $N_{cg,hz}$ and $N_{cg,vt}$.

$$R_{trans} = \frac{S_{c,max} \times 8}{\sum_{f=1}^{N_f} N_{cg}(f)} \quad \text{(C.5)}$$

In Formula (C.5), $S_{c,max}$ is the maximum size of the entire codestream in bytes from SOC to EOC, including all markers. In case that the L_{cod} field of the picture header is not equal to zero, $S_{c,max}$ shall equal the L_{cod} value.

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

$$S_{c,max} = \left\lfloor \frac{W_f \times H_f \times B_{target}}{8} \right\rfloor \quad \text{(C.6)}$$

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

$$\sum_{\tau=0}^{t} w_{dec}(\tau) \leq \left\lfloor (t - t_{channel,start}) \times R_{trans} \right\rfloor \quad \text{(C.7)}$$

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 interruption.

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

$$\sum_{\tau=0}^{t} w_{\text{dec}}(\tau) = \lfloor (t - t_{\text{channel,start}}) \times R_{\text{trans}} \rfloor \qquad \text{(C.8)}$$

The value of $t_{\text{channel,start}}$ depends on the system configuration, and in particular on the encoder. In a full system, $t_{\text{channel,start}}$ 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 (`$t_{\text{channel,start}}$`)`	Wait until transmission channel starts to write bits
`while (!end_of_stream) {`	Repeat until the source stops sending bits
`  write_bits (`$\lfloor R_{\text{trans}} + r \rfloor$`)`	Write $\lfloor R_{\text{trans}} + r \rfloor$ bits to the decoder smoothing buffer
`  r = `R_{trans}` + r - `$\lfloor R_{\text{trans}} + r \rfloor$	Update the number of fractional bits
`  wait (1)`	Wait for one cycle
`}`	Continue sending data

C.4.3 Relation between the two transmission channel models

In case an encoder generates only codestreams having the maximum size $S_{\text{c,max}}$, both channel models defined in C.4.1 and C.4.2 are effectively the same due to the definition of R_{trans} in C.4.1, Formula (C.5). 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 model

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

$$l^{*}_{\text{dec,max}} = S_{\text{sbo}} + \begin{cases} \infty & T_{\text{bmd}} = 0 \\ \min\left(l^{\text{cbr}}_{\text{dec,max}}, \lfloor R_{\text{trans}} \times W_{\text{f}} / N_{\text{p,cg}} \times \Delta T_{\text{max,lines}} \rfloor\right) & T_{\text{bmd}} = 1 \\ l^{\text{cbr}}_{\text{dec,max}} & T_{\text{bmd}} = 2 \end{cases} \qquad \text{(C.9)}$$

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

NOTE $T_{\text{bmd}} = 1$ 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 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 C.5:

$l_{\text{dec,max}}^{\text{cbr}} \in \mathbb{N}$ 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. Can be infinite, if the level is `Unrestricted`, or the sublevel is `Unrestricted`.

NOTE 1 The actual number of bits that can be stored in the decoder smoothing buffer is given by C.4.1, Formula (C.7).

NOTE 2 In practice, in the case that $l_{\text{dec,max}}^{\text{cbr}} = \infty$, the buffer model does not apply at all. This happens when the level is `Unrestricted`, or the sublevel is `Unrestricted`.

$\Delta T_{\text{max,lines}}$ 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 C.4.2 and 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
$l_{\text{dec,max}}^{\text{cbr}}$	$N_{\text{sbu}} \times S_{\text{sbu}}$ (see A.4)
$\Delta T_{\text{max,lines}}$	N_{sbu}, see Table A.1, Table A.2, Table A.3, Table A.4, Table A.5, Table A.6, and Table A.7
S_{sbo}	see Table A.1, Table A.2, Table A.3, Table A.4, Table A.5, Table A.6, and Table A.7

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 $D_{\text{c2d}} \in \text{N}$ such that B.4, Formula (B.6), B.4, Formula (B.7), B.4, Formula (B.8) and C.2, Formula (C.2) are valid for all cycles t, when the temporal behaviour of the channel and decoder models corresponds to the specifications in C.2, C.3, C.4, C.5 and C.6.

In case the transmission channel uses variable bit rate as defined in C.4.1, C.4.1, Formula (C.5) only establishes an upper bound on the number of bits $\sum_{\tau=0}^{t} w_{\text{dec}}(\tau)$ written to the decoder smoothing buffer. For a conforming codestream, there shall exist a function $w_{\text{dec}}(\tau)$ that is consistent with C.4.1, Formula (C.5) and for which B.4, Formula (B.6), B.4, Formula (B.7), B.4, Formula (B.8) and C.2, Formula (C.2) are true.

C.8.2 Conformance of a sequence of codestreams

This subclause defines the conformance of a sequence $C = [C(1), \cdots, C(n)]$ of codestreams to a buffer model instance. For that, define the symbol $t_{\text{channel,start}}(i)$ as the clock cycle where the transmission channel writes the first bit of codestream $C(i)$. Further, define the symbol $D_{\text{c2d}}(1)$ to be the delay between $t_{\text{channel,start}}(1)$ and the start of decoding of codestream $C(1)$. Moreover, define the symbol $F_{\text{first}}(i)$ to be the first codestream fragment of codestream $C(i)$ and $F_{\text{last}}(i)$ to be the last codestream fragment of $C(i)$.

Define $S_{c,max}(i)$ to be the maximum size in bytes of codestream $C(i)$, and define $N_f(i)$ 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 C.8.1, and in addition, the following shall hold:

$$\forall 1 \leq i < n : t_{dec,start}(F_{first}(i+1)) = t_{dec,read}(F_{last}(i)) + 1 \quad \text{(C.10)}$$

and

$$\forall 1 \leq i < n : \frac{S_{c,max}(i) \times 8}{\sum_{f=1}^{N_f(i)} N_{cg}(f)} = R_{trans} \quad \text{(C.11)}$$

with R_{trans} being constant for all i.

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 C.2. However, it is then the responsibility of the implementation to take all necessary measures to ensure that this implementation can 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 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 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 subclauses 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 a latency bound of conforming encoder-decoder systems.

D.2 Encoder model

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

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

Let $S_{\mathrm{bits}}(f)$ 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 model pseudo code

Operation	Notes
`f = 1`	Reset the codestream fragment counter
`while(!end_of_stream) {`	Repeat until the codestream is interrupted
`wait (` $N_{\mathrm{cg}}(f)$ `- 1)`	Wait for $N_{\mathrm{cg}}(f) - 1$ clock cycles
`write_bits (` $S_{\mathrm{bits}}(f)$ `)`	Write $S_{\mathrm{bits}}(f)$ to the smoothing buffer
`wait(1)`	Wait one clock cycle
`f = f + 1`	Advance to the next codestream fragment
`}`	End of while

D.3 Buffer relations

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

$$l_{enc}(t) = \sum_{f=1}^{\infty} S_{bits}(f) \times \xi(t - t_{enc,write}(f)) - ((t + 1 - t_{channel,start}) \times R_{trans}) \times \xi(t - t_{channel,start}) \quad \text{(D.2)}$$

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

$$t_{dec,read}(f) = t_{dec,start}(1) + t_{enc,write}(f) = t_{channel,start} + D_{c2d} + t_{enc,write}(f) \quad \text{(D.3)}$$

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

$$l_{dec}(t_{dec,start}(1) + t) = \lfloor (t_{dec,start}(1) + t - t_{channel,start} + 1) \times R_{trans} \rfloor - \sum_{f=1}^{\infty} S_{bits}(f) \times \xi(t - t_{enc,write}(f)) \quad \text{(D.4)}$$

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

From this, it follows that

$$l_{enc}(t) + l_{dec}(t_{dec,start}(1) + t) = \lfloor (t_{dec,start}(1) + t - t_{channel,start} + 1) \times R_{trans} \rfloor - \lfloor (t_{dec,start}(1) + t - t_{channel,start} + 1) \times R_{trans} \rfloor \times \xi(t - t_{channel,start}) \quad \text{(D.5)}$$

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

$$\begin{aligned} & l_{enc}(t) + l_{dec}(t_{dec,start}(1) + t) = l_{sum} + \varepsilon_1(t) \in \mathrm{N} \\ & |\varepsilon_1(t)| < 1 \\ & l_{sum} = t_{dec,start}(1) \times R_{trans} \\ & R_{trans} \in \mathrm{N} \Rightarrow \varepsilon_1(t) = 0 \end{aligned} \quad \text{(D.6)}$$

This is equivalent to

$$\begin{aligned} & l_{enc}(t) + l_{dec}(t_{dec,start}(1) + t) = \lceil l_{sum} \rceil + \varepsilon_2(t) \in \mathrm{N} \\ & \varepsilon_2(t) \in \{0, -1\} \\ & R_{trans} \in \mathrm{N} \Rightarrow \varepsilon_2(t) = 0 \end{aligned} \quad \text{(D.7)}$$

NOTE 2 $\varepsilon_1(t)$ and $\varepsilon_2(t)$ represent mathematically small values that trend towards 0.