

---

---

**Information technology — MPEG video  
technologies —**

**Part 4:  
Video tool library**

*Technologies de l'information — Technologies vidéo MPEG —  
Partie 4: Bibliothèque d'outils vidéo*

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23002-4:2018



STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23002-4:2018



**COPYRIGHT PROTECTED DOCUMENT**

© ISO/IEC 2018

All rights reserved. Unless otherwise specified, or required in the context of its implementation, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO at the address below or ISO's member body in the country of the requester.

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

Published in Switzerland

## Contents

Page

Foreword.....	viii
Introduction.....	x
1 Scope.....	1
2 Normative references.....	1
3 Terms and definitions.....	2
4 FU description convention.....	2
4.1 FU interface.....	2
4.2 FU IDs.....	3
4.3 Token pool.....	7
4.4 Array data order.....	10
4.5 Input ports (reset_i, init_i, start_i).....	10
4.6 FU block diagram notations.....	10
4.7 Conventions.....	11
4.8 Granular FU.....	11
5 General-purpose FUs.....	11
5.1 Syntax parsing.....	11
5.1.1 Generic syntax parser.....	11
5.1.2 Algo_Byte2bit.....	11
5.1.3 Mgnt_Select_MB_4.....	12
5.1.4 Mgnt_Merger420.....	12
5.1.5 Mgnt_Select_MB_8.....	12
5.1.6 Mgnt_SynP_DEMUX.....	13
5.1.7 Mgnt_SynP_BytePreprocessor.....	13
5.2 General processing FUs.....	14
5.2.1 Algo_InverseQuantization1D.....	14
5.2.2 Algo_InverseQuantizationND.....	15
5.2.3 Algo_InversePrediction1D.....	18
5.2.4 Algo_InversePredictionND.....	20
5.2.5 Algo_ED_AD_StaticBit.....	24
5.2.6 Algo_ED_VLD.....	26
5.2.7 Algo_ED_AD_AdaptiveBit.....	27
5.2.8 Algo_ED_BitPrecision.....	28
5.2.9 Algo_ED_AD.....	31
5.2.10 Algo_ED_AD_EG.....	36
5.2.11 Algo_ED_4bitsD.....	42
5.2.12 Algo_ED_FixedLength.....	44
5.2.13 Algo_LookUpTable1D.....	45
5.2.14 Algo_ContextModeling.....	46
5.2.15 Algo_simpleMath_2op.....	48
5.2.16 Mgnt_Replicate_1_2.....	50
5.2.17 Mgnt_Replicate_1_4.....	50
5.2.18 Mgnt_Replicate_1_8.....	51
5.2.19 Mgnt_MUX_2_1.....	52
5.2.20 Mgnt_MUX_4_1.....	53
5.2.21 Mgnt_MUX_8_1.....	55
5.2.22 Mgnt_DEMUX_1_2.....	56
5.2.23 Mgnt_DEMUX_1_4.....	57
5.2.24 Mgnt_DEMUX_1_8.....	59
5.2.25 Mgnt_ExtractSegment.....	60
5.2.26 Mgnt_ProviderValue.....	61
5.2.27 Mgnt_RepeatSegment.....	62
5.2.28 Mgnt_ExtractBytes.....	64
5.2.29 Mgnt_ExtractBits.....	65

5.2.30	Mgnt_Provider1D .....	66
5.2.31	Mgnt_Provider2D .....	67
6	FUs for MPEG-4 Simple Profile .....	69
6.1	General .....	69
6.2	Syntax parsing .....	69
6.2.1	Algo_SynP .....	69
6.2.2	Mgnt_BlockExpand .....	69
6.2.3	Mgnt_Splitter420B .....	70
6.2.4	Mgnt_Splitter420MV .....	70
6.2.5	Algo_MVR_MedianOfThreeLeftAndTopAndTopRight .....	71
6.2.6	Mgnt_Splitter_420_TYPE .....	71
6.2.7	Algo_VLDtableB6_MPEG4Part2 .....	72
6.2.8	Algo_VLDtableB7_MPEG4Part2 .....	72
6.2.9	Algo_VLDtableB8_MPEG4Part2 .....	73
6.2.10	Algo_VLDtableB12_MPEG4Part2 .....	73
6.2.11	Algo_VLDtableB13_MPEG4Part2 .....	73
6.2.12	Algo_VLDtableB14_MPEG4Part2 .....	74
6.2.13	Algo_VLDtableB15_MPEG4Part2 .....	74
6.2.14	Algo_VLDtableB16_MPEG4Part2 .....	75
6.2.15	Algo_VLDtableB17_MPEG4Part2 .....	75
6.3	Texture decoding .....	76
6.3.1	Algo_IQ_QSAndQmatrixMp4vOrH263Scaler .....	76
6.3.2	Algo_DCRAddr_ThreeLeftTop_8x8 .....	76
6.3.3	Algo_DCRAddr_ThreeLeftTop_16x16 .....	77
6.3.4	Algo_DCRInvPred_CHROMA_8x8 .....	77
6.3.5	Algo_DCRInvPred_LUMA_16x16 .....	78
6.3.6	Algo_IS_ZigzagOrAlternateHorizontalVertical_8x8 .....	79
6.3.7	Algo_IAP_AdaptiveHorizontalOrVerticalPred_8x8 .....	79
6.3.8	Algo_IAP_AdaptiveHorizontalOrVerticalPred_16x16 .....	80
6.3.9	Algo_IDCT2D_ISOIEC_23002_1 .....	80
6.3.10	Mgnt_DCSplit .....	81
6.4	Motion compensation .....	81
6.4.1	Mgnt_FB_w_Address_8x8 .....	81
6.4.2	Mgnt_FB_w_Address_16x16 .....	82
6.4.3	Algo_PictureReconstruction_Saturation .....	82
6.4.4	Algo_Interp_HalfpelBilinearRoundingControl .....	83
7	FUs for MPEG-4 AVC Constrained Baseline Profile .....	84
7.1	General .....	84
7.2	Syntax parsing .....	84
7.2.1	Algo_NALU .....	84
7.2.2	Algo_SynP .....	84
7.2.3	Algo_BlockExpand .....	85
7.2.4	Algo_BlockSplit .....	85
7.2.5	Algo_IntraPred_Split .....	86
7.2.6	Algo.Parser_I_PCM .....	86
7.2.7	Algo_DemuxParserInfoForBlocks_Chroma .....	87
7.2.8	Algo_DemuxParserInfoForBlocks_Luma .....	87
7.3	Texture decoding .....	88
7.3.1	Algo_IS_Zigzag_4x4 .....	88
7.3.2	Algo_DCR_Hadamard_LUMA_IHT1d .....	88
7.3.3	Algo_Transpose4x4 .....	89
7.3.4	Algo_DCR_Hadamard_LUMA_Reordering .....	89
7.3.5	Algo_DCR_Hadamard_LUMA_Scaling .....	90
7.3.6	Algo_DCR_Hadamard_CHROMA .....	90
7.3.7	Algo_IT4x4_1d .....	90
7.3.8	Algo_IT4x4_Addshift .....	91
7.3.9	Algo_IntraPred_LUMA_16x16 .....	91
7.3.10	Algo_IntraPred_LUMA_4x4 .....	92

7.3.11	Algo_Merge_4x4_to_16x16 .....	92
7.3.12	Algo_IQ_QSAndSLAndIDCTScaler_4x4 .....	92
7.3.13	Mgnt_IQ_INTRA16x16.....	93
7.3.14	Algo_IntraPred_4x4_to_8x8 .....	93
7.3.15	Algo_IntraPred_CHROMA .....	94
7.3.16	Mgnt_Intra16x16 .....	94
7.3.17	Mgnt_Intra4x4 .....	95
7.3.18	Mgnt_IQ_Chroma.....	95
7.3.19	Mgnt_Buffer_Neighbour_FullMb.....	95
7.3.20	Mgnt_Buffer_Neighbour_YxY.....	96
7.3.21	Algo_Merge_4x4_to_16x16_norasterscan .....	96
7.3.22	Algo_Split_16x16_to_4x4_norasterscan.....	97
7.4	Motion compensation.....	97
7.4.1	Algo_Interp_EighthPelBilinear .....	97
7.4.2	Algo_Interp_SeparableSixTapQuarterPel .....	98
7.4.3	Algo_Interp_Reord .....	98
7.4.4	Algo_MvLXReconstr .....	99
7.4.5	Mgnt_DPB .....	99
7.4.6	Algo_MMCO .....	100
7.4.7	AlgoRefList.....	100
7.4.8	Mgnt_InterPred .....	101
7.4.9	Algo_RefldxttoFrameNum .....	101
7.5	Filtering .....	102
7.5.1	Mgnt_DBF_AdaptiveFilter.....	102
7.5.2	Algo_DBF_AdaptiveFilter .....	102
7.5.3	Algo_MvComponentReorder .....	103
7.6	Renderer .....	103
7.6.1	Mgnt_POC.....	103
7.6.2	Mgnt_BufferRender .....	103
7.6.3	Mgnt_Merger420_AVC.....	104
8	FUs for MPEG-4 AVC Progressive High Profile .....	104
8.1	General.....	104
8.1.1	Overview .....	104
8.1.2	Algo_SynP .....	104
8.1.3	Algo_BlockExpand.....	106
8.1.4	Algo_DemuxParserInfoForBlocks_Luma .....	106
8.2	Texture decoding .....	107
8.2.1	Algo_IS_Zigzag_8x8.....	107
8.2.2	Algo_IQ_QSAndSLAndIDCTScaler_8x8 .....	107
8.2.3	Algo_IIT_8x8 .....	107
8.2.4	Algo_IntraPred_LUMA_8x8.....	108
8.2.5	Mgnt_Intra_8x8.....	108
8.2.6	Algo_Merge_8x8_to_16x16 .....	109
8.2.7	Algo_DCR_Hadamard_CHROMA.....	109
8.2.8	Algo_DCR_Hadamard_LUMA_Scaling.....	110
8.2.9	Algo_IQ_QSAndSLAndIDCTScaler_4x4 .....	110
8.2.10	Algo_Merge_8x8_to_16x16_norasterscan .....	111
8.2.11	Algo_Split_16x16_to_8x8_norasterscan.....	111
8.2.12	Mgnt_I4x4_I8x8_demux.....	111
8.2.13	Mgnt_I4x4_I8x8_mux.....	112
8.3	Motion compensation.....	112
8.3.1	Algo_GeneratePredWeight.....	112
8.3.2	Mgnt_SelectMvpLX .....	113
8.3.3	Algo_MvLXReconstr .....	113
8.3.4	Algo_MvBuffer .....	114
8.3.5	Mgnt_SelectMvpLX.....	115
8.3.6	Algo_FrameNumToPocList.....	115
8.4	Filtering .....	115

8.4.1	Algo_DBF_AdaptiveFilter .....	115
8.4.2	Algo_MvComponentReorder .....	116
9	MPEG-4 Part 16 SC3DMC decoder specific FUs .....	117
9.1	General .....	117
9.2	Algo_ExtractMask_SC3DMC .....	117
9.3	MPEG-4 SC3DMC TFAN Specific FUs .....	118
9.3.1	Algo_DecodeConnectivity_TFAN .....	118
9.4	MPEG-4 SC3DMC SVA Specific FUs .....	123
9.4.1	Algo_ContextModeling_SVA_nType .....	123
9.4.2	Algo_ContextModeling_SVA_Indexes .....	125
9.4.3	Algo_ContextModeling_SVA_Vertex_Attribute .....	127
9.4.4	Algo_DecodeConnectivity_SVA .....	130
9.4.5	Algo_ExtractFaceDirection_SVA .....	134
9.4.6	Algo_Connectivity_InversePrediction_SVA .....	135
10	FUs for HEVC Main Profile .....	137
10.1	General .....	137
10.2	Syntax parsing .....	137
10.2.1	Algo_SynP .....	137
10.3	Texture decoding .....	138
10.3.1	Algo_IntraPrediction .....	138
10.3.2	selectCU .....	138
10.4	Motion compensation .....	139
10.4.1	Algo_InterPrediction .....	139
10.4.2	Mgnt_DecodedPictureBuffer .....	139
10.4.3	Algo_GenerateRefList .....	140
10.4.4	Algo_MvComponentPred .....	140
10.5	Filtering .....	141
10.5.1	Algo_SaoFilter .....	141
10.5.2	Algo_GenerateBs .....	141
10.5.3	Algo_DeblockingFilter .....	142
10.5.4	Algo_QpGen .....	142
10.6	MD5 check .....	142
10.6.1	Mgnt_MD5SplitInfo .....	142
10.6.2	Mgnt_padding .....	143
10.6.3	Mgnt_MD5Shifter .....	143
10.6.4	Mgnt_MD5Compute .....	143
10.7	Inverse transforms .....	144
10.7.1	Mgnt_IT_Splitter .....	144
10.7.2	Mgnt_IT_Merger .....	144
10.7.3	Mgnt_Block_Merger .....	145
10.7.4	Mgnt_Transpose_4x4 .....	145
10.7.5	Mgnt_Transpose_8x8 .....	145
10.7.6	Mgnt_Transpose_16x16 .....	146
10.7.7	Mgnt_Transpose_32x32 .....	146
10.7.8	Mgnt_Transpose_32x32 .....	146
10.7.9	Algo_IT4x4_1d .....	146
10.7.10	Algo_IT8x8_1d .....	147
10.7.11	Algo_IT16x16_1d .....	147
10.7.12	Algo_IT32x32_1d .....	147
10.7.13	Algo_invDST4x4_1st .....	148
10.7.14	Algo_invDST4x4_2nd .....	148
Annex A (normative)	Naming convention of FU .....	149
Annex B (informative)	FU network examples .....	151
Annex C (informative)	FNL of MPEG-4 AVC Progressive High Profile decoder .....	186
Annex D (normative)	Granular FUs concept .....	218

<b>Annex E (informative) Granular FUs .....</b>	<b>219</b>
<b>Annex F (normative) Parallelogram prediction.....</b>	<b>236</b>
<b>Annex G (informative) Network specifications .....</b>	<b>240</b>
<b>Annex H (informative) Patent statement.....</b>	<b>241</b>
<b>Annex I (informative) Instantiation of generic syntax parser FU from BSD.....</b>	<b>242</b>
<b>Bibliography.....</b>	<b>261</b>

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23002-4:2018

## 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. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

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 ISO documents 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](http://www.iso.org/directives)).

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

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

For an explanation on 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 the following URL: [www.iso.org/iso/foreword.html](http://www.iso.org/iso/foreword.html).

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

This third edition cancels and replaces the second edition (ISO/IEC 23002-4:2014), which has been technically revised.

It also incorporates ISO/IEC 23002-4:2014/Amd.1:2014, ISO/IEC 23002-4:2014/Amd.2:2015, and ISO/IEC 23002-4:2014/DAMd.3<sup>1</sup>.

The changes compared to the previous edition are as follows.

- FU description convention described in Clause 4 has been technically updated.
- Functional unit (FU) and FU network (FN) descriptions for graphics tool library (GTL) have been added according to ISO/IEC 23002-4:2014/Amd.1:2014. Clause 5, Clause 9, Annexes D, E, F, and G have been added.
- FU and FN descriptions for high efficiency video coding (HEVC) have been added according to ISO/IEC 23002-4:2014/Amd.2:2015. Clause 10 has been added and Annex B has been updated.

---

<sup>1</sup> Draft amendment approved in DAM ballot but not published.

- FU and FN descriptions for parser instantiation from bitstream syntax description (BSD) have added. Clause 5 has updated and Annex I has added.
- A patent statement has been added in Annex H.

A list of all parts in the ISO/IEC 23002 series can be found on the ISO website.

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23002-4:2018

## Introduction

This document defines the MPEG video tool library, which contains tools drawn from existing MPEG coding standards, such as ISO/IEC 14496-2 and ISO/IEC 14496-10, and ISO/IEC 23001-4 defines the methods capable of describing codec configurations in the reconfigurable video coding (RVC) framework.

This document primarily addresses reconfigurable video aspects and will only focus on the description of representation of video codec configurations under the RVC framework, but could be extended to a more generic reconfigurable media coding (RMC) framework.

The objective of RVC is to offer a framework that is capable of configuring and specifying video codecs as a collection of “higher level” modules by using video coding tools. The video coding tools are defined in video tool libraries. This d defines the MPEG video tool library. The RVC framework principle could also support non-MPEG tool libraries, provided that their developers have taken care to obey the appropriate rules of operation.

For the purpose of framework deployment, an appropriate description is needed to describe configurations of decoders composed of or instantiated from a subset of video tools from either one or more libraries. As illustrated in Figure 1, the configuration information consists of

- bitstream syntax description, and
- network of functional units (FUs) description (also referred to as the decoder configuration)

that together constitute the entire decoder description.

Bitstreams of existing MPEG standards are specified by specific syntax structures and decoders are composed of various coding tools. Therefore, RVC includes support for bitstream syntax descriptions as well as video coding tools. As depicted in Figure 1, a typical RVC decoder requires two types of information, namely the decoder description and the encoded media (e.g. video bitstreams) data.

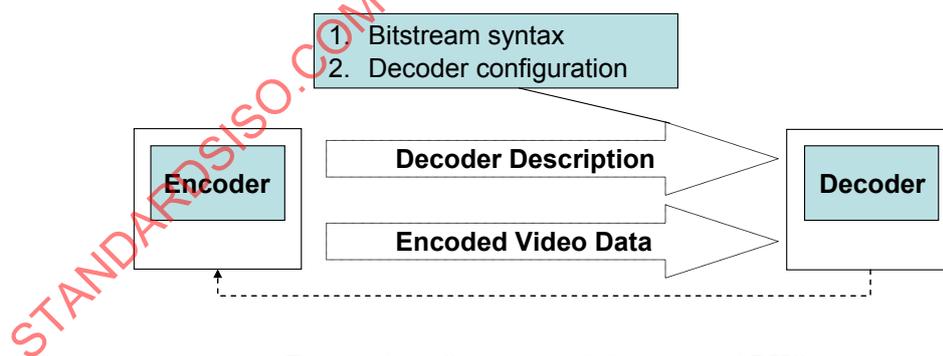
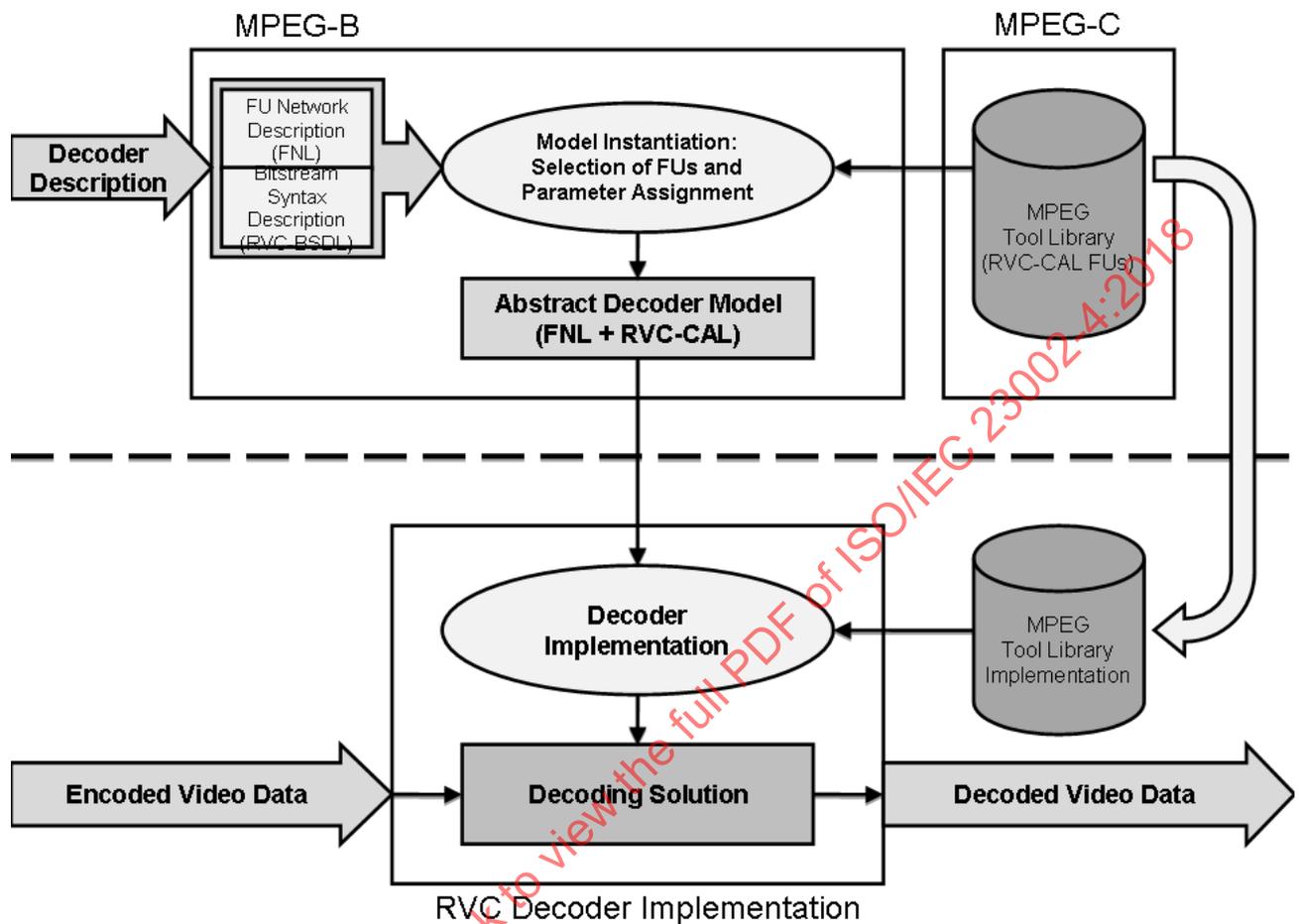


Figure 1 — Conceptual diagram of RVC

A more detailed description of the RVC decoder is illustrated in Figure 2. As shown in Figure 2, the decoder description is required for the configuration of a RVC decoder. The Bitstream Syntax Description (BSD) and FU Network Description (FND) (which compose the Decoder Description) are used to configure or compose an abstract decoder model (ADM) which is instantiated through the selection of FUs from tool libraries optionally with proper parameter assignment. Such ADM constitutes the behavioral reference model used in setting up a decoding solution under the RVC framework. The process of yielding a decoding solution may vary depending on the technologies used for the desired

implementations. Examples of the instantiation of an ADM and generation of proprietary decoding solutions can be found in ISO/IEC 23001-4.



**Figure 2 — Graphical representation of the process for setting up a decoding solution under the RVC framework**

Within the RVC framework, the decoder description describes a particular decoder configuration and consists of the FND and the BSD. The FND describes the connectivity of the network of FUs used to form a decoder whereas the parsing process for the bitstream syntax is implicitly described by the BSD. These two descriptions are specified using two standard XML-based languages or dialects.

- Functional unit network language (FNL) is a language that describes the FND, known also as “network of FUs”. The FNL specified normatively within the scope of the RVC framework is provided in ISO/IEC 23001-4.
- Bitstream syntax description language (BSDL), standardized in ISO/IEC 23001-5 (MPEG-B Part 5), describes the bitstream syntax and the parsing rules. A pertinent subset of this BSDL named RVC-BSDL is defined within the scope of the current RVC framework. This RVC-BSDL also includes possibilities for further extensions, which are necessary to provide complete description of video bitstreams. RVC-BSDL specified normatively within the scope of the RVC framework is provided in ISO/IEC 23001-4.

The decoder configuration specified using FNL, together with the specification of the bitstream syntax using RVC-BSDL fully specifies the ADM and provides an “executable” model of the RVC decoder description.

The instantiated ADM includes the information about the selected FUs and how they should be connected. As already mentioned, the FND with the network connection information is expressed by using FNL. Furthermore, the RVC framework specifies and uses a dataflow-oriented language called RVC-CAL for describing FUs' behavior. The normative specification of RVC-CAL is provided in ISO/IEC 23001-4. The ADM is the behavioural model that should be referred to in order to implement any RVC conformant decoder. Any RVC compliant decoding solution/implementation can be achieved by using proprietary non-normative tools and mechanisms that yield decoders that behave equivalent to the RVC ADM.

The decoder description, the MPEG tool library, and the associated instantiation of an ADM are normative. More precisely, the ADM is intended to be normative in terms of a behavioural model. In other words, what is normative is the input/output behaviour of the complete ADM, as well as the input/output behaviour of all the FUs that are included in the ADM.

A statement concerning patents is included in Annex H of this document.

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23002-4:2018

# Information technology — MPEG video technologies —

## Part 4: Video tool library

### 1 Scope

This document defines the description of the MPEG video tool library (VTL) based on the decoder description specified in ISO/IEC 23001-4. This tool library defines the specification of FUs, which are sufficient to build complete decoding solutions according to the following coding standards:

- ISO/IEC 14496-2 (MPEG-4 Simple Profile),
- ISO/IEC 14496-10 (MPEG-4 AVC Constrained Baseline Profile and Progressive High Profile),
- ISO/IEC 14496-16 (MPEG-4 SC3DMC), and
- ISO/IEC 23008-2 (HEVC Main Profile).

The objective of ISO/IEC 23001-4 is to define the general framework principles, and this document defines the MPEG VTL that includes relevant tools (or FUs) from the existing MPEG coding standards. Each FU is defined in the form of a textual description, which can be found in 4.1. The input and output behaviour follows the conventions described in Clause 5, Clause 6, and Clause 7.

This document compliant implementations can be designed using any software or hardware language and components. The reference software for the textual specification of FUs is written in RVC-CAL language of which a formal syntax is provided in ISO/IEC 23001-4.

### 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 14496-2, *Information technology — Coding of audio-visual objects — Part 2: Visual*

ISO/IEC 14496-10, *Information technology — Coding of audio-visual objects — Part 10: Advanced Video Coding*

ISO/IEC 14496-16, *Information technology — Coding of audio-visual objects — Part 16: Animation Framework eXtension (AFX)*

ISO/IEC 23001-4, *Information technology — MPEG systems technologies — Part 4: Codec configuration representation*

ISO/IEC 23008-2, *Information technology — High efficiency coding and media delivery in heterogeneous environments — Part 2: High efficiency video coding*

### 3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 23001-4 apply.

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

- IEC Electropedia: available at <http://www.electropedia.org/>
- ISO Online browsing platform: available at <http://www.iso.org/obp>

### 4 FU description convention

#### 4.1 FU interfaces

As shown in Table 1, each FU is described with the following elements.

- **FU Name:** Name to represent the functional unit in this specification. The name of the FU is normative and follows the naming convention described in Annex A.
- **Description:** Textual explanation to describe the functionality of the FU. The description should be concise. The precise normative behaviour of the algorithm (input/output, timing etc.) is specified by the RVC-CAL reference code in ISO/IEC 23002-4:2014/Amd 1:2014.
- **Profiles@levels supported:** The profiles@level supported for this functional unit. It may append that a given range of values makes the FU behave for a given profile@level and another range of values makes the FU behave for another profile@level.
- **Input:** A token that is entering the FU through the designated input port. The token type refers to the token pool described in 4.3. The “name” field indicates the input port.
- **Output:** A token that is coming out of the FU through the designated output port. The “name” field indicates the output port.
- **Parameter (optional):** Parameters are optionally described to adjust the behaviour of the FU. All the parameters shall be specified with name, description and range.
- In several FU diagrams, the ports are named with a trailing “\_i” for the input port type and with a trailing “\_o” for the output port type.
- Some FU diagrams contain as well the Finite State Machine diagram. The following conventions apply: INPUT - the action of reading a token or a set of tokens from the input port, OUTPUT - the action of writing the token or a set of tokens to an output port.
- “Parameter” is set at network configuration stage (cannot be changed during the process) and it is characteristic for each FU.
- Token RANGE: describes the mathematical interval for the token value

#### EXAMPLE

Token RANGE: { 0, 1} – binary value.

Token RANGE: [ 0 .. N ], value  $\in$  [ 0, N ] – real values, closed interval.

- All the FUs require the data to be in little-endian format.

**Table 1 — Template of description of an FU (example)**

<b>FU Name</b>	e.g. Algo_IDCT2D_ISOIEC_23002_1	
<b>Description</b>	<p>e.g. This module computes the 8×8 Inverse Discrete Cosine Transform (IDCT) defined as</p> $f(x,y) = \frac{2}{N} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} C(u)C(v)F(u,v) \cos\left(\frac{(2x+1)u\pi}{2N}\right) \cos\left(\frac{(2y+1)v\pi}{2N}\right)$ <p>with <math>u, v, x, y = 0, 1, 2, \dots, N-1</math>  where <math>x, y</math> are spatial coordinates in the sample domain  <math>u, v</math> are coordinates in the transform domain</p> $C(u), C(v) = \begin{cases} \frac{1}{\sqrt{2}} & \text{for } u, v = 0 \\ 1 & \text{otherwise} \end{cases}$ <p>It inputs a list of 64 coefficients and outputs a list of 64 decoded coefficients.</p>	
<b>Profiles@levels supported</b>	e.g. MPEG-4 SP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
e.g. X	e.g. BLOCK token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
e.g. Y	e.g. BLOCK token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>

## 4.2 FU IDs

FU of the specific functionality is identified by its unique identification number. Table 2 lists IDs and names of all FUs in VTL. IDs and names are used in FND to select FUs.

**Table 2 — List of FUs and their IDs**

ID	FU Name
1	org.sc29.wg11.common.Algo_SynP_Generic
2	org.sc29.wg11.mpeg4.part2.sp.parser.Algo_SynP
3	org.sc29.wg11.mpeg4.part2.sp.parser.Mgnt_BlockExpand
4	org.sc29.wg11.mpeg4.part2.sp.parser.Mgnt_Splitter420B
5	org.sc29.wg11.mpeg4.part2.sp.parser.Mgnt_Splitter420MV
6	org.sc29.wg11.mpeg4.part2.sp.parser.Algo_MVR_MedianOfThreeLeftAndTopAndTopRight
7	org.sc29.wg11.mpeg4.part2.sp.parser.Algo_MVSequence_LeftAndTopAndTopRight
9	org.sc29.wg11.mpeg4.part2.sp.parser.Mgnt_Splitter_420_TYPE
10	org.sc29.wg11.mpeg4.part2.sp.parser.vlc.Algo_VLDtableB6_MPEG4Part2
11	org.sc29.wg11.mpeg4.part2.sp.parser.vlc.Algo_VLDtableB7_MPEG4Part2
12	org.sc29.wg11.mpeg4.part2.sp.parser.vlc.Algo_VLDtableB8_MPEG4Part2
13	org.sc29.wg11.mpeg4.part2.sp.parser.vlc.Algo_VLDtableB12_MPEG4Part2
14	org.sc29.wg11.mpeg4.part2.sp.parser.vlc.Algo_VLDtableB13_MPEG4Part2
15	org.sc29.wg11.mpeg4.part2.sp.parser.vlc.Algo_VLDtableB14_MPEG4Part2

ID	FU Name
16	org.sc29.wg11.mpeg4.part2.sp.parser.vlc.Algo_VLDtableB15_MPEG4Part2
17	org.sc29.wg11.mpeg4.part2.sp.parser.vlc.Algo_VLDtableB16_MPEG4Part2
18	org.sc29.wg11.mpeg4.part2.sp.parser.vlc.Algo_VLDtableB17_MPEG4Part2
19	org.sc29.wg11.mpeg4.part2.sp.texture.Algo_IQ_QSAndQmatrixMp4vOrH263Scaler
20	org.sc29.wg11.mpeg4.part2.sp.texture.dc_reconstruction.Algo_DCRAAddr_ThreeLeftTop_8x8
21	org.sc29.wg11.mpeg4.part2.sp.texture.dc_reconstruction.Algo_DCRAAddr_ThreeLeftTop_16x16
22	org.sc29.wg11.mpeg4.part2.sp.texture.dc_reconstruction.Algo_DCRInvPred_CHROMA_8x8
23	org.sc29.wg11.mpeg4.part2.sp.texture.dc_reconstruction.Algo_DCRInvPred_LUMA_16x16
24	org.sc29.wg11.mpeg4.part2.sp.texture.Algo_IS_ZigzagOrAlternateHorizontalVertical_8x8
25	org.sc29.wg11.mpeg4.part2.sp.texture.Algo_IAP_AdaptiveHorizontalOrVerticalPred_8x8
26	org.sc29.wg11.mpeg4.part2.sp.texture.Algo_IAP_AdaptiveHorizontalOrVerticalPred_16x16
27	org.sc29.wg11.mpeg4.part2.sp.texture.Algo_IDCT2D_ISOIEC_23002_1
28	org.sc29.wg11.mpeg4.part2.sp.texture.Mgmt_DCSplit
29	org.sc29.wg11.mpeg4.part2.sp.motion.Mgmt_FB_w_Address_8x8
30	org.sc29.wg11.mpeg4.part2.sp.motion.Mgmt_FB_w_Address_16x16
31	org.sc29.wg11.mpeg4.part2.sp.motion.Algo_PictureReconstruction_Saturation
32	org.sc29.wg11.mpeg4.part2.sp.motion.Algo_Interp_HalfpelBilinearRoundingControl
33	org.sc29.wg11.mpeg4.part10.cbp.parser.Algo_NALU
34	org.sc29.wg11.mpeg4.part10.cbp.parser.Algo_SynP
35	org.sc29.wg11.mpeg4.part10.cbp.parser.Algo_BlockExpand
36	org.sc29.wg11.mpeg4.part10.cbp.parser.Algo_BlockSplit
37	org.sc29.wg11.mpeg4.part10.cbp.parser.Algo_IntraPred_Split
38	org.sc29.wg11.mpeg4.part10.cbp.parser.Algo_Parser_I_PCM
39	org.sc29.wg11.mpeg4.part10.cbp.selectMacroblock.Algo_DemuxParserInfoForBlocks_Chroma
40	org.sc29.wg11.mpeg4.part10.cbp.selectMacroblock.Algo_DemuxParserInfoForBlocks_Luma
41	org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_IS_Zigzag_4x4
42	org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_DCR_Hadamard_LUMA_IHT1d
43	org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_Transpose4x4
44	org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_DCR_Hadamard_LUMA_Reordering
45	org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_DCR_Hadamard_LUMA_Scaling
46	org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_DCR_Hadamard_CHROMA
47	org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_IT4x4_1d
48	org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_IT4x4_Addshift
49	org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_IntraPred_LUMA_16x16
50	org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_IntraPred_LUMA_4x4
51	org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_Merge_4x4_to_16x16

ID	FU Name
52	org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_IQ_QSAndSLAndIDCTScaler_4x4
53	org.sc29.wg11.mpeg4.part10.cbp.Residual.Mgnt_IQ_INTRA16x16
54	org.sc29.wg11.mpeg4.part10.cbp.intraPred.Algo_Merge_4x4_to_8x8
55	org.sc29.wg11.mpeg4.part10.cbp.intraPred.Algo_IntraPred_CHROMA
56	org.sc29.wg11.mpeg4.part10.cbp.intraPred.Mgnt_Intra_16x16
57	org.sc29.wg11.mpeg4.part10.cbp.intraPred.Mgnt_Intra_4x4
58	org.sc29.wg11.mpeg4.part10.cbp.intraPred.Mgnt_IQ_Chroma
59	org.sc29.wg11.mpeg4.part10.cbp.intraPred.Mgnt_Buffer_Neighbour_FullMb
60	org.sc29.wg11.mpeg4.part10.cbp.intraPred.Mgnt_Buffer_Neighbour_YxY
61	org.sc29.wg11.mpeg4.part10.cbp.intraPred.Algo_Split_16x16_to_4x4_norasterscan
62	org.sc29.wg11.mpeg4.part10.cbp.intraPred.Algo_Merge_4x4_to_16x16_norasterscan
63	org.sc29.wg11.mpeg4.part10.cbp.interPred.Algo_Interp_Bilinear
64	org.sc29.wg11.mpeg4.part10.cbp.interPred.Algo_Interp_SeparableSixTapQuarterPel
65	org.sc29.wg11.mpeg4.part10.cbp.interPred.Algo_Interp_Reord
66	org.sc29.wg11.mpeg4.part10.cbp.interPred.Algo_MvLXReconstr
67	org.sc29.wg11.mpeg4.part10.cbp.interPred.Mgnt_DPB
68	org.sc29.wg11.mpeg4.part10.cbp.interPred.Algo_MMCO
69	org.sc29.wg11.mpeg4.part10.cbp.interPred.Algo_RefIdxtoFrameNum
70	org.sc29.wg11.mpeg4.part10.cbp.interPred.Algo_RefList
71	org.sc29.wg11.mpeg4.part10.cbp.interPred.Mgnt_InterPred
72	org.sc29.wg11.mpeg4.part10.cbp.deblockingFilter.Mgnt_DBF_AdaptiveFilter
73	org.sc29.wg11.mpeg4.part10.cbp.deblockingFilter.Algo_DBF_AdaptiveFilter
74	org.sc29.wg11.mpeg4.part10.cbp.display.Mgnt_POC
75	org.sc29.wg11.mpeg4.part10.cbp.display.Mgnt_BufferRender
76	org.sc29.wg11.common.Algo_Add_PixSat
77	org.sc29.wg11.common.Mgnt_Select_MB_4
78	org.sc29.wg11.common.Algo_Byte2bit
79	org.sc29.wg11.common.Algo_Mgnt_Merger420
80	org.sc29.wg11.common.Mgnt_Select_MB_8
81	org.sc29.wg11.mpeg4.part10.cbp.display.Mgnt_Merger420_AVC
82	org.sc29.wg11.mpeg4.part10.php.parser.Algo_SynP
83	org.sc29.wg11.mpeg4.part10.php.parser.Algo_BlockExpand
84	org.sc29.wg11.mpeg4.part10.php.selectMacroblock.Algo_DemuxParserInfoForBlocks_Luma
85	org.sc29.wg11.mpeg4.part10.php.Residual.Algo_IS_Zigzag_8x8
86	org.sc29.wg11.mpeg4.part10.php.Residual.Algo_IQ_QSAndSLAndIDCTScaler_8x8
87	org.sc29.wg11.mpeg4.part10.php.Residual.Algo_IIT_8x8
88	org.sc29.wg11.mpeg4.part10.php.Intrapred.Algo_IntraPred_LUMA_8x8
89	org.sc29.wg11.mpeg4.part10.php.Intrapred.Mgnt_Intra_8x8
90	org.sc29.wg11.mpeg4.part10.php.Residual.Algo_Merge_8x8_to_16x16
91	org.sc29.wg11.mpeg4.part10.php.Residual.Algo_DCR_Hadamard_CHROMA
92	org.sc29.wg11.mpeg4.part10.php.Residual.Algo_DCR_Hadamard_LUMA_Scaling
93	org.sc29.wg11.mpeg4.part10.php.Residual.Algo_IQ_QSAndSLAndIDCTScaler_4x4

ID	FU Name
94	org.sc29.wg11.mpeg4.part10.php.Intrapred.Algo_Merge_8x8_to_16x16_norasterscan
95	org.sc29.wg11.mpeg4.part10.php.Intrapred.Algo_Split_16x16_to_8x8_no_rasterscan
96	org.sc29.wg11.mpeg4.part10.php.Residual.Mgnt_I4x4_I8x8_demux
97	org.sc29.wg11.mpeg4.part10.php.Residual.Mgnt_I4x4_I8x8_mux
98	org.sc29.wg11.mpeg4.part10.php.interPred.Algo_GeneratePredWeight
99	org.sc29.wg11.mpeg4.part10.php.interPred.Mgnt_SelectMvpLX
100	org.sc29.wg11.mpeg4.part10.php.interPred.Algo_MvLXReconstr
101	org.sc29.wg11.mpeg4.part10.php.interPred.Algo_MvBuffer
102	org.sc29.wg11.mpeg4.part10.php.interPred.Mgnt_SelectMvpLX
103	org.sc29.wg11.mpeg4.part10.php.interPred.Mgnt_SelectRefIdx
104	org.sc29.wg11.mpeg4.part10.php.interPred.Algo_FrameNumToPocList
105	org.sc29.wg11.mpeg4.part10.php.deblockingFilter.Algo_DBF_AdaptiveFilter
106	org.sc29.wg11.mpeg4.part10.php.deblockingFilter.Algo_MvComponentReorder
107	org.sc29.wg11.mpeg4.part16.Algo_Parser_SC3DMC
108	org.sc29.wg11.mpeg4.part16.Algo_InverseQuantization1D
109	org.sc29.wg11.mpeg4.part16.Algo_InverseQuantizationND
110	org.sc29.wg11.mpeg4.part16.Algo_InversePrediction1D
111	org.sc29.wg11.mpeg4.part16.Algo_InversePredictionND
112	org.sc29.wg11.mpeg4.part16.Algo_ED_AD_StaticBit
113	org.sc29.wg11.mpeg4.part16.Algo_ED_AD_AdaptiveBit
114	org.sc29.wg11.mpeg4.part16.Algo_ED_VLD
115	org.sc29.wg11.mpeg4.part16.Algo_ED_BitPrecision
116	org.sc29.wg11.mpeg4.part16.Algo_ED_AD
117	org.sc29.wg11.mpeg4.part16.Algo_ED_AD_EG
118	org.sc29.wg11.mpeg4.part16.Algo_ContextModeling
119	org.sc29.wg11.mpeg4.part16.Algo_ContextModeling_SVA_nType
120	org.sc29.wg11.mpeg4.part16.Algo_ContextModeling_SVA_Indexes
121	org.sc29.wg11.mpeg4.part16.Algo_ContextModeling_SVA_Vertex_Attribute
122	org.sc29.wg11.mpeg4.part16.Algo_ED_4bitsD
123	org.sc29.wg11.mpeg4.part16.Algo_ED_FixedLength
124	org.sc29.wg11.mpeg4.part16.Algo_LookUpTable1D
125	org.sc29.wg11.mpeg4.part16.Algo_DecodeConnectivity_SVA
126	org.sc29.wg11.mpeg4.part16.Algo_DecodeConnectivity_TFAN
127	org.sc29.wg11.mpeg4.part16.Algo_ExtractMask_SC3DMC
128	org.sc29.wg11.mpeg4.part16.Algo_ExtractFaceDirection_SVA
129	org.sc29.wg11.mpeg4.part16.Algo_simpleMath_2op
130	org.sc29.wg11.mpeg4.part16.Algo_Connectivity_InversePrediction_SVA
131	org.sc29.wg11.mpeg4.part16.Mgnt_Replicate_1_2
132	org.sc29.wg11.mpeg4.part16.Mgnt_Replicate_1_4
133	org.sc29.wg11.mpeg4.part16.Mgnt_Replicate_1_8
134	org.sc29.wg11.mpeg4.part16.Mgnt_MUX_2_1
135	org.sc29.wg11.mpeg4.part16.Mgnt_MUX_4_1
136	org.sc29.wg11.mpeg4.part16.Mgnt_MUX_8_1
137	org.sc29.wg11.mpeg4.part16.Mgnt_DEMUX_1_2
138	org.sc29.wg11.mpeg4.part16.Mgnt_DEMUX_1_4
139	org.sc29.wg11.mpeg4.part16.Mgnt_DEMUX_1_8
140	org.sc29.wg11.mpeg4.part16.Mgnt_ExtractSegment
141	org.sc29.wg11.mpeg4.part16.Mgnt_ProviderValue
142	org.sc29.wg11.mpeg4.part16.Mgnt_RepeatSegment

ID	FU Name
143	org.sc29.wg11.mpeg4.part16.Mgnt_ExtractBytes
144	org.sc29.wg11.mpeg4.part16.Mgnt_ExtractBits
145	org.sc29.wg11.mpeg4.part16.Mgnt_Provider1D
146	org.sc29.wg11.mpeg4.part16.Mgnt_Provider2D
147	org.sc29.wg11.mpeg4.part2.main.intra.selectCU
148	org.sc29.wg11.mpeg4.part2.main.inter.Algo_InterPrediction
149	org.sc29.wg11.mpeg4.part2.main.inter.Mgnt_DecodedPictureBuffer
150	org.sc29.wg11.mpeg4.part2.main.inter.Algo_GenerateRefList
151	org.sc29.wg11.mpeg4.part2.main.inter.Algo_MvComponentPred
152	org.sc29.wg11.mpeg4.part2.main.Filters.Algo_SaoFilter
153	org.sc29.wg11.mpeg4.part2.main.Filters.Algo_GenerateBs
154	org.sc29.wg11.mpeg4.part2.main.Filters.Algo_DeblockingFilter
155	org.sc29.wg11.mpeg4.part2.main.Filters.Algo_QpGen
156	org.sc29.wg11.mpeg4.part2.common.Mgnt_MD5SplitInfo
157	org.sc29.wg11.mpeg4.part2.common.Mgnt_padding
158	org.sc29.wg11.mpeg4.part2.common.Mgnt_MD5Shifter
159	org.sc29.wg11.mpeg4.part2.common.Mgnt_MD5Compute
160	org.sc29.wg11.mpeg4.part2.main.IT.Mgnt_IT_Splitter
161	org.sc29.wg11.mpeg4.part2.main.IT.Mgnt_IT_Merger
162	org.sc29.wg11.mpeg4.part2.main.IT.Mgnt_Block_Merger
163	org.sc29.wg11.mpeg4.part2.main.IT.Mgnt_Transpose_4x4
164	org.sc29.wg11.mpeg4.part2.main.IT.Mgnt_Transpose_8x8
165	org.sc29.wg11.mpeg4.part2.main.IT.Mgnt_Transpose_16x16
166	org.sc29.wg11.mpeg4.part2.main.IT.Mgnt_Transpose_32x32
167	org.sc29.wg11.mpeg4.part2.main.IT.Mgnt_Transpose_32x32
168	org.sc29.wg11.mpeg4.part2.main.IT.Algo_IT4x4_1d
169	org.sc29.wg11.mpeg4.part2.main.IT.Algo_IT8x8_1d
170	org.sc29.wg11.mpeg4.part2.main.IT.Algo_IT16x16_1d
171	org.sc29.wg11.mpeg4.part2.main.IT.Algo_IT32x32_1d
172	org.sc29.wg11.mpeg4.part2.main.IT.Algo_invDST4x4_1st
173	org.sc29.wg11.mpeg4.part2.main.IT.Algo_invDST4x4_2nd

### 4.3 Token pool

Every token is listed in the “token pool” that is the table of managing all tokens used in VTL. To facilitate the feasibility of connections among input and output ports of different FUs described in this specification, Table 3 lists all data elements (called “token”, which is used throughout this document). The ID field here is informative and used for easy lookup.

**Table 3 — List of all token types that are used in the descriptions of FUs in this subclause**

ID & Name	Description
<b>1 BIT</b>	Token which value is 0 or 1. The bits belongs to the non-decoded bitstream.
<b>2 ACKNOWLEDGMENT</b>	Boolean token (True or False) indicating a acknowledgment. True means it is ok, False, not ok.
<b>3 MCBPC</b>	Token representing the MCBPC element of syntax
<b>4 CBPY</b>	Token representing the CBPY element of syntax
<b>5 DCT_DC_SIZE</b>	Token representing the element of syntax DCT_DC_SIZE
<b>6 DCT_DC_DIFF</b>	Token representing the element of syntax DCT_DC_DIFF

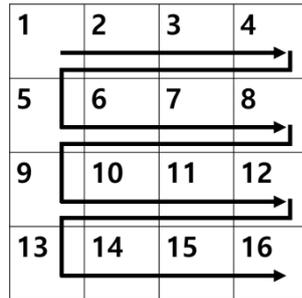
ID & Name	Description
<b>7 RUN</b>	Token representing the RUN value in the decoding of the DCT coefficients
<b>8 VALUE</b>	Token representing the VALUE value in the decoding of the DCT coefficients
<b>9 LAST</b>	Token representing the LAST value in the decoding of the DCT coefficients
<b>10 MEM_ADDRESS</b>	Token representing an address in the memory of the frames
<b>11 MEM_WIDTH</b>	Token representing the width in pixel of the data to read or write in the memory of the frames starting from the MEM_ADDRESS information
<b>12 MEM_HEIGHT</b>	Token representing the height in pixel of the data to read or write in the memory of the frames starting from the MEM_ADDRESS information
<b>13 MEM_DATA</b>	Token representing a data stored in the memory of the frames
<b>14 WIDTH</b>	Token representing the width value of video frame in pixels
<b>15 HEIGHT</b>	Token representing the height value of video frame in pixels
<b>16 SIZE</b>	Token representing the size of the current frame in number of macroblock in width and number of macroblock in height
<b>17 DC</b>	Tokens representing the DC coefficients. Each token represent one coefficient
<b>18 AC</b>	Token representing AC coefficients without DC coefficients
<b>19 BLOCK</b>	Token representing BLOCK that consists of 8×8 pixels
<b>20 MB</b>	Token representing a macroblock that consists of BLOCKs
<b>21 MVD</b>	Tokens representing the motion vector differences decoded by the syntax parsing process
<b>22 MV</b>	Tokens representing the coordinates of the motion vectors
<b>23 QUANT</b>	Token representing the QUANT value of quantization
<b>24 COORDINATE</b>	Token representing coordinates of block or macroblocks
<b>25 DISPLACEMENT</b>	Token representing the displacement between pixels (e.g., half- or quarter-pixel)
<b>26 SIGN</b>	Token representing a sign.
<b>27 ROUND</b>	Boolean token (True or False) indicating whether rounding is to be made or not
<b>28 BTYPE</b>	Integer token (2 = INTRA), (1 = INTER), (15 = NEW VOP)
<b>29 ACCODED</b>	Boolean token (True or False) indicating whether AC is coded or not
<b>30 ACPRED</b>	Boolean token (True or False) indicating whether AC prediction is made or not
<b>31 ACPRED_DIR</b>	Token representing the order of prediction of the AC coefficients
<b>32 MOTION</b>	Boolean token (True or False) indicating whether motion predication is made or not
<b>33 FOURMV</b>	Boolean token (True or False) indicating whether FOURMV is to be used or not

ID & Name	Description
34 F_CODE	Token representing a value of FCODE of VOP to specify the range of motion vectors
35 RBSP	Token representing the data in the Raw Byte Sequence Payload
36 NAL_SIZE	Token representing the size in byte of a Network Abstraction Layer unit
37 SUB_MB_TYPE	Token representing sub mb type value for sub partition
38 FRAME_NUM	Token representing the frame number of the current picture to decode
39 FRAME_TO_READ	Token representing the frame to read
40 PART_SIZE	Token representing the size in pixel for a partition of a macroblock, first the width of the partition, then the height
41 REF_ID	Token representing the identification of the decoded reference frame in memory
42 MB_ID	Token representing the number which identified a macroblock in a frame in position x and y. The macroblocks are counted in a frame using raster scan order.
43 POC	Token representing the index of the frame to display
44 REF_ORDER	Token representing the index of frames to store into long frame reference and short term reference
45 MMCO	Token representing the order of the index for frame to store in memory
46 PRED_MODE_INTRA	Token representing the prediction mode of an intra macroblock
47 MB_TYPE	Token representing the type of prediction used by a macroblock (Intra, Intra 4×4, Inter, new frame IDR, non IDR or new slice )
48 SLICE_DBF_PARAM	Tokens representing for each slice if deblocking filter is enabled and the offset used in accessing the $\alpha$ and $tC0$ deblocking filter tables for filtering operations and the offset used in accessing the $\beta$ deblocking filter table for filtering operations for each slice
49 MB_DBF_PARAM	Tokens representing for each MB the offset used in accessing the $\beta$ deblocking filter table for filtering operations
50 CBP_BLK	Token representing which of the sixteen 4×4 luma blocks of a macroblock may contain non-zero transform coefficient levels
51 SELECT	Token representing which blocks will be selected (from 0-3)
52 ST_LIST	Token representing the short term reference list
53 ST_LIST_SZ	Token representing the size of the short term reference list
54 DEL_LIST	Token representing the list of reference frames that have to be deleted
55 BOOLEAN	Token which value is 0 or 1
56 SIGN	Token which value is 0 or 1
57 FLAG	Token which value is 0 or 1
58 UINT_2	Unsigned integer on 2 bits
59 UINT_4	Unsigned integer on 4 bits

ID & Name	Description
60 UINT_8	Unsigned integer on 8 bits
61 UINT_16	Unsigned integer on 16 bits
62 UINT_32	Unsigned integer on 32 bits

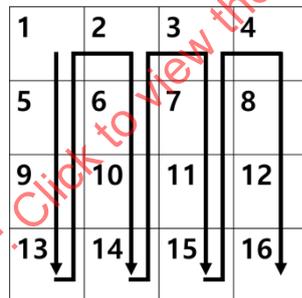
**4.4 Array data order**

— **Row based:** The data is processed or sent in a sequential order, row by row.



**Figure 3 — Example of row-based array data order**

— **Column based:** The data is processed or sent in a sequential order, column by column.



**Figure 4 — Example of column-based array data order**

**4.5 Input ports (reset\_i, init\_i, start\_i)**

An FU does not have an outside synchronization signal or synchronization mechanism. These ports are used for the purpose of changing the values of the local variables to default or initialization values.

**4.6 FU block diagram notations**

The notation [EMBED] defines a part of the main FSM schematic that is described as a separate schematic (for complexity reasons). The [EMBED] schematic is an integrated part of the main FSM schematic.

The notation [MODULE] defines a part of the main FSM schematic that is defined as a separate FU. The module schematic is integrated in the main schematic with the entire FU logic, except the “START” FSM state. The INPUT/OUTPUT states do not read or write values from the ports, they refer to local variables relative to the FU that embeds the other schematic.

## 4.7 Conventions

The significance of the “sign” port values is:

Value	Significance
0	negative
1	positive

The significance of the “flag” values is:

Value	Significance
0	false
1	true

## 4.8 Granular FU

When a decoder is implemented using the FUs, two or more instances of FUs can be grouped and can be dealt as a single instance of FU with bigger granularity. This is called Granular FU, and the use of Granular FU during implementation is normatively allowed. More detailed description about Granular FU concept is given in Annex D, and informative examples of Granular FU can be found from Annex E.

## 5 General-purpose FUs

### 5.1 Syntax parsing

#### 5.1.1 Generic syntax parser

<b>FU Name</b>	Algo_SynP_Generic	
<b>Description</b>	This is a generic syntax parser that needs BSD as an input. Input and output port will be defined as the information in the BSD.	
<b>Profiles@levels supported</b>	All (Generic)	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>

#### 5.1.2 Algo\_Byte2bit

<b>FU Name</b>	Algo_Byte2bit	
<b>Description</b>	This module will take 8 bits in once and will output them one by one.	
<b>Profiles@levels supported</b>	All (Generic)	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
BYTE	BIT token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
BITS	BIT token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.common		

## 5.1.3 Mgnt\_Select\_MB\_4

<b>FU Name</b>	Mgnt_Select_MB_4	
<b>Description</b>	This module selects data from one of its 4 inputs and sends it to its output (X) according to select information.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
IN_0	MB token	
IN_1	MB token	
IN_2	MB token	
IN_3	MB token	
SELECT	SELECT token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
OUT	MB token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
MB_WIDTH	Size in pixel of a macroblock	[8;16]
<b>Package</b>		
package org.sc29.wg11.common		

## 5.1.4 Mgnt\_Merger420

<b>FU Name</b>	Mgnt_Merger420	
<b>Description</b>	This module merges data from YUV to compose a MB 16×16.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
Y	BLOCK token	
U	BLOCK token	
V	BLOCK token	
Select	SELECT token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
YUV	MB token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
MB_WIDTH	Size in pixel of a macroblock	[8;16]
<b>Package</b>		
package org.sc29.wg11.common		

## 5.1.5 Mgnt\_Select\_MB\_8

<b>FU Name</b>	Mgnt_Select_MB_8	
<b>Description</b>	This module selects data from one of its 8 inputs and sends it to its output according to select information.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	

Input		
Name	Token	
IN_0	MB token	
IN_1	MB token	
IN_2	MB token	
IN_3	MB token	
IN_4	MB token	
IN_5	MB token	
IN_6	MB token	
IN_7	MB token	
SELECT	SELECT token	
Output		
Name	Token	
OUT	MB token	
Parameter		
Name	Description	Range
MB_WIDTH	Size in pixel of a macroblock	[8;16]
Package		
package org.sc29.wg11.common		

### 5.1.6 Mgnt\_SynP\_DEMUX

<b>FU Name</b>	Mgnt_SynP_DEMUX	
<b>Description</b>	<p>This DEMUX FU can be used to distribute serialized output through a number of different output ports. A potential use of this FU is distribution of output tokens derived from the generic syntax parser FU with serial output.</p> <p>The value of PORT_ID is used to designate the output port which the data from PORT_DATA will be sent.</p> <p>The output port will be defined as the information in the BSD or the FN.</p> <p>NOTE FU descriptions for the Informative examples of this FU can be found in Annex I.</p>	
<b>Profiles@levels supported</b>	All (Generic)	
Input		
Name	Token	
PORT_ID	BIT token	
PORT_DATA	BIT token	
Output		
Name	Token	
Parameter		
Name	Description	Range

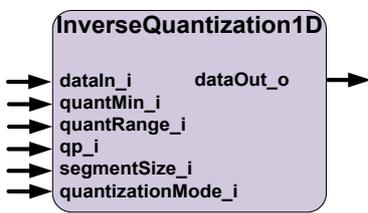
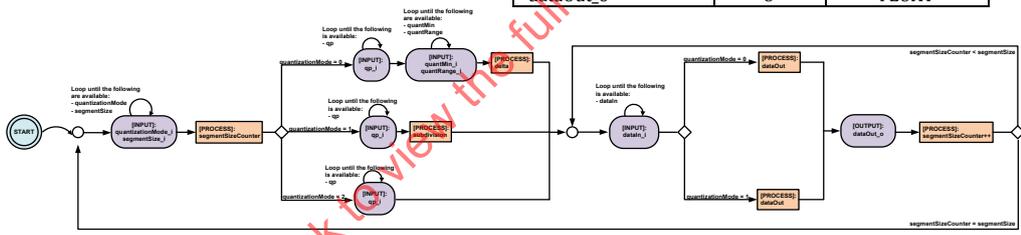
### 5.1.7 Mgnt\_SynP\_BytePreprocessor

<b>FU Name</b>	Mgnt_SynP_BytePreprocessor	
<b>Description</b>	<p>This FU pre-processes the input bitstream to remove the emulation prevention codes (i.e. decodes NAL RBSP structure). The output of this FU is a raw encoded video bistream.</p>	
<b>Profiles@levels supported</b>	HEVC Main Profile	
Input		
Name	Token	
BITS_IN	BIT token	

<b>Output</b>		
<b>Name</b>	<b>Token</b>	
BITS_OUT	BIT token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>	package org.sc29.wg11.mpegH.part2.main.synParser	

## 5.2 General processing FUs

### 5.2.1 Algo\_InverseQuantization1D

<b>FU Name</b>	Algo_InverseQuantization1D																								
<b>Description</b>	 <table border="1" data-bbox="861 739 1340 1030"> <thead> <tr> <th>Port Name</th> <th>Direction (I/O)</th> <th>Token TYPE</th> </tr> </thead> <tbody> <tr> <td>dataIn_i</td> <td></td> <td>INT8, INT16, INT32, INT64</td> </tr> <tr> <td>qp_i</td> <td>I</td> <td>UINT_32</td> </tr> <tr> <td>quantMin_i</td> <td>I</td> <td>FLOAT</td> </tr> <tr> <td>quantRange_i</td> <td>I</td> <td>FLOAT</td> </tr> <tr> <td>segmentSize_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>quantizationMode_i</td> <td>I</td> <td>UINT_2</td> </tr> <tr> <td>dataOut_o</td> <td>O</td> <td>FLOAT</td> </tr> </tbody> </table>  <p><b>Inverse Quantization Process:</b>  <b>START:</b>          INPUT_SEGMENT_PARAM:          INPUT:              quantizationMode              segmentSize          SegmentSizeCounter = 0          IF quantizationMode = 0              INPUT:                  qp                  quantMin                  quantRange              IF ( quantRange &gt; 0.0 )                  delta = ( ( 1 &lt;&lt; qp ) - 1 ) / quantRange              ELSE                  delta = 1.0              IF quantizationMode = 1                  INPUT:                      qp              INPUT_DATA_IN:              INPUT:                  dataIn              IF quantizationMode = 0                  PROCESS:                      dataOut = quantMin + ( dataIn / delta )              IF quantizationMode = 1                  PROCESS:                      dataOut = dataIn / qp              OUTPUT                  dataOut              SegmentSizeCounter ++</p>	Port Name	Direction (I/O)	Token TYPE	dataIn_i		INT8, INT16, INT32, INT64	qp_i	I	UINT_32	quantMin_i	I	FLOAT	quantRange_i	I	FLOAT	segmentSize_i	I	UINT8, UINT16, UINT32, UINT64	quantizationMode_i	I	UINT_2	dataOut_o	O	FLOAT
	Port Name	Direction (I/O)	Token TYPE																						
dataIn_i		INT8, INT16, INT32, INT64																							
qp_i	I	UINT_32																							
quantMin_i	I	FLOAT																							
quantRange_i	I	FLOAT																							
segmentSize_i	I	UINT8, UINT16, UINT32, UINT64																							
quantizationMode_i	I	UINT_2																							
dataOut_o	O	FLOAT																							

	<p>IF SegmentSizeCounter &lt; segmentSize GOTO INPUT_DATA_IN ELSE GOTO INPUT_SEGMENT_PARAM</p> <p>The following table contains the quantization types index used in the Inverse Quantization 1D FU:</p> <table border="1"> <thead> <tr> <th>Name of quantization mode</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td>Uniform Quantization</td> <td>0</td> </tr> <tr> <td>Uniform Texture Quantization</td> <td>1</td> </tr> </tbody> </table> <p>The "Inverse Quantization" is an algorithm (step by step procedures) that allows a set of data to be represented with a limited set of values that are associated with its nearest representative. For a number of "segmentSize" of input data (dataIn), it uses the same set of quantMin, quantRange and quantValue to produce a set of output data (dataOut) of size "segmentSize".</p>	Name of quantization mode	Value	Uniform Quantization	0	Uniform Texture Quantization	1
Name of quantization mode	Value						
Uniform Quantization	0						
Uniform Texture Quantization	1						
<b>ISO Standards using the FU</b>	ISO/IEC 14496-16						
<b>Profiles@levels supported</b>	All (Generic)						
<b>Input</b>							
<b>Name</b>	<b>Token</b>						
dataIn_i	INT8, INT16, INT32, INT64						
qp_i	UINT_32						
quantMin_i	FLOAT						
quantRange_i	FLOAT						
segmentSize_i	UINT8, UINT16, UINT32, UINT64						
quantizationMode_i	UINT_2						
<b>Output</b>							
<b>Name</b>	<b>Token</b>						
dataOut_o	FLOAT						
<b>Package</b>							
package org.sc29.wg11.mpeg4.part16							

### 5.2.2 Algo\_InverseQuantizationND

<b>FU Name</b>	Algo_InverseQuantizationND																								
<b>Description</b>	<div style="display: flex; align-items: flex-start;"> <div style="margin-right: 20px;"> </div> <table border="1"> <thead> <tr> <th>Port Name</th> <th>Direction (I/O)</th> <th>Token TYPE</th> </tr> </thead> <tbody> <tr> <td>dataIn_i</td> <td>I</td> <td>INT8, INT16, INT32, INT64</td> </tr> <tr> <td>qp_i</td> <td>I</td> <td>UINT_32</td> </tr> <tr> <td>quantMin_i</td> <td>I</td> <td>FLOAT</td> </tr> <tr> <td>quantRange_i</td> <td>I</td> <td>FLOAT</td> </tr> <tr> <td>segmentSize_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>quantizationMode_i</td> <td>I</td> <td>UINT_2</td> </tr> <tr> <td>dataOut_o</td> <td>O</td> <td>FLOAT</td> </tr> </tbody> </table> </div> 	Port Name	Direction (I/O)	Token TYPE	dataIn_i	I	INT8, INT16, INT32, INT64	qp_i	I	UINT_32	quantMin_i	I	FLOAT	quantRange_i	I	FLOAT	segmentSize_i	I	UINT8, UINT16, UINT32, UINT64	quantizationMode_i	I	UINT_2	dataOut_o	O	FLOAT
Port Name	Direction (I/O)	Token TYPE																							
dataIn_i	I	INT8, INT16, INT32, INT64																							
qp_i	I	UINT_32																							
quantMin_i	I	FLOAT																							
quantRange_i	I	FLOAT																							
segmentSize_i	I	UINT8, UINT16, UINT32, UINT64																							
quantizationMode_i	I	UINT_2																							
dataOut_o	O	FLOAT																							

```

Inverse Quantization Process:

$$dimQ = \begin{cases} dimD, & \text{if homogeneous}Q = 0 \\ 1, & \text{if homogeneous}Q = 1 \end{cases}$$


START:
INPUT_SEGMENT_PARAM
INPUT:
    quantizationMode
    segmentSize
SegmentSizeCounter = 0
IF quantizationMode = 0
    INPUT:
        qp
        quantMin [ dimQ ]
        quantRange [ dimQ ]
    WHILE dimQ_counter < dimQ
        IF ( quantRange [ dimQ_counter ] > 0.0 )
            delta [ dimQ_counter ] = ( ( 1 << qp ) - 1 ) / quantRange [ dimQ_counter ]
        ELSE
            delta [ dimQ_counter ] = 1.0;
        dimQ_counter++
IF quantizationMode = 1
    INPUT:
        Qp [ dimQ ]
IF quantizationMode = 2
    INPUT:
        qp
    PROCESS:
        Subdivision = (qp - 3) / 2

INPUT_DATA_IN:
INPUT:
    dataIn [ dimQ ]
IF quantizationMode = 0
    PROCESS:
        dataOut = quantMin [ segmentSizeCounter%dimD ] + (dataIn / delta [segmentSizeCounter%dimD])
IF quantizationMode = 1
    PROCESS: EMBED Code 2 Normal
IF quantizationMode = 2
    PROCESS:
        dataOut = dataIn / qp [ segmentSizeCounter % dimD ]
OUTPUT
    dataOut
SegmentSizeCounter ++
IF SegmentSizeCounter < segmentSize
    GOTO INPUT_DATA_IN
ELSE
    GOTO INPUT_SEGMENT_PARAM

EMBED: Code 2 Normal
Mask = ( 1 << ( 2 * subdivision ) ) - 1
tricode = data & mask;

// Find y coordinate by solving 2nd degree equation
factor = 1 << subdivision
y = factor - sqrt ( (factor^2) - tricode )
tricode = tricode + ( y * ( y - (2 * factor) ) )
x = tricode / 2
upsideDown = tricode % 2

// Calculate coordinates for all vertices in triangle
v1x = x + upsideDown
v1y = y + upsideDown
v2x = x + 1
v2y = y
v3x = x
v3y = y + 1

// Calculate coordinates of barycenter
invMaxCoord = 1 / factor
normal [ 0 ] = (v1x + v2x + v3x) * invMaxCoord
    
```

STANDARD.PDF.COM : Click to view the full PDF of ISO/IEC 23002-4:2018

	<pre> normal [ 1 ] = (v1y + v2y + v3y) * invMaxCoord normal [ 2 ] = 3 - normal [ 0 ] - normal [ 1 ]  // Flip component signs if necessary octantCode = ( data &gt;&gt; 2 * subdivision ) &amp; 0x7 if (octantCode &amp; 0x4)     normal [ 0 ] = (-1) * normal [ 0 ] if (octantCode &amp; 0x2)     normal [ 1 ] = (-1) * normal [ 1 ] if (octantCode &amp; 0x1)     normal [ 2 ] = (-1) * normal [ 2 ]  invNorm:= 1 / sqrt ( (normal[ 0 ])² + (normal [ 1 ])² + (normal [ 2 ])²);  //Write the 3 output values normal [ 0 ] = normal [ 0 ] * invNorm normal [ 1 ] = normal [ 1 ] * invNorm normal [ 2 ] = normal [ 2 ] * invNorm  dataOut = normal                 </pre> <p>The following table contains the quantization types index used in the Inverse Quantization ND FU:</p> <table border="1" data-bbox="395 748 916 857"> <thead> <tr> <th>Name of quantization mode</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td>Uniform Quantization</td> <td>0</td> </tr> <tr> <td>Normal Quantization</td> <td>1</td> </tr> <tr> <td>Uniform Texture Quantization</td> <td>2</td> </tr> </tbody> </table> <p>The "Inverse Quantization" is an algorithm (step by step procedures) that allows a set of data to be represented with a limited set of values that are associated with its nearest representative. For a number of "segmentSize" x "dimD" of input data (dataIn), it uses the same set of quantMin, quantRange and quantValue of size "dimD" to produce a set of output data (dataOut) of size segmentSize" x "dimD". For each set of size "dimD" of input data (dataIn) it uses the corresponding value of quantMin, quantRange and quantValue.</p>	Name of quantization mode	Value	Uniform Quantization	0	Normal Quantization	1	Uniform Texture Quantization	2
Name of quantization mode	Value								
Uniform Quantization	0								
Normal Quantization	1								
Uniform Texture Quantization	2								
<b>ISO Standards using the FU</b>	ISO/IEC 14496-16								
<b>Profiles@levels supported</b>	All (Generic)								
<b>Input</b>									
<b>Name</b>	<b>Token</b>								
dataIn_i	INT8, INT16, INT32, INT64								
qp_i	UINT_32								
quantMin_i	FLOAT								
quantRange_i	FLOAT								
segmentSize_i	UINT8, UINT16, UINT32, UINT64								
quantizationMode_i	UINT_2								
<b>Output</b>									
<b>Name</b>	<b>Token</b>								
dataOut_o	FLOAT								
<b>Parameter</b>									
<b>Name</b>	<b>Description</b>	<b>Type/range</b>							
dimD	Describes the number of tokens of type dataIn_i that are consumed at each firing. This parameter is set at the network configuration level.	Type: Integer Range: [1 .. 2 <sup>5</sup> ]							
homogeneousQ	Describes the number of tokens of type quantRange_i, quantMin_i and quantValue_i that are necessary for the inverse quantization process. This parameter is set at the network configuration level. The number of tokens is equal to dimD if this parameter is 0 and the number of tokens is equal to 1 if this parameter is 1	Type: Boolean Range: {0,1}							
<b>Package</b>									
package org.sc29.wg11.mpeg4.part16									

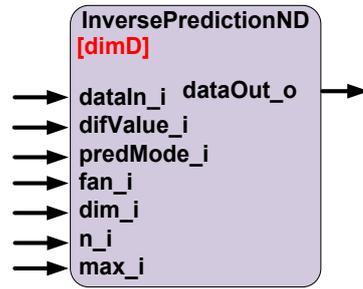


	<p>Inverse Prediction Process:</p> <p>Switch ( predMode )</p> <pre> {   Case 0: NP – No Prediction     dataOut[j] = dataIn[j],    ∀j ∈ {0 .. N – 1}    Case 1: Diff – Differential Prediction     dataOut[j] = {       dataIn[j], ∀j ∈ 0       dataIn[j] + dataOut[j – 1], ∀j ∈ {1 .. N – 1}    Case 2: XOR – based prediction     dataOut[j] = {       dataIn[j], ∀j ∈ 0       dataIn[j] ⊗ dataOut[j – 1], ∀j ∈ {1 .. N – 1}    Case 3: Adaptive Prediction     dataOut[j] = {       dataIn[j], if difValue[j] = 0 or j = 0       dataIn[j] + dataOut[j – difValue[j]], otherwise    Case 4: Circular Differential Prediction     dataOut[j] = {       dataIn[j], ∀j ∈ 0       d, if dataIn[j] &lt; dataIn[j – 1], where d = (...)       –d, otherwise     d = {       dataIn[j – 1] + M<sub>d</sub> – dataIn[j], if dataIn[j] &gt; dataIn[j – 1]       dataIn[j] + M<sub>d</sub> – dataIn[j – 1], otherwise    Case 5: Parallelogram Inverse Prediction    a = fan ( 0 )   if a not visited     d = 0     WHILE d &lt; dim       dataOut [ a x dim + d ] = dataIn [ a x dim + d ]       d = d + 1     Mark a as visited   If number of vertexes &gt; 3     b = fan ( 1 )     if b not visited       d = 0       WHILE d &lt; dim         dataOut [ b x dim + d ] = dataIn [ b x dim + d ] + dataOut [ a x dim + d ]         d = d + 1       Mark b as visited     c = fan ( 2 )     if c not visited       init h, t<sub>a</sub>       WHILE t<sub>a</sub> = (l,j,k) ∈ Triangles incident to vertex a         If ( i=a &amp; j=b &amp; k≠c and k not visited ) h = k, break         If ( i=a &amp; k=b &amp; j≠c and j not visited ) h = k, break         If ( j=a &amp; i=b &amp; k≠c and k not visited ) h = k, break         If ( j=a &amp; k=b &amp; i≠c and i not visited ) h = k, break         If ( k=a &amp; i=b &amp; j≠c and j not visited ) h = k, break         If ( k=a &amp; b=b &amp; i≠c and i not visited ) h = k, break       If h not set         d = 0         WHILE d &lt; dim           dataOut [ c x dim + d ] = dataIn [ c x dim + d ] + dataOut [ b x dim + d ]           d = d + 1         else           d = 0           WHILE d &lt; dim             dataOut [ c x dim + d ] =               dataIn [ c x dim + d ] + dataOut [ b x dim + d ]               – dataOut [ h x dim + d ] + dataOut [ a x dim + d ]            d = d + 1 </pre>
--	---

	<pre> Mark c as visited g = 3, d = 0 WHILE g &lt; fanSize   q = fan ( g )   if q not visited     WHILE d &lt; dim       dataOut [ q x dim + d ] =         dataIn [ q x dim + d ] + dataOut [ c x dim + d ]         - dataOut [ b x dim + d ] + dataOut [ a x dim + d ]        d = d + 1     b = c     c = q </pre> <p>The following table contains the prediction types index used in the Inverse Prediction 1D FU:</p> <table border="1"> <thead> <tr> <th>Name of prediction mode</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td>No Prediction</td> <td>0</td> </tr> <tr> <td>Differential Prediction</td> <td>1</td> </tr> <tr> <td>XOR based prediction</td> <td>2</td> </tr> <tr> <td>Adaptive Differential Prediction</td> <td>3</td> </tr> <tr> <td>Circular Differential Prediction</td> <td>4</td> </tr> <tr> <td>Parallelogram Inverse Prediction</td> <td>5</td> </tr> </tbody> </table> <p>The detailed description of the inverse parallelogram prediction is described in Annex F.</p>	Name of prediction mode	Value	No Prediction	0	Differential Prediction	1	XOR based prediction	2	Adaptive Differential Prediction	3	Circular Differential Prediction	4	Parallelogram Inverse Prediction	5
Name of prediction mode	Value														
No Prediction	0														
Differential Prediction	1														
XOR based prediction	2														
Adaptive Differential Prediction	3														
Circular Differential Prediction	4														
Parallelogram Inverse Prediction	5														
<b>ISO Standards using the FU</b>	ISO/IEC 14496-16														
<b>Profiles@levels supported</b>	All (Generic)														
<b>Input</b>															
<b>Name</b>	<b>Token</b>														
dataIn_i	INT8, INT16, INT32, INT64														
difValue_i	INT8, INT16, INT32, INT64														
predMode_i	UINT_4														
fan_i	INT8, INT16, INT32, INT64														
dim_i	UINT8, UINT16, UINT32, <del>UINT64</del>														
n_i	INT8, INT16, INT32, INT64														
max_i	UINT8, UINT16, <del>UINT32, UINT64</del>														
<b>Output</b>															
<b>Name</b>	<b>Token</b>														
dataOut_o	INT8, INT16, INT32, INT64														
<b>Parameter</b>															
<b>Name</b>	<b>Description</b>	<b>Range</b>													
<b>Package</b>															
package org.sc29.wg11.mpeg4.part16															

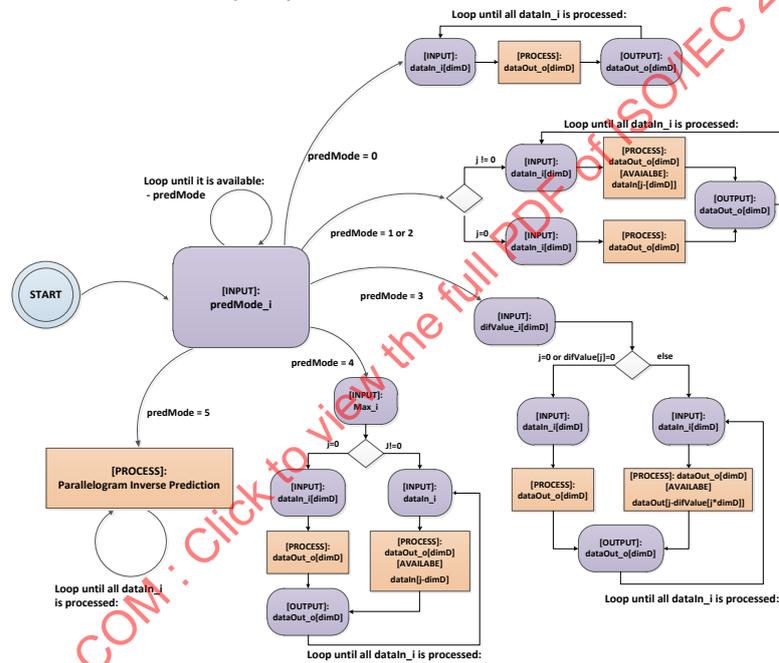
5.2.4 Algo\_InversePredictionND

<b>FU Name</b>	Algo_InversePredictionND
<b>Description</b>	This FU allows five modes of inverse prediction. Not all the ports are used for each prediction mode. This FU can handle a number of dimD input tokens at a time.

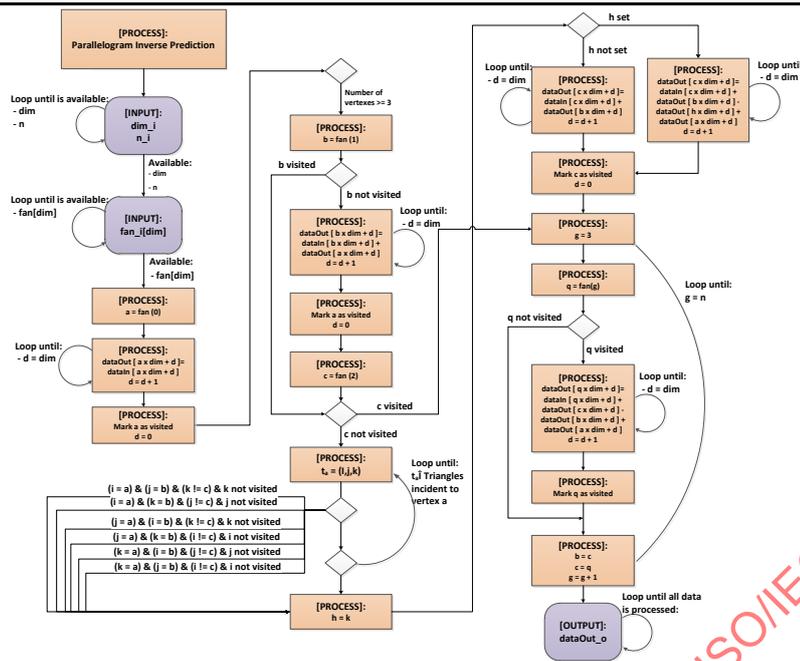


Port Name	Direction (I/O)	Token RANGE
dataIn_i	I	INT8, INT16, INT32, INT64
difValue_i	I	INT8, INT16, INT32, INT64
predMode_i	I	UINT_4
fan_i	I	INT8, INT16, INT32, INT64
dim_i	I	UINT8, UINT16, UINT32, UINT64
n_i	I	INT8, INT16, INT32, INT64
max_i	I	UINT8, UINT16, UINT32, UINT64
dataOut_o	O	INT8, INT16, INT32, INT64

Process Schematic (FSM):



Process Parallelogram Inverse Prediction Schematic (FSM):



Inverse Prediction ND Process:  
Switch( predMode )

```

{
  Case 0: NP – No Prediction
    dataOut[j] = dataIn[j],    ∀j ∈ {0 .. (N × dimD) – 1}

  Case 1: Diff – Differential Prediction
    dataOut[j] = {
      dataIn[j], ∀j ∈ 0
      dataIn[j] + dataOut[j – dimD], ∀j ∈ {1 .. (N × dimD) – 1}
    }

  Case 2: XOR – based prediction
    dataOut[j] = {
      dataIn[j], ∀j ∈ 0
      dataIn[j] ⊗ dataOut[j – dimD], ∀j ∈ {1 .. (N × dimD) – 1}
    }

  Case 3: Adaptive Prediction
    dataOut[j] = {
      dataIn[j], if difValue[j] = 0 or j = 0
      dataIn[j] + dataOut[j – difValue[j] × dimD], otherwise
    }

  Case 4: Circular Differential Prediction
    dataOut[j] = {
      dataIn[j], if difValue[j] = 0 or j = 0
      dataIn[j] + dataOut[j – difValue[j] × dimD], otherwise
    }
    d = {
      dataIn[j – dimD] + Md – dataIn[j], if dataIn[j] > dataIn[j – dimD]
      dataIn[j] + Md – dataIn[j – dimD], otherwise
    }

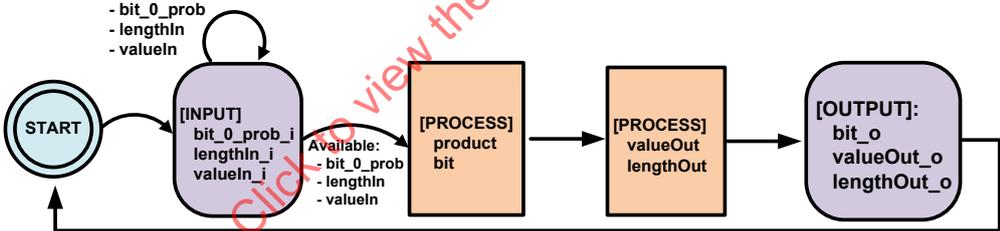
  Case 5: Parallelogram Inverse Prediction
    a = fan ( 0 )
    if a not visited
      d = 0
      WHILE d < dim
        dataOut [ a x dim + d ] = dataIn [ a x dim + d ]
        d = d + 1
      Mark a as visited
    If number of vertexes > 3
      b = fan ( 1 )
      if b not visited
        d = 0
        WHILE d < dim
  
```

STANDARDSISO.COM · Click to view the full PDF of ISO/IEC 23002-4:2018

	<pre> dataOut [ b x dim + d ] = dataIn [ b x dim + d ] + dataOut [ a x dim + d ] d = d + 1 Mark b as visited c = fan ( 2 ) if c not visited     init h, t<sub>a</sub>     WHILE t<sub>a</sub> = (l,j,k) ∈ Triangles incident to vertex a         If ( i=a &amp; j=b &amp; k≠c and k not visited ) h = k, break         If ( i=a &amp; k=b &amp; j≠c and j not visited ) h = k, break         If ( j=a &amp; i=b &amp; k≠c and k not visited ) h = k, break         If ( j=a &amp; k=b &amp; i≠c and i not visited ) h = k, break         If ( k=a &amp; i=b &amp; j≠c and j not visited ) h = k, break         If ( k=a &amp; b=b &amp; i≠c and i not visited ) h = k, break     If h not set         d = 0         WHILE d &lt; dim             dataOut [ c x dim + d ] = dataIn [ c x dim + d ] + dataOut [ b x dim + d ]             d = d + 1         else             d = 0             WHILE d &lt; dim                 dataOut [ c x dim + d ] =                     dataIn [ c x dim + d ] + dataOut [ b x dim + d ]                     - dataOut [ h x dim + d ] + dataOut [ a x dim + d ]                 d = d + 1             Mark c as visited             g = 3, d = 0             WHILE g &lt; fanSize                 q = fan ( g )                 if q not visited                     WHILE d &lt; dim                         dataOut [ q x dim + d ] =                             dataIn [ q x dim + d ] + dataOut [ c x dim + d ]                             - dataOut [ b x dim + d ] + dataOut [ a x dim + d ]                         d = d + 1                     b = c                     c = q </pre> <p>The following table contains the prediction types index used in the Inverse Prediction ND FU:</p> <table border="1"> <thead> <tr> <th>Name of prediction mode</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td>No Prediction</td> <td>0</td> </tr> <tr> <td>Differential Prediction</td> <td>1</td> </tr> <tr> <td>XOR based prediction</td> <td>2</td> </tr> <tr> <td>Adaptive Differential Prediction</td> <td>3</td> </tr> <tr> <td>Circular Differential Prediction</td> <td>4</td> </tr> <tr> <td>Parallelogram Inverse Prediction</td> <td>5</td> </tr> </tbody> </table> <p>The detailed description of the inverse parallelogram prediction is described in Annex F.</p>	Name of prediction mode	Value	No Prediction	0	Differential Prediction	1	XOR based prediction	2	Adaptive Differential Prediction	3	Circular Differential Prediction	4	Parallelogram Inverse Prediction	5
Name of prediction mode	Value														
No Prediction	0														
Differential Prediction	1														
XOR based prediction	2														
Adaptive Differential Prediction	3														
Circular Differential Prediction	4														
Parallelogram Inverse Prediction	5														
<b>ISO Standards using the FU</b>	ISO/IEC 14496-16														
<b>Profiles@levels supported</b>	All (Generic)														
<b>Input</b>															
<b>Name</b>	<b>Token</b>														
dataIn_i	INT8, INT16, INT32, INT64														
difValue_i	INT8, INT16, INT32, INT64														
predMode_i	UINT_4														
fan_i	INT8, INT16, INT32, INT64														
dim_i	UINT8, UINT16, UINT32, UINT64														
n_i	INT8, INT16, INT32, INT64														
max_i	UINT8, UINT16, UINT32, UINT64														
<b>Output</b>															
<b>Name</b>	<b>Token</b>														
dataOut_o	INT8, INT16, INT32, INT64														

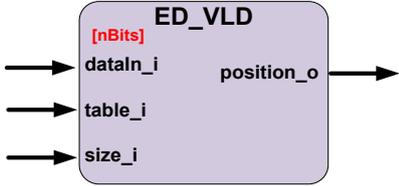
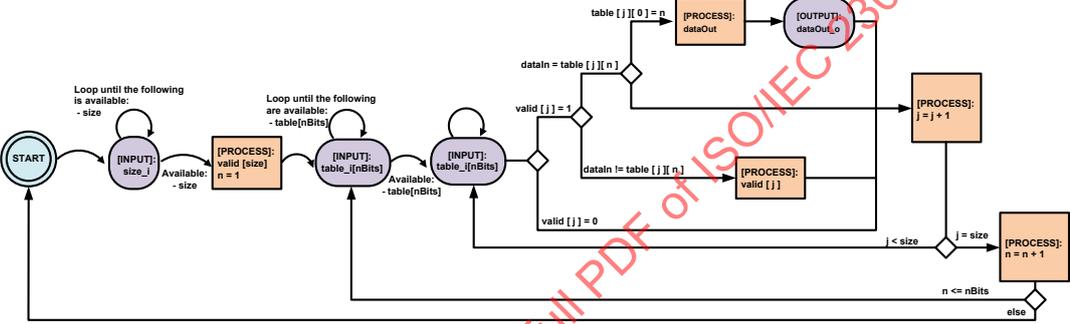
Parameter		
Name	Description	Range
dimD	Describes the number of tokens of type dataIn_i that are consumed at each firing. This parameter is set at the network configuration level.	Type: Integer Range: [1 .. 2 <sup>5</sup> ]
Package		
package org.sc29.wg11.mpeg4.part16		

### 5.2.5 Algo\_ED\_AD\_StaticBit

FU Name	Algo_ED_AD_StaticBit																					
Description	<p>This FU describes the arithmetic decoding process based on a static bit model.</p>  <table border="1" data-bbox="861 654 1359 1012"> <thead> <tr> <th>Port Name</th> <th>Direction (I/O)</th> <th>Token RANGE</th> </tr> </thead> <tbody> <tr> <td>Bit_0_prob_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>valueIn_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>length_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>valueOut_o</td> <td>O</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>lengthOut_o</td> <td>O</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>bit_o</td> <td>O</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> </tbody> </table> <p>Process Schematic (FSM):</p>  <pre> ED_AD_StaticBit Process: START INPUT:   Bit_0_prob_i   lengthIn_i   valueIn_i product = bit_0_prob * ( lengthIn &gt;&gt; BM_LengthShift ) bit = ( valueIn &gt;= product ) if ( bit == 0 )   valueOut = valueIn   lengthOut = product else   valueOut = valueIn - product   lengthOut = lengthIn - product OUTPUT:   valueOut_o   lengthOut_o GOTO START                     </pre>	Port Name	Direction (I/O)	Token RANGE	Bit_0_prob_i	I	UINT8, UINT16, UINT32, UINT64	valueIn_i	I	UINT8, UINT16, UINT32, UINT64	length_i	I	UINT8, UINT16, UINT32, UINT64	valueOut_o	O	UINT8, UINT16, UINT32, UINT64	lengthOut_o	O	UINT8, UINT16, UINT32, UINT64	bit_o	O	UINT8, UINT16, UINT32, UINT64
	Port Name	Direction (I/O)	Token RANGE																			
Bit_0_prob_i	I	UINT8, UINT16, UINT32, UINT64																				
valueIn_i	I	UINT8, UINT16, UINT32, UINT64																				
length_i	I	UINT8, UINT16, UINT32, UINT64																				
valueOut_o	O	UINT8, UINT16, UINT32, UINT64																				
lengthOut_o	O	UINT8, UINT16, UINT32, UINT64																				
bit_o	O	UINT8, UINT16, UINT32, UINT64																				
ISO Standards using the FU	ISO/IEC 14496-16																					
Profiles@levels supported	All (Generic)																					
Input																						
Name	Token																					
Bit_0_prob_i	UINT8, UINT16, UINT32, UINT64																					
valueIn_i	UINT8, UINT16, UINT32, UINT64																					

length_i	UINT8, UINT16, UINT32, UINT64	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
valueOut_o	UINT8, UINT16, UINT32, UINT64	
lengthOut_o	UINT8, UINT16, UINT32, UINT64	
bit_o	UINT8, UINT16, UINT32, UINT64	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
AC_MinLength	Describes the threshold for renormalization. This parameter is set at the network configuration level.	Type:Unsigned Integer Range: [1 .. 2 <sup>32</sup> ]
BM_LengthShift	Describes the length bits discarded before multiplication. This parameter is set at the network configuration level.	Type:Unsigned Integer Range: [1 .. 2 <sup>5</sup> ]
<b>Package</b>		
package org.sc29.wg11.mpeg4.part16		

5.2.6 Algo\_ED\_VLD

<p><b>FU Name</b></p>	<p>Algo_ED_VLD</p>															
<p><b>Description</b></p>	<p>This FU describes the Variable Length Decoding process.</p> <div style="display: flex; justify-content: space-around; align-items: flex-start;"> <div style="text-align: center;">  </div> <table border="1" style="border-collapse: collapse; text-align: center;"> <thead> <tr style="background-color: #4F81BD; color: white;"> <th>Port Name</th> <th>Direction (I/O)</th> <th>Token RANGE</th> </tr> </thead> <tbody> <tr> <td>dataIn_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>table_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>size_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>position_o</td> <td>O</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> </tbody> </table> </div> <p>Process Schematic (FSM):</p>  <pre> ED_VLD Process: START SET valid[size] = 1 SET n = 1 DO     DO         If valid[j] = 1             If dataIn=table[j][n]                 If table[j][0] = n                     dataOut = j                 else                     valid[j] = 0             j = j + 1         WHILE j &lt; size         n = n + 1     WHILE n &lt; nBits     GOTO START     </pre> <p>The matrix tokens have to be send column based.</p>	Port Name	Direction (I/O)	Token RANGE	dataIn_i	I	UINT8, UINT16, UINT32, UINT64	table_i	I	UINT8, UINT16, UINT32, UINT64	size_i	I	UINT8, UINT16, UINT32, UINT64	position_o	O	UINT8, UINT16, UINT32, UINT64
Port Name	Direction (I/O)	Token RANGE														
dataIn_i	I	UINT8, UINT16, UINT32, UINT64														
table_i	I	UINT8, UINT16, UINT32, UINT64														
size_i	I	UINT8, UINT16, UINT32, UINT64														
position_o	O	UINT8, UINT16, UINT32, UINT64														
<p><b>ISO Standards using the FU</b></p>	<p>ISO/IEC 14496-16</p>															
<p><b>Profiles@levels supported</b></p>	<p>All (Generic)</p>															
<p><b>Input</b></p>																
<p><b>Name</b></p>	<p><b>Token</b></p>															
<p>dataIn_i</p>	<p>UINT8, UINT16, UINT32, UINT64</p>															
<p>table_i</p>	<p>UINT8, UINT16, UINT32, UINT64</p>															
<p>size_i</p>	<p>UINT8, UINT16, UINT32, UINT64</p>															
<p><b>Output</b></p>																
<p><b>Name</b></p>	<p><b>Token</b></p>															
<p>position_o</p>	<p>UINT8, UINT16, UINT32, UINT64</p>															

Parameter		
Name	Description	Range
nBits	Describes the length of the bits used for the search algorithm. This parameter is set at the network configuration level.	Type: Integer Range: [1 .. 2 <sup>5</sup> ]
Package		
package org.sc29.wg11.mpeg4.part16		

### 5.2.7 Alog\_ED\_AD\_AdaptiveBit

FU Name	Algo_ED_AD_AdaptiveBit																					
	<p>This FU describes the arithmetic decoding process based on a adaptive bit model as presented in [3,4].</p> <div style="display: flex; justify-content: space-around;"> <div style="border: 1px solid black; padding: 5px; background-color: #e0e0e0;"> <p><b>ED_AD_AdaptiveBit</b>                      [AC_MinLength]                      [BM_LengthShift]                      [scaleMax]</p> <p>valueIn_i    valueOut_o                      lengthIn_i    lengthOut_o                      reset_i    bit_o</p> </div> <table border="1"> <thead> <tr> <th>Port Name</th> <th>Direction n (I/O)</th> <th>Token RANGE</th> </tr> </thead> <tbody> <tr> <td>valueIn_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>lengthIn_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>reset_i</td> <td>I</td> <td>BOOLEAN</td> </tr> <tr> <td>valueOut_o</td> <td>O</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>lengthOut_o</td> <td>O</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>bit_o</td> <td>O</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> </tbody> </table> </div> <p>Process Schematic (FSM):</p> <p><b>ED_AD_AdaptiveBit Process:</b></p> <pre> START If reset = 1     bit_0_count = 1;     bit_count = 2;     bit_0_prob = 1U &lt;&lt; (BM_LengthShift - 1);     update_cycle = bits_until_update = 4;  product = bit_0_prob * ( lengthIn &gt;&gt; BM_LengthShift ) bit = ( valueIn &gt;= product )  if ( bit == 0 )     valueOut = valueIn     lengthOut = product     ++bit_0_count else     valueOut = valueIn - product     lengthOut = lengthIn - product  if (bit_count += update_cycle) &gt; BM_MaxCount     bit_count = (bit_count + 1) &gt;&gt; 1     bit_0_count = (bit_0_count + 1) &gt;&gt; 1     if bit_0_count = bit_count         ++bit_count scale = scaleMax / bit_count; bit_0_prob = (bit_0_count * scale) &gt;&gt; (31 - BM_LengthShift); update_cycle = (5 * update_cycle) &gt;&gt; 2; if update_cycle &gt; 64     update_cycle = 64; bits_until_update = update_cycle; GOTO START     </pre>	Port Name	Direction n (I/O)	Token RANGE	valueIn_i	I	UINT8, UINT16, UINT32, UINT64	lengthIn_i	I	UINT8, UINT16, UINT32, UINT64	reset_i	I	BOOLEAN	valueOut_o	O	UINT8, UINT16, UINT32, UINT64	lengthOut_o	O	UINT8, UINT16, UINT32, UINT64	bit_o	O	UINT8, UINT16, UINT32, UINT64
Port Name	Direction n (I/O)	Token RANGE																				
valueIn_i	I	UINT8, UINT16, UINT32, UINT64																				
lengthIn_i	I	UINT8, UINT16, UINT32, UINT64																				
reset_i	I	BOOLEAN																				
valueOut_o	O	UINT8, UINT16, UINT32, UINT64																				
lengthOut_o	O	UINT8, UINT16, UINT32, UINT64																				
bit_o	O	UINT8, UINT16, UINT32, UINT64																				
Description																						

<b>ISO Standards using the FU</b>	ISO/IEC 14496-16	
<b>Profiles@levels supported</b>	All (Generic)	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
valueIn_i	UINT8, UINT16, UINT32, UINT64	
length_i	UINT8, UINT16, UINT32, UINT64	
reset_i	BOOLEAN	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
valueOut_o	UINT8, UINT16, UINT32, UINT64	
lengthOut_o	UINT8, UINT16, UINT32, UINT64	
bit_o	UINT8, UINT16, UINT32, UINT64	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
AC_MinLength	Describes the threshold for renormalization. This parameter is set at the network configuration level.	Type:Unsigned Integer Range: [1 .. 2 <sup>32</sup> ]
BM_LengthShift	Describes the length bits discarded before multiplication. This parameter is set at the network configuration level.	Type:Unsigned Integer Range: [1 .. 2 <sup>5</sup> ]
scaleMax	Describes the max value to compute the scaled bit 0 probability. This parameter is set at the network configuration level.	Type:Unsigned Integer Range: [1 .. 2 <sup>32</sup> ]
<b>Package</b>		
package org.sc29.wg11.mpeg4.part16		

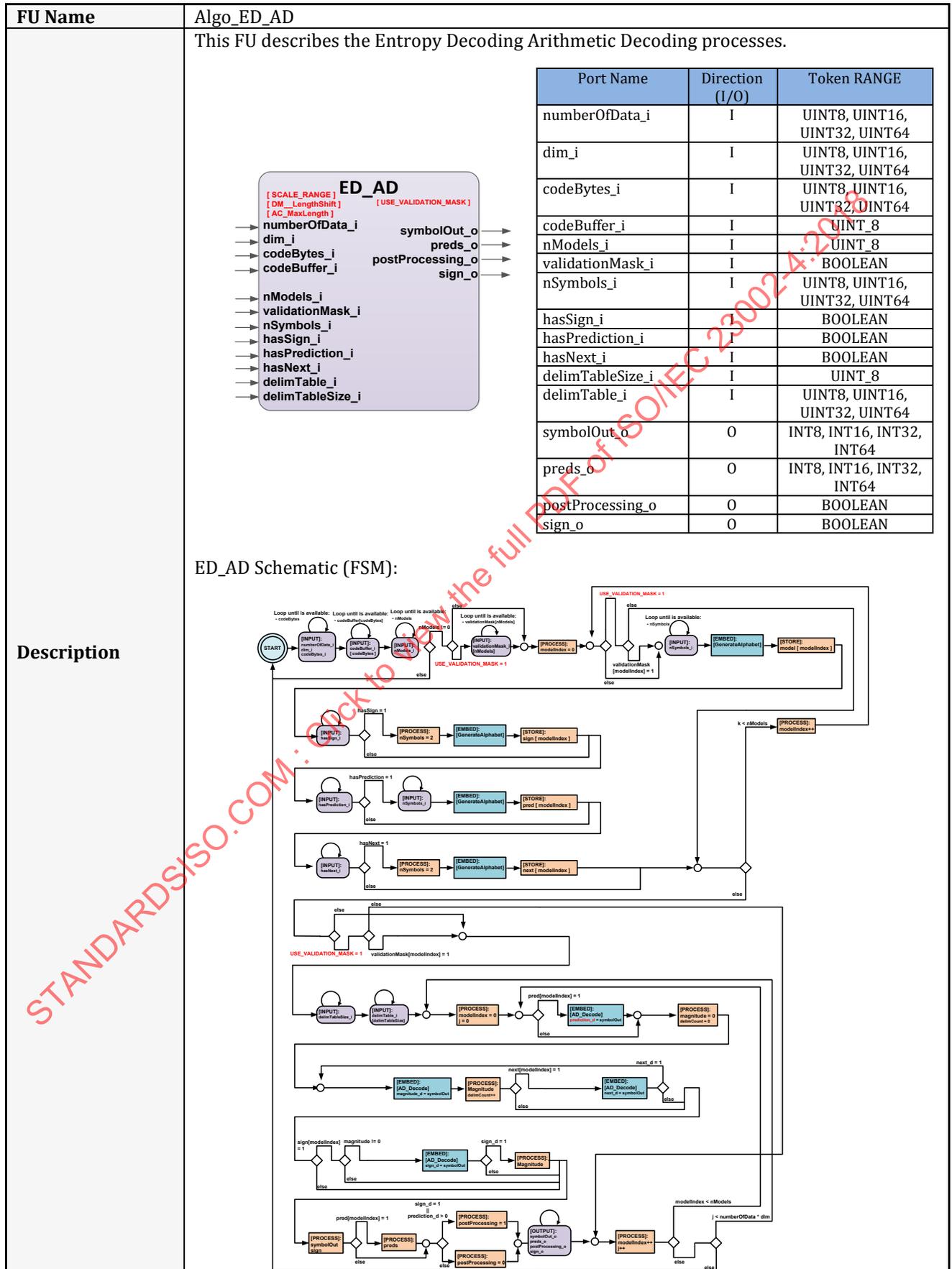
5.2.8 Algo\_ED\_BitPrecision

<b>FU Name</b>	Algo_ED_BitPrecision This FU describes the Entropy Decoding Bit Precision processes.																																								
<b>Description</b>		<table border="1"> <thead> <tr> <th>Port Name</th> <th>Direction (I/O)</th> <th>Token RANGE</th> </tr> </thead> <tbody> <tr> <td>dataIn_i</td> <td>I</td> <td>UINT_8</td> </tr> <tr> <td>numeorOfData_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>dim_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>prefixSize_i</td> <td>I</td> <td>UINT_8</td> </tr> <tr> <td>hasSign_i</td> <td>I</td> <td>BOOLEAN</td> </tr> <tr> <td>hasPrediction_i</td> <td>I</td> <td>BOOLEAN</td> </tr> <tr> <td>BPLTable_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>BPLTableSize_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>dataOut_o</td> <td>O</td> <td>INT8, INT16, INT32, INT64</td> </tr> <tr> <td>preds_o</td> <td>O</td> <td>INT8, INT16, INT32, INT64</td> </tr> <tr> <td>postProcessing_o</td> <td>O</td> <td>BOOLEAN</td> </tr> <tr> <td>sign_o</td> <td>O</td> <td>BOOLEAN</td> </tr> </tbody> </table>	Port Name	Direction (I/O)	Token RANGE	dataIn_i	I	UINT_8	numeorOfData_i	I	UINT8, UINT16, UINT32, UINT64	dim_i	I	UINT8, UINT16, UINT32, UINT64	prefixSize_i	I	UINT_8	hasSign_i	I	BOOLEAN	hasPrediction_i	I	BOOLEAN	BPLTable_i	I	UINT8, UINT16, UINT32, UINT64	BPLTableSize_i	I	UINT8, UINT16, UINT32, UINT64	dataOut_o	O	INT8, INT16, INT32, INT64	preds_o	O	INT8, INT16, INT32, INT64	postProcessing_o	O	BOOLEAN	sign_o	O	BOOLEAN
	Port Name	Direction (I/O)	Token RANGE																																						
dataIn_i	I	UINT_8																																							
numeorOfData_i	I	UINT8, UINT16, UINT32, UINT64																																							
dim_i	I	UINT8, UINT16, UINT32, UINT64																																							
prefixSize_i	I	UINT_8																																							
hasSign_i	I	BOOLEAN																																							
hasPrediction_i	I	BOOLEAN																																							
BPLTable_i	I	UINT8, UINT16, UINT32, UINT64																																							
BPLTableSize_i	I	UINT8, UINT16, UINT32, UINT64																																							
dataOut_o	O	INT8, INT16, INT32, INT64																																							
preds_o	O	INT8, INT16, INT32, INT64																																							
postProcessing_o	O	BOOLEAN																																							
sign_o	O	BOOLEAN																																							
	<p>ED_BitPrecision Schematic (FSM):</p> <p>ED_BitPrecision process: START</p>																																								

	<pre> INPUT: BPLTableSize_i INPUT: BPLTable_i [ BPLTableSize ] INPUT: numberOfData_i INPUT: dim_i IF STATIC_CONTEXT = 1     INPUT: hasSign_i     INPUT: hasPrediction_i IF STATIC_PREFIX_SIZE = 1     INPUT: prefixSize_i j = 0 PROCESS_CHUNK: WHILE j &lt; numberOfData * dim     IF STATIC_PREFIX_SIZE = 0         INPUT: prefixSize_i     IF STATIC_CONTEXT = 0         INPUT: hasSign_i         INPUT: hasPrediction_i     IF hasSign = 0 &amp; hasPrediction = 0         INPUT: dataIn { prefixSize }         nBPL = dataIn         IF nBPL &gt; 2             INPUT: dataIn { nBPL - 1 }             nPayload = dataIn         ELSE             nPayload = 0         difValue = BPLTable [ nBPL ] + nPayload         dataOut = difValue         postProcessing = 0         sign = 0         OUTPUT: dataOut         OUTPUT: postProcessing_o         OUTPUT: sign_o         ELSE IF hasSign = 1 &amp; hasPrediction = 0             INPUT: dataIn { prefixSize }             nBPL = dataIn             if nBPL &gt; 2                 INPUT: dataIn { nBPL - 1 }                 nPayload = dataIn             ELSE                 nPayload = 0             difValue = BPLTable [ nBPL ] + nPayload             IF difValue != 0                 INPUT: dataIn { 1 }                 nSign = dataIn             ELSE                 nSign = 1             sign = nSign             postProcessing = 1             dataOut = difValue             OUTPUT: dataOut             OUTPUT: postProcessing_o             OUTPUT: sign_o         ELSE IF hasSign = 1 &amp; hasPrediction = 1             signDef[2] = { 1,-1 }             predsOut [ j ] = 0             DO                 INPUT: dataIn { 2 }                 nPred = dataIn                 preds += nPred                 WHILE nPred != 3             INPUT: dataIn { prefixSize }             nBPL = dataIn             IF ( nBPL &gt; 2 )                 INPUT: dataIn { nBPL - 1 }                 nPayload = dataIn             ELSE                 nPayload = 0             difValue = BPLTable [ nBPL ] + nPayload             dataOut = difValue             IF preds != 0                 postProcessing = 1             ELSE                 postProcessing = 0 </pre>
--	---

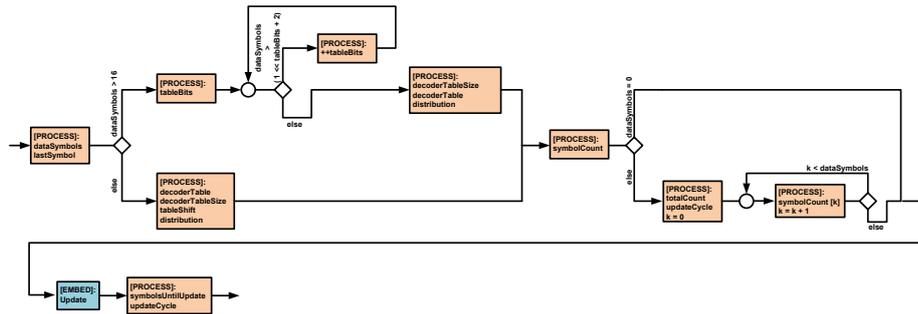
	<pre> IF postProcessing != 0     sign = signDef [ difValue % 2 ] ELSE     sign = 1 OUTPUT: dataOut_o OUTPUT: preds_o OUTPUT: postProcessing_o OUTPUT: sign_o ELSE IF hasSign = 1 &amp; hasPrediction = 0     IF STATIC_CONTEXT = 0         PROCESS_CHUNK     ELSE         GOTO START GOTO START                 </pre> <p>An input of size {N} has the meaning of a input of a N-size bit value.                  NOTE The case when both hasSign=1 and hasPrediction=0 is not considered. The behaviour in this case is to return to the START stage (if the parameter STATIC_CONTEXT=1) or to process the next data (if the parameter STATIC_CONTEXT=0).</p>	
<b>ISO Standards using the FU</b>	ISO/IEC 14496-16	
<b>Profiles@levels supported</b>	All (Generic)	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
dataIn_i	UINT_8	
numOfData_i	UINT8, UINT16, UINT32, UINT64	
dim_i	UINT8, UINT16, UINT32, UINT64	
prefixSize_i	UINT_8	
hasSign_i	BOOLEAN	
hasPrediction_i	BOOLEAN	
BPLTable_i	UINT8, UINT16, UINT32, UINT64	
BPLTableSize_i	UINT8, UINT16, UINT32, UINT64	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
dataOut_o	INT8, INT16, INT32, INT64	
preds_o	INT8, INT16, INT32, INT64	
postProcessing_o	BOOLEAN	
sign_o	BOOLEAN	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
STATIC_CONTEXT	Describes whether the context data (hasSign and hasPrediction) is read for every processing iteration. If set to 1, the context data is read only once and reused during the process, if set to 0, the context data is read for every processing iteration. This parameter is set at the network configuration level.	Type:Boolean Range: {0,1}
STATIC_PREFIX_SIZE	Describes whether the prefix size value is read for every processing iteration. If set to 1, the prefix value is read only once and reused during the process, if set to 0, the prefix value is read for every processing iteration. This parameter is set at the network configuration level.	Type:Boolean Range: {0,1}
<b>Package</b>		
package org.sc29.wg11.mpeg4.part16		

5.2.9 Algo\_ED\_AD

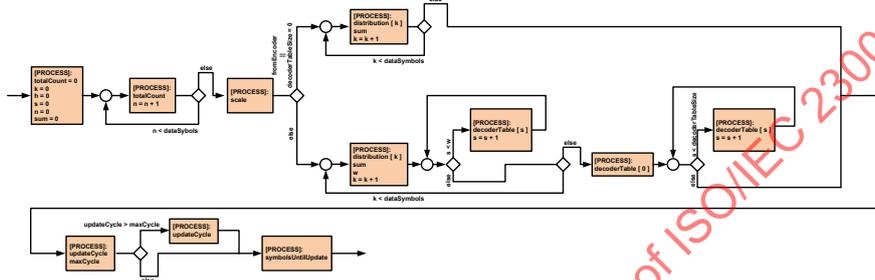


Description

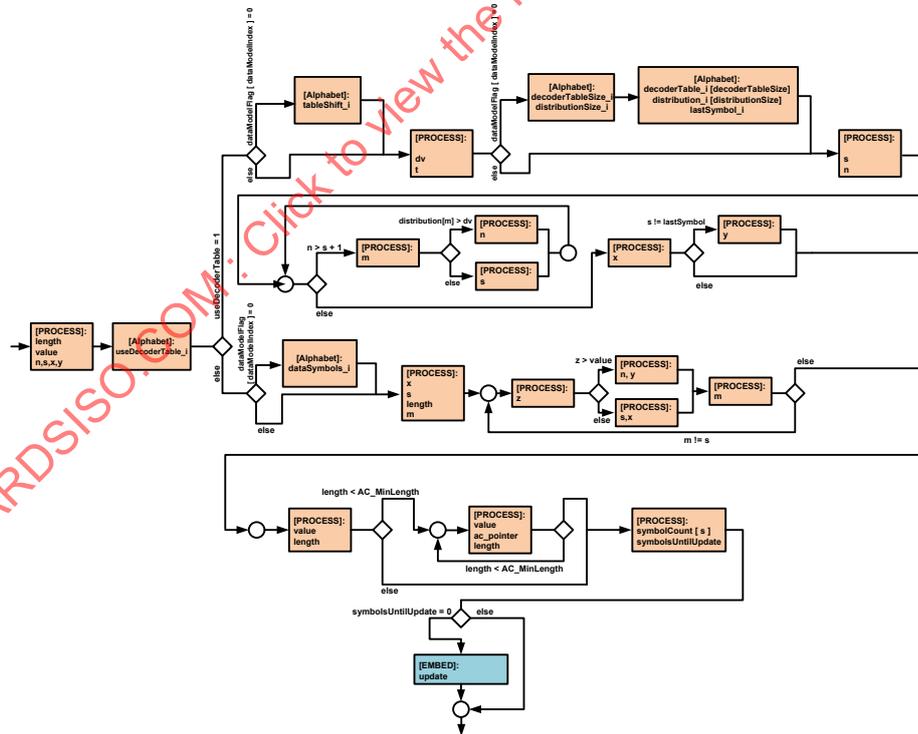
Generate Alphabet Process Schematic (FSM):



Update - Process Schematic (FSM):



Generate Alphabet Process:  
Arithmetic Decode Adaptive Data - Process Schematic (FSM):



ED\_AD process:  
 START  
 INPUT: numberOfData\_i  
 INPUT: dim\_i  
 INPUT: codeBytes\_i  
 INPUT: codeBuffer\_i[codeBytes]  
 INPUT: nModels\_i  
 if nModels != 0

```

if USE_VALIDATION_MASK = 1
  INPUT: validationMask_i [ nModels ]
modelIndex = 0
k = 0
DO
  if USE_VALIDATION_MASK = 1
    if validationMask [ modelIndex ] = 1
      GOTO MODEL_CONFIG
    else
      GOTO NEXT_MODEL_CONFIG
  else
    GOTO MODEL_CONFIG

MODEL_CONFIG:
  INPUT: nSymbols_i
  EMBED: Generate Alphabet
  STORE: model [ modelIndex ]
  INPUT: hasSign_i
  if hasSign = 1
    nSymbols = 2
    EMBED: Generate Alphabet
    STORE: sign [ modelIndex ]
  INPUT: hasPrediction_i
  if hasPrediction = 1
    INPUT: nSymbols_i
    EMBED: Generate Alphabet
    STORE: pred [ modelIndex ]
  INPUT: hasNext_i
    nSymbols = 2
    EMBED: Generate Alphabet
    STORE: next [ modelIndex ]
  k = k + 1
  WHILE k < nModels
INPUT: delimTableSize_i
INPUT: delimTable_i [ delimTableSize ]

INIT:
modelIndex = 0
j = 0
DECODE:
  if USE_VALIDATION_MASK = 1
    if validationMask [ modelIndex ] = 1
      GOTO MODEL_PROCESS
    else
      GOTO NEXT_MODEL
  else
    GOTO MODEL_PROCESS

MODEL_PROCESS:
  if pred[modelIndex] = 1
    EMBED: AD_Decode (prediction_d = symbolOut)
  magnitude = 0
  delimCount = 0
  HAS_NEXT:
    EMBED: AD_DECODE (magnitude_d = symbolOut)
    Magnitude
    delimCount++
    if next [ modelIndex ] = 1
      EMBED: AD_DECODE (next_d = symbolOut)
    if next_d = 1
      GOTO HAS_NEXT
    if sign[modelIndex] = 1
      if magnitude != 0
        EMBED: AD_DECODE (sign_d = symbolCount)
    symbolOut = magnitude
    sign = sign_d
    if pred[modelIndex] = 1
      preds = prediction_d
    if sign_d = 1 || prediction_d > 0
      postProcessing = 1
    else
      postProcessing = 0
  OUTPUT: symbolOut_o

```

```

OUTPUT: preds_o
OUTPUT: postProcessing_o
OUTPUT: sign_o

NEXT_MODEL:
    modelIndex++
    j++

    if modelIndex < nModels
        GOTO DECODE
    if j < numberOfData * dim
        GOTO INIT
GOTO START
    
```

**Generate Alphabet Process:**

```

dataSymbols = nSymbols
lastSymbol = nSymbols - 1
if dataSymbols > 16
    tableBits = 3
    WHILE dataSymbols > ( 1 << (tableBits + 2) )
        ++tableBits
    tableSize = 1 << tableBits
    tableShift = DM_LengthShift - tableBits
    distribution [ 2 * dataSymbols + tableSize + 2 ]
    decoderTable = distribution + 2 * dataSymbols
else
    decoderTable = 0
    tableSize = 0
    tableShift = 0
    distribution [ 2 * dataSymbols ]
symbolCount = distribution + dataSymbols
if dataSymbols != 0
    totalCount = 0
    updateCycle = dataSymbols
    k = 0
    WHILE k < dataSymbols
        symbolCount [ k ] = 1
        k = k + 1
    EMBED: Update
    symbolsUntilUpdate = ( dataSymbols + 6 ) >> 1
    updateCycle = ( dataSymbols + 6 ) >> 1
    
```

**Update - Process:**

```

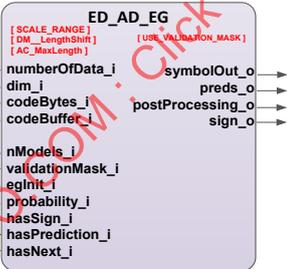
totalCount = 0
k = 0
h = 0
s = 0
n = 0
sum = 0
WHILE n < dataSymbols
    totalCount += ( symbolCount [ n ] = (symbolCount [ n ] + 1) >> 1 )
    scale = SCALE_RANGE / totalCount
    if fromEncofer = true | tableSize = 0
        WHILE k < dataSymbols
            distribution [ k ] = ( scale * sum ) >> ( 31 - DM_LengthShift )
            sum += symbolCount [ k ]
            k = k + 1
    else
        WHILE k < dataSymbols
            distribution [ k ] = ( scale * sum ) >> ( 31 - DM_LengthShift )
            sum += symbolCount [ k ]
            w = distribution [ k ] >> tableShift
            WHILE s < w
                decoderTable [ ++s ] = k - 1
            decoderTable [ 0 ] = 0
            WHILE s <= tableSize
                decoderTable [ ++s ] = dataSymbols - 1
        updateCycle = ( 5 * updateCycle ) >> 2
        maxCycle = ( dataSymbols + 6 ) << 3
        if ( updateCycle > maxCycle )
            updateCycle = maxCycle
        symbolsUntilUpdate = updateCycle
    
```

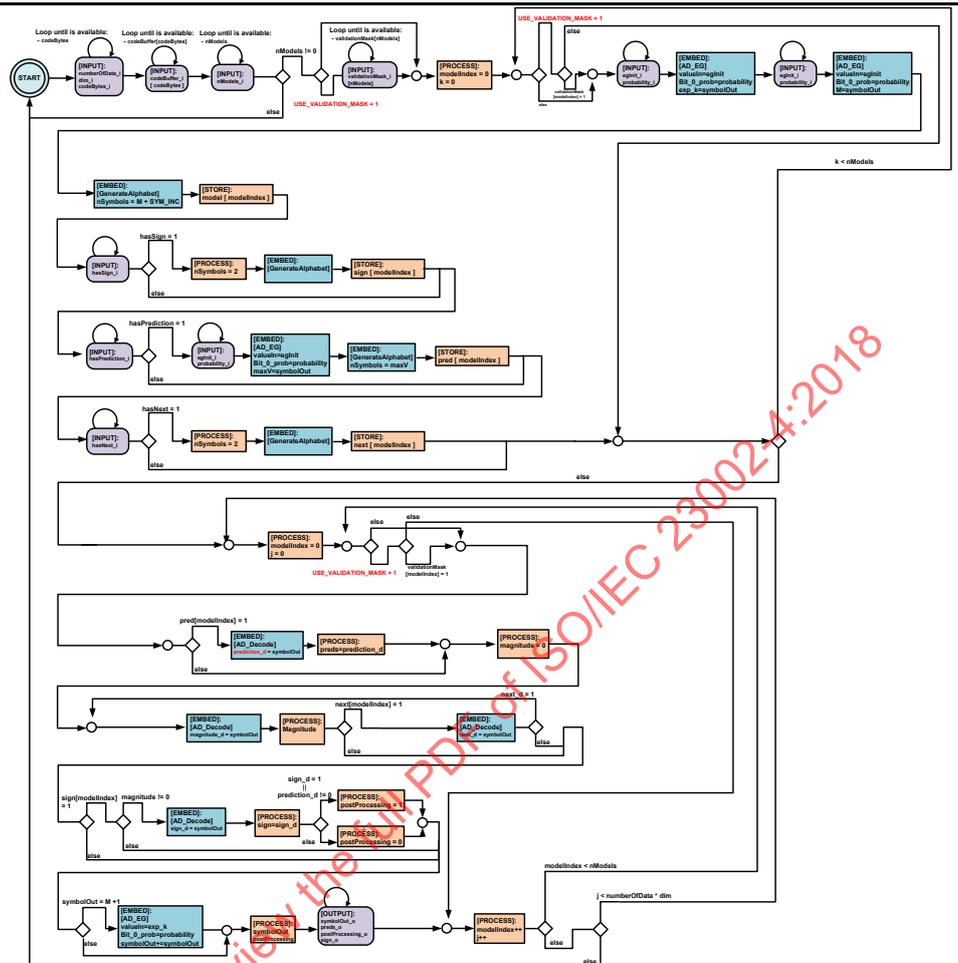
STANDARDS360.COM : Click to view the full PDF of ISO/IEC 23002-4:2018

	<p>Arithmetic Decode Adaptive Data Process:</p> <pre> codeBytes = 0 WHILE codeBytes &lt;= 0     INPUT: codeBytes length = 0 value = 0 n = 0 s = 0 x = 0 y = 0 if useDecoderTable = 1     if dataModelFlag [ dataModelIndex ] = 0         INPUT: tableShift         dv = value / ( length &gt;&gt;= DM_LengthShift )         t = dv &gt;&gt; tableShift         s = decoderTable [ t ]         n = decoderTable [ t+1 ] + 1         WHILE n &gt; s + 1             m = ( s + n ) &gt;&gt; 1             if distribution [ m ] &gt; dv                 n = m             else                 s = m         x = distribution [ s ] * length         if s != lastSymbol             y = distribution [ s + 1 ] * length     else         x = 0         s = 0         length &gt;&gt;= DM_LengthShift         m = ( n = dataSymbols ) &gt;&gt; 1         DO             z = length * distribution [ m ]             if z &gt; value                 n = m                 y = z             else                 s = m                 x = z             WHILE m = ((s+n) &gt;&gt; 1) != s         value -= x         length = y - x         if length &lt; AC_MinLength             DO                 value = (value &lt;&lt; 8)   ++ac_pointer                 WHILE (length &lt;&lt;= 8) &lt; AC_MinLength                     ++symbolCount [ s ]             if -- symbolsUntilUpdate = 0                 EMBED: Update                     </pre>
<p><b>ISO Standards using the FU</b></p>	<p>ISO/IEC 14496-16</p>
<p><b>Profiles@levels supported</b></p>	<p>All (Generic)</p>
<p><b>Input</b></p>	
<p><b>Name</b></p>	<p><b>Token</b></p>
<p>numberOfData_i</p>	<p>UINT8, UINT16, UINT32, UINT64</p>
<p>dim_i</p>	<p>UINT8, UINT16, UINT32, UINT64</p>
<p>codeBytes_i</p>	<p>UINT8, UINT16, UINT32, UINT64</p>
<p>codeBuffer_i</p>	<p>UINT_8</p>
<p>nModels_i</p>	<p>UINT_8</p>
<p>validationMask_i</p>	<p>BOOLEAN</p>
<p>nSymbols_i</p>	<p>UINT8, UINT16, UINT32, UINT64</p>
<p>hasSign_i</p>	<p>BOOLEAN</p>
<p>hasPrediction_i</p>	<p>BOOLEAN</p>
<p>hasNext_i</p>	<p>BOOLEAN</p>
<p>delimTableSize_i</p>	<p>UINT_8</p>

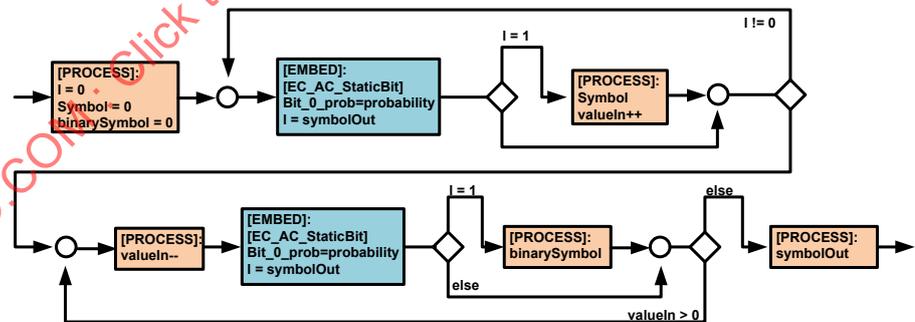
delimTable_i	UINT8, UINT16, UINT32, UINT64	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
symbolOut_o	INT8, INT16, INT32, INT64	
preds_o	INT8, INT16, INT32, INT64	
postProcessing_o	BOOLEAN	
sign_o	BOOLEAN	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Type / Range</b>
AC_MaxLength	Describes the maximum AC interval. This parameter is set at the network configuration level.	Type:Unsigned Integer Range: [1 .. 2 <sup>32</sup> ]
DM_LengthShift	Describes the number of bits discarded before multiplication. This parameter is set at the network configuration level.	Type:Unsigned Integer Range: [1 .. 2 <sup>32</sup> ]
SCALE_RANGE	Describes the range for the scale value. This parameter is set at the network configuration level.	Type:Unsigned Integer Range: [1 .. 2 <sup>32</sup> ]
USE_VALIDATION_MASK	Indicates whether to use the validation mask input port or not.(0 - NO, 1 - YES)	Type:Unsigned Integer Range: {0,1}
<b>Package</b>		
package org.sc29.wg11.mpeg4.part16		

5.2.10 Algo\_ED\_AD\_EG

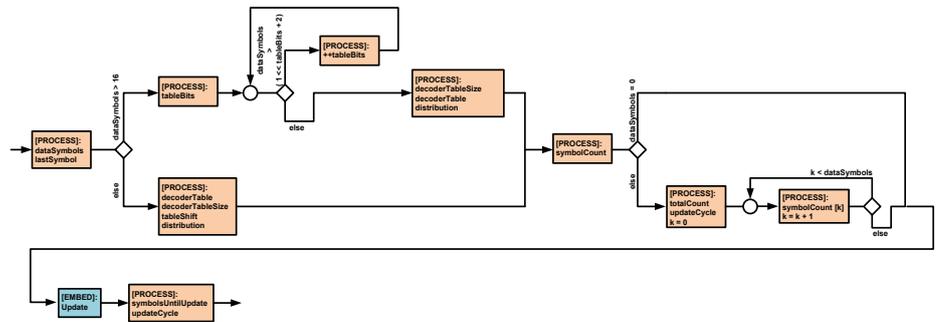
<b>FU Name</b>	Algo_ED_AD_EG																																																
<b>Description</b>	<p>This FU describes the Entropy Decoding Arithmetic Decoding Exponential Golomb processes.</p> 																																																
	<table border="1"> <thead> <tr> <th>Port Name</th> <th>Direction (I/O)</th> <th>Token RANGE</th> </tr> </thead> <tbody> <tr> <td>numberOfData_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>dim_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>codeBytes_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>codeBuffer_i</td> <td>I</td> <td>UINT_8</td> </tr> <tr> <td>nModels_i</td> <td>I</td> <td>UINT_8</td> </tr> <tr> <td>validationMask_i</td> <td>I</td> <td>BOOLEAN</td> </tr> <tr> <td>eglmit_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>probability_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>hasSign_i</td> <td>I</td> <td>BOOLEAN</td> </tr> <tr> <td>hasPrediction_i</td> <td>I</td> <td>BOOLEAN</td> </tr> <tr> <td>hasNext_i</td> <td>I</td> <td>BOOLEAN</td> </tr> <tr> <td>symbolOut_o</td> <td>O</td> <td>INT8, INT16, INT32, INT64</td> </tr> <tr> <td>preds_o</td> <td>O</td> <td>INT8, INT16, INT32, INT64</td> </tr> <tr> <td>postProcessing_o</td> <td>O</td> <td>BOOLEAN</td> </tr> <tr> <td>sign_o</td> <td>O</td> <td>BOOLEAN</td> </tr> </tbody> </table>		Port Name	Direction (I/O)	Token RANGE	numberOfData_i	I	UINT8, UINT16, UINT32, UINT64	dim_i	I	UINT8, UINT16, UINT32, UINT64	codeBytes_i	I	UINT8, UINT16, UINT32, UINT64	codeBuffer_i	I	UINT_8	nModels_i	I	UINT_8	validationMask_i	I	BOOLEAN	eglmit_i	I	UINT8, UINT16, UINT32, UINT64	probability_i	I	UINT8, UINT16, UINT32, UINT64	hasSign_i	I	BOOLEAN	hasPrediction_i	I	BOOLEAN	hasNext_i	I	BOOLEAN	symbolOut_o	O	INT8, INT16, INT32, INT64	preds_o	O	INT8, INT16, INT32, INT64	postProcessing_o	O	BOOLEAN	sign_o	O
Port Name	Direction (I/O)	Token RANGE																																															
numberOfData_i	I	UINT8, UINT16, UINT32, UINT64																																															
dim_i	I	UINT8, UINT16, UINT32, UINT64																																															
codeBytes_i	I	UINT8, UINT16, UINT32, UINT64																																															
codeBuffer_i	I	UINT_8																																															
nModels_i	I	UINT_8																																															
validationMask_i	I	BOOLEAN																																															
eglmit_i	I	UINT8, UINT16, UINT32, UINT64																																															
probability_i	I	UINT8, UINT16, UINT32, UINT64																																															
hasSign_i	I	BOOLEAN																																															
hasPrediction_i	I	BOOLEAN																																															
hasNext_i	I	BOOLEAN																																															
symbolOut_o	O	INT8, INT16, INT32, INT64																																															
preds_o	O	INT8, INT16, INT32, INT64																																															
postProcessing_o	O	BOOLEAN																																															
sign_o	O	BOOLEAN																																															
	ED_AD_EG Schematic (FSM):																																																



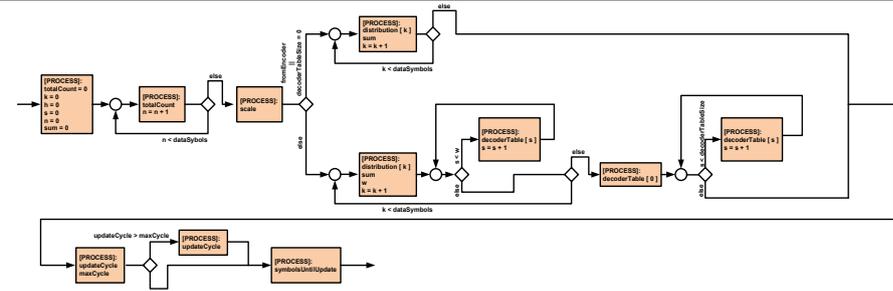
ED\_AD\_EG Process Schematic (FSM):



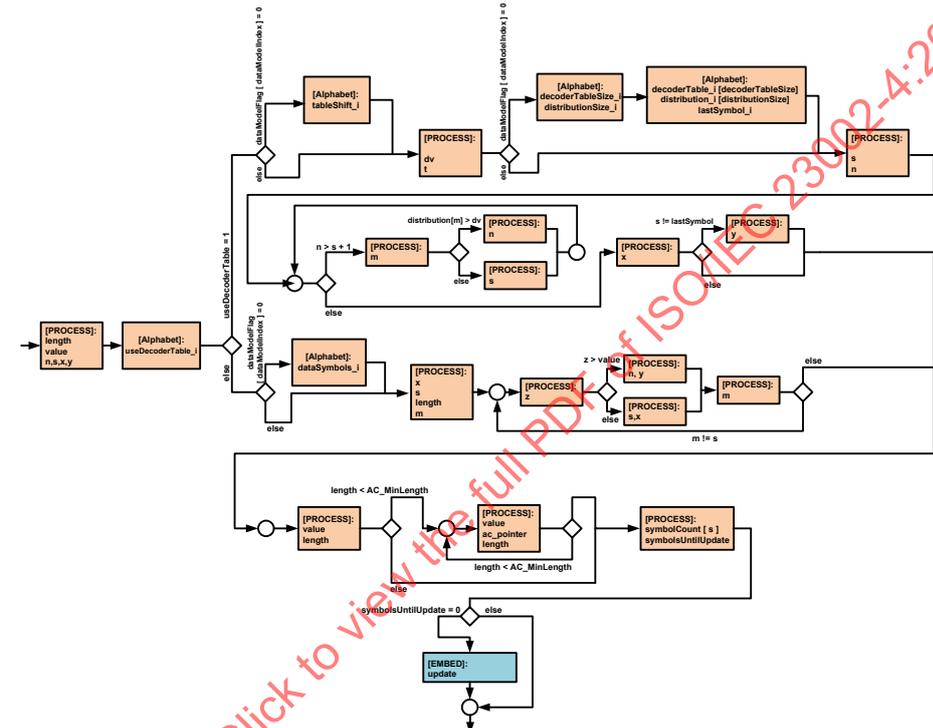
Generate Alphabet Process Schematic (FSM):



Update - Process Schematic (FSM):



Arithmetic Decode Adaptive Data - Process Schematic (FSM):



```

ED_AD_EG process:
START
INPUT: numberOfData_i
INPUT: dim_i
INPUT: codeBytes_i
INPUT: delimTableSize_i
INPUT: codeBuffer_i[codeBytes]
INPUT: delimTable_i[delimTableSize]
INPUT: nModels_i
if USE_VALIDATION_MASK = 1
    INPUT: validationMask[nModels]
if nModels != 0
    modelIndex = 0
    k = 0
    INPUT: egInit_i
    INPUT: probability_i
    EMBED: AD_EG (valueIn=egInit, Bit_0_prob=probability,exp_k=symbolOut)
    INPUT: egInit_i
    INPUT: probability_i
    EMBED: AD_EG (valueIn=egInit, Bit_0_prob=probability,exp_k=symbolOut)
DO
    if USE_VALIDATION_MASK = 1
        if validationMask [nModels] = 1
            EMBED: Generate Alphabet (nSymbols=M+SYM_INC)
            STORE: model [modelIndex]
            INPUT: hasSign_i
            if hasSign = 1
                nSymbols = 2
                EMBED: Generate Alphabet
                STORE: sign [modelIndex]
    
```

	<pre> INPUT: hasPrediction_i if hasPrediction = 1   INPUT: egInit_i   INPUT: probability_i   EMBED: AD_EG (valueIn=egInit, Bit_0_prob=probability,maxV=symbolOut)   EMBED: Generate Alphabet (nSymbols=maxV)   STORE: pred [ modelIndex ] INPUT: hasNext_i nSymbols = 2 EMBED: Generate Alphabet STORE: next [ modelIndex ] k = k + 1 WHILE k &lt; nModels  j = 0 DECODE: modelIndex = 0 NEXT_MODEL:   if USE_VALIDATION_MASK = 1     if validationMask [nModel] = 1       if pred[modelIndex] = 1         EMBED: AD_Decode (prediction_d = symbolOut)         magnitude = 0       HAS_NEXT:         EMBED: AD_DECODE (magnitude_d = symbolOut)         Magnitude         if next [ modelIndex ] = 1           EMBED: AD_DECODE (next_d = symbolOut)         if next_d = 1           GOTO HAS_NEXT         if sign[modelIndex] = 1           if magnitude != 0             EMBED: AD_DECODE (sign_d = symbolCount)           EMBED: AD_EG (valueIn=exp_k, bit_0_prob=probability, symbolOut+=symbolOut)           symbolOut = magnitude           sign = sign_d           if pred[modelIndex] = 1             preds = prediction_d           if sign_d = 1    prediction_d != 0             postProcessing = 1           else             postProcessing = 0           OUTPUT: symbolOut_o           OUTPUT: predst_o           OUTPUT: postProcessing_o           OUTPUT: sign_o           modelIndex++           if modelIndex &lt; nModels             GOTO NEXT_MODEL           j++           if j &lt; numberOfData * dim             GOTO DECODE  GOTO START  AD_EG Process: START l = 0 symbol = 0 binarySymbol = 0 DO   EMBED: EC_AC_StaticBit (bit_0_prob=probability,l = symbolOut)   if l = 1     symbol += (1&lt;&lt;valueIn)     valueIn++   WHILE l != 0   WHILE valueIn—     EMBED: EC_AC_StaticBit (bit_0_prob=probability,l = symbolOut)     if l = 1       binarySymbol  = (1&lt;&lt;valueIn)   symbolOut = symbol + binarySymbol  Generate Alphabet Process: dataSymbols = nSymbols </pre>
--	---

```

lastSymbol = nSymbols - 1
if dataSymbols > 16
    tableBits = 3
    WHILE dataSymbols > ( 1 << (tableBits + 2) )
        ++tableBits
    tableSize = 1 << tableBits
    tableShift = DM_LengthShift - tableBits
    distribution [ 2 * dataSymbols + tableSize + 2 ]
    decoderTable = distribution + 2 * dataSymbols
else
    decoderTable = 0
    tableSize = 0
    tableShift = 0
    distribution [ 2 * dataSymbols ]
symbolCount = distribution + dataSymbols
if dataSymbols != 0
    totalCount = 0
    updateCycle = dataSymbols
    k = 0
    WHILE k < dataSymbols
        symbolCount [ k ] = 1
        k = k + 1
    EMBED: Update
    symbolsUntilUpdate = ( dataSymbols + 6 ) >> 1
    updateCycle = ( dataSymbols + 6 ) >> 1

```

#### Update – Process:

```

totalCount = 0
k = 0
h = 0
s = 0
n = 0
sum = 0
WHILE n < dataSymbols
    totalCount += ( symbolCount [ n ] = (symbolCount [ n ] + 1) >> 1 )
scale = SCALE_RANGE / totalCount
if fromEncofer = true | tableSize = 0
    WHILE k < dataSymbols
        distribution [ k ] = ( scale * sum ) >> ( 31 - DM_LengthShift )
        sum += symbolCount [ k ]
        k = k + 1
else
    WHILE k < dataSymbols
        distribution [ k ] = ( scale * sum ) >> ( 31 - DM_LengthShift )
        sum += symbolCount [ k ]
        w = distribution [ k ] >> tableShift
        WHILE s < w
            decoderTable [++s] = k - 1
        decoderTable [ 0 ] = 0
        WHILE s <= tableSize
            decoderTable [++s] = dataSymbols - 1
        updateCycle = ( 5 * updateCycle ) >> 2
        maxCycle = ( dataSymbols + 6 ) << 3
        if ( updateCycle > maxCycle )
            updateCycle = maxCycle
        symbolsUntilUpdate = updateCycle

```

#### Arithmetic Decode Adaptive Data Process:

```

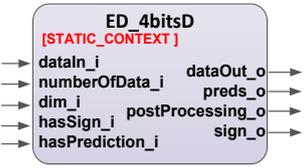
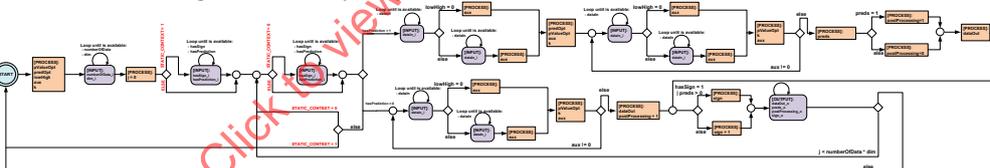
codeBytes = 0
WHILE codeBytes <= 0
    INPUT: codeBytes
length = 0
value = 0
n = 0
s = 0
x = 0
y = 0
if useDecoderTable = 1
    if dataModelFlag [ dataModelIndex ] = 0
        INPUT: tableShift
        dv = value / ( length >>= DM_LengthShift )
        t = dv >> tableShift

```

	<pre> s = decoderTable [ t ] n = decoderTable [ t+1 ] + 1 WHILE n &gt; s + 1     m = ( s + n ) &gt;&gt; 1     if distribution [ m ] &gt; dv         n = m     else         s = m x = distribution [ s ] * length if s != lastSymbol     y = distribution [ s + 1 ] * length else     x = 0     s = 0 length &gt;&gt;= DM_LengthShift m = ( n = dataSymbols ) &gt;&gt; 1 DO     z = length * distribution [ m ]     if z &gt; value         n = m         y = z     else         s = m         x = z     WHILE m = ((s+n) &gt;&gt; 1) != s value -= x length = y - x if length &lt; AC_MinLength DO     value = (value &lt;&lt; 8)   ++ac_pointer     WHILE (length &lt;= 8) &lt; AC_MinLength ++symbolCount [ s ] if -- symbolsUntilUpdate = 0     EMBED: Update         </pre>	
<b>ISO Standards using the FU</b>	ISO/IEC 14496-16	
<b>Profiles@levels supported</b>	All (Generic)	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
numberOfData_i	UINT8, UINT16, UINT32, UINT64	
dim_i	UINT8, UINT16, UINT32, UINT64	
codeBytes_i	UINT8, UINT16, UINT32, UINT64	
codeBuffer_i	UINT_8	
nModels_i	UINT_8	
validationMask_i	BOOLEAN	
eglnit_i	UINT8, UINT16, UINT32, UINT64	
probability_i	UINT8, UINT16, UINT32, UINT64	
hasSign_i	BOOLEAN	
hasPrediction_i	BOOLEAN	
hasNext_i	BOOLEAN	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
symbolOut_o	INT8, INT16, INT32, INT64	
preds_o	INT8, INT16, INT32, INT64	
postProcessing_o	BOOLEAN	
sign_o	BOOLEAN	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Type / Range</b>
AC_MaxLength	Describes the maximum AC interval. This parameter is set at the network configuration level.	Type:Unsigned Integer Range: [1 .. 2 <sup>32</sup> ]

DM_LengthShift	Describes the number of bits discarded before multiplication. This parameter is set at the network configuration level.	Type:Unsigned Integer Range: [1 .. 2 <sup>32</sup> ]
SCALE_RANGE	Describes the range for the scale value. This parameter is set at the network configuration level.	Type:Unsigned Integer Range: [1 .. 2 <sup>32</sup> ]
USE_VALIDATION_MASK	Indicates whether to use the validation mask input port or not.(0 – NO, 1 – YES)	Type:Unsigned Integer Range: {0,1}
<b>Package</b>		
package org.sc29.wg11.mpeg4.part16		

5.2.11 Algo\_ED\_4bitsD

<b>FU Name</b>	Algo_ED_4bitsD																														
<b>Description</b>	<p>This FU describes the Entropy Decoding - 4 bits Decoding processes.</p> <div style="display: flex; justify-content: space-around; align-items: flex-start;"> <div style="text-align: center;">  <p><b>ED_4bitsD</b> [STATIC_CONTEXT]</p> </div> <table border="1" data-bbox="805 761 1300 1108"> <thead> <tr> <th>Port Name</th> <th>Direction (I/O)</th> <th>Token RANGE</th> </tr> </thead> <tbody> <tr> <td>dataIn_i</td> <td>I</td> <td>UINT_8</td> </tr> <tr> <td>numberOfData_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>dim_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>hasSign_i</td> <td>I</td> <td>BOOLEAN</td> </tr> <tr> <td>hasPrediction_i</td> <td>I</td> <td>BOOLEAN</td> </tr> <tr> <td>dataOut_o</td> <td>O</td> <td>INT8, INT16, INT32, INT64</td> </tr> <tr> <td>preds_o</td> <td>O</td> <td>INT8, INT16, INT32, INT64</td> </tr> <tr> <td>postProcessing_o</td> <td>O</td> <td>BOOLEAN</td> </tr> <tr> <td>sign_o</td> <td>O</td> <td>BOOLEAN</td> </tr> </tbody> </table> </div> <p>ED 4 bits Decoding Schematic (FSM):</p>  <pre> ED 4 bits Decoding process: START signDef = { -1, 1 } pValueOpt = 0 predOpt = 0 lowHigh = 0 aux = 0 k = 0 INPUT:   numberOfData_i   dim_i IF STATIC_CONTEXT = 1   INPUT:     hasSign_i     hasPrediction_i j = 0 DO   IF STATIC_CONTEXT = 1     INPUT:       hasSign_i       hasPrediction_i   IF hasPrediction = 1     INPUT:       dataIn_i     IF lowHigh = 0       aux = dataIn &amp; 15     ELSE       INPUT:     </pre>	Port Name	Direction (I/O)	Token RANGE	dataIn_i	I	UINT_8	numberOfData_i	I	UINT8, UINT16, UINT32, UINT64	dim_i	I	UINT8, UINT16, UINT32, UINT64	hasSign_i	I	BOOLEAN	hasPrediction_i	I	BOOLEAN	dataOut_o	O	INT8, INT16, INT32, INT64	preds_o	O	INT8, INT16, INT32, INT64	postProcessing_o	O	BOOLEAN	sign_o	O	BOOLEAN
Port Name	Direction (I/O)	Token RANGE																													
dataIn_i	I	UINT_8																													
numberOfData_i	I	UINT8, UINT16, UINT32, UINT64																													
dim_i	I	UINT8, UINT16, UINT32, UINT64																													
hasSign_i	I	BOOLEAN																													
hasPrediction_i	I	BOOLEAN																													
dataOut_o	O	INT8, INT16, INT32, INT64																													
preds_o	O	INT8, INT16, INT32, INT64																													
postProcessing_o	O	BOOLEAN																													
sign_o	O	BOOLEAN																													

```

        dataIn_i
        aux = dataIn & 240
        lowHigh = (lowHigh+1)%2
        predOpt = aux & 7
        pValueOpt = 0
        aux >>= 3
        pValueOpt += aux
        WHILE aux
            INPUT:
                dataIn_i
            IF lowHigh = 0
                aux = dataIn & 15
            ELSE
                INPUT:
                    dataIn_i
                    aux = dataIn & 240
                    lowHigh = (lowHigh+1)%2
                    pValueOpt += (aux&7) << k
                    k += 3
                    aux >>= 3
                    pValueOpt += aux << k
        IF preds > 0
            postProcessing = 1
        ELSE
            postProcessing = 0
        preds = predOpt
        IF hasSign = 1 | preds > 0
            sign = signDef [ pValueOpt % 2 ]
        ELSE
            sign = 1
        dataOut = pValueOpt
        OUTPUT: dataOut_o
        OUTPUT: preds_o
        OUTPUT: postProcessing_o
        OUTPUT: sign_o

    ELSE IF hasPrediction = 0
        k = 0
        pValueOpt = 0
        DO
            INPUT:
                dataIn_i
            IF lowHigh = 0
                aux = dataIn & 15
            ELSE
                INPUT:
                    dataIn_i
                    aux = dataIn & 240
                    lowHigh = (lowHigh+1)%2
                    pValueOpt += (aux&7) << k
                    k += 3
                    aux >>= 3
                    pValueOpt += aux << k
            WHILE aux
                dataOut = pValueOpt
            IF hasSign = 1
                sign = signDef [ pValueOpt % 2 ]
            ELSE
                sign = 1
            postProcessing = 1
            OUTPUT: dataOut_o
            OUTPUT: postProcessing_o
            OUTPUT: sign_o
        j++
    ELSE
        IF STATIC_CONTEXT = 1
            GOTO START
        WHILE j < numberOfData * dim
            GOTO START

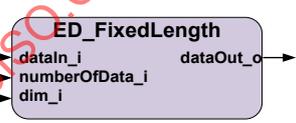
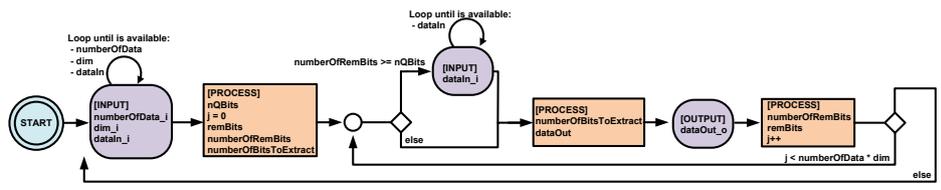
```

## NOTE

The following cases are not considered:  
 — hasSign=0, hasPrediction=0;

	— hasSign=0, hasPrediction=1. The behaviour in this case is to return to the START stage (if the parameter STATIC_CONTEXT=1) or to process the next data (if the parameter STATIC_CONTEXT=0).	
<b>ISO Standards using the FU</b>	ISO/IEC 14496-16	
<b>Profiles@levels supported</b>	All (Generic)	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
dataIn_i	UINT_8	
numOfData_i	UINT8, UINT16, UINT32, UINT64	
dim_i	UINT8, UINT16, UINT32, UINT64	
hasSign_i	BOOLEAN	
hasPrediction_i	BOOLEAN	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
dataOut_o	INT8, INT16, INT32, INT64	
preds_o	INT8, INT16, INT32, INT64	
postProcessing_o	BOOLEAN	
sign_o	BOOLEAN	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Type / Range</b>
STATIC_CONTEXT	Describes whether the context data (hasSign and hasPrediction) is read for every processing iteration. If set to 1, the context data is read only once and reused during the process, if set to 0, the context data is read for every processing iteration. This parameter is set at the network configuration level.	Type:Boolean Range: {0,1}
<b>Package</b>		
package org.sc29.wg11.mpeg4.part16		

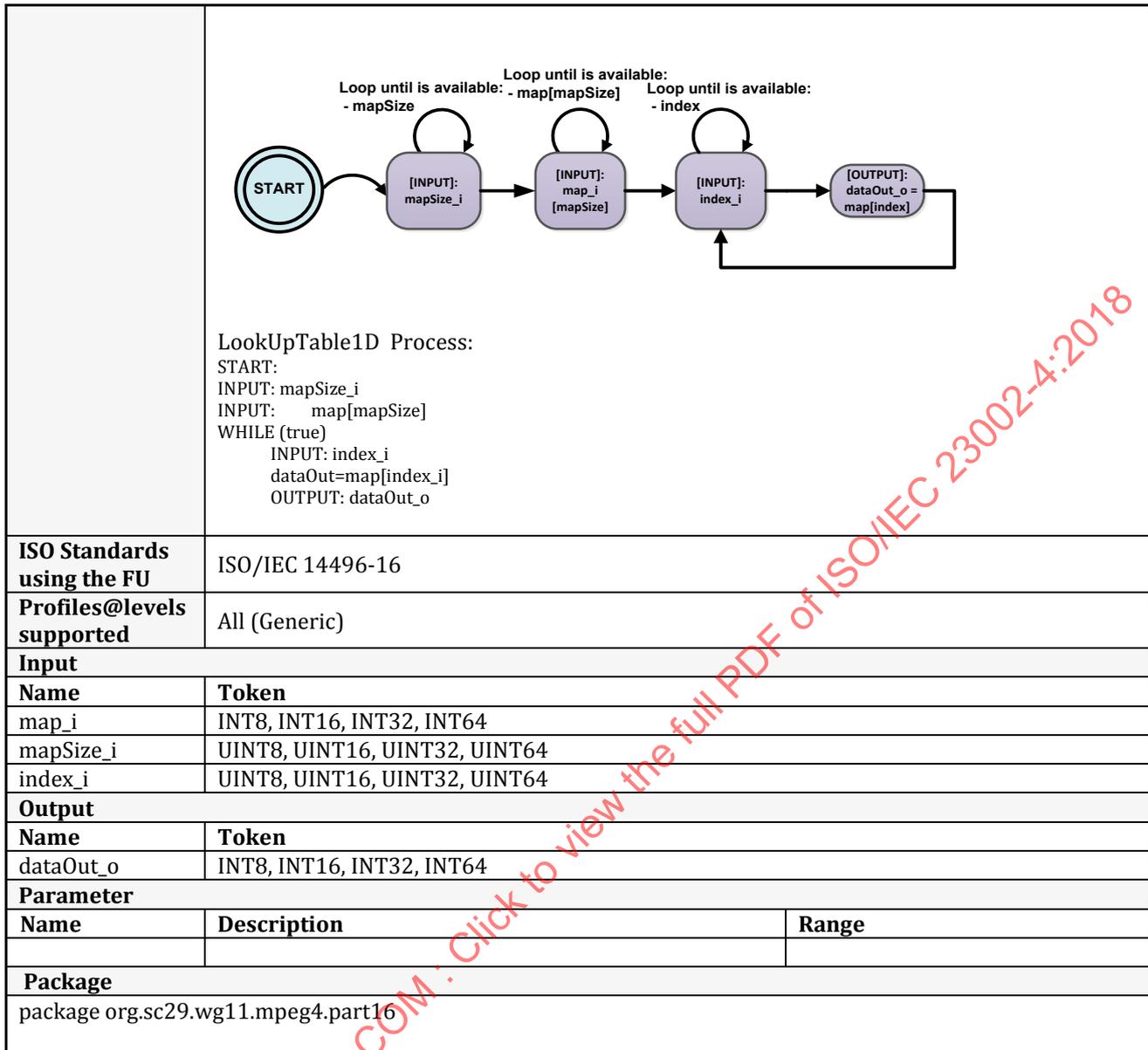
5.2.12 Algo\_ED\_FixedLength

<b>FU Name</b>	Algo_ED_FixedLength This FU describes the Entropy Decoding Fixed Length processes.															
<b>Description</b>	<div style="display: flex; justify-content: space-around; align-items: flex-start;"> <div style="text-align: center;">  </div> <table border="1" style="border-collapse: collapse; text-align: center;"> <thead> <tr style="background-color: #4a86e8; color: white;"> <th>Port Name</th> <th>Direction (I/O)</th> <th>Token RANGE</th> </tr> </thead> <tbody> <tr> <td>dataIn_i</td> <td>I</td> <td>UINT_8</td> </tr> <tr> <td>numOfData_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>dim_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>dataOut_o</td> <td>O</td> <td>INT8, INT16, INT32, INT64</td> </tr> </tbody> </table> </div> <p>ED Fixed Length Decoding Schematic (FSM):</p>  <p>ED Fixedlength Decoding process:          START          INPUT:          numberOfData_i          dim_i</p>	Port Name	Direction (I/O)	Token RANGE	dataIn_i	I	UINT_8	numOfData_i	I	UINT8, UINT16, UINT32, UINT64	dim_i	I	UINT8, UINT16, UINT32, UINT64	dataOut_o	O	INT8, INT16, INT32, INT64
Port Name	Direction (I/O)	Token RANGE														
dataIn_i	I	UINT_8														
numOfData_i	I	UINT8, UINT16, UINT32, UINT64														
dim_i	I	UINT8, UINT16, UINT32, UINT64														
dataOut_o	O	INT8, INT16, INT32, INT64														

	<pre> predMode_i dataIn_i nQBits = dataIn j = 0 remBits = 0 numberOfRemBits = 0 numberOfBitsToExtract = 0 DO   If numberOfRemBits &gt;= nQBits     INPUT:       dataIn_i       numberOfBitsToExtract = nQBits - numberOfRemBits       dataOut = remBits   ( dataIn[numberOfBitsToExtract] )       numberOfRemBits = nQBits - numberOfBitsToExtract       remBits = numberOfRemBits from dataIn       j++     WHILE j &lt; numberOfData * dim   GOTO START </pre> <p>NOTE  remBits are the remaining bits from the last iterative operation.  numberOfRemBits is the number of remaining bits from the last iterative operation.  numberOfBitsToExtract is the number of necessary bits to complete a number of nQBits.</p>	
<b>ISO Standards using the FU</b>	ISO/IEC 14496-16	
<b>Profiles@levels supported</b>	All (Generic)	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
dataIn_i	UINT_8	
numerOfData_i	UINT8, UINT16, UINT32, UINT64	
dim_i	UINT8, UINT16, UINT32, UINT64	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
dataOut_o	INT8, INT16, INT32, INT64	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>

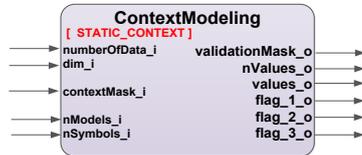
5.2.13 Algo\_LookUpTable1D

<b>FU Name</b>	Algo_LookUpTable1D															
<b>Description</b>	The FU implements the Look-Up-Table concept by using an 1-dimensional array as a table.															
	<pre> graph LR     map_i --&gt; LookUpTable1D     mapSize_i --&gt; LookUpTable1D     index_i --&gt; LookUpTable1D     LookUpTable1D --&gt; dataOut_o </pre>	<table border="1"> <thead> <tr> <th>Port Name</th> <th>Direction (I/O)</th> <th>Token RANGE</th> </tr> </thead> <tbody> <tr> <td>map_i</td> <td>I</td> <td>INT8, INT16, INT32, INT64</td> </tr> <tr> <td>mapSize_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>index_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>dataOut_o</td> <td>O</td> <td>INT8, INT16, INT32, INT64</td> </tr> </tbody> </table>	Port Name	Direction (I/O)	Token RANGE	map_i	I	INT8, INT16, INT32, INT64	mapSize_i	I	UINT8, UINT16, UINT32, UINT64	index_i	I	UINT8, UINT16, UINT32, UINT64	dataOut_o	O
Port Name	Direction (I/O)	Token RANGE														
map_i	I	INT8, INT16, INT32, INT64														
mapSize_i	I	UINT8, UINT16, UINT32, UINT64														
index_i	I	UINT8, UINT16, UINT32, UINT64														
dataOut_o	O	INT8, INT16, INT32, INT64														
	LookUpTable1D Process Schematic (FSM):															



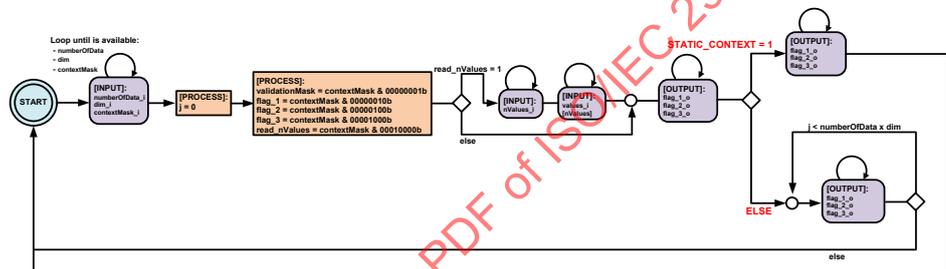
5.2.14 Algo\_ContextModeling

<b>FU Name</b>	Algo_ContextModeling
<b>Description</b>	This FU describes context generation to be used in the entropy decoding units. It generates codec specific data sets based on the input values.



Port Name	Direction (I/O)	Token RANGE
numberOfData_i	I	UINT8, UINT16, UINT32, UINT64
dim_i	I	UINT8, UINT16, UINT32, UINT64
contextMask_i	I	UINT_8
nValues_i	I	UINT8, UINT16, UINT32, UINT64
values_i	I	INT8, INT16, INT32, INT64
validationMask_o	O	BOOLEAN
nValues_o	O	UINT8, UINT16, UINT32, UINT64
values_o	O	INT8, INT16, INT32, INT64
flag_1_o	O	BOOLEAN
flag_2_o	O	BOOLEAN
flag_3_o	O	BOOLEAN

ContextModeling Schematic (FSM):



ContextModeling process:

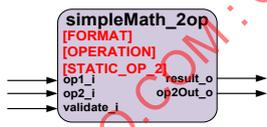
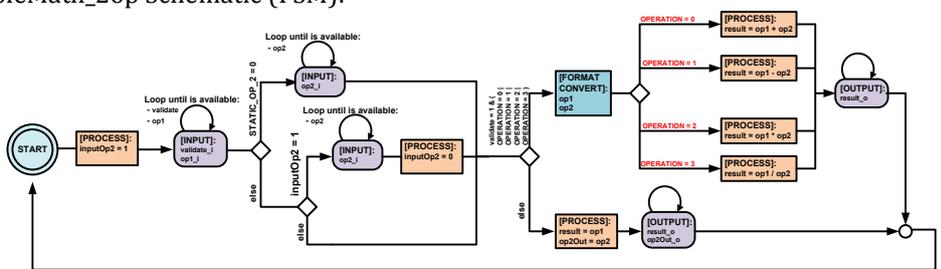
```

START
INPUT: numberOfData_i
INPUT: dim_i
INPUT: ccontextMask_i
j = 0
validationMask = contextMask & 00000001b
flag_1 = contextMask & 00000010b
flag_2 = contextMask & 00000100b
flag_3 = contextMask & 00001000b
read_nValues = contextMask & 00010000b
if read_nValues = 1
    INPUT: nValues_i
    INPUT: values_i [nValues]
    OUTPUT: validationMask_o
    OUTPUT: nValues_o
    OUTPUT: values_o [nValues]
    outputCounter = 0
    IF STATIC_CONTEXT = 1
        DO
            OUTPUT: flag_1_o
            OUTPUT: flag_2_o
            OUTPUT: flag_3_o
            outputCounter ++
            WHILE outputCounter <= nValues
        ELSE
            DO
                outputCounter = 0
                DO
                    OUTPUT: flag_1_o
                    OUTPUT: flag_2_o
                    OUTPUT: flag_3_o
                    WHILE outputCounter <= nValues
                j = j + 1
                WHILE j < numberOfData * dim
            GOTO START
    
```

Bit position	7	6	5	4	3	2	1	0
	X	X	X	bit	bit	bit	bit	bit
Flag	X	X	X	Read N Values	flag_3	flag_3	flag_1	Validation Mask

<b>ISO Standards using the FU</b>	ISO/IEC 14496-16	
<b>Profiles@levels supported</b>	All (Generic)	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
numberOfData_i	UINT8, UINT16, UINT32, UINT64	
dim_i	UINT8, UINT16, UINT32, UINT64	
contextMask_i	UINT_8	
nValues_i	UINT8, UINT16, UINT32, UINT64	
values_i	INT8, INT16, INT32, INT64	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
validationMask_o	BOOLEAN	
nValues_o	UINT8, UINT16, UINT32, UINT64	
values_o	INT8, INT16, INT32, INT64	
flag_1_o	BOOLEAN	
flag_2_o	BOOLEAN	
flag_3_o	BOOLEAN	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part16		

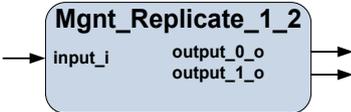
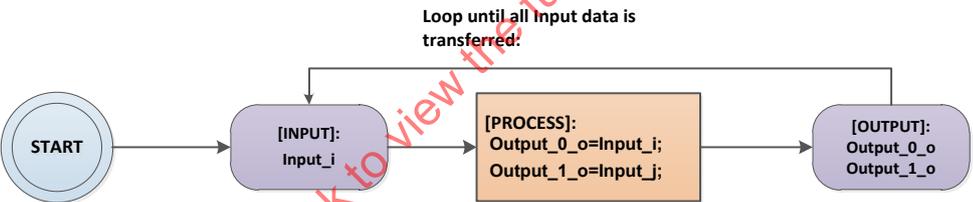
5.2.15 Algo\_simpleMath\_2op

<b>FU Name</b>	Algo_simpleMath_2op																		
<b>Description</b>	<p>This FU describes the simple mathematical operation on 2 operands.</p> <div style="display: flex; align-items: center;">  <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>Port Name</th> <th>Direction (I/O)</th> <th>Token RANGE</th> </tr> </thead> <tbody> <tr> <td>op1_i</td> <td>I</td> <td>INT8, INT16, INT32, INT64, UINT8, UINT16, UINT32, UINT64, FLOAT</td> </tr> <tr> <td>op2_i</td> <td>I</td> <td>INT8, INT16, INT32, INT64, UINT8, UINT16, UINT32, UINT64, FLOAT</td> </tr> <tr> <td>validate_i</td> <td>I</td> <td>BOOLEAN</td> </tr> <tr> <td>result_o</td> <td>O</td> <td>INT8, INT16, INT32, INT64, UINT8, UINT16, UINT32, UINT64, FLOAT</td> </tr> <tr> <td>op2Out_o</td> <td>O</td> <td>INT8, INT16, INT32, INT64, UINT8, UINT16, UINT32, UINT64, FLOAT</td> </tr> </tbody> </table> </div> <p>simpleMath_2op Schematic (FSM):</p>  <p>simpleMath_2op process:</p> <pre> START inputOp2 = 1 INPUT: validate_i     </pre>	Port Name	Direction (I/O)	Token RANGE	op1_i	I	INT8, INT16, INT32, INT64, UINT8, UINT16, UINT32, UINT64, FLOAT	op2_i	I	INT8, INT16, INT32, INT64, UINT8, UINT16, UINT32, UINT64, FLOAT	validate_i	I	BOOLEAN	result_o	O	INT8, INT16, INT32, INT64, UINT8, UINT16, UINT32, UINT64, FLOAT	op2Out_o	O	INT8, INT16, INT32, INT64, UINT8, UINT16, UINT32, UINT64, FLOAT
Port Name	Direction (I/O)	Token RANGE																	
op1_i	I	INT8, INT16, INT32, INT64, UINT8, UINT16, UINT32, UINT64, FLOAT																	
op2_i	I	INT8, INT16, INT32, INT64, UINT8, UINT16, UINT32, UINT64, FLOAT																	
validate_i	I	BOOLEAN																	
result_o	O	INT8, INT16, INT32, INT64, UINT8, UINT16, UINT32, UINT64, FLOAT																	
op2Out_o	O	INT8, INT16, INT32, INT64, UINT8, UINT16, UINT32, UINT64, FLOAT																	

	<pre> op1_i IF STATIC_OP_2 = 0   INPUT: op2_i ELSE   IF inputOp2 = 1     INPUT: op2_i     inputOp2 = 0   IF validate = 1 &amp; ( OPERATION = 0   OPERATION = 1   OPERATION = 2   OPERATION = 3 )     FORMAT CONVERT: op1     FORMAT CONVERT: op2     SWITCH ( OPERATION )       CASE 0:         result = op1 + op2         break       CASE 1:         result = op1 - op2         break       CASE 2:         result = op1 * op2         break       CASE 3:         result = op1 / op2         break     OUTPUT: result_o   ELSE     result = op1     op2Out = op2     OUTPUT: result_o     OUTPUT: op2Out_o   GOTO START </pre> <p>The following table contains the interpretation of the FORMAT codes:</p> <table border="1"> <thead> <tr> <th>FORMAT</th> <th>Interpretation</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Unsigned Integer</td> </tr> <tr> <td>1</td> <td>Signed Integer</td> </tr> <tr> <td>2</td> <td>Fixed Point</td> </tr> <tr> <td>3</td> <td>Floating Point (IEEE 754)</td> </tr> </tbody> </table> <p>The following table contains the supported operations:</p> <table border="1"> <thead> <tr> <th>Index</th> <th>Operation</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>Addition ( + )</td> </tr> <tr> <td>1</td> <td>Subtraction ( - )</td> </tr> <tr> <td>2</td> <td>Multiplication ( * )</td> </tr> <tr> <td>3</td> <td>Division ( / )</td> </tr> </tbody> </table> <p>NOTE If the validate_i flag is true (value 1), one of the operations above are applied to the 2 operands. Else, the operands are repeated at the output ports (result_o = op1 and op2Out_o = op2).</p>		FORMAT	Interpretation	0	Unsigned Integer	1	Signed Integer	2	Fixed Point	3	Floating Point (IEEE 754)	Index	Operation	0	Addition ( + )	1	Subtraction ( - )	2	Multiplication ( * )	3	Division ( / )
FORMAT	Interpretation																					
0	Unsigned Integer																					
1	Signed Integer																					
2	Fixed Point																					
3	Floating Point (IEEE 754)																					
Index	Operation																					
0	Addition ( + )																					
1	Subtraction ( - )																					
2	Multiplication ( * )																					
3	Division ( / )																					
<b>ISO Standards using the FU</b>	ISO/IEC 14496-16																					
<b>Profiles@levels supported</b>	All (Generic)																					
<b>Input</b>																						
<b>Name</b>	<b>Token</b>																					
op1_i	INT8, INT16, INT32, INT64, UINT8, UINT16, UINT32, UINT64, FLOAT																					
op2_i	INT8, INT16, INT32, INT64, UINT8, UINT16, UINT32, UINT64, FLOAT																					
validate_i	BOOLEAN																					
<b>Output</b>																						
<b>Name</b>	<b>Token</b>																					
result_o	INT8, INT16, INT32, INT64, UINT8, UINT16, UINT32, UINT64, FLOAT																					
op2Out_o	INT8, INT16, INT32, INT64, UINT8, UINT16, UINT32, UINT64, FLOAT																					
<b>Parameter</b>																						
<b>Name</b>	<b>Description</b>	<b>Range</b>																				
FORMAT	Describes the input operands number format representation. This parameter is set at the network configuration level.	Type:Unsigned Integer Range: {0,1,2,3}																				

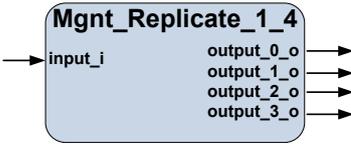
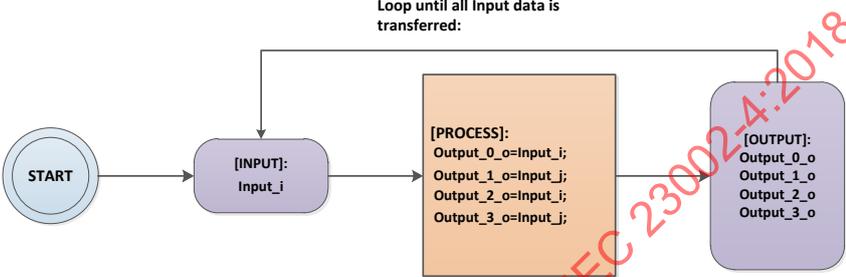
<b>OPERATION</b>	Describes the index of the operation to be applied. This parameter is set at the network configuration level.	Type:Unsigned Integer Range: {0,1,2,3}
<b>STATIC_OP_2</b>	If it is set to 0, the FU expects that the input op2 will be read for each computation. If it is set to 1, the op 2 will be read just once and then used for any of the following operations. This parameter is set at the network configuration level.	Type:Unsigned Integer Range: {0,1}
<b>Package</b>		
package org.sc29.wg11.mpeg4.part16		

5.2.16 Mgnt\_Replicate\_1\_2

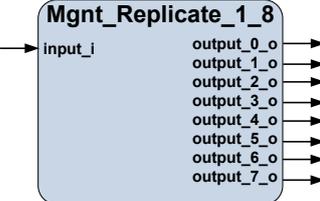
<b>FU Name</b>	Mgnt_Replicate_1_2	
<b>Description</b>	The output is generated by replicating the transferred input data. The detailed process is described in the above FSM.	
	Block diagram:	Ports description
		Port Constraints: - the output ports have to be of the same data type as the input port
Process Schematic (FSM):		
<b>ISO Standards using the FU</b>	ISO/IEC 14496-16	
<b>Profiles@levels supported</b>	All (Generic)	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
input_i	(Any type of token)	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
output_0_o	(Any type of token identical with input port)	
output_1_o	(Any type of token identical with input port)	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part16		

5.2.17 Mgnt\_Replicate\_1\_4

<b>FU Name</b>	Mgnt_Replicate_1_4
<b>Description</b>	The output is generated by replicating the transferred input data. The detailed process is described in the above FSM.

	<p>Block diagram:</p>  <p>Ports description</p> <p>Port Constraints: - the output ports have to be of the same data type as the input port</p> <p>Process Schematic (FSM):</p> 	
<p><b>ISO Standards using the FU</b></p>	<p>ISO/IEC 14496-16</p>	
<p><b>Profiles@levels supported</b></p>	<p>All (Generic)</p>	
<p><b>Input</b></p>		
<p><b>Name</b></p>	<p><b>Token</b></p>	
<p>input_i</p>	<p>(Any type of token)</p>	
<p><b>Output</b></p>		
<p><b>Name</b></p>	<p><b>Token</b></p>	
<p>output_0_o</p>	<p>(Any type of token identical with input port)</p>	
<p>output_1_o</p>	<p>(Any type of token identical with input port)</p>	
<p>output_2_o</p>	<p>(Any type of token identical with input port)</p>	
<p>output_3_o</p>	<p>(Any type of token identical with input port)</p>	
<p><b>Parameter</b></p>		
<p><b>Name</b></p>	<p><b>Description</b></p>	<p><b>Range</b></p>
<p><b>Package</b></p>		
<p>package org.sc29.wg11.mpeg4.part16</p>		

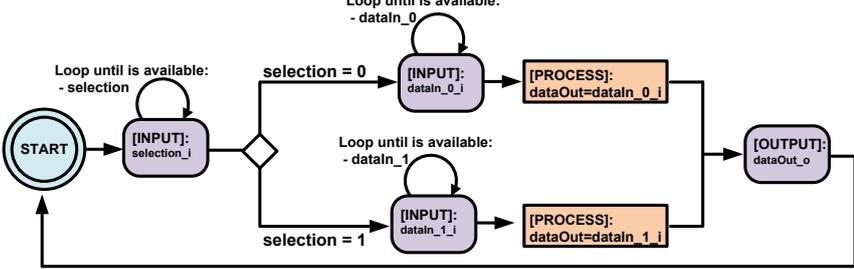
**5.2.18 Mgnt\_Replicate\_1\_8**

<p><b>FU Name</b></p> <p><b>Description</b></p>	<p>Mgnt_Replicate_1_8</p> <p>The output is generated by replicating the transferred input data. The detailed process is described in the above FSM.</p> <p>Block diagram:</p>  <p>Ports description</p> <p>Port Constraints: - the output ports have to be of the same data type as the input port</p>
---	---

	<p>Process Schematic (FSM):</p>	
<p><b>ISO Standards using the FU</b></p>	<p>ISO/IEC 14496-16</p>	
<p><b>Profiles@levels supported</b></p>	<p>All (Generic)</p>	
<p><b>Input</b></p>		
<p><b>Name</b></p>	<p><b>Token</b></p>	
<p>input_i</p>	<p>(Any type of token)</p>	
<p><b>Output</b></p>		
<p><b>Name</b></p>	<p><b>Token</b></p>	
<p>output_0_o</p>	<p>(Any type of token identical with input port)</p>	
<p>output_1_o</p>	<p>(Any type of token identical with input port)</p>	
<p>output_2_o</p>	<p>(Any type of token identical with input port)</p>	
<p>output_3_o</p>	<p>(Any type of token identical with input port)</p>	
<p>output_4_o</p>	<p>(Any type of token identical with input port)</p>	
<p>output_5_o</p>	<p>(Any type of token identical with input port)</p>	
<p>output_6_o</p>	<p>(Any type of token identical with input port)</p>	
<p>output_7_o</p>	<p>(Any type of token identical with input port)</p>	
<p><b>Parameter</b></p>		
<p><b>Name</b></p>	<p><b>Description</b></p>	<p><b>Range</b></p>
<p><b>Package</b></p>		
<p>package org.sc29.wg11.mpeg4.part16</p>		

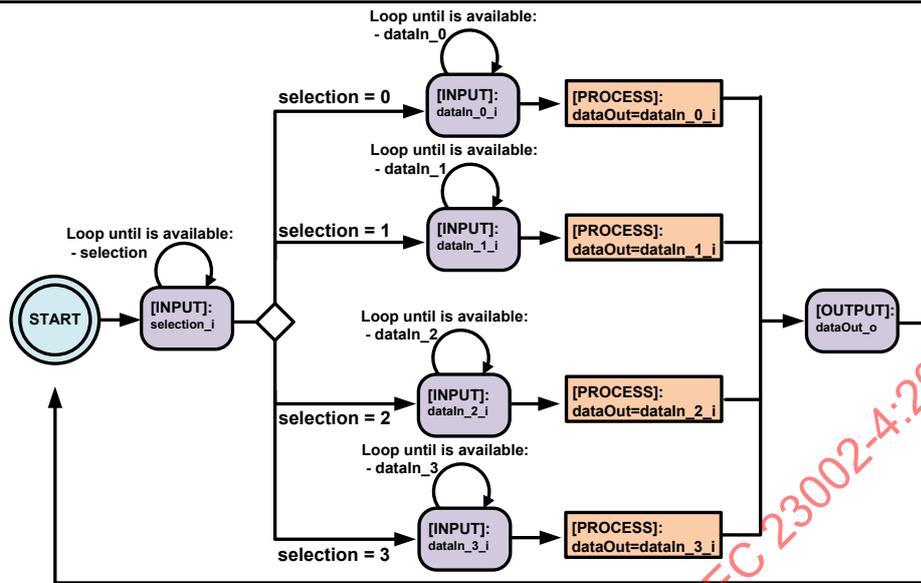
5.2.19 Mgnt\_MUX\_2\_1

<p><b>FU Name</b></p>	<p>Mgnt_MUX_2_1</p>	
<p><b>Description</b></p>		<p>Port Constraints:</p> <ul style="list-style-type: none"> <li>- the dataIn ports have to be of the same data type</li> <li>- the dataOut port has to be of the same data type as the dataIn input ports</li> <li>- the selection port is of type bit.</li> </ul> <p>MUX_2_1 Schematic (FSM):</p>

	 <p>MUX_2_1 Process:</p> <pre> START INPUT: selection SWITCH selection CASE 0:   INPUT: dataIn_0_i   dataOut = dataIn_0 CASE 1:   INPUT: dataIn_1_i   dataOut = dataIn_1 OUTPUT: dataOut_o GOTO START     </pre>	
<b>ISO Standards using the FU</b>	ISO/IEC 14496-16	
<b>Profiles@levels supported</b>	All (Generic)	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
dataIn_0_i	(Any type of token identical with dataOut_o port)	
dataIn_1_i	(Any type of token identical with dataOut_o port)	
Selection_i	BIT	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
dataOut_o	(Any type of token)	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part16		

### 5.2.20 Mgnt\_MUX\_4\_1

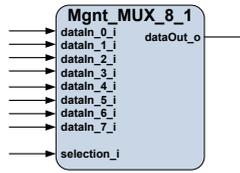
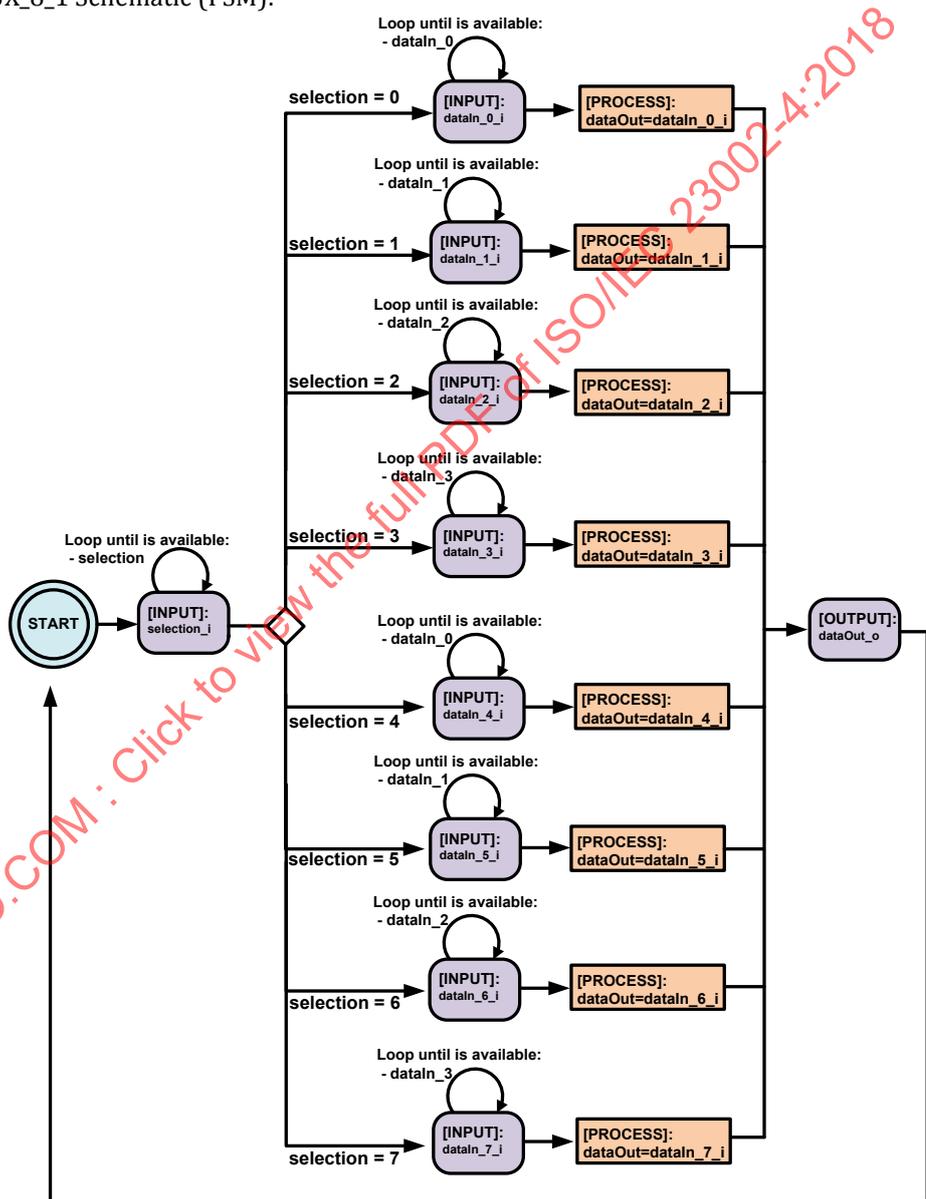
<b>FU Name</b>	Mgnt_MUX_4_1	
<b>Description</b>		<p>Port Constraints:</p> <ul style="list-style-type: none"> <li>- the dataIn ports have to be of the same data type</li> <li>- the dataOut port has to be of the same data type as the dataIn input ports</li> <li>- the selection port is of type unsigned integer, represented on 2 bits.</li> </ul> <p>MUX_4_1 Schematic (FSM):</p>



MUX\_4\_1 Process:  
 START  
 INPUT: selection  
 SWITCH selection  
 CASE 0:  
   INPUT: dataIn\_0\_i  
   dataOut = dataIn\_0  
 CASE 1:  
   INPUT: dataIn\_1\_i  
   dataOut = dataIn\_1  
 CASE 2:  
   INPUT: dataIn\_2\_i  
   dataOut = dataIn\_2  
 CASE 3:  
   INPUT: dataIn\_3\_i  
   dataOut = dataIn\_3  
 OUTPUT: dataOut\_o  
 GOTO START

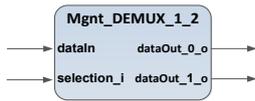
ISO Standards using the FU	ISO/IEC 14496-16	
Profiles@levels supported	All (Generic)	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
dataIn_0_i	(Any type of token identical with dataOut_o port)	
dataIn_1_i	(Any type of token identical with dataOut_o port)	
dataIn_2_i	(Any type of token identical with dataOut_o port)	
dataIn_3_i	(Any type of token identical with dataOut_o port)	
Selection_i	BIT	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
dataOut_o	(Any type of token)	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part16		

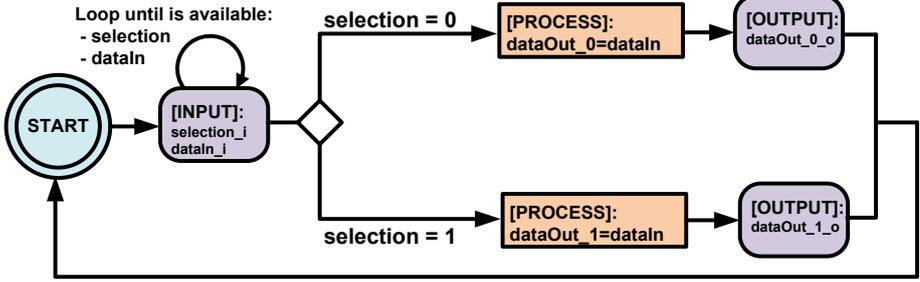
5.2.21 Mgnt\_MUX\_8\_1

FU Name	Mgnt_MUX_8_1
<p>Description</p>	<div style="display: flex; justify-content: space-between;"> <div style="width: 25%;">  </div> <div style="width: 70%;"> <p>Port Constraints:</p> <ul style="list-style-type: none"> <li>- the dataIn ports have to be of the same data type</li> <li>- the dataOut port has to be of the same data type as the dataIn input ports</li> <li>- the selection port is of type unsigned integer, represented on 3 bits.</li> </ul> </div> </div> <p>MUX_8_1 Schematic (FSM):</p>  <p>MUX_8_1 Process:</p> <pre> START INPUT: selection SWITCH selection CASE 0:   INPUT: dataIn_0_i   dataOut = dataIn_0 CASE 1:   INPUT: dataIn_1_i   dataOut = dataIn_1 CASE 2:   INPUT: dataIn_2_i   dataOut = dataIn_2 CASE 3:   INPUT: dataIn_3_i   dataOut = dataIn_3 CASE 4:   INPUT: dataIn_4_i   dataOut = dataIn_4 CASE 5:   INPUT: dataIn_5_i   dataOut = dataIn_5 CASE 6:   INPUT: dataIn_6_i   dataOut = dataIn_6 CASE 7:   INPUT: dataIn_7_i   dataOut = dataIn_7     </pre>

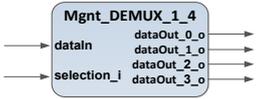
	<p>INPUT: dataIn_2_i dataOut = dataIn_2</p> <p>CASE 3: INPUT: dataIn_3_i dataOut = dataIn_3</p> <p>CASE 4: INPUT: dataIn_4_i dataOut = dataIn_4</p> <p>CASE 5: INPUT: dataIn_5_i dataOut = dataIn_5</p> <p>CASE 6: INPUT: dataIn_6_i dataOut = dataIn_6</p> <p>CASE 7: INPUT: dataIn_7_i dataOut = dataIn_7</p> <p>OUTPUT: dataOut_o GOTO START</p>	
<b>ISO Standards using the FU</b>	ISO/IEC 14496-16	
<b>Profiles@levels supported</b>	All (Generic)	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
dataIn_0_i	(Any type of token identical with dataOut_o port)	
dataIn_1_i	(Any type of token identical with dataOut_o port)	
dataIn_2_i	(Any type of token identical with dataOut_o port)	
dataIn_3_i	(Any type of token identical with dataOut_o port)	
dataIn_4_i	(Any type of token identical with dataOut_o port)	
dataIn_5_i	(Any type of token identical with dataOut_o port)	
dataIn_6_i	(Any type of token identical with dataOut_o port)	
dataIn_7_i	(Any type of token identical with dataOut_o port)	
Selection_i	BIT	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
dataOut_o	(Any type of token)	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part16		

5.2.22 Mgnt\_DEMUX\_1\_2

<b>FU Name</b>	Mgnt_DEMUX_1_2	
<b>Description</b>	 <p>Port Constraints:</p> <ul style="list-style-type: none"> <li>- the dataOut ports have to be of the same data type</li> <li>- the dataIn port has to be of the same data type as the dataOut ports</li> <li>- the selection port is of type bit.</li> </ul>	<p>DEMUX_1_2 Schematic (FSM):</p>

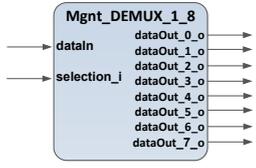
	 <p>DEMUX_1_2 Process:</p> <pre> START INPUT: selection INPUT: dataIn SWITCH selection CASE 0:     dataOut_0 = dataIn     OUTPUT: dataOut_0 CASE 1:     dataOut_1 = dataIn     OUTPUT: dataOut_1 GOTO START         </pre>	
<b>ISO Standards using the FU</b>	ISO/IEC 14496-16	
<b>Profiles@levels supported</b>	All (Generic)	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
dataIn	(Any type of token)	
selection_i	BIT	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
dataOut_0_o	(Any type of token identical with dataIn port)	
dataOut_1_o	(Any type of token identical with dataIn port)	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part16		

5.2.23 Mgnt\_DEMUX\_1\_4

<b>FU Name</b>	Mgnt_DEMUX_1_4	
<b>Description</b>		<p>Port Constraints:</p> <ul style="list-style-type: none"> <li>- the dataOut ports have to be of the same data type</li> <li>- the dataIn port has to be of the same data type as the dataOut ports</li> <li>- the selection port is of type unsigned integer represented on 2 bits.</li> </ul> <p>DEMUX_1_4 Schematic (FSM):</p>

	<p>Loop until is available: - selection - dataIn</p> <p>START</p> <p>[INPUT]: selection_i dataIn_i</p> <p>selection = 0 → [PROCESS]: dataOut_0=dataIn → [OUTPUT]: dataOut_0_o</p> <p>selection = 1 → [PROCESS]: dataOut_1=dataIn → [OUTPUT]: dataOut_1_o</p> <p>selection = 2 → [PROCESS]: dataOut_2=dataIn → [OUTPUT]: dataOut_2_o</p> <p>selection = 3 → [PROCESS]: dataOut_3=dataIn → [OUTPUT]: dataOut_3_o</p> <p>DEMUX_1_4 Process: START INPUT: selection INPUT: dataIn SWITCH selection CASE 0: dataOut_0 = dataIn OUTPUT: dataOut_0 CASE 1: dataOut_1 = dataIn OUTPUT: dataOut_1 CASE 2: dataOut_2 = dataIn OUTPUT: dataOut_2 CASE 3: dataOut_3 = dataIn OUTPUT: dataOut_3 GOTO START</p>
<p><b>ISO Standards using the FU</b></p>	<p>ISO/IEC 14496-16</p>
<p><b>Profiles@levels supported</b></p>	<p>All (Generic)</p>
<p><b>Input</b></p>	
<p><b>Name</b></p>	<p><b>Token</b></p>
<p>dataIn</p>	<p>(Any type of token)</p>
<p>selection_i</p>	<p>BIT</p>
<p><b>Output</b></p>	
<p><b>Name</b></p>	<p><b>Token</b></p>
<p>dataOut_0_o</p>	<p>(Any type of token identical with dataIn port)</p>
<p>dataOut_1_o</p>	<p>(Any type of token identical with dataIn port)</p>
<p>dataOut_2_o</p>	<p>(Any type of token identical with dataIn port)</p>
<p>dataOut_3_o</p>	<p>(Any type of token identical with dataIn port)</p>
<p><b>Parameter</b></p>	
<p><b>Name</b></p>	<p><b>Description</b></p>
<p></p>	<p><b>Range</b></p>
<p><b>Package</b></p>	
<p>package org.sc29.wg11.mpeg4.part16</p>	

5.2.24 Mgnt\_DEMUX\_1\_8

FU Name	Mgnt_DEMUX_1_8
<p style="text-align: center;"><b>Description</b></p>	<div style="display: flex; justify-content: space-between;"> <div style="width: 30%;">  </div> <div style="width: 65%;"> <p>Port Constraints:</p> <ul style="list-style-type: none"> <li>- the dataOut ports have to be of the same data type</li> <li>- the dataIn port has to be of the same data type as the dataOut ports</li> <li>- the selection port is of type unsigned integer represented on 3 bits.</li> </ul> </div> </div> <p>DEMUX_1_8 Schematic (FSM):</p> <p>Loop until is available:</p> <ul style="list-style-type: none"> <li>- selection</li> <li>- dataIn</li> </ul> <p>DEMUX_1_8 Process:</p> <pre> START INPUT: selection INPUT: dataIn SWITCH selection CASE 0:   dataOut_0 = dataIn   OUTPUT: dataOut_0 CASE 1:   dataOut_1 = dataIn   OUTPUT: dataOut_1 CASE 2:   dataOut_2 = dataIn   OUTPUT: dataOut_2 CASE 3:   dataOut_3 = dataIn   OUTPUT: dataOut_3 CASE 4:   dataOut_4 = dataIn   OUTPUT: dataOut_4 CASE 5:   dataOut_5 = dataIn   OUTPUT: dataOut_5 </pre>

	CASE 6: dataOut_6 = dataIn OUTPUT: dataOut_6 CASE 7: dataOut_7 = dataIn OUTPUT: dataOut_7 GOTO START	
<b>ISO Standards using the FU</b>	ISO/IEC 14496-16	
<b>Profiles@levels supported</b>	All (Generic)	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
dataIn	(Any type of token)	
selection_i	BIT	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
dataOut_0_o	(Any type of token identical with dataIn port)	
dataOut_1_o	(Any type of token identical with dataIn port)	
dataOut_2_o	(Any type of token identical with dataIn port)	
dataOut_3_o	(Any type of token identical with dataIn port)	
dataOut_4_o	(Any type of token identical with dataIn port)	
dataOut_5_o	(Any type of token identical with dataIn port)	
dataOut_6_o	(Any type of token identical with dataIn port)	
dataOut_7_o	(Any type of token identical with dataIn port)	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part16		

5.2.25 Mgnt\_ExtractSegment

<b>FU Name</b>	Mgnt_ExtractSegment						
<b>Description</b>	This FU extracts a piece of data from the input of size embedded in the input data, outputs the size of the segment, the segment data and repeats the rest of the input on the output port.						
	The size of the segment is embedded in the beginning of the input data and it is represented using a number of sizeBytes of type unsigned integer. The computed size is outputted as unsigned integer on the extractedDataSize_o output port, the extracted data segment is outputted on the extractedDataOut_o port and the rest of the input data is repeated on the repeatedDataOut_o port.						
	EXAMPLE <table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 33%; text-align: center;">extractedDataSize</td> <td style="width: 33%; text-align: center;">extractedData</td> <td style="width: 33%; text-align: center;">repeatedData</td> </tr> <tr> <td style="text-align: center;">size = sizeBytes</td> <td style="text-align: center;">size = extractedDataSize</td> <td style="text-align: center;">size = rest of input</td> </tr> </table>		extractedDataSize	extractedData	repeatedData	size = sizeBytes	size = extractedDataSize
extractedDataSize	extractedData	repeatedData					
size = sizeBytes	size = extractedDataSize	size = rest of input					

	<table border="1" data-bbox="821 257 1353 537"> <thead> <tr> <th>Port Name</th> <th>Direction (I/O)</th> <th>Token RANGE</th> </tr> </thead> <tbody> <tr> <td>dataIn_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>sizeBytes_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>extractedDataOut_o</td> <td>O</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>extractedDataSize_o</td> <td>O</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>repeatedDataOut_o</td> <td>O</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> </tbody> </table>	Port Name	Direction (I/O)	Token RANGE	dataIn_i	I	UINT8, UINT16, UINT32, UINT64	sizeBytes_i	I	UINT8, UINT16, UINT32, UINT64	extractedDataOut_o	O	UINT8, UINT16, UINT32, UINT64	extractedDataSize_o	O	UINT8, UINT16, UINT32, UINT64	repeatedDataOut_o	O	UINT8, UINT16, UINT32, UINT64
Port Name	Direction (I/O)	Token RANGE																	
dataIn_i	I	UINT8, UINT16, UINT32, UINT64																	
sizeBytes_i	I	UINT8, UINT16, UINT32, UINT64																	
extractedDataOut_o	O	UINT8, UINT16, UINT32, UINT64																	
extractedDataSize_o	O	UINT8, UINT16, UINT32, UINT64																	
repeatedDataOut_o	O	UINT8, UINT16, UINT32, UINT64																	
<b>ISO Standards using the FU</b>	ISO/IEC 14496-16																		
<b>Profiles@levels supported</b>	All (Generic)																		
<b>Input</b>																			
<b>Name</b>	<b>Token</b>																		
dataIn_i	UINT8, UINT16, UINT32, UINT64																		
sizeBytes_i	UINT8, UINT16, UINT32, UINT64																		
<b>Output</b>																			
<b>Name</b>	<b>Token</b>																		
extractedDataOut_o	UINT8, UINT16, UINT32, UINT64																		
extractedDataSize_o	UINT8, UINT16, UINT32, UINT64																		
repeatedDataOut_o	UINT8, UINT16, UINT32, UINT64																		
<b>Parameter</b>																			
<b>Name</b>	<b>Description</b>	<b>Range</b>																	
<b>Package</b>																			
package org.sc29.wg11.mpeg4.part16																			

5.2.26 Mgnt ProviderValue

<b>FU Name</b>	Mgnt_ProviderValue						
<b>Description</b>	<table border="1" data-bbox="847 1693 1362 1794"> <thead> <tr> <th>Port Name</th> <th>Direction (I/O)</th> <th>Token RANGE</th> </tr> </thead> <tbody> <tr> <td>dataOut_o</td> <td>O</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> </tbody> </table> <p>ProviderValue Process Schematic (FSM):</p>	Port Name	Direction (I/O)	Token RANGE	dataOut_o	O	UINT8, UINT16, UINT32, UINT64
Port Name	Direction (I/O)	Token RANGE					
dataOut_o	O	UINT8, UINT16, UINT32, UINT64					

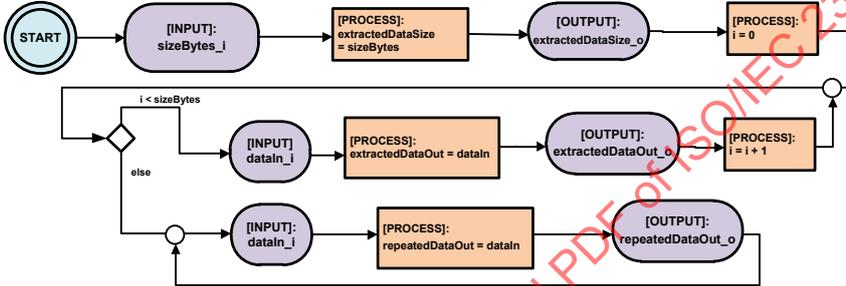
	 <p>ProviderValue Process:                  START:                  dataOut = PARAM                  OUTPUT:                  dataOut_o</p>	
<b>ISO Standards using the FU</b>	ISO/IEC 14496-16	
<b>Profiles@levels supported</b>	All (Generic)	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
dataOut_o	UINT8, UINT16, UINT32, UINT64	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
PARAM	Describes the value of the parameter that is outputted one time as dataOut. It is set at the network configuration level.	
<b>Package</b>		
package org.sc29.wg11.mpeg4.part16		

5.2.27 Mgnt\_RepeatSegment

<b>FU Name</b>	Mgnt_RepeatSegment																
<b>Description</b>	This FU describes how to repetitively transfer the given data.																
		<table border="1"> <thead> <tr> <th>Port Name</th> <th>Direction (I/O)</th> <th>Token RANGE</th> </tr> </thead> <tbody> <tr> <td>dataIn_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>repetition_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>numofdata_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>dataOut_o</td> <td>O</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> </tbody> </table>	Port Name	Direction (I/O)	Token RANGE	dataIn_i	I	UINT8, UINT16, UINT32, UINT64	repetition_i	I	UINT8, UINT16, UINT32, UINT64	numofdata_i	I	UINT8, UINT16, UINT32, UINT64	dataOut_o	O	UINT8, UINT16, UINT32, UINT64
Port Name	Direction (I/O)	Token RANGE															
dataIn_i	I	UINT8, UINT16, UINT32, UINT64															
repetition_i	I	UINT8, UINT16, UINT32, UINT64															
numofdata_i	I	UINT8, UINT16, UINT32, UINT64															
dataOut_o	O	UINT8, UINT16, UINT32, UINT64															
RepeatSegment Process Schematic (FSM):																	

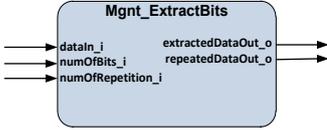
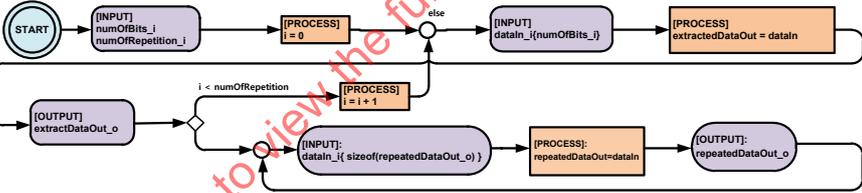
	<pre> graph TD     START((START)) --&gt; Input1([INPUT] numofdata)     Input1 --&gt; Input2([INPUT] dataIn)     Input2 --&gt; Process1([PROCESS] data[i] = dataIn)     Process1 --&gt; Dec1{i &lt; numofdata}     Dec1 --&gt; Yes  Input2     Dec1 --&gt; No  Input3([INPUT] repetition)     Input3 --&gt; Dec2{i &lt; repetition}     Dec2 --&gt; Yes  Process1     Dec2 --&gt; No  Process2([PROCESS] cnt = 0 i = 0)     Process2 --&gt; Dec3{cnt &lt; numofdata}     Dec3 --&gt; Yes  Process1     Dec3 --&gt; No  Output1([OUTPUT] dataOut = data[cnt])     Output1 --&gt; Process3([PROCESS] cnt = cnt + 1 i = i + 1)     Process3 --&gt; Dec3     </pre> <p>RepeatSegment Process:          START:          INPUT: numofdata          WHILE i &lt; numofdata            INPUT: dataIn            data[i] = dataIn          INPUT: repetition          WHILE i &lt; repetition            WHILE cnt &lt; numofdata              dataOut = data[cnt]            OUTPUT: dataOut</p>	
<b>ISO Standards using the FU</b>	ISO/IEC 14496-16	
<b>Profiles@levels supported</b>	All (Generic)	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
dataIn_i	UINT8, UINT16, UINT32, UINT64	
repetition_i	UINT8, UINT16, UINT32, UINT64	
numofdata_i	UINT8, UINT16, UINT32, UINT64	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
dataOut_o	UINT8, UINT16, UINT32, UINT64	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part16		

5.2.28 Mgnt\_ExtractBytes

FU Name	Mgnt_ExtractBytes																		
<p><b>Description</b></p>	<div style="display: flex; justify-content: space-around; align-items: flex-start;"> <div style="text-align: center;">  </div> <table border="1" style="border-collapse: collapse; text-align: center;"> <thead> <tr style="background-color: #4a86e8; color: white;"> <th>Port Name</th> <th>Direction (I/O)</th> <th>Token RANGE</th> </tr> </thead> <tbody> <tr> <td>dataIn_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>sizeBytes_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>extractedDataOut_o</td> <td>O</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>repeatedDataOut_o</td> <td>O</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>extractedDataSize_o</td> <td>O</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> </tbody> </table> </div> <p style="margin-top: 20px;">Mgnt_ExtractBytes Schematic:</p>  <p style="margin-top: 20px;">Mgnt_ExtractBytes Process:</p> <pre> START:   INPUT:     sizeBytes_i   extractedDataSize = sizeBytes   OUTPUT:     extractedDataSize_o   i = 0   EXTRACT_DATA   IF i &lt; sizeBytes     INPUT:       dataIn_i       extractedDataOut = dataIn;     OUTPUT:       extractedDataOut_o     i = i + 1     GOTO EXTRACT_DATA   REPEATED_DATA   INPUT:     dataIn_i     repeatedDataOut = dataIn;   OUTPUT:     repeatedDataOut_o   GOTO REPEATED_DATA     </pre>	Port Name	Direction (I/O)	Token RANGE	dataIn_i	I	UINT8, UINT16, UINT32, UINT64	sizeBytes_i	I	UINT8, UINT16, UINT32, UINT64	extractedDataOut_o	O	UINT8, UINT16, UINT32, UINT64	repeatedDataOut_o	O	UINT8, UINT16, UINT32, UINT64	extractedDataSize_o	O	UINT8, UINT16, UINT32, UINT64
	Port Name	Direction (I/O)	Token RANGE																
	dataIn_i	I	UINT8, UINT16, UINT32, UINT64																
	sizeBytes_i	I	UINT8, UINT16, UINT32, UINT64																
	extractedDataOut_o	O	UINT8, UINT16, UINT32, UINT64																
repeatedDataOut_o	O	UINT8, UINT16, UINT32, UINT64																	
extractedDataSize_o	O	UINT8, UINT16, UINT32, UINT64																	
<p><b>ISO Standards using the FU</b></p>	<p>ISO/IEC 14496-16</p>																		
<p><b>Profiles@levels supported</b></p>	<p>All (Generic)</p>																		
<p><b>Input</b></p>																			
<p><b>Name</b></p>	<p><b>Token</b></p>																		
<p>dataIn_i</p>	<p>UINT8, UINT16, UINT32, UINT64</p>																		
<p>sizeBytes_i</p>	<p>UINT8, UINT16, UINT32, UINT64</p>																		
<p><b>Output</b></p>																			
<p><b>Name</b></p>	<p><b>Token</b></p>																		
<p>extractedDataOut_o</p>	<p>UINT8, UINT16, UINT32, UINT64</p>																		
<p>repeatedDataOut_o</p>	<p>UINT8, UINT16, UINT32, UINT64</p>																		

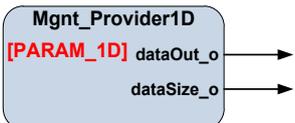
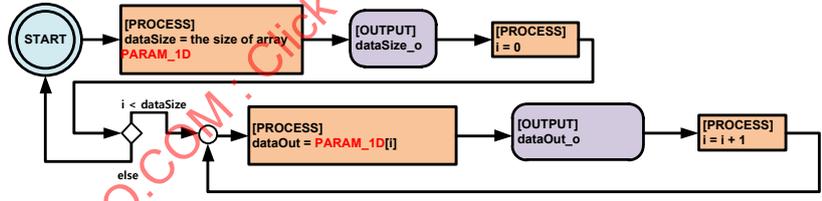
extractedDataSize_o	UINT8, UINT16, UINT32, UINT64	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part16		

5.2.29 Mgmt\_ExtractBits

<b>FU Name</b>	Mgmt_ExtractBits																		
	<p>This FU describes how to repetitively extract a given number of bits and output the values, sequentially. The rest of the input data is repeated at the output port. Internally, the bit alignment is done after finishing the bits extraction operation.</p>																		
	 <table border="1" data-bbox="876 719 1414 987"> <thead> <tr> <th>Port Name</th> <th>Direction (I/O)</th> <th>Token RANGE</th> </tr> </thead> <tbody> <tr> <td>dataIn_i</td> <td>1</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>numOfBits_i</td> <td>1</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>numOfRepetition_i</td> <td>1</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>extractedValuesOut_o</td> <td>0</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>repeatedDataOut_o</td> <td>0</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> </tbody> </table>	Port Name	Direction (I/O)	Token RANGE	dataIn_i	1	UINT8, UINT16, UINT32, UINT64	numOfBits_i	1	UINT8, UINT16, UINT32, UINT64	numOfRepetition_i	1	UINT8, UINT16, UINT32, UINT64	extractedValuesOut_o	0	UINT8, UINT16, UINT32, UINT64	repeatedDataOut_o	0	UINT8, UINT16, UINT32, UINT64
Port Name	Direction (I/O)	Token RANGE																	
dataIn_i	1	UINT8, UINT16, UINT32, UINT64																	
numOfBits_i	1	UINT8, UINT16, UINT32, UINT64																	
numOfRepetition_i	1	UINT8, UINT16, UINT32, UINT64																	
extractedValuesOut_o	0	UINT8, UINT16, UINT32, UINT64																	
repeatedDataOut_o	0	UINT8, UINT16, UINT32, UINT64																	
	<p>ExtractBits Process Schematic (FSM):</p> 																		
<b>Description</b>	<p>ExtExtractBits Process:</p> <p>START:</p> <p>INPUT:</p> <p>numOfBits_i</p> <p>numOfRepetition_i</p> <p>i = 0</p> <p>EXTRACT BITS</p> <p>INPUT:</p> <p>dataIn_i{numOfBits}</p> <p>extractedDataOut = dataIn</p> <p>OUTPUT:</p> <p>extractedDataOut_o</p> <p>IF i &lt; numOfRepetition</p> <p>i++</p> <p>GOTO EXTRACT BITS</p> <p>REPEATED_DATA_OUT</p> <p>INPUT:</p> <p>dataIn_i{ sizeof (repeatedDataOut) }</p> <p>repeatedDataOut = dataIn</p> <p>OUTPUT:</p> <p>repeatedDataOut_o</p> <p>GOTO REPEATED_DATA_OUT</p> <p>NOTE The "sizeof" operation is used to obtain the size of the token in bits (bus width). The way to use the "sizeof" operation is as follows:</p> <p>— sizeof( name of port ).</p>																		
<b>ISO Standards using the FU</b>	ISO/IEC 14496-16																		
<b>Profiles@levels</b>	All (Generic)																		

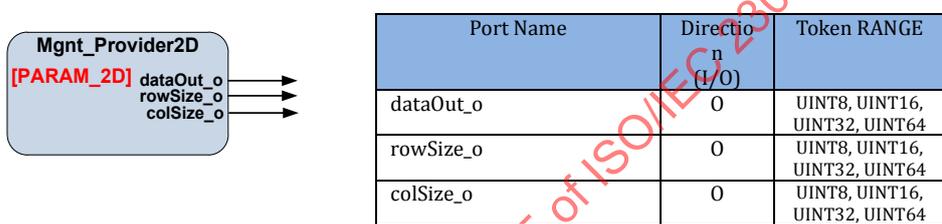
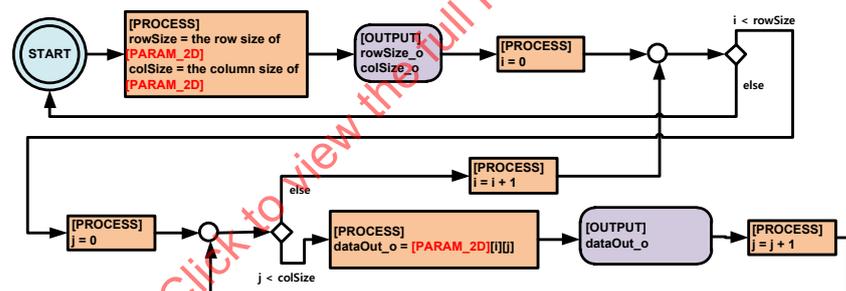
<b>supported</b>		
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
dataIn_i	UINT8, UINT16, UINT32, UINT64	
repetition_i	UINT8, UINT16, UINT32, UINT64	
numOfRepetition_i	UINT8, UINT16, UINT32, UINT64	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
extractedValuesOut_o	UINT8, UINT16, UINT32, UINT64	
repeatedDataOut_o	UINT8, UINT16, UINT32, UINT64	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part16		

5.2.30 Mgnt\_Provider1D

<b>FU Name</b>	Mgnt_Provider1D									
<b>Description</b>	<p>This FU describes how to provide 1D array values from FU network description.</p>  <table border="1" data-bbox="726 996 1300 1164"> <thead> <tr> <th>Port Name</th> <th>Direction (I/O)</th> <th>Token RANGE</th> </tr> </thead> <tbody> <tr> <td>dataOut_o</td> <td>O</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>dataSize_o</td> <td>O</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> </tbody> </table> <p>Provider1D Process Schematic (FSM):</p>  <p>Provider1D Process:</p> <pre> START:   dataSize = the size of array PARAM_1D OUTPUT:   dataSize_o   i = 0   // READ 1D TABLE   IF i &lt; dataSize     dataOut = PARAM_1D[i]     OUTPUT:       dataOut_o     i = i + 1   GOTO READ 1D TABLE                 </pre>	Port Name	Direction (I/O)	Token RANGE	dataOut_o	O	UINT8, UINT16, UINT32, UINT64	dataSize_o	O	UINT8, UINT16, UINT32, UINT64
	Port Name	Direction (I/O)	Token RANGE							
dataOut_o	O	UINT8, UINT16, UINT32, UINT64								
dataSize_o	O	UINT8, UINT16, UINT32, UINT64								
<b>ISO Standards using the FU</b>	ISO/IEC 14496-16									
<b>Profiles@levels supported</b>	All (Generic)									
<b>Output</b>										
<b>Name</b>	<b>Token</b>									

dataOut_o	UINT8, UINT16, UINT32, UINT64	
dataSize_o	UINT8, UINT16, UINT32, UINT64	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
PARAM_1D	It is stored internally as an array, loaded at the network configuration stage.	{{[0, 2 <sup>32</sup> -1], ..., [0, 2 <sup>32</sup> -1]}}
<b>Package</b>		
package org.sc29.wg11.mpeg4.part16		

### 5.2.31 Mgmt\_Provider2D

<b>FU Name</b>	Mgmt_Provider2D
<b>Description</b>	<p>This FU describes how to provide 2D values in the row-base manner.</p>  <p>Provider2D Process Schematic (FSM):</p>  <p>Provider2D Process:</p> <pre> START:   rowSize = the row size of 2-dimensional array PARAM_2D   colSize = the column size of 2-dimensional array PARAM_2D OUTPUT:   rowSize_o   colSize_o   i = 0 ROW_LOOP IF i &lt; rowSize   j = 0   COL_LOOP   IF i &lt; colSize     dataOut = PARAM_2D[i][j]     OUTPUT:       dataOut_o     j = j + 1     GOTO COL_LOOP   ELSE     i = i + 1     GOTO ROW_LOOP ELSE GOTO START           </pre>
<b>ISO Standards using the FU</b>	ISO/IEC 14496-16
<b>Profiles@levels supported</b>	All (Generic)

Output		
Name	Token	
dataOut_o	UINT8, UINT16, UINT32, UINT64	
rowSize_o	UINT8, UINT16, UINT32, UINT64	
colSize_o	UINT8, UINT16, UINT32, UINT64	
Parameter		
Name	Description	Range
PARAM_2D	It is stored internally as 2-dimensional array, loaded at the network configuration stage.	$\{[0, 2^{32}-1], \dots, [0, 2^{32}-1]\}$ , $\{[0, 2^{32}-1], \dots, [0, 2^{32}-1]\}$ ... $\{[0, 2^{32}-1], \dots, [0, 2^{32}-1]\}$
Package		
package org.sc29.wg11.mpeg4.part16		

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23002-4:2018

## 6 FUs for MPEG-4 Simple Profile

### 6.1 General

FUs for building MPEG-4 Simple Profile decoder are described in this clause.

### 6.2 Syntax parsing

#### 6.2.1 Algo\_SynP

<b>FU Name</b>	Algo_SynP	
<b>Description</b>	This module is parsing all syntax element from an MPEG-4 Simple Profile (SP) bitstream.	
<b>Profiles@levels supported</b>	MPEG-4 SP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
BITS	BIT token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
ROUND	ROUND token	
F_CODE	F_CODE token	
MV	MV token	
RUN	RUN token	
VALUE	VALUE token	
LAST	LAST token	
WIDTH	WIDTH token	
HEIGHT	HEIGHT token	
QP	QUANT token	
ACCODED	ACCODED token	
MOTION	MOTION token	
ACPRED	ACPRED token	
BTYPE	BTYPE token	
FOURMV	FOURMV token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part2.sp.parser		

#### 6.2.2 Mgnt\_BlockExpand

<b>FU Name</b>	Mgnt_BlockExpand	
<b>Description</b>	This module is composing a 8×8 block with RUN, VALUE and LAST syntax elements from the parser.	
<b>Profiles@levels supported</b>		
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
RUN	RUN token	
VALUE	VALUE token	
LAST	LAST token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	

OUT	BLOCK token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part2.sp.parser		

### 6.2.3 Mgnt\_Splitter420B

<b>FU Name</b>	Mgnt_Splitter420B	
<b>Description</b>	This module split each 8×8 block in 4 Y, 1 U and 1V depending on ACCODED and the BTYPE values.	
<b>Profiles@levels supported</b>	MPEG-4 SP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
B	BLOCK token	
ACCODED	ACCODED token	
BTYPE	BTYPE token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
B_Y	BLOCK token	
B_U	BLOCK token	
B_V	BLOCK token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part2.sp.parser		

### 6.2.4 Mgnt\_Splitter420MV

<b>FU Name</b>	Mgnt_SplitterMV	
<b>Description</b>	This module split each 6 MV token in 4 Y, 1 U and 1V when MOTION is true.	
<b>Profiles@levels supported</b>	MPEG-4 SP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
MV	MV token	
MOTION	MOTION token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
MV_Y	MV token	
MV_U	MV token	
MV_V	MV token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part2.sp.parser		

## 6.2.5 Algo\_MVR\_MedianOfThreeLeftAndTopAndTopRight

<b>FU Name</b>	Algo_MVSequence_LeftAndTopAndTopRight	
<b>Description</b>	This module computes the sequence of coordinates of the different blocks necessary for the prediction of the motion vectors. From the type of encoding of the block (given by the FOURMV, MOTION, VOPMODE, and WIDTH), the FU generates the coordinates of the blocks (on the A port) which will be used by the FU charged of reconstructing the motion vectors for each 8×8 block.	
<b>Profiles@levels supported</b>	MPEG-4 SP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
FOURMV	FOURMV token	
MOTION	MOTION token	
BTYPE	BTYPE token	
WIDTH	WIDTH token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
A	COORDINATE token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part2.sp.parser		

## 6.2.6 Mgnt\_Splitter\_420\_TYPE

<b>FU Name</b>	Mgnt_Splitter_420_TYPE	
<b>Description</b>	This module distributes each VOPMODE, ACCODED, ACPRED, and MOTION for each Y, U, and V components sequentially.	
<b>Profiles@levels supported</b>	MPEG-4 SP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
BTYPE	BTYPE token	
MOTION	MOTION token	
ACCODED	ACCODED token	
ACPRED	ACPRED token	
QP	QUANT token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
BTYPE_Y	BTYPE token	
BTYPE_U	BTYPE token	
BTYPE_V	BTYPE token	
MOTION_Y	MOTION token	
MOTION_U	MOTION token	
MOTION_V	MOTION token	
ACCODED_Y	ACCODED token	
ACCODED_U	ACCODED token	
ACCODED_V	ACCODED token	
ACPRED_Y	ACPRED token	
ACPRED_U	ACPRED token	
ACPRED_V	ACPRED token	

QP_Y	QUANT token
QP_U	QUANT token
QP_V	QUANT token
<b>Parameter</b>	
<b>Name</b>	<b>Description</b>
<b>Package</b>	
package org.sc29.wg11.mpeg4.part2.sp.parser	

6.2.7 Algo\_VLDtableB6\_MPEG4Part2

<b>FU Name</b>	Algo_VLDtableB6_MPEG4Part2	
<b>Description</b>	This Functional Unit decodes the mcbpc element of syntax of an MPEG-4 part 2 conforming bitstream. It is in the case of intra mode. It decodes the bits according to ISO/IEC 14496-2:2004, Table B.6.	
<b>Profiles@levels supported</b>	MPEG-4 SP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
BITS	BIT token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
FINISH	ACKNOWLEDGMENT token	
DATA	MCBPC token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part2.sp.parser.vlc		

6.2.8 Algo\_VLDtableB7\_MPEG4Part2

<b>FU Name</b>	Algo_VLDtableB7_MPEG4Part2	
<b>Description</b>	This Functional Unit decodes the mcbpc element of syntax of an MPEG-4 conforming bitstream. It is in the case of inter mode. It decodes the bits according to ISO/IEC 14496-2:2004, Table B.7.	
<b>Profiles@levels supported</b>	MPEG-4 SP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
BITS	BIT token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
FINISH	ACKNOWLEDGMENT token	
DATA	MCBPC token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part2.sp.parser.vlc		

## 6.2.9 Algo\_VLDtableB8\_MPEG4Part2

<b>FU Name</b>	Algo_VLDtableB8_MPEG4Part2	
<b>Description</b>	This Functional Unit decodes the cbpy element of syntax of an MPEG-4 conforming bitstream. It decodes the bits according to ISO/IEC 14496-2:2004, Table B.8.	
<b>Profiles@levels supported</b>	MPEG-4 SP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
BITS	BIT token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
FINISH	ACKNOWLEDGMENT token	
DATA	CBPY token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part2.sp.parser.vlc		

## 6.2.10 Algo\_VLDtableB12\_MPEG4Part2

<b>FU Name</b>	Algo_VLDtableB12_MPEG4Part2	
<b>Description</b>	This Functional Unit decodes the elements of syntax relative to the motion vectors in an MPEG-4 conformant bitstream ("horizontal_mv_data" and "vertical_mv_data"). It decodes the bits according to ISO/IEC 14496-2:2004, Table B.12.	
<b>Profiles@levels supported</b>	MPEG-4 SP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
BITS	BIT token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
FINISH	ACKNOWLEDGMENT token	
DATA	MVD token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part2.sp.parser.vlc		

## 6.2.11 Algo\_VLDtableB13\_MPEG4Part2

<b>FU Name</b>	Algo_VLDtableB13_MPEG4Part2	
<b>Description</b>	This Functional Unit decodes the "dct_dc_size" element of syntax in an MPEG-4 conforming bitstream. The decoding applies only for the luminance macroblocks. It decodes the bits according to ISO/IEC 14496-2:2004, Table B.13.	
<b>Profiles@levels supported</b>	MPEG-4 SP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
BITS	BIT token	

<b>Output</b>		
<b>Name</b>	<b>Token</b>	
FINISH	ACKNOWLEDGMENT token	
DATA	DCT_DC_SIZE token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part2.sp.parser.vlc		

6.2.12 Algo\_VLDtableB14\_MPEG4Part2

<b>FU Name</b>	Algo_VLDtableB14_MPEG4Part2	
<b>Description</b>	This Functional Unit decodes the “dct_dc_size” element of syntax in an MPEG-4 conforming bitstream. The decoding applies only for the chrominance macroblocks. It decodes the bits according to ISO/IEC 14496-2:2004, Table B.14.	
<b>Profiles@levels supported</b>	MPEG-4 SP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
BITS	BIT token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
FINISH	ACKNOWLEDGMENT token	
DATA	DCT_DC_SIZE token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part2.sp.parser.vlc		

6.2.13 Algo\_VLDtableB15\_MPEG4Part2

<b>FU Name</b>	Algo_VLDtableB15_MPEG4Part2	
<b>Description</b>	This Functional Unit decodes the “dct_dc_differencial” element of syntax in an MPEG-4 conforming bitstream. It decodes the bits according to ISO/IEC 14496-2:2004, Table B.15.	
<b>Profiles@levels supported</b>	MPEG-4 SP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
BITS	BIT token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
FINISH	ACKNOWLEDGMENT token	
DATA	DCT_DC_DIFF token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part2.sp.parser.vlc		

## 6.2.14 Algo\_VLDtableB16\_MPEG4Part2

<b>FU Name</b>	Algo_VLDtableB16_MPEG4Part2	
<b>Description</b>	This Functional Unit decodes the INTRA coefficients as elements of syntax in an MPEG-4 conforming bitstream. It decodes the bits according to ISO/IEC 14496-2:2004, Table B.16.	
<b>Profiles@levels supported</b>	MPEG-4 SP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
BITS	BIT token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
FINISH	ACKNOWLEDGMENT token	
RUN	RUN token	
VALUE	VALUE token	
LAST	LAST token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part2.sp.parser.vlc		

## 6.2.15 Algo\_VLDtableB17\_MPEG4Part2

<b>FU Name</b>	Algo_VLDtableB17_MPEG4Part2	
<b>Description</b>	This Functional Unit decodes the INTER coefficients as elements of syntax in an MPEG-4 conforming bitstream. It decodes the bits according to ISO/IEC 14496-2:2004, Table B.16.	
<b>Profiles@levels supported</b>	MPEG-4 SP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
BITS	BIT token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
FINISH	ACKNOWLEDGMENT token	
RUN	RUN token	
VALUE	VALUE token	
LAST	LAST token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part2.sp.parser.vlc		

### 6.3 Texture decoding

#### 6.3.1 Algo\_IQ\_QSAndQmatrixMp4vOrH263Scaler

<b>FU Name</b>	Algo_IQ_QSAndQmatrixMp4vOrH263Scaler	
<b>Description</b>	This module computes inverse quantization of AC for 8×8 blocks. Supports both MPEG and H.263 Inverse Quantization modes.	
<b>Profiles@levels supported</b>	MPEG-4 SP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
DC	DC token	
AC	AC token	
QP	QUANT token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
OUT	BLOCK token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part2.sp.texture		

#### 6.3.2 Algo\_DCRAddr\_ThreeLeftTop\_8x8

<b>FU Name</b>	Algo_DCRAddr_ThreeLeftTop_8x8	
<b>Description</b>	This module calculates the addresses of the three neighboring blocks for the current 8×8 block used for DC prediction. If any of the neighbors is not coded for some reason (either outside frame boundaries or skipped by encoder), the address is set as zero. Otherwise the lowest two bits specify the 8×8 component in the macroblock and the other higher order bits specify the macroblock index using circular buffer addressing.	
<b>Profiles@levels supported</b>	MPEG-4 SP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
BTYPE	BTYPE token	
WIDTH	WIDTH token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
A	COORDINATE token	
B	COORDINATE token	
C	COORDINATE token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part2.sp.texture.dc_reconstruction		

## 6.3.3 Algo\_DCRAddr\_ThreeLeftTop\_16x16

<b>FU Name</b>	Algo_DCRAddr_ThreeLeftTop_16x16	
<b>Description</b>	<p>This module calculates the addresses of the three neighboring blocks for the current 8×8 block used for DC prediction. This module manages groups of four 8×8 blocks as follows:</p> <pre> +-----+   0   1   +-----+   2   3   +-----+ </pre> <p>If any of the neighbors is not coded for some reason (either outside frame boundaries or skipped by encoder), the address is set as zero. Otherwise the lowest two bits specify the 8×8 component in the macroblock and the other higher order bits specify the macroblock index using circular buffer addressing.</p>	
<b>Profiles@levels supported</b>	MPEG-4 SP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
BTYPE	BTYPE token	
WIDTH	WIDTH token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
A	COORDINATE token	
B	COORDINATE token	
C	COORDINATE token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part2.sp.texture.dc_reconstruction		

## 6.3.4 Algo\_DCRInvPred\_CHROMA\_8x8

<b>FU Name</b>	Algo_DCRInvPred_CHROMA_8x8	
<b>Description</b>	<p>This module reconstructs the DC coefficient of the current 8×8 block based on the gradients between neighboring block DC coefficients (see ISO/IEC 14496-2). This module also forwards the decoded prediction direction to the inverse AC prediction (IAP) module and a pointer to the neighboring block used for the prediction. Since inverse quantization is necessary to reconstruct the DC coefficient, the decoded quantization parameter is forwarded to the inverse quantization module. This FU applies for 8×8 blocks of chrominance.</p>	
<b>Profiles@levels supported</b>	MPEG-4 SP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
A	COORDINATE token	
B	COORDINATE token	
C	COORDINATE token	
ACCODED	ACCODED token	
ACPRED	ACPRED token	
QFS_DC	DC token	
QP	QUANT token	
BTYPE	BTYPE token	

WIDTH	WIDTH token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
QF_DC	DC token	
PTR	COORDINATE token	
AC_PRED_DIR	ACPRED_DIR token	
SIGNED	SIGN token	
QUANT	QUANT token	
PREV_QUANT	QUANT token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part2.sp.texture.dc_reconstruction		

6.3.5 Algo\_DCRInvPred\_LUMA\_16x16

<b>FU Name</b>	Algo_DCRInvPred_LUMA_16x16	
<b>Description</b>	This module reconstructs the DC coefficient of the current 8×8 block based on the gradients between neighboring block DC coefficients (see ISO/IEC 14496-2). This module also forwards the decoded prediction direction to the inverse AC prediction (IAP) module and a pointer to the neighboring block used for the prediction. Since inverse quantization is necessary to reconstruct the DC coefficient, the decoded quantization parameter is forwarded to the inverse quantization module. This FU applies for four 8×8 blocks of luminance.	
<b>Profiles@levels supported</b>	MPEG-4 SP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
A	COORDINATE token	
B	COORDINATE token	
C	COORDINATE token	
ACCODED	ACCODED token	
ACPRED	ACPRED token	
QFS_DC	DC token	
QP	QUANT token	
BTYPE	BTYPE token	
WIDTH	WIDTH token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
QF_DC	DC token	
PTR	COORDINATE token	
AC_PRED_DIR	ACPRED_DIR token	
SIGNED	SIGN token	
QUANT	QUANT token	
PREV_QUANT	QUANT token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part2.sp.texture.dc_reconstruction		

## 6.3.6 Algo\_IS\_ZigzagOrAlternateHorizontalVertical\_8x8

<b>FU Name</b>	Algo_IS_ZigzagOrAlternateHorizontalVertical_8x8	
<b>Description</b>	This module inverts the one-dimensional array of coefficients ordered in zigzag (AC_PRED_DIR=0), alternate vertical (AC_PRED_DIR=1) or alternate horizontal (AC_PRED_DIR=2) scan to 2D raster order. It inputs a list of 64 integer coefficients (one per 8×8 block) and outputs the ordered list of integer according to the value of the token AC_PRED_DIR.	
<b>Profiles@levels supported</b>	MPEG-4 SP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
AC_PRED_DIR	ACPRED_DIR token	
QFS_AC	AC token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
PQF_AC	AC token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part2.sp.texture		

## 6.3.7 Algo\_IAP\_AdaptiveHorizontalOrVerticalPred\_8x8

<b>FU Name</b>	Algo_IAP_AdaptiveHorizontalOrVerticalPred_8x8	
<b>Description</b>	This module computes inverse AC prediction for specific AC coefficients of 8×8 blocks that have been flagged in the bitstream as coded in this fashion. It inputs a list of 63 AC coefficients received in a horizontal raster manner after being re-ordered by the inverse scan block and the addresses of the 8×8 block used for the prediction in the encoder. It outputs a list of 63 reconstructed AC coefficients sent in a horizontal raster man. The AC_PRED_DIR token communicates the direction of prediction: <ul style="list-style-type: none"> <li>— AC_PRED_DIR = -2 =&gt; NEWVOP flag</li> <li>— AC_PRED_DIR = -1 =&gt; An un-coded block so skip inverse AC prediction</li> <li>— AC_PRED_DIR = 0 =&gt; No inverse AC prediction but use zigzag inverse scan</li> <li>— AC_PRED_DIR = 1 =&gt; Prediction form the left and use alternate vertical scan</li> <li>— AC_PRED_DIR = 2 =&gt; Prediction form the top and use alternate horizontal scan</li> </ul>	
<b>Profiles@levels supported</b>	MPEG-4 SP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
PQF_AC	AC token	
PTR	COORDINATE token	
AC_PRED_DIR	ACPRED_DIR token	
QP	QUANT token	
PREV_QP	QUANT token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
QF_AC	AC token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part2.sp.texture		

6.3.8 Algo\_IAP\_AdaptiveHorizontalOrVerticalPred\_16x16

<b>FU Name</b>	Algo_IAP_AdaptiveHorizontalOrVerticalPred_16x16	
<b>Description</b>	<p>This module computes inverse AC prediction for specific AC coefficients of a 16×16 blocks that have been flagged in the bitstream as coded in this fashion. It inputs four lists of 63 AC coefficients received in a horizontal raster manner after being re-ordered by the inverse scan block and the addresses of the four 8×8 block used for the prediction in the encoder. It outputs four lists of 63 reconstructed AC coefficients sent in a horizontal raster man. The AC_PRED_DIR token communicates the direction of prediction:</p> <ul style="list-style-type: none"> <li>— AC_PRED_DIR = -2 =&gt; NEWVOP flag</li> <li>— AC_PRED_DIR = -1 =&gt; An un-coded block so skip inverse AC prediction</li> <li>— AC_PRED_DIR = 0 =&gt; No inverse AC prediction but use zig zag inverse scan</li> <li>— AC_PRED_DIR = 1 =&gt; Prediction form the left and use alternate vertical scan</li> <li>— AC_PRED_DIR = 2 =&gt; Prediction form the top and use alternate horizontal scan</li> </ul>	
<b>Profiles@levels supported</b>	MPEG-4 SP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
PQF_AC	AC token	
PTR	COORDINATE token	
AC_PRED_DIR	ACPRED_DIR token	
QP	QUANT token	
PREV_QP	QUANT token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
QF_AC	AC token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part2.sp.texture		

6.3.9 Algo\_IDCT2D\_ISOIEC\_23002\_1

<b>FU Name</b>	Algo_IDCT2D_ISOIEC_23002_1	
<b>Description</b>	<p>This module computes the 8×8 Inverse Discrete Cosine Transform (IDCT) defined as</p> $f(x, y) = \frac{2}{N} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} C(u)C(v)F(u, v) \cos \frac{(2x+1)u\pi}{2N} \cos \frac{(2y+1)v\pi}{2N}$ <p>with <math>u, v, x, y = 0, 1, 2, \dots, N-1</math>          where <math>x, y</math> are spatial coordinates in the sample domain  <math>u, v</math> are coordinates in the transform domain</p> $C(u), C(v) = \begin{cases} \frac{1}{\sqrt{2}} & \text{for } u, v = 0 \\ 1 & \text{otherwise} \end{cases}$ <p>It inputs a list of 64 coefficients and outputs a list of 64 decoded coefficients.</p>	
<b>Profiles@levels supported</b>	MPEG-4 SP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
IN	BLOCK token	
SIGNED	SIGN token	

Output		
Name	Token	
OUT	BLOCK token	
Parameter		
Name	Description	Range
Package		
package org.sc29.wg11.mpeg4.part2.sp.texture		

### 6.3.10 Mgnt\_DCSplit

<b>FU Name</b>	Mgnt_DCSplit	
<b>Description</b>	This module separates the DC coefficient from the AC coefficients. It takes as an input a list of 64 tokens and outputs on the DC port the DC coefficient (the first one) and the AC coefficient (the other 63) on the AC port.	
<b>Profiles@levels supported</b>	MPEG-4 SP	
Input		
Name	Token	
IN	BLOCK token	
Output		
Name	Token	
AC	AC token	
DC	DC token	
Parameter		
Name	Description	Range
Package		
package org.sc29.wg11.mpeg4.part2.sp.texture		

## 6.4 Motion compensation

### 6.4.1 Mgnt\_FB\_w\_Address\_8x8

<b>FU Name</b>	Mgnt_FB_Address_8x8	
<b>Description</b>	This module generates addresses for the frame buffer block. Write addresses (WA) are used to save the current pixels for retrieval and read addresses (RA) are used to retrieve interpolation pixel values. This module is a frame buffer. It saves data WD at address WA. It outputs the data RD located at address RA.	
<b>Profiles@levels supported</b>	MPEG-4 SP	
Input		
Name	Token	
HEIGHT	HEIGHT token	
MOT	MOTION token	
MV	MV token	
ROUND	ROUND token	
VOPMODE	BTYPE token	
WIDTH	WIDTH token	
WD	MB token	
Output		
Name	Token	

RA	MEM_ADDRESS token	
RD	MB token	
HALFPEL	DISPLACEMENT token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part2.sp.texture.motion		

### 6.4.2 Mgnt\_FB\_w\_Address\_16x16

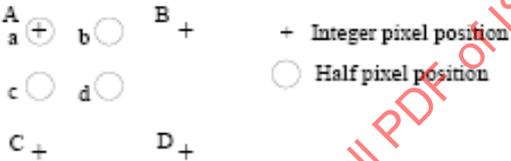
<b>FU Name</b>	Mgnt_FB_w_Address_16x16	
<b>Description</b>	This module generates addresses for the frame buffer block. Write addresses (WA) are used to save the current pixels for retrieval and read addresses (RA) are used to retrieve interpolation pixel values. This module is a frame buffer. It saves data WD at address WA. It outputs the data RD located at address RA.	
<b>Profiles@levels supported</b>	MPEG-4 SP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
HEIGHT	HEIGHT token	
MOT	MOTION token	
MV	MV token	
ROUND	ROUND token	
BTYPE	BTYPE token	
WIDTH	WIDTH token	
WD	MB token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
RA	MEM_ADDRESS token	
RD	MB token	
HALFPEL	DISPLACEMENT token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part2.sp.motion		

### 6.4.3 Algo\_PictureReconstruction\_Saturation

<b>FU Name</b>	Algo_PictureReconstruction_Saturation	
<b>Description</b>	This module adds texture pixels (TEX) and MC prediction pixels (MOT) in order to output the decoded pixels. For each ACCODED and WIDTH token inputted and according to the type of encoding of the block under consideration, the FU consumes either: <ul style="list-style-type: none"> <li>— one MB token from the MOT input if the block is only a “motion” block;</li> <li>— one MB token from the TEX input if the block is only “texture” block;</li> <li>— one MB token from TEX input and one MB token from MOT input otherwise.</li> </ul>	
<b>Profiles@levels supported</b>	MPEG-4 SP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
MOT	MB token	

TEX	MB token	
ACCODED	ACCODED token	
BTYPE	BTYPE token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
VID	MB token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part2.sp.motion		

#### 6.4.4 Algo\_Interp\_HalfpelBilinearRoundingControl

<b>FU Name</b>	Algo_Interp_HalfpelBilinearRoundingControl	
<b>Description</b>	<p>This module interpolates the pixels in case of a displacement between two frames of half a pixel. This is done according to the following scheme:</p>  <p> <math>a = A,</math>  <math>b = (A + B + 1 - \text{rounding\_control}) / 2</math>  <math>c = (A + C + 1 - \text{rounding\_control}) / 2,</math>  <math>d = (A + B + C + D + \text{rounding\_control}) / 4</math> </p>	
<b>Profiles@levels supported</b>	MPEG-4 SP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
RD	MB token	
halfpel	DISPLACEMENT token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
MOT	MB token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part2.sp.motion		

## 7 FUs for MPEG-4 AVC Constrained Baseline Profile

### 7.1 General

FUs for building MPEG-4 AVC (ISO/IEC 14496-10) Constrained Baseline Profile decoder are described in this clause.

### 7.2 Syntax parsing

#### 7.2.1 Algo\_NALU

<b>FU Name</b>	Algo_NALU	
<b>Description</b>	This module removes emulation_prevention_three_byte (0x03) and send to its output Rbsp bytes and the number of rbsp bytes between 2 NAL units.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
bits8	BIT token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
bits_rbsp8	Rbsp token	
nb_rbsp_byte	NAL_SIZE token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>

#### 7.2.2 Algo\_SynP

<b>FU Name</b>	Algo_SynP	
<b>Description</b>	This module analyses a sequence of tokens and realizes Parsing for MPEG-4 AVC syntax.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
BYTE	BIT token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
REF_REORDERING	REF_ORDER token uint (size=31)	
NB_REF_FRAME_I0	BIT token uint (size=5)	
MMCO	MMCO token uint(size=31)	
NUM_FREF_FRAME	NUM_REF_FRAME token uint(size=31)	
FRAME_NUM	FRAME_NUM token uint(size=31)	
REF_TYPE_I0	BIT token	
MAX_FRAME_NUM	BIT token uint(size=31)	
SLICE_DBF_PARAM	BIT token uint(size=6)	
CONSTRAINED_IFLAG	BIT token	
DBP_MAX_SZ	BIT token uint(size=6)	
PIC_SIZE_IN_MB	SIZE token	
POC	POC token	
MB_LOCATION	MB_ID token	
MB_TYPE	MB_TYPE token	
CBP_BLK	CBP_BLK token	
I_PCM	MB token	
SUB_MB_TYPE	SUB_MB_TYPE token	

INTRA_PRED_MODE	PRED_MODE_INTRA token	
INTRA_PRED_MODE_C	PRED_MODE_INTRA token	
RUN	RUN token	
VALUE	VALUE token	
LAST	LAST token	
QP	QUANT token	
QP_Cb	QUANT token	
QP_Cr	QUANT token	
MB_PRED_REF_IDX_10	REF_ID token	
MB_PRED_MVD_10	MVD token	
SUB_MB_PRED_REF_IDX_10	REF_ID token	
MB_PRED_MVD_10	MVD token	
WIDTH	WIDTH token	
HEIGHT	HEIGHT token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.cbp.parser		

### 7.2.3 Algo\_BlockExpand

<b>FU Name</b>	Algo_BlockExpand	
<b>Description</b>	This module decodes the DCT coefficients as elements of syntax in a MPEG-4 AVC Constrained BP conformant bitstream.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
LAST	LAST token	
RUN	RUN token	
VALUE	VALUE token	
MB_TYPE	MB_TYPE token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
BLOCK	BLOCK token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.cbp.parser		

### 7.2.4 Algo\_BlockSplit

<b>FU Name</b>	Algo_BlockSplit	
<b>Description</b>	This module splits DCT coefficient from LUMA/CHROMA into the DC and AC CHROMA/LUMA coefficients.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
BLOCK	BLOCK token	
MB_TYPE	MB_TYPE token	

Output		
Name	Token	
BLOCK_U_AC	AC token	
BLOCK_U_DC	DC token	
BLOCK_V_AC	AC token	
BLOCK_V_DC	DC token	
BLOCK_Y_AC	AC token	
BLOCK_Y_DC	DC token	
Parameter		
Name	Description	Range
Package		
package org.sc29.wg11.mpeg4.part10.cbp.parser		

### 7.2.5 Algo\_IntraPred\_Split

<b>FU Name</b>	Algo_IntraPred_Split	
<b>Description</b>	This module sends information token of the prediction type (Mb_Type for LUMA and Mb_TypeC for CHROMA) and mode (PredMode) used by the current macroblock according to MbIntraFlag and IntraPredMode Input.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
Input		
Name	Token	
MB_TYPE	MB_TYPE token	
INTRA_PRED_MODE	PRED_MODE_INTRA token	
MB_LOCATION	MB_ID token	
CONSTRAINED_IFLAG	BIT token	
Output		
Name	Token	
PRED_MODE	PRED_MODE_INTRA token	
Parameter		
Name	Description	Range
Package		
package org.sc29.wg11.mpeg4.part10.cbp.parser		

### 7.2.6 Algo\_Parser\_I\_PCM

<b>FU Name</b>	Algo_Parser_I_PCM	
<b>Description</b>	This module splits an I_PCM macroblocs into Y, U and V I_PCM blocks.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
Input		
Name	Token	
I_PCM	MB token	
Output		
Name	Token	
I_PCM_Y	MB token	
I_PCM_U	MB token	
I_PCM_V	MB token	

Parameter		
Name	Description	Range
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.cbp.parser		

### 7.2.7 Algo\_DemuxParserInfoForBlocks\_Chroma

<b>FU Name</b>	Algo_DemuxParserInfoForBlocks_Chroma	
<b>Description</b>	This module sends information for all Chroma prediction modules (intra 8×8, inter).	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
COEF_ACR	MB token	
MB_LOCATION	MB_ID token	
MB_TYPE	MB_TYPE token	
CONSTRAINED_IFLAG	BIT token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
COEF_ACR_INTRA	BLOCK token	
NEIGHBOUR_INTRA	COORDINATE token	
COEF_ACR_INTER	BLOCK token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.cbp.selectMacroblock		

### 7.2.8 Algo\_DemuxParserInfoForBlocks\_Luma

<b>FU Name</b>	Algo_DemuxParserInfoForBlocks_Luma	
<b>Description</b>	This module sends information for all Luma prediction modules (intra 4×4, intra 16×16, inter).	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
COEF_ACR	MB token	
MB_LOCATION	MB_ID token	
MB_TYPE	MB_TYPE token	
PRED_MODE	PRED_INTRA_MODE token	
CONSTRAINED_IFLAG	BIT token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
COEF_ACR_INTRA4	BLOCK token	
NEIGHBOUR_INTRA4	COORDINATE token	
PRED_MODE4	PRED_INTRA_MODE token	
COEF_ACR_INTRA16	BLOCK token	
NEIGHBOUR_INTRA16	COORDINATE token	
PRED_MODE16	PRED_INTRA_MODE token	

COEF_ACR_INTER	BLOCK token	
SELECT	SELECT token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.cbp.selectMacroblock		

### 7.3 Texture decoding

#### 7.3.1 Algo\_IS\_Zigzag\_4x4

<b>FU Name</b>	Algo_IS_Zigzag_4x4	
<b>Description</b>	This module inverts the one-dimensional array of coefficients ordered in zigzag scan to 2D raster order. It inputs a list of 16 integer coefficients (one per 4x4 block) and outputs the ordered list of integer values.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
X	BLOCK token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
Y	BLOCK token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.cbp.Residual		

#### 7.3.2 Algo\_DCR\_Hadamard\_LUMA\_IHT1d

<b>FU Name</b>	Algo_DCR_Hadamard_LUMA_IHT1d	
<b>Description</b>	This module computes 1 dimensional 4x4 Inverse Hadamard Transform for DC luminance coefficients of an intra 16x16 prediction block as a part of 2 dimensional 4x4 Inverse Hadamard Transform.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
X	DC token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
Y	DC token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.cbp.Residual		

### 7.3.3 Algo\_Transpose4x4

<b>FU Name</b>	Algo_Transpose4x4	
<b>Description</b>	This module transposes 4×4 integer array.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
X	DC token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
Y	DC token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.cbp.Residual		

### 7.3.4 Algo\_DCR\_Hadamard\_LUMA\_Reordering

<b>FU Name</b>	Algo_DCR_Hadamard_LUMA_Reordering	
<b>Description</b>	<p>This module reorders 4×4 DC luminance coefficients of an intra 16×16 prediction macroblock from raster scan order to the order of block number specified in MPEG-4 AVC specification.</p> <p>The order of block number in a MB</p> <pre> +---+---+---+---+   0   1   4   5   +---+---+---+---+   2   3   6   7   +---+---+---+---+   8   9  12  13   +---+---+---+---+  10  11  14  15   +---+---+---+---+ </pre>	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
X	DC token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
Y	DC token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.cbp.Residual		

## 7.3.5 Algo\_DCR\_Hadamard\_LUMA\_Scaling

<b>FU Name</b>	Algo_DCR_Hadamard_LUMA_Scaling	
<b>Description</b>	This module computes inverse quantization of DC luminance coefficients for intra 16×16 prediction macroblock.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
I	DC token	
QP	QUANT token	
MB_TYPE	MB_TYPE token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
O	DC token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.cbp.Residual		

## 7.3.6 Algo\_DCR\_Hadamard\_CHROMA

<b>FU Name</b>	Algo_DCR_Hadamard_CHROMA	
<b>Description</b>	This module computes 2 dimensional 2×2 Inverse Hadamard Transform and inverse quantization for DC chrominance coefficients of a macroblock.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
IN	DC token	
QP	QUANT token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
OUT	DC token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.cbp.Residual		

## 7.3.7 Algo\_IT4x4\_1d

<b>FU Name</b>	Algo_IT4x4_1d	
<b>Description</b>	This module computes 1 dimensional 4×4 Inverse Integer Transform for 4×4 block coefficients according to MPEG-4 AVC specification as a part of 2 dimensional 4×4 Inverse Integer Transform.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
X	BLOCK token	

<b>Output</b>		
<b>Name</b>	<b>Token</b>	
Y	BLOCK token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.Residual		

### 7.3.8 Algo\_IT4x4\_Addshift

<b>FU Name</b>	Algo_IT4x4_Addshift	
<b>Description</b>	This module computes right shifting of input integer value after adding value of 32 as a part of 2 dimensional 4×4 Inverse Integer Transform specified in MPEG-4 AVC specification.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
X	BLOCK token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
Y	BLOCK token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.Residual		

### 7.3.9 Algo\_IntraPred\_LUMA\_16x16

<b>FU Name</b>	Algo_IntraPred_LUMA_16x16	
<b>Description</b>	This module computes intra 16×16 predicted block for a 16×16 luminance block according to MPEG-4 AVC intra 16×16 prediction.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
Y_LEFT	COORDINATE token	
Y_UP	COORDINATE token	
Y_UP_LEFT	COORDINATE token	
AVAIL	ACKNOWLEDGMENT token	
PRED_MODE	PRED_MODE_INTRA token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
MPR	MB token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.cbp.intraPred		

7.3.10 Algo\_IntraPred\_LUMA\_4x4

<b>FU Name</b>	Algo_IntraPred_LUMA_4x4	
<b>Description</b>	This module computes intra 4×4 predicted block for a 4×4 luminance block according to MPEG-4 AVC intra 4×4 prediction.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
Y_LEFT	COORDINATE token	
Y_UP	COORDINATE token	
Y_UP_LEFT	COORDINATE token	
AVAIL	ACKNOWLEDGMENT token	
PRED_MODE	PRED_MODE_INTRA token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
MPR	MB token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.cbp.intraPred		

7.3.11 Algo\_Merge\_4x4\_to\_16x16

<b>FU Name</b>	Algo_Merge_4x4_to_16x16	
<b>Description</b>	This module merge 4×4 blocks send in rasterscan order into a 16×16 macroblocks.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
X	MB token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
Y	MB token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.cbp.Residual		

7.3.12 Algo\_IQ\_QSAndSLAndIDCTScaler\_4x4

<b>FU Name</b>	Algo_IQ_QSAndSLAndIDCTScaler_4x4	
<b>Description</b>	This module computes inverse quantization of 16 luminance 4×4 blocks and 4 chrominance 4×4 blocks for Cb and Cr components within a macroblock.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
IN	BLOCK token	
QP	QUANT token	

IS_DC	BIT token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
OUT	BLOCK token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
NB4x4	Number of block 4×4 in a block	[4; 16]
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.cbp.Residual		

### 7.3.13 Mgnt\_IQ\_INTRA16x16

<b>FU Name</b>	Mgnt_IQ_INTRA16x16	
<b>Description</b>	This module combines DC coefficients, if available, with AC coefficients within a 16×16 block.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
COEF_DC	BLOCK token	
MB_TYPE	MB_TYPE token	
COEF_AC	BLOCK token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
IS_DC	BIT token	
0	BLOCK token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.cbp.Residual		

### 7.3.14 Algo\_IntraPred\_4x4\_to\_8x8

<b>FU Name</b>	Algo_IntraPred_4x4_to_8x8	
<b>Description</b>	This module merge 4×4 blocks send in rasterscan order into a 8×8 macroblocks.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
X	MB token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
Y	MB token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.cbp.intraPred		

7.3.15 Algo\_IntraPred\_CHROMA

<b>FU Name</b>	Algo_IntraPred_CHROMA	
<b>Description</b>	This module computes an intra chroma predicted block for a chrominance block according to MPEG-4 AVC intra chroma prediction.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
C_LEFT	COORDINATE token	
C_UP	COORDINATE token	
C_UP_LEFT	COORDINATE token	
AVAIL	ACKNOWLEDGMENT token	
PredMode	PRED_MODE_INTRA token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
MPR	MB token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.cbp.intraPred		

7.3.16 Mgnt\_Intra16x16

<b>FU Name</b>	Mgnt_Intra_16x16	
<b>Description</b>	This module activates and sends the value of pixels needed for a full macrobloc intra prediction to output Y_LEFT, Y_UP, Y_UP_LEFT and the available edge for intra prediction to ouput. Values of needed pixels are received from EDGE input.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
EDGE	MB token	
NEIGHBOUR	BIT token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
AVAIL	BIT token	
Y_LEFT	COORDINATE token	
Y_UP	COORDINATE token	
Y_UP_LEFT	COORDINATE token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
MB_WIDTH	Size in pixel of macroblocks	[8;16]
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.cbp.intraPred		

## 7.3.17 Mgnt\_Intra4x4

<b>FU Name</b>	Mgnt_Intra_4x4	
<b>Description</b>	This module activates and sends the value of pixels needed for a 4×4 macroblock intra prediction to output Y_LEFT4, Y_UP4, Y_UP_LEFT4, the available edge for intra prediction to output AVAIL and the prediction mode to output PRED_MODE4. Values of needed pixels are received from EDGE for neighbouring 16×16 pixels and MB_4x4 input for neighbouring 4×4 pixels.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
NEIGHBOUR	BIT token	
EDGE	MB token	
MB_4X4	MB token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
AVAIL	BIT token	
Y_LEFT4	COORDINATE token	
Y_UP4	COORDINATE token	
Y_UP_LEFT4	COORDINATE token	
PRED_MODE4	PRED_MODE_INTRA token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.cbp.intraPred		

## 7.3.18 Mgnt\_IQ\_Chroma

<b>FU Name</b>	Mgnt_IQ_Chroma	
<b>Description</b>	This module adds the DC coefficient to the AC coefficients.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
COEF_DC	BLOCK token	
COEF_AC	BLOCK token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
O	BLOCK token	
IS_DC	BIT token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.cbp.intraPred		

## 7.3.19 Mgnt\_Buffer\_Neighbour\_FullMb

<b>FU Name</b>	Mgnt_Buffer_Neighbour_FullMb	
<b>Description</b>	This module stores bottom and right edge from macroblock receive in input MB_IN and	

	send to EDGE according to CurrMbAddr and Mb_Type when full macroblock intra-prediction is needed.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
MB_LOCATION	MB_ID token	
MB_IN	MB token	
MB_TYPE	MB_TYPE token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
EDGE	MB token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
IsChroma	If this FU operates on chrominance	boolean
MB_WIDTH	Size in pixel of macroblocks	[8;16]
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.cbp.intraPred		

7.3.20 Mgnt\_Buffer\_Neighbour\_YxY

<b>FU Name</b>	Mgnt_Buffer_Neighbour_YxY	
<b>Description</b>	This module stores bottom and right egde from macrobloc receive in input MB_IN and send to EDGE according to CurrMbAddr and Mb_Type when 4x4 macroblock intra-prediction is needed.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
MB_LOCATION	MB_ID token	
MB_IN	MB token	
MB_TYPE	MB_TYPE token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
EDGE	MB token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
IsIntra4x4	If this FU is used for intra 4x4 blocks	boolean
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.cbp.intraPred		

7.3.21 Algo\_Merge\_4x4\_to\_16x16\_norasterscan

<b>FU Name</b>	Algo_Merge_4x4_to_16x16_norasterscan	
<b>Description</b>	This module merges 4x4 blocks send in norasterscan order into a 16x16 macroblocks.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
X	MB token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	

Y	MB token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.cbp.intraPred		

### 7.3.22 Algo\_Split\_16x16\_to\_4x4\_norasterscan

<b>FU Name</b>	Algo_Split_16x16_to_4x4_norasterscan	
<b>Description</b>	This module splits 16×16 blocks send in non rasterscan order into 4×4 blocks.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
X	MB token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
Y	MB token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.cbp.intraPred		

## 7.4 Motion compensation

### 7.4.1 Algo\_Interp\_EighthPelBilinear

<b>FU Name</b>	Algo_Interp_Bilinear	
<b>Description</b>	This FU performs fractional chroma sample interpolation with bilinear filter. Interpolating a sample in a fractional position requires at most 2×2 integer samples around the interpolated location.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
RD	MEM_DATA token	
PARTSZ	PART_SIZE token	
MV	MV token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
INTERP	MB token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.cbp.interPred		

7.4.2 Algo\_Interp\_SeparableSixTapQuarterPel

<b>FU Name</b>	Algo_Interp_SeparableSixTapQuarterPel	
<b>Description</b>	This FU performs fractional luma sample interpolation with separable 6-tap FIR linear phase filter. Interpolating a sample in a fractional position requires at most 6x6 integer samples around the interpolated location:	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
RD	MEM_DATA token	
PARTSZ	PART_SIZE token	
MV	MV token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
INTERP	MB token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.cbp.interPred		

7.4.3 Algo\_Interp\_Reord

<b>FU Name</b>	Algo_Interp_Reord	
<b>Description</b>	This module reconstructs an inter prediction macroblock from the partition of inter prediction.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
INTERP	MB token	
MB_TYPE	MB_TYPE token	
SUB_MB_TYPE	SUB_MB_TYPE token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
MBPRED	MB token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
SzSidePerPart	If CHROMA then 2 else /*LUMA*/ 4 end	[2; 4]
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.cbp.interPred		

## 7.4.4 Algo\_MvLXReconstr

<b>FU Name</b>	Algo_MvLXReconstr	
<b>Description</b>	This FU finds a motion vector predictor by using reference index and MV of neighbouring blocks (left, top, top right). This process is adaptively switched according to the partition size of the current block and availability of the neighbouring blocks. It also adds the residual motion vector.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
MB_LOCATION	MB_ID token	
MB_TYPE	MB_TYPE token	
SUB_MB_TYPE	SUB_MB_TYPE token	
MB_PRED_REF_IDX	REF_ID token	
SUB_MB_PRED_REF_IDX	REF_ID token	
MB_PRED_MVD	MVD token	
SUB_MB_PRED_MVD	MVD token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
MV_OUT	MV token	
LOCATION	MB_ID token	
IS_NEW_MB_PIC	BIT token	
IS_NEW_MB_SLICE	BIT token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
IsPredL0	Only PredL0 if true else PredL1 or PredL0	boolean
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.cbp.interPred		

## 7.4.5 Mgnt\_DPB

<b>FU Name</b>	Mgnt_DPB	
<b>Description</b>	This module stores decoded pictures input from WD port according to the macroblock address (CurrMbAddr), and sends parts of a selected stored picture output through RD port for inter prediction, according to the prediction type (Mb_Type), current position of the macroblock (CurrMbAddr), a motion vecteur (MV) and an id of frame (RefIdx). This module also reorganizes and erases the picture stored into its internal memory according to RefList input.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
PIC_SIZE_IN_MB	SIZE token	
MB_TYPE	MB_TYPE token	
FRAME_NUM	FRAME_NUM token	
FRAME_TO_READ	FRAME_TO_READ token	
DEL_LIST	DEL_LIST token	
ENABLE_READ	BIT token	
WD	MB token	
RA	RA token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
RD	MEM_DATA token	
<b>Parameter</b>		

Name	Description	Range
MB_WIDTH	Width and Height in pixel of a macroblock	[8;16]
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.cbp.interPred		

#### 7.4.6 Algo\_MMCO

<b>FU Name</b>	Algo_MMCO	
<b>Description</b>	This modules marks the index of frames to store as long terme reference, short terme reference, to delete or reorganize in memory. RefReordering indicates whether or not a reordering of the frame stored in memory is necessary.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
MMCO	MMCO token	
FRAME_NUM	FRAM_NUM token	
MAX_FRAME_NUM	MAX_FRAME_NUM token	
NUM_REF_FRAME	NUM_REF_FRAME token	
POC	POC token	
MB_TYPE	MB_TYPE token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
DEL_LIST	DEL_LIST token	
LT_LIST	LT_LIST token	
ST_LIST	ST_LIST token	
ST_LIST_SZ	ST_LIST_SZ token	
EMPTY_DBP	BIT token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.cbp.interPred		

#### 7.4.7 AlgoRefList

<b>FU Name</b>	Algo_RefList	
<b>Description</b>	This FU manages short term and long term reference lists of AVC. This FU is managed by Algo_MMCO.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
LT_LIST	LT_LIST token	
ST_LIST	ST_LIST token	
ST_LIST_SZ	ST_LIST_SZ token	
NB_REF_FRAME	NB_REF_FRAME token	
POC	POC token	
REF_TYPE_LIST	REF_TYPE_LIST token	
EMPTY_DPB	BIT token	
FRAME_NUM	FRAM_NUM token	

MAX_FRAME_NUM	MAX_FRAME_NUM token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
REF_LIST	REF_LIST token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.cbp.interPred		

#### 7.4.8 Mgnt\_InterPred

<b>FU Name</b>	Mgnt_InterPred	
<b>Description</b>	This FU issues an address and a reference frame ID to the luma frame buffer to fetch necessary data for the luma fractional sample interpolation. It also notifies the luma fractional sample interpolation FU which mode is selected via Frac.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
MV	MV token	
LOCATION	MB_ID token	
PARTSZ	PART_SIZE token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
ADDR_L	MEM_ADDRESS token	
ADDR_C	MEM_ADDRESS token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.cbp.interPred		

#### 7.4.9 Algo\_RefIdxtoFrameNum

<b>FU Name</b>	Algo_RefIdxToFrameNum	
<b>Description</b>	This module computes the frame to read for each MV.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
REF_LIST	REF_LIST token	
REF_IDX	REF_ID token	
IS_NEW_MB	BIT token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
READ_FRAME	FRAME_TO_READ token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.cbp.interPred		

## 7.5 Filtering

### 7.5.1 Mgnt\_DBF\_AdaptiveFilter

<b>FU Name</b>		Mgnt_DBF_AdaptiveFilter
<b>Description</b>		This FU implements deblocking filter without supporting MBAFF
<b>Profiles@levels supported</b>		MPEG-4 AVC Constrained BP
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
CBP_BLK	CBP_BLK token	
MB_LOCATION	MEM_WIDTH token	
SLICE_DBF_PARAM	SLICE_DBF_PARAM token	
MB_TYPE	MB_TYPE token	
MV	MV token	
QP_Y	QUANT token	
QP_Cb	QUANT token	
QP_Cr	QUANT token	
READ_FRAME	REF_ID token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
BS	BS token	
MB_DBF_PARAM_Y	MB_DBF_PARAM token	
MB_DBF_PARAM_U	MB_DBF_PARAM token	
MB_DBF_PARAM_V	MB_DBF_PARAM token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.cbp.deblockingFilter		

### 7.5.2 Algo\_DBF\_AdaptiveFilter

<b>FU Name</b>		Algo_DBF_AdaptiveFilter
<b>Description</b>		This FU implements deblocking filter.
<b>Profiles@levels supported</b>		MPEG-4 AVC Constrained BP
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
MB_TYPE	MB_TYPE token	
BS	BS token	
MB_LOCATION	MB_ID token	
PIC_SIZE_IN_MB	MEM_HEIGHT token	
MB_DBF_PARAM	MB_DBF_PARAM token	
CBP_BLK	CBP_BLK token	
MB_IN	MB token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
MB_OUT	MB token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
MbWidth	Size in pixel of a macroblock	[8;16]
ChromaEdgeFlag	LUMA = false, CHROMA=true	bool

<b>Package</b>
package org.sc29.wg11.mpeg4.part10.cbp.deblockingFilter

### 7.5.3 Algo\_MvComponentReorder

<b>FU Name</b>	Algo_MvComponentReorder	
<b>Description</b>	This FU reorganizes Motion vector for the deblocking filter	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
MB_TYPE	MB_TYPE token	
MV	MV token	
SUB_MB_TYPE	SUB_MB_TYPE token	
READ_FRAME	REF_ID token	
<b>Output</b>		

## 7.6 Renderer

### 7.6.1 Mgnt\_POC

<b>FU Name</b>	Mgnt_POC	
<b>Description</b>	This module select the order of the frame to display according to POC input.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
MB_TYPE	MB_TYPE token	
POC	POC token	
DBP_MAX_SZ	DBP_MAX_SZ token	
EMPTY_DPB	BIT token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
POC_DISPLAY	POC token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.cbp.display		

### 7.6.2 Mgnt\_BufferRender

<b>FU Name</b>	Mgnt_BufferRender	
<b>Description</b>	Based on the POC, this module will buffer and display the picture in the right order.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
MB_TYPE	MB_TYPE token	
POC	POC token	
POC_DISPLAY	POC token	
WD	MB token	

<b>Output</b>		
<b>Name</b>	<b>Token</b>	
0	MB token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
MbWidth	Size in pixel of a macroblock	[8;16]
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.cbp.display		

### 7.6.3 Mgnt\_Merger420\_AVC

<b>FU Name</b>	Mgnt_Merger420_AVC	
<b>Description</b>	This module merges data from YUV to compose a MB 16×16.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Constrained BP	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
Y	BLOCK token	
U	BLOCK token	
V	BLOCK token	
Select	SELECT token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
YUV	MB token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.cbp.display		

## 8 FUs for MPEG-4 AVC Progressive High Profile

### 8.1 General

#### 8.1.1 Overview

FUs for building MPEG-4 AVC High Profile decoder are described in this clause.

#### 8.1.2 Algo\_SynP

<b>FU Name</b>	Algo_SynP	
<b>Description</b>	This module analyses a sequence of tokens and realizes Parsing for MPEG-4 AVC syntax.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Progressive High Profile	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
BYTE	BIT token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
MMCO	MMCO token uint(size=31)	
NUM_FREF_FRAME	NUM_REF_FRAME token uint(size=31)	
FRAME_NUM	FRAME_NUM token uint(size=31)	

NB_REF_FRAME_I0	BIT token uint(size=5)	
REF_TYPE_I0	BIT token	
REF_REORDERING_I0	REF_ORDER token uint(size=31)	
NB_REF_FRAME_I1	BIT token uint(size=5)	
REF_TYPE_I1	BIT token	
REF_REORDERING_I1	REF_ORDER token uint(size=31)	
MAX_FRAME_NUM	BIT token uint(size=31)	
SLICE_DBF_PARAM	BIT token uint(size=6)	
CONSTRAINED_IFLAG	BIT token	
DBP_MAX_SZ	BIT token uint(size=6)	
PIC_SIZE_IN_MB	SIZE token	
POC	POC token	
SCALING_LIST_Y	SCALING_LIST token	
SCALING_LIST_U	SCALING_LIST token	
SCALING_LIST_V	SCALING_LIST token	
MB_LOCATION	MB_ID token	
MB_TYPE	MB_TYPE token	
CBP_BLK	CBP_BLK token	
I_PCM	MB token	
SUB_MB_TYPE	SUB_MB_TYPE token	
INTRA_PRED_MODE	PRED_MODE_INTRA token	
INTRA_PRED_MODE_C	PRED_MODE_INTRA token	
RUN	RUN token	
VALUE	VALUE token	
LAST	LAST token	
QP	QUANT token	
QP_Cb	QUANT token	
QP_Cr	QUANT token	
MB_PRED_REF_IDX_I0	REF_ID token	
MB_PRED_MVD_I0	MVD token	
SUB_MB_PRED_REF_IDX_I0	REF_ID token	
MB_PRED_MVD_I0	MVD token	
MB_PRED_REF_IDX_I1	REF_ID token	
MB_PRED_MVD_I1	MVD token	
SUB_MB_PRED_REF_IDX_I1	REF_ID token	
MB_PRED_MVD_I1	MVD token	
DIRECT_SPATIAL_MV_PRED_FLAG	BIT token	
TRANSFORM_SIZE_8x8_FLAG	BIT token	
DIRECT_8X8_INTERFERENCE_FLAG	BIT token	
WEIGHTED_PRED_IDC	BIT token uint(size=2)	
WP_PARAM	BIT token uint(size=9)	
WIDTH	WIDTH token	
HEIGHT	HEIGHT token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.php.parser		

### 8.1.3 Algo\_BlockExpand

<b>FU Name</b>	Algo_BlockExpand	
<b>Description</b>	This module decodes the DCT coefficients as elements of syntax in a MPEG-4 AVC Progressive High Profile conforming bitstream (conforming with CALVC and CABAC).	
<b>Profiles@levels supported</b>	MPEG-4 AVC Progressive High Profile	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
LAST	LAST token	
RUN	RUN token	
VALUE	VALUE token	
MB_TYPE	MB_TYPE token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
BLOCK	BLOCK token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.php.parser		

### 8.1.4 Algo\_DemuxParserInfoForBlocks\_Luma

<b>FU Name</b>	Algo_DemuxParserInfoForBlocks_Luma	
<b>Description</b>	This module adds information to package org.sc29.wg11.mpeg4.part10.php.selectMacroblock. Algo_DemuxParserInfoForBlocks_Luma by adding support for intra 8×8 decoding blocks.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Progressive High Profile	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
COEF_ACR	MB token	
MB_LOCATION	MB_ID token	
MB_TYPE	MB_TYPE token	
PRED_MODE	PRED_INTRA_MODE token	
CONSTRAINED_IFLAG	BIT token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
COEF_ACR_INTRA4	BLOCK token	
NEIGHBOUR_INTRA4	COORDINATE token	
PRED_MODE4	PRED_INTRA_MODE token	
COEF_ACR_INTRA16	BLOCK token	
NEIGHBOUR_INTRA16	COORDINATE token	
PRED_MODE16	PRED_INTRA_MODE token	
COEF_ACR_INTRA8	BLOCK token	
NEIGHBOUR_INTRA8	COORDINATE token	
PRED_MODE8	PRED_INTRA_MODE token	
COEF_ACR_INTER	BLOCK token	
SELECT	SELECT token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>

<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.php.selectMacroblock		

## 8.2 Texture decoding

### 8.2.1 Algo\_IS\_Zigzag\_8x8

<b>FU Name</b>	Algo_IS_Zigzag_8x8	
<b>Description</b>	This module computes inverse zigzag of an 8×8 block.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Progressive High Profile	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
X	BLOCK token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
Y	BLOCK token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.php.Residual		

### 8.2.2 Algo\_IQ\_QSAndSLAndIDCTScaler\_8x8

<b>FU Name</b>	Algo_IQ_QSAndSLAndIDCTScaler_8x8	
<b>Description</b>	This module computes inverse quantization of 8×8 blocks with scaling list (weight scale) following by scaling of the 8×8 Inverse Integer Cosine Transform.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Progressive High Profile	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
IN	BLOCK token	
QP	QUANT token	
MB_TYPE	MB_TYPE token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
OUT	BLOCK token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.php.Residual		

### 8.2.3 Algo\_IIT\_8x8

<b>FU Name</b>	Algo_IIT_8x8	
<b>Description</b>	This module computes 2D IIT (Inverse Integer Transform) of 8×8 blocks. 2D IIT is achieved by row column decomposition of the coefficients.	

<b>Profiles@levels supported</b>	MPEG-4 AVC Progressive High Profile	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
X	BLOCK token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
Y	BLOCK token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.php.Residual		

### 8.2.4 Algo\_IntraPred\_LUMA\_8x8

<b>FU Name</b>	Algo_IntraPred_LUMA_8x8	
<b>Description</b>	This module computes intra 8x8 predicted block for a 8x8 luma block according to MPEG-4 AVC intra 8x8 prediction.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Progressive High Profile	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
Y_LEFT	COORDINATE token	
Y_UP	COORDINATE token	
Y_UP_LEFT	COORDINATE token	
Y_UP_RIGHT	COORDINATE token	
AVAIL	ACKNOWLEDGMENT token	
PRED_MODE	PRED_MODE_INTRA token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
MPR	MB token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.php.intraPred		

### 8.2.5 Mgnt\_Intra\_8x8

<b>FU Name</b>	Mgnt_Intra_8x8	
<b>Description</b>	This module activates and sends samples needed for a 8x8 Macroblock intra prediction to output Y_LEFT8, Y_UP8, Y_UP_LEFT8, Y_UP_RIGHT8, the available edge for intra prediction to output AVAIL and the prediction mode to output PRED_MODE4. Values of needed samples are received from EDGE for neighbouring 16x16 block samples and MB_8x8 input for neighbouring 8x8 block samples.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Progressive High Profile	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
NEIGHBOUR	BIT token	
EDGE	MB token	

MB_8X8	MB token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
AVAIL	BIT token	
Y_LEFT8	COORDINATE token	
Y_UP8	COORDINATE token	
Y_UP_LEFT8	COORDINATE token	
Y_UP_RIGHT8	COORDINATE token	
PRED_MODE8	PRED_MODE_INTRA token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.php.intraPred		

### 8.2.6 Algo\_Merge\_8x8\_to\_16x16

<b>FU Name</b>	Algo_Merge_8x8_to_16x16	
<b>Description</b>	This module merges 8×8 blocks send in raster scan order into a 16×16 block.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Progressive High Profile	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
X	MB token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
Y	MB token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.php.Residual		

### 8.2.7 Algo\_DCR\_Hadamard\_CHROMA

<b>FU Name</b>	Algo_DCR_Hadamard_CHROMA	
<b>Description</b>	This module overrides FU from Constrained Baseline Profile to support scaling list in Progressive High Profile.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Progressive High Profile	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
IN	DC token	
QP	QUANT token	
MB_TYPE	MB_TYPE token	
SCALING_LIST	SCALING_LIST token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
OUT	DC token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>

<b>Package</b>
package org.sc29.wg11.mpeg4.part10.php.Residual

### 8.2.8 Algo\_DCR\_Hadamard\_LUMA\_Scaling

<b>FU Name</b>	Algo_DCR_Hadamard_LUMA_Scaling	
<b>Description</b>	This module adds the support of scaling list to org.sc29.wg11.mpeg4.part10.php.Residual.Algo_DCR_Hadamard_LUMA_Scaling.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Progressive High Profile	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
I	DC token	
QP	QUANT token	
MB_TYPE	MB_TYPE token	
SCALING_LIST	SCALING_LIST token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
O	DC token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.php.Residual		

### 8.2.9 Algo\_IQ\_QSAndSLAndIDCTScaler\_4x4

<b>FU Name</b>	Algo_IQ_QSAndSLAndIDCTScaler_4x4	
<b>Description</b>	This module adds the support of non default scaling list to org.sc29.wg11.mpeg4.part10.php.Residual.Algo_IQ_QSAndSLAndIDCTScaler_4x4.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Progressive High Profile	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
IN	BLOCK token	
QP	QUANT token	
IS_DC	BIT token	
MB_TYPE	MB_TYPE token	
SCALING_LIST	SCALING_LIST token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
OUT	BLOCK token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
NB4x4	Number of block 4x4 in a block	[4; 16]
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.php.Residual		

### 8.2.10 Algo\_Merge\_8x8\_to\_16x16\_norasterscan

<b>FU Name</b>	Algo_Merge_8x8_to_16x16_norasterscan	
<b>Description</b>	This module merges 8×8 blocks sent in non raster scan order into a 16×16 block.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Progressive High Profile	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
X	MB token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
Y	MB token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.php.intraPred		

### 8.2.11 Algo\_Split\_16x16\_to\_8x8\_norasterscan

<b>FU Name</b>	Algo_Split_16x16_to_8x8_norasterscan	
<b>Description</b>	This module splits a 16x16 block sent in non-raster scan order into 8×8 blocks.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Progressive High Profile	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
X	MB token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
Y	MB token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.php.intraPred		

### 8.2.12 Mgnt\_I4x4\_I8x8\_demux

<b>FU Name</b>	Mgnt_I4x4_I8x8_demux	
<b>Description</b>	This module splits the information to compute the residual of a 4×4 block or a 8×8 block.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Progressive High Profile	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
COEF_AC	MB token	
MB_TYPE	MB_TYPE token	
QP	QUANT token	
SCALING_LIST	SCALING_LIST token	
TRANSFORM_SIZE_8x8_FLAG	BIT token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
COEF_AC_4x4	MB token	
MB_TYPE_4x4	MB_TYPE token	

QP_4x4	QUANT token	
SCALING_LIST_4x4	SCALING_LIST token	
COEF_AC_8x8	MB token	
MB_TYPE_8x8	MB_TYPE token	
QP_8x8	QUANT token	
SCALING_LIST_8x8	SCALING_LIST token	
TRANSFORM_SIZE_8x8_FLAG_0	BIT token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.php.intraPred		

### 8.2.13 Mgnt\_I4x4\_I8x8\_mux

<b>FU Name</b>	Mgnt_I4x4_I8x8_mux	
<b>Description</b>	This module reorders the information with the residual coming from an 4x4 block or a 8x8 block.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Progressive High Profile	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
COEF_AC_4x4	MB token	
COEF_AC_8x8	MB token	
TRANSFORM_SIZE_8x8_FLAG	BIT token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
COEF_AVC	MB token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.php.intraPred		

## 8.3 Motion compensation

### 8.3.1 Algo\_GeneratePredWeight

<b>FU Name</b>	Algo_GeneratePredWeight
<b>Description</b>	This FU performs weighted sum of two block partitions if necessary.
<b>Profiles@levels supported</b>	MPEG-4 AVC Progressive High Profile
<b>Input</b>	
<b>Name</b>	<b>Token</b>
IS_NEW_MB_L0	BIT token
IS_NEW_MB_L1	BIT token
POC_LISTX	POC_LIST token
POC_IS_LT	BIT token
REF_IDX_L0	REF_ID token
REF_IDX_L1	REF_ID token
SELECT_LIST	BIT token
WP_PRED_IDC	BIT token uint(size=2)
WP_PARAM	BIT token uint(size=9)
<b>Output</b>	

Name	Token	
WP_PARAM_Y	BIT token uint(size=9)	
WP_PARAM_U	BIT token uint(size=9)	
WP_PARAM_V	BIT token uint(size=9)	
<b>Parameter</b>		
Name	Description	Range
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.php.interPred		

### 8.3.2 Mgnt\_SelectMvpLX

<b>FU Name</b>	Mgnt_SelectMvpLX	
<b>Description</b>	This FU computes MV for each list.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Progressive High Profile	
<b>Input</b>		
Name	Token	
MB_TYPE	MB_TYPE token	
SUB_MB_TYPE	SUB_MB_TYPE token	
MV_L0	MVD token	
LOCATION_L0	MB_ID token	
FRAME_L0	FRAME_TO_READ token	
MV_L1	MVD token	
FRAME_L1	FRAME_TO_READ token	
LOCATION_L1	MB_ID token	
DIRECT_PRED_L0	BIT token	
DIRECT_PRED_L1	BIT token	
REF_IDX_L0	REF_ID token	
REF_IDX_L1	REF_ID token	
<b>Output</b>		
Name	Token	
MV	MVD token	
READ_FRAME		
LOCATION		
SELECT_LIST		
REF_IDX	MVD token	
<b>Parameter</b>		
Name	Description	Range
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.php.interPred		

### 8.3.3 Algo\_MvLXReconstr

<b>FU Name</b>	Algo_MvLXReconstr	
<b>Description</b>	This FU finds a motion vector predictor by using reference index and MV of neighbouring blocks (left, top, top right). This process is adaptively switched according to the partition size of the current block and availability of the neighbouring blocks. It also adds the residual motion vector.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Progressvie High Profile	
<b>Input</b>		
Name	Token	
MB_LOCATION	MB_ID token	
MB_TYPE	MB_TYPE token	

SUB_MB_TYPE	SUB_MB_TYPE token	
MB_PRED_REF_IDX	REF_ID token	
SUB_MB_PRED_REF_IDX	REF_ID token	
MB_PRED_MVD	MVD token	
SUB_MB_PRED_MVD	MVD token	
COL_ZERO_FLAG	BIT token	
TEMPORAL_PRED_INFO	BIT token int(size=32)	
DIRECT_SPATIAL_MV_FLAG	BIT token	
DIRECT_PRED_LX_I	BIT token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
MV_OUT	MV token	
LOCATION	MB_ID token	
IS_NEW_MB_PIC	BIT token	
IS_NEW_MB_SLICE	BIT token	
DIRECT_PRED_LX_O	BIT token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
IsPredL0	Only PredL0 if true else PredL1 or PredL0	boolean
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.php.interPred		

### 8.3.4 Algo\_MvBuffer

<b>FU Name</b>	Algo_MvBuffer	
<b>Description</b>	This FU manages motion vectors for an entire frame when List1 is used.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Progressvie High Profile	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
MB_LOCATION	MB_ID token	
MB_TYPE	MB_TYPE token	
SUB_MB_TYPE	SUB_MB_TYPE token	
REF_IDX	REF_ID token	
MV	MVD token	
DIRECT8x8_INFERENCE_FLAG	BIT token	
DIRECT_SPATIAL_MV_FLAG	BIT token	
EMPTY_DPB	BIT token	
FRAME_NUM	FRAME_NUM token	
POC	POC token	
DEL_LIST	DEF_LIST token	
LT	LT_LIST token	
REF_LIST0	REF_LIST token	
REF_LIST1	REF_LIST token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
COL_ZERO_FLAG	BIT token	
TEMPORAL_PRED_INFO	BIT token int (size=32)	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
IsPredL0	Only PredL0 if true else PredL1 or PredL0	boolean
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.php.interPred		

### 8.3.5 Mgnt\_SelectMvpLX

<b>FU Name</b>	Mgnt_SelectReflDx	
<b>Description</b>	This FU selects the frame to read depending on the list.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Progressvie High Profile	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
SELECT_LIST	BIT token	
READ_FRAME_L0	READ_FRAME token	
READ_FRAME_L1	READ_FRAME token	
IS_NEW_MB_L0	BIT token	
IS_NEW_MB_L1	BIT token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
READ_FRAME	READ_FRAME token	
READ_4MB_PRED	BIT token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.php.interPred		

### 8.3.6 Algo\_FrameNumToPocList

<b>FU Name</b>	Algo_FrameNumToPocList	
<b>Description</b>	This FU generates a list of POC for the weighted prediction FU.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Progressvie High Profile	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
DEL_LIST	DEL_LIST token	
EMPTY_DPB	BIT token	
FRAME_NUM	FRAME_NUM token	
MB_TYPE	MB_TYPE token	
LT	LT_LIST token	
POC	POC token	
REF_LIST0	REF_LIST token	
REF_LIST1	REF_LIST token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
POC_LISTX	POC token	
POC_IS_LT	BIT token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.php.interPred		

## 8.4 Filtering

### 8.4.1 Algo\_DBF\_AdaptiveFilter

<b>FU Name</b>	Algo_DBF_AdaptiveFilter
<b>Description</b>	This FU replaces deblocking filter from org.sc29.wg11.mpeg4.part10.cbp.deblockingFilter. Algo_DBF_AdaptiveFilter

	in order to support deblocking of 8×8 blocks.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Progressive High Profile	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
MB_TYPE	MB_TYPE token	
BS	BS token	
MB_LOCATION	MB_ID token	
PIC_SIZE_IN_MB	MEM_HEIGHT token	
MB_DBF_PARAM	MB_DBF_PARAM token	
CBP_BLK	CBP_BLK token	
MB_IN	MB token	
TRANSFORM_SIZE_8x8_FLAG	BIT token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
MB_OUT	MB token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
MB_WIDTH	Size in samples of a Macroblock width	[8;16]
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.php.deblockingFilter		

### 8.4.2 Algo\_MvComponentReorder

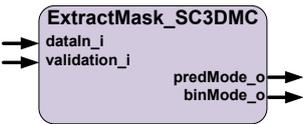
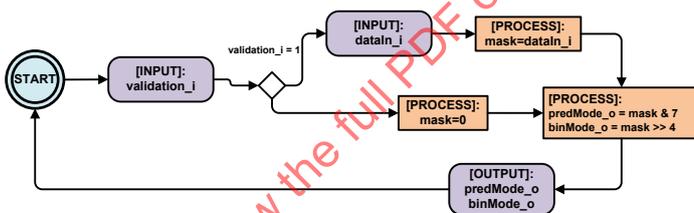
<b>FU Name</b>	Algo_MvComponentReorder	
<b>Description</b>	This FU reorganizes motion vectors for the deblocking filter.	
<b>Profiles@levels supported</b>	MPEG-4 AVC Progressive High Profile	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
MB_TYPE	MB_TYPE token	
SUB_MB_TYPE	SUB_MB_TYPE token	
MV	MV token	
SUB_MB_TYPE	SUB_MB_TYPE token	
READ_FRAME	REF_ID token	
SELECT_LIST	BIT token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
MV	MV token	
READ_FRAME_DBF	REF_ID token	
PARTSZ	PART_SIZE token	
REF_IDX_TAB	REF_IDX token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part10.php.deblockingFilter		

## 9 MPEG-4 Part 16 SC3DMC decoder specific FUs

### 9.1 General

The specific FUs for building MPEG-4 Part 16 (ISO/IEC 14496-16) SC3DMC decoder are described in this clause.

### 9.2 Algo\_ExtractMask\_SC3DMC

FU Name	Algo_ExtractMask_SC3DMC																																										
<p><b>Description</b></p>	<div style="display: flex; justify-content: space-around; align-items: flex-start;"> <div style="text-align: center;">  </div> <table border="1" style="border-collapse: collapse; text-align: center;"> <thead> <tr style="background-color: #4a86e8; color: white;"> <th>Port Name</th> <th>Direction (I/O)</th> <th>Token RANGE</th> </tr> </thead> <tbody> <tr> <td>dataIn_i</td> <td>I</td> <td>UINT_8</td> </tr> <tr> <td>validation_i</td> <td>I</td> <td>BOOLEAN</td> </tr> <tr> <td>predMode_o</td> <td>O</td> <td>UINT_8</td> </tr> <tr> <td>binMode_o</td> <td>O</td> <td>UINT_8</td> </tr> </tbody> </table> </div> <p style="text-align: center; margin-top: 10px;">Algo_ExtractMask_SC3DMC Schematic:</p> <div style="text-align: center;">  </div> <p style="margin-top: 20px;">Algo_ExtractMask_SC3DMC Process:</p> <pre> START: INPUT: validation_i IF validation_i = 1 INPUT: dataIn_i PROCESS: mask = dataIn_i; ELSE PROCESS: mask = 0 PROCESS: predMode_o = mask &amp; 7; binMode_o = mask &gt;&gt; 4; OUTPUT: predMode_o binMode_o GOTO START     </pre> <p>NOTE:</p> <p>SC3DMC mask description:</p> <table border="1" style="border-collapse: collapse; text-align: center; width: 100%;"> <thead> <tr> <th>Bit position</th> <th>7</th> <th>6</th> <th>5</th> <th>4</th> <th>3</th> <th>2</th> <th>1</th> <th>0</th> </tr> </thead> <tbody> <tr> <td></td> <td>bit</td> <td>bit</td> <td>bit</td> <td>bit</td> <td>bit</td> <td>bit</td> <td>bit</td> <td>bit</td> </tr> <tr> <td>Flag</td> <td colspan="4">Binarization Mode</td> <td>x</td> <td colspan="3">Prediction Mode</td> </tr> </tbody> </table>	Port Name	Direction (I/O)	Token RANGE	dataIn_i	I	UINT_8	validation_i	I	BOOLEAN	predMode_o	O	UINT_8	binMode_o	O	UINT_8	Bit position	7	6	5	4	3	2	1	0		bit	Flag	Binarization Mode				x	Prediction Mode									
	Port Name	Direction (I/O)	Token RANGE																																								
dataIn_i	I	UINT_8																																									
validation_i	I	BOOLEAN																																									
predMode_o	O	UINT_8																																									
binMode_o	O	UINT_8																																									
Bit position	7	6	5	4	3	2	1	0																																			
	bit	bit	bit	bit	bit	bit	bit	bit																																			
Flag	Binarization Mode				x	Prediction Mode																																					
<p><b>ISO Standards using the FU</b></p>	<p>ISO/IEC 14496-16</p>																																										

<b>Profiles@levels supported</b>	MPEG-4 Part 16 SC3DMC	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
dataIn_i	UINT_8	
validation_i	BOOLEAN	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
predMode_o	UINT_8	
binMode_o	UINT_8	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part16		

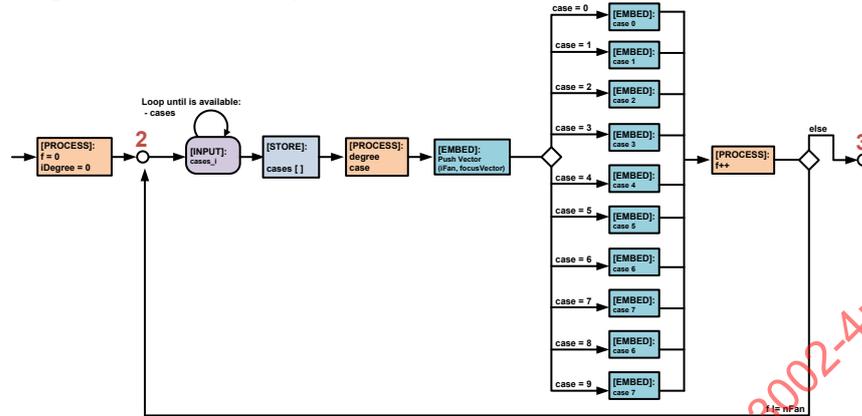
### 9.3 MPEG-4 SC3DMC TFAN Specific FUs

#### 9.3.1 Algo\_DecodeConnectivity\_TFAN

<b>FU Name</b>	Algo_DecodeConnectivity_TFAN																								
<b>Description</b>	<div style="display: flex; justify-content: space-between;"> <div style="width: 30%;"> </div> <div style="width: 65%;"> <table border="1"> <thead> <tr> <th>Port Name</th> <th>Direction (I/O)</th> <th>Token RANGE</th> </tr> </thead> <tbody> <tr> <td>nFans_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>fans_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>degrees_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>cases_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>ops_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>vertices_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>dataOut_o</td> <td>O</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> </tbody> </table> </div> </div> <p><b>Decode Connectivity TFAN FSM:</b></p> <p><b>Get Number of FANS and Max TFANS dim:</b></p> <p><b>Get all constricted vertices adjacent to the focused vertex:</b></p> <p><b>Sort the L vector:</b> The implementation of any algorithm that is able to sort an array.</p> <p><b>Eliminate the duplicates in the L vector:</b> The implementation of any algorithm that is able</p>	Port Name	Direction (I/O)	Token RANGE	nFans_i	I	UINT8, UINT16, UINT32, UINT64	fans_i	I	UINT8, UINT16, UINT32, UINT64	degrees_i	I	UINT8, UINT16, UINT32, UINT64	cases_i	I	UINT8, UINT16, UINT32, UINT64	ops_i	I	UINT8, UINT16, UINT32, UINT64	vertices_i	I	UINT8, UINT16, UINT32, UINT64	dataOut_o	O	UINT8, UINT16, UINT32, UINT64
Port Name	Direction (I/O)	Token RANGE																							
nFans_i	I	UINT8, UINT16, UINT32, UINT64																							
fans_i	I	UINT8, UINT16, UINT32, UINT64																							
degrees_i	I	UINT8, UINT16, UINT32, UINT64																							
cases_i	I	UINT8, UINT16, UINT32, UINT64																							
ops_i	I	UINT8, UINT16, UINT32, UINT64																							
vertices_i	I	UINT8, UINT16, UINT32, UINT64																							
dataOut_o	O	UINT8, UINT16, UINT32, UINT64																							

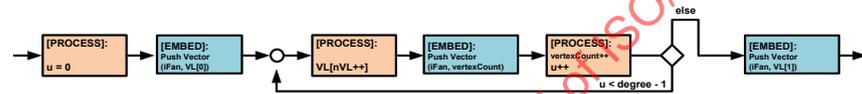
to eliminate the duplicates from a sorted array.

Compute tfans according to the defined cases:

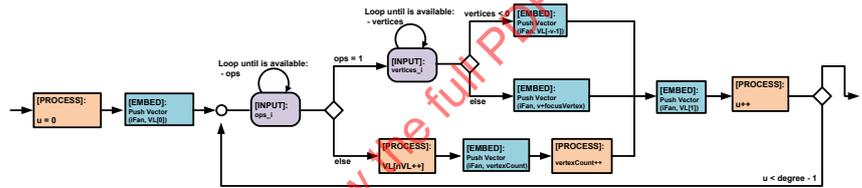


Cases:

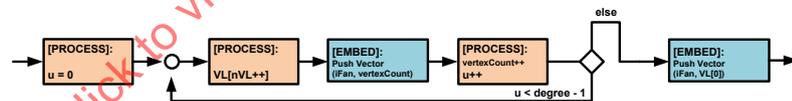
CASE = 0



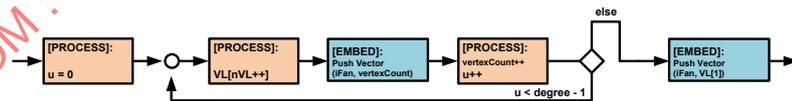
CASE = 1



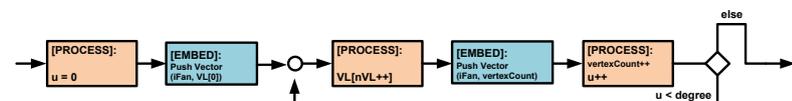
CASE = 2



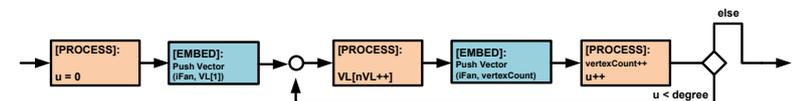
CASE = 3



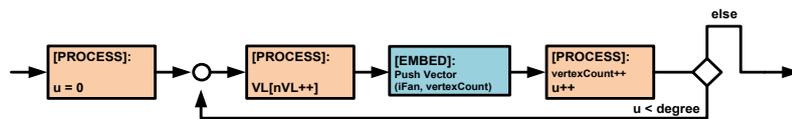
CASE = 4



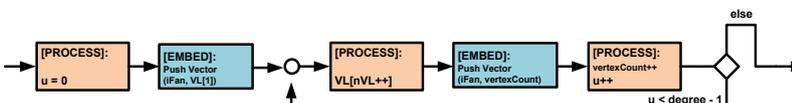
CASE = 5



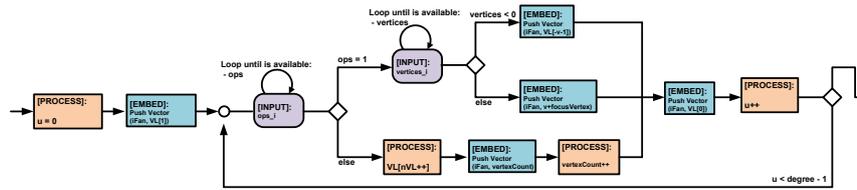
CASE = 6



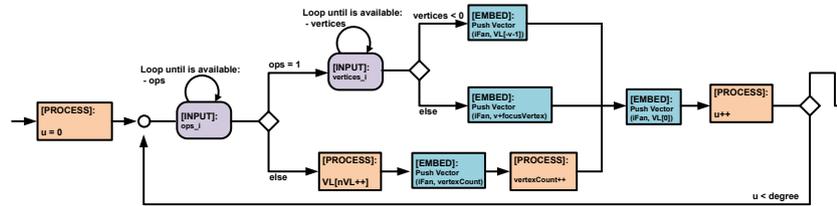
CASE = 7



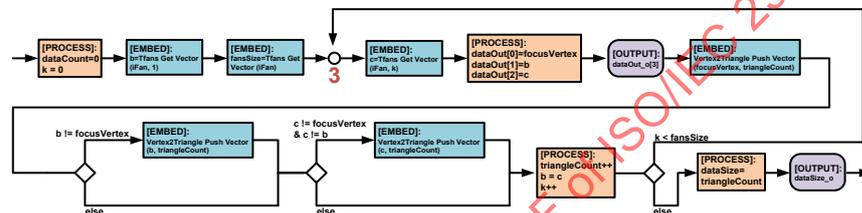
CASE = 8



CASE = 9



Compute dataOut:



Decode Connectivity TFAN Process:

EMBED: Get Number of FANS and Max TFANS dim

```

j = 0
INPUT: nFans
WHILE j < nFans
    INPUT: fans[]
    Number_FANS += fans[j]
    k = 0
    WHILE k < Number_FANS
        INPUT: degree[]
        IF Max_TFANS_dim < degree + 1
            Max_TFANS_dim = degree + 1
        k = k + 1
    j = j + 1
    
```

EMBED: Get all congested vertices adjacent to the focused vertex

```

focusVertex = 0
iFan = 0
j = 0
k = 0
if focusVertex = vertexCount
    vertexCount++
nFan = fans [
nL = 0
EMBED: vector size = Vertex2Triangle Get Vector ( focusVertex )
WHILE j != vector size
    t = EMBED: vector size = Vertex2Triangle Get Vector (focusVertex,j)
    WHILE k != 3
        v = data[t*3+k]
        IF v > focusVertex
            L[nL]=v
            nL++
        k++
    j++
    
```

EMBED: Compute tfans according to the defined cases

```

f = 0
iDegree = 0
iCase = 0
iOps=0
iVertices=0
WHILE f != nFan
    
```

STANDARD.PDF ISO.COM: Only to view the full PDF ONLINE ISO/IEC 23002-4:2018

```

INPUT: cases[]
degree = degree[iDegree]
case = cases[iCase]
EMBED: tFan Push Vector (iFan,focusVector)
SWITCH case
  case = 0: EMBED: CASE 0
  case = 1: EMBED: CASE 1
  case = 2: EMBED: CASE 2
  case = 3: EMBED: CASE 3
  case = 4: EMBED: CASE 4
  case = 5: EMBED: CASE 5
  case = 6: EMBED: CASE 6
  case = 7: EMBED: CASE 7
  case = 8: EMBED: CASE 8
  case = 9: EMBED: CASE 9
f++

EMBED: CASE 0
u=0
EMBED: tFan Push Vector (iFan,VL[0])
WHILE u != degree-1
  VL[nVL++]=vertexCount
  EMBED: tFan Push Vector (iFan,vertexCount)
  u++
EMBED: tFan Push Vector (iFan,VL[1])

EMBED: CASE 1
u=0
iOps=0
iVertices=0
EMBED: tFan Push Vector (iFan,VL[0])
WHILE u != degree-1
  INPUT: ops[]
  IF ops[iOps] = 1
    INPUT: vertices[]
    v = vertices[iVertices]
    IF v < 0
      EMBED: tFan Push Vector (iFan,VL[-v-1])
    ELSE
      EMBED: tFan Push Vector (iFan,v+focusVertex)
  ELSE
    VL[nVL++]=vertexCount
    EMBED: tFan Push Vector (iFan,vertexCount)
    nVL++
    vertexCount++
  u++
EMBED: tFan Push Vector (iFan,VL[1])

EMBED: CASE 2
u=0
WHILE u != degree-1
  VL[nVL]=vertexCount
  EMBED: tFan Push Vector (iFan,vertexCount)
  nVL++
  vertexCount++
  u++
EMBED: tFan Push Vector (iFan,VL[0])

EMBED: CASE 3
u=0
WHILE u != degree-1
  VL[nVL]=vertexCount
  EMBED: tFan Push Vector (iFan,vertexCount)
  nVL++
  vertexCount++
  u++
EMBED: tFan Push Vector (iFan,VL[1])

EMBED: CASE 4
EMBED: tFan Push Vector (iFan,VL[0])
u=0
WHILE u != degree
  VL[nVL]=vertexCount

```

	<pre> EMBED: tFan Push Vector (iFan,vertexCount) nVL++ vertexCount++ u++  EMBED: CASE 5 EMBED: tFan Push Vector (iFan,VL[1]) u=0 WHILE u != degree   VL[nVL]=vertexCount   EMBED: tFan Push Vector (iFan,vertexCount)   nVL++   vertexCount++   u++  EMBED: CASE 6 u=0 WHILE u != degree   VL[nVL]=vertexCount   EMBED: tFan Push Vector (iFan,vertexCount)   nVL++   vertexCount++   u++  EMBED: CASE 7 EMBED: tFan Push Vector (iFan,VL[1]) u=0 WHILE u != degree-1   VL[nVL]=vertexCount   EMBED: tFan Push Vector (iFan,vertexCount)   nVL++   vertexCount++   u++ EMBED: tFan Push Vector (iFan,VL[0])  EMBED: CASE 8 u=0 iOps=0 iVertices=0 EMBED: tFan Push Vector (iFan,VL[1]) WHILE u != degree-1   INPUT: ops[]   IF ops[iOps] = 1     INPUT: vertices[]     v = vertices[iVertices]     IF v &lt; 0       EMBED: tFan Push Vector (iFan,VL[-v-1])     ELSE       EMBED: tFan Push Vector (iFan,v+focusVertex)   ELSE     VL[nVL++]=vertexCount     EMBED: tFan Push Vector (iFan,vertexCount)     nVL++     vertexCount++   u++ EMBED: tFan Push Vector (iFan,VL[0])  EMBED: CASE 9 u=0 WHILE u != degree   INPUT: ops[]   IF ops[iOps] = 1     INPUT: vertices[]     v = vertices[iVertices]     IF v &lt; 0       EMBED: tFan Push Vector (iFan,VL[-v-1])     ELSE       EMBED: tFan Push Vector (iFan,v+focusVertex)   ELSE     VL[nVL++]=vertexCount     EMBED: tFan Push Vector (iFan,vertexCount)     nVL++     vertexCount++ </pre>
--	---

STANDARDS4SO.COM. Click to view the full PDF of ISO/IEC 23002-4:2018

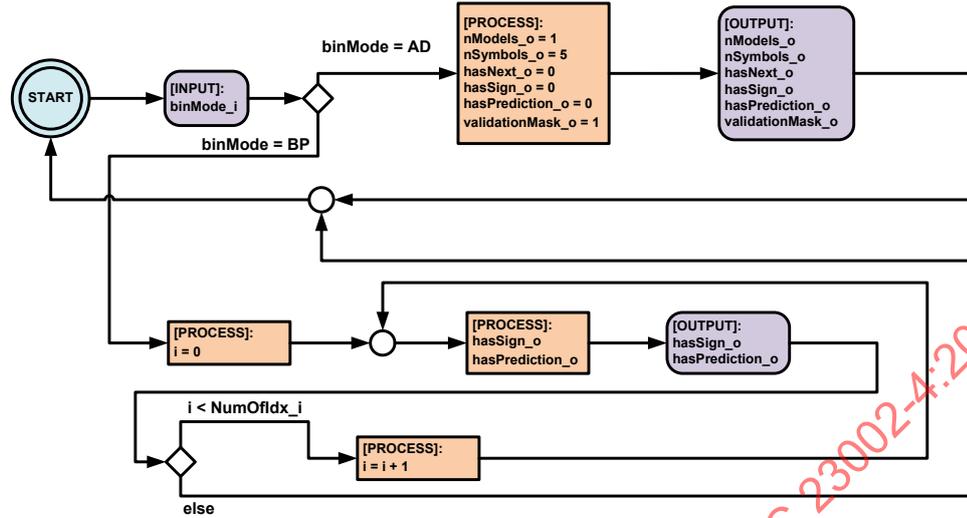
	<pre> u++ EMBED: Compute dataOut k=2 dataCount=0 b=EMBED: tFan Get Vector (iFan,1) fansSize=EMBED: tFan Get Vector (iFan) WHILE k &lt; fansSize   c=EMBED: tFan Get Vector (iFan,k)   dataOut[0]=focusVertex   dataOut[1]=b   dataOut[2]=c   EMBED:Vertex2Triangle Push Vector(focusVertex,triangleCount)   IF b != focusVertex     EMBED:Vertex2Triangle Push Vector(b,triangleCount)   IF c != focusVertex &amp; c!= b     EMBED:Vertex2Triangle Push Vector(c,triangleCount)   triangleCount++   b=c   k++ OUTPUT: dataOut[] OUTPUT: dataSize         </pre>	
<b>ISO Standards using the FU</b>	ISO/IEC 14496-16	
<b>Profiles@levels supported</b>	MPEG-4 Part 16 SC3DMC TFAN	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
nFans_i	UINT8, UINT16, UINT32, UINT64	
fans_i	UINT8, UINT16, UINT32, UINT64	
degrees_i	UINT8, UINT16, UINT32, UINT64	
cases_i	UINT8, UINT16, UINT32, UINT64	
ops_i	UINT8, UINT16, UINT32, UINT64	
vertices_i	UINT8, UINT16, UINT32, UINT64	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
dataOut_o	UINT8, UINT16, UINT32, UINT64	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part16		

## 9.4 MPEG-4 SC3DMC SVA Specific FUs

### 9.4.1 Algo\_ContextModeling\_SVA\_nType

<b>FU Name</b>	Algo_ContextModeling_SVA_nType																													
<b>Description</b>	<p>This FU describes how to generate the context model of “nType” in the arithmetic decoding process.</p> <div style="display: flex; align-items: flex-start;"> <div style="border: 1px solid black; padding: 5px; margin-right: 20px;"> <p><b>Algo_ContextModeling_nType</b></p> <p>→ NumOfIdx_i</p> <p>→ binMode_i</p> <p>nModels_o →</p> <p>validationMask_o →</p> <p>nSymbols_o →</p> <p>hasSign_o →</p> <p>hasPrediction_o →</p> <p>hasNext_o →</p> </div> <table border="1"> <thead> <tr> <th>Port Name</th> <th>Direction (I/O)</th> <th>Token RANGE</th> </tr> </thead> <tbody> <tr> <td>NumOfIdx_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>binMode_i</td> <td>I</td> <td>UINT_8</td> </tr> <tr> <td>nModels_o</td> <td>O</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>validationMask_o</td> <td>O</td> <td>BOOLEAN</td> </tr> <tr> <td>nSymbols_o</td> <td>O</td> <td>UINT32,UINT64</td> </tr> <tr> <td>hasSign_o</td> <td>O</td> <td>BOOLEAN</td> </tr> <tr> <td>hasPrediction_o</td> <td>O</td> <td>BOOLEAN</td> </tr> <tr> <td>hasNext_o</td> <td>O</td> <td>BOOLEAN</td> </tr> </tbody> </table> </div>			Port Name	Direction (I/O)	Token RANGE	NumOfIdx_i	I	UINT8, UINT16, UINT32, UINT64	binMode_i	I	UINT_8	nModels_o	O	UINT8, UINT16, UINT32, UINT64	validationMask_o	O	BOOLEAN	nSymbols_o	O	UINT32,UINT64	hasSign_o	O	BOOLEAN	hasPrediction_o	O	BOOLEAN	hasNext_o	O	BOOLEAN
Port Name	Direction (I/O)	Token RANGE																												
NumOfIdx_i	I	UINT8, UINT16, UINT32, UINT64																												
binMode_i	I	UINT_8																												
nModels_o	O	UINT8, UINT16, UINT32, UINT64																												
validationMask_o	O	BOOLEAN																												
nSymbols_o	O	UINT32,UINT64																												
hasSign_o	O	BOOLEAN																												
hasPrediction_o	O	BOOLEAN																												
hasNext_o	O	BOOLEAN																												

ContextModeling\_SVA\_nType Process Schematic (FSM):



ContextModeling\_SVA\_nType Process:

```

START
INPUT:
  numOfIdx_i
  binMode_i
IF binMode = AD
  nModels = 1
  nSymbols = 5
  hasNext = 0
  hasSign = 0
  hasPrediction = 0
  validationMask = 1
OUTPUT:
  nModels_o
  nSymbols_o
  hasNext_o
  hasSign_o
  hasPrediction_o
  validationMask_o
GOTO START

ELSE IF binMode = BP
  i = 0
  BP_PROCESS:
    hasSign = 0
    hasPrediction = 0
  OUTPUT:
    hasSign_o
    hasPrediction_o
  IF i < numOfIdx_i
    i++
    GOTO BP_PROCESS
  ELSE GOTO START
    
```

<b>ISO Standards using the FU</b>	ISO/IEC 14496-16
<b>Profiles@levels supported</b>	MPEG-4 Part 16 SC3DMC SVA
<b>Input</b>	
<b>Name</b>	<b>Token</b>
NumOfIdx_i	UINT8, UINT16, UINT32, UINT64
binMode_i	UINT_8
<b>Output</b>	
<b>Name</b>	<b>Token</b>
nModels_o	UINT8, UINT16, UINT32, UINT64
validationMask_o	BOOLEAN
nSymbols_o	UINT32,UINT64

hasSign_o	BOOLEAN
hasPrediction_o	BOOLEAN
hasNext_o	BOOLEAN
<b>Parameter</b>	
<b>Name</b>	<b>Description</b>
<b>Package</b>	
package org.sc29.wg11.mpeg4.part16	

### 9.4.2 Algo\_ContextModeling\_SVA\_Indexes

<p><b>FU Name</b></p> <p><b>Description</b></p>	<p><b>Algo_ContextModeling_SVA_Indexes</b></p> <p>This FU describes how to generate context models of “connectivity” depending on the nType and the binarization mode.</p> <div style="display: flex; justify-content: space-around; align-items: center;"> <div data-bbox="470 869 790 1041"> </div> <table border="1" data-bbox="845 739 1364 1153"> <thead> <tr> <th>Port Name</th> <th>Direction (I/O)</th> <th>Token RANGE</th> </tr> </thead> <tbody> <tr> <td>nType_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>numOfIdx_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>fdMode_i</td> <td>I</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>binMode_i</td> <td>I</td> <td>UINT_8</td> </tr> <tr> <td>nModels_o</td> <td>O</td> <td>UINT_8</td> </tr> <tr> <td>validationMask_o</td> <td>O</td> <td>BOOLEAN</td> </tr> <tr> <td>nSymbols_o</td> <td>O</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> <tr> <td>hasSign_o</td> <td>O</td> <td>BOOLEAN</td> </tr> <tr> <td>hasPrediction_o</td> <td>O</td> <td>BOOLEAN</td> </tr> <tr> <td>hasNext_o</td> <td>O</td> <td>BOOLEAN</td> </tr> <tr> <td>modelIndex_o</td> <td>O</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> </tbody> </table> </div>	Port Name	Direction (I/O)	Token RANGE	nType_i	I	UINT8, UINT16, UINT32, UINT64	numOfIdx_i	I	UINT8, UINT16, UINT32, UINT64	fdMode_i	I	UINT8, UINT16, UINT32, UINT64	binMode_i	I	UINT_8	nModels_o	O	UINT_8	validationMask_o	O	BOOLEAN	nSymbols_o	O	UINT8, UINT16, UINT32, UINT64	hasSign_o	O	BOOLEAN	hasPrediction_o	O	BOOLEAN	hasNext_o	O	BOOLEAN	modelIndex_o	O	UINT8, UINT16, UINT32, UINT64
	Port Name	Direction (I/O)	Token RANGE																																		
nType_i	I	UINT8, UINT16, UINT32, UINT64																																			
numOfIdx_i	I	UINT8, UINT16, UINT32, UINT64																																			
fdMode_i	I	UINT8, UINT16, UINT32, UINT64																																			
binMode_i	I	UINT_8																																			
nModels_o	O	UINT_8																																			
validationMask_o	O	BOOLEAN																																			
nSymbols_o	O	UINT8, UINT16, UINT32, UINT64																																			
hasSign_o	O	BOOLEAN																																			
hasPrediction_o	O	BOOLEAN																																			
hasNext_o	O	BOOLEAN																																			
modelIndex_o	O	UINT8, UINT16, UINT32, UINT64																																			

```

hasNext
hasSign
hasPrediction
delimTableSize
delimTable
i = 0
AD VALIDATION
IF i==0
    nType = 1
ELSE
    INPUT:
        nType_i
    IF nType = 0
        validationMask = {1,1,1,1,0,0,0,0,0,0,0}
    ELSE IF nType = 1
        validationMask = {0,0,1,0,0,0,1,0,0,0,1,0}
    ELSE IF nType = 2
        validationMask = {1,0,1,0,0,0,1,1,0,0,0,0}
    ELSE IF nType = 3
        validationMask = {0,0,1,0,0,0,1,0,0,0,1,0}
    ELSE IF nType = 4
        validationMask = {0,1,0,1,0,0,0,0,0,0,0,0}
    OUTPUT:
        validationMask_o
        IF i < numOfIdx
            i = i + 1
            GOTO AD VALIDATION
        ELSE GOTO START
ELSE IF binMode = BP
    i = 0
    BP PROCESS
    INPUT:
        FMode_i
        numOfIdx_i
    IF i = 0
        nType = 1
    ELSE
        INPUT: nType_i
    IF FMode = 0
        IF nType = 0
            nModels = 3
            hasSign = {0,1,0}
            hasPrediction = {0,1,0}
            modelIndex = {0,2,3}
        ELSE IF nType = 1
            nModels = 3
            hasSign = {1,1,1}
            hasPrediction = {1,1,1}
            modelIndex = {2,2,2}
        ELSE IF nType = 2
            nModels = 4
            hasSign = {0,1,1,0}
            hasPrediction = {0,1,1,0}
            modelIndex = {0,2,2,3}
        ELSE IF nType = 3
            nModels = 3
            hasSign = {1,1,1}
            hasPrediction = {1,1,1}
            modelIndex = {2,2,2}
        ELSE IF nType = 4
            nModels = 1
            hasSign = {0}
            hasPrediction = {0}
            modelIndex = {3}
    ELSE IF FMode = 1
        IF nType = 0
            nModels = 4
            hasSign = {0,0,1,0}
            hasPrediction = {0,0,1,0}
            modelIndex = {0,1,2,3}
        ELSE IF nType = 1
            nModels = 3

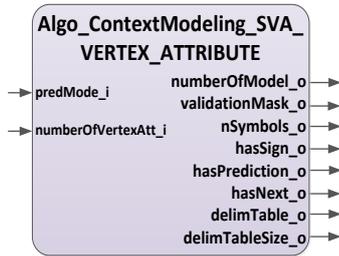
```

STANDARDSISO.COM Click to view the full PDF of ISO/IEC 23002-4:2018

	<pre> hasSign = {1,1,1} hasPrediction = {1,1,1} modelIndex = {2,2,2} ELSE IF nType = 2   nModels = 4   hasSign = {0,1,1,0}   hasPrediction = {0,1,1,0}   modelIndex = {0,2,2,3} ELSE IF nType = 3   nModels = 3   hasSign = {1,1,1}   hasPrediction = {1,1,1}   modelIndex = {2,2,2} ELSE IF nType = 4   nModels = 2   hasSign = {0,0}   hasPrediction = {0,0}   modelIndex = {1, 3}  OUTPUT:   nModels_o   hasSign_o   hasPrediction_o   modelIndex_o  IF i &lt; numOfIdx   i = i + 1   GOTO BP PROCESS ELSE GOTO START         </pre>	
<b>ISO Standards using the FU</b>	ISO/IEC 14496-16	
<b>Profiles@levels supported</b>	MPEG-4 Part 16 SC3DMC SVA	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
nType_i	UINT8, UINT16, UINT32, UINT64	
numOfIdx_i	UINT8, UINT16, UINT32, UINT64	
fdMode_i	UINT8, UINT16, UINT32, UINT64	
binMode_i	UINT_8	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
nModels_o	UINT_8	
validationMask_o	BOOLEAN	
nSymbols_o	UINT8, UINT16, UINT32, UINT64	
hasSign_o	BOOLEAN	
hasPrediction_o	BOOLEAN	
hasNext_o	BOOLEAN	
modelIndex_o	UINT8, UINT16, UINT32, UINT64	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part16		

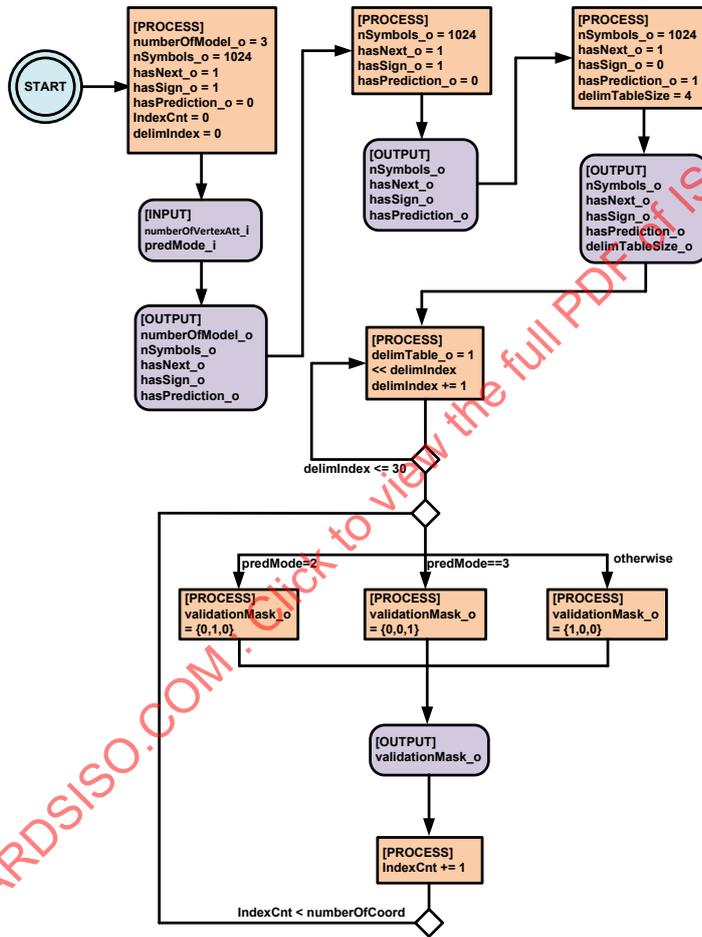
### 9.4.3 Algo\_ContextModeling\_SVA\_Vertex\_Attribute

<b>FU Name</b>	Algo_ContextModeling_SVA_Vertex_Attribute
<b>Description</b>	This FU describes how to generate the context model of “vertex attribute” such as coordinate, normal, color, etc., in the arithmetic decoding process.



Port Name	Direction (I/O)	Token RANGE
predMode_i	I	UINT_4
numberOfCoord_i	I	UINT8, UINT16, UINT32, UINT64
numberOfModel_o	O	UINT8, UINT16, UINT32, UINT64
validationMask_o	O	BOOLEAN
nSymbols_o	O	UINT8, UINT16, UINT32, UINT64
hasSign_o	O	BOOLEAN
hasPrediction_o	O	BOOLEAN
hasNext_o	O	BOOLEAN
delimTable_o	O	UINT8, UINT16, UINT32, UINT64
delimTableSize_o	O	UINT8, UINT16, UINT32, UINT64

ContextModeling\_SVA\_Vertex\_Attribute Process Schematic (FSM):



ContextModeling\_SVA\_Vertex\_Attribute Process:

```

numberOfModel_o = 3
IndexCnt = 0

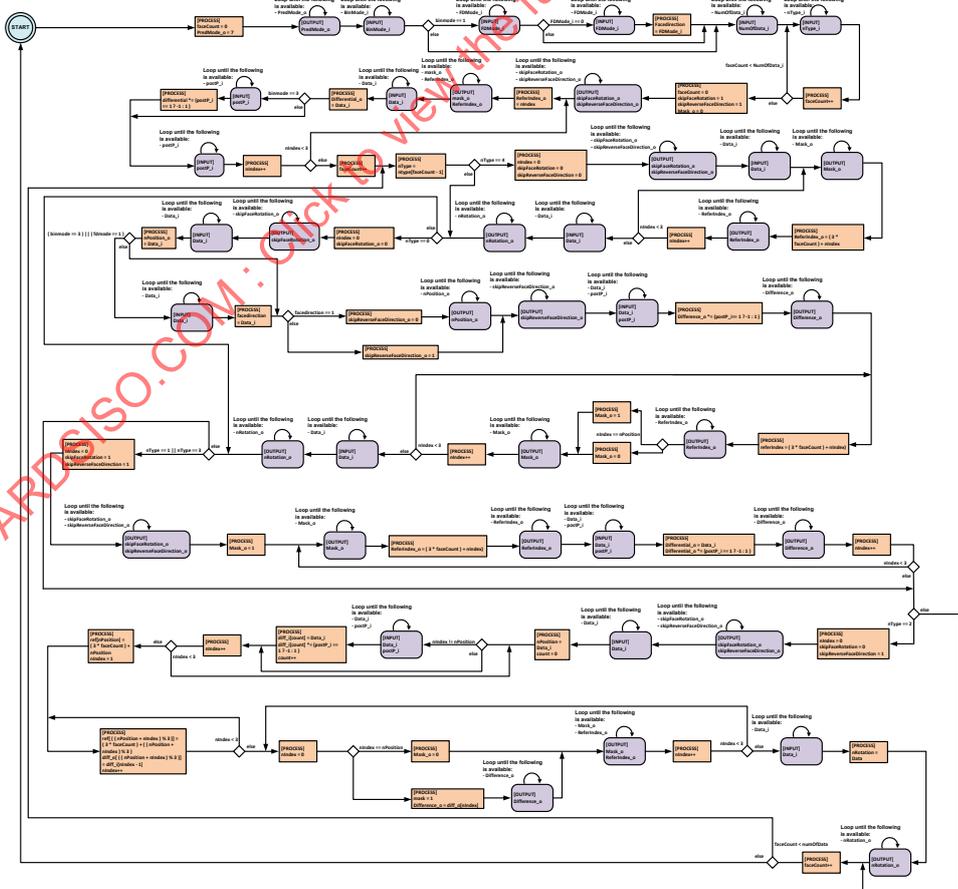
// Data type 1 symbol
nSymbols_o = 1024
hasNext_o = 1
hasSign_o = 1
hasPrediction_o = 0

// Data type 2 symbol
nSymbols_o = 1024
    
```

	<pre> hasNext_o = 1 hasSign_o = 0 hasPrediction_o = 0  // Data type 3 symbol nSymbols_o = 1024 hasNext_o = 1 hasSign_o = 0 hasPrediction_o = 1  delimTableSize_o = 4 for ( delimIndex = 0; delimIndex &lt;= 30; delimIndex += 10 )     delimTable_o = 1 &lt;&lt; delimIndex     for ( ; IndexCnt &lt; numberOfVertxAtt; IndexCnt++ ) {         if ( predMode == 2 )             validationMask_o = {0, 1, 0}         else if ( predMode == 3 )             validationMask_o = {0, 0, 1}         else             validationMask_o = {1, 0, 0}     } </pre>	
<b>ISO Standards using the FU</b>	ISO/IEC 14496-16	
<b>Profiles@levels supported</b>	MPEG-4 Part 16 SC3DMC SVA	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
predMode_i	UINT_4	
numberOfCoord_i	UINT8, UINT16, UINT32, UINT64	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
numberOfModel_o	UINT8, UINT16, UINT32, UINT64	
validationMask_o	BOOLEAN	
nSymbols_o	UINT8, UINT16, UINT32, UINT64	
hasSign_o	BOOLEAN	
hasPrediction_o	BOOLEAN	
hasNext_o	BOOLEAN	
delimTable_o	UINT8, UINT16, UINT32, UINT64	
delimTableSize_o	UINT8, UINT16, UINT32, UINT64	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part16		

9.4.4 Algo\_DecodeConnectivity\_SVA

Port Name	Direction (I/O)	Token RANGE
nType_i	I	UINT8, UINT16, UINT32, UINT64
NumOfData_i	I	UINT8, UINT16, UINT32, UINT64
Data_i	I	UINT8, UINT16, UINT32, UINT64
postP_i	I	BOOLEAN
BinMode_i	I	UINT 8
FDMode_i	I	UINT8, UINT16, UINT32, UINT64
Mask_o	O	UINT8, UINT16, UINT32, UINT64
Difference_o	O	UINT8, UINT16, UINT32, UINT64
ReferIndex_o	O	UINT8, UINT16, UINT32, UINT64
PredMode_o	O	UINT 8
skipFaceRotation_o	O	BOOLEAN
skipReverseFaceDirection_o	O	BOOLEAN
nPosition_o	O	UINT8, UINT16, UINT32, UINT64
nRotation_o	O	UINT8, UINT16, UINT32, UINT64

**DecodingConnectivity\_SVA Process Schematic (FSM):**

The diagram illustrates the state machine logic for decoding connectivity. It starts at a 'START' state and proceeds through various processing blocks (PROCESS) and output blocks (OUTPUT). The process involves multiple nested loops, such as 'Loop until the following is available: - FaceIndex\_i', and decision points based on conditions like 'FaceIndex\_i < nType\_i'. The final outputs include Mask\_o, Difference\_o, ReferIndex\_o, PredMode\_o, skipFaceRotation\_o, skipReverseFaceDirection\_o, nPosition\_o, and nRotation\_o.

Description

ConnectivityDecoding SVA Process:

```

START
faceCount = 0
PredMode_o = 7
OUTPUT:PredMode_o
INPUT:BinMode_i

IF ( binmode == 1 )
INPUT:FDMode_i
IF ( FDMode_i == 0 )
INPUT:FDMode_i
facedirection = FDMode_i;
ENDIF
ENDIF

INPUT:NumOfData_i
while ( faceCount < NumOfData_i )
INPUT:nType_i
faceCount++
ENDWHILE

faceCount = 0;
skipFaceRotation = 1;
skipReverseFaceDirection = 1;
mask_o = 0
OUTPUT:skipFaceRotation_o
OUTPUT:skipReverseFaceDirection_o
mask_o = 0
WHILE ( nIndex < 3 )
ReferIndex_o = nIndex
OUTPUT:Mask_o
OUTPUT:ReferIndex_o
INPUT:Data_i
Differential_o = Data_i

IF ( binmode == 3 )
INPUT:postP_i
Differential_o *= (postP_i == 1 ? -1 : 1);
ENDIF

OUTPUT:Difference_o
nIndex++
ENDIF

faceCount++

WHILE ( faceCount < numOfData ) {
nType = ntype[faceCount - 1];

IF ( nType == 4 )
nIndex = 0;
skipFaceRotation = 0;
skipReverseFaceDirection = 0;
OUTPUT:skipFaceRotation_o
OUTPUT:skipReverseFaceDirection_o
INPUT:Data_i
WHILE ( nIndex < 3 ) {
OUTPUT:Mask_o
ReferIndex_o = ( 3 * faceCount ) + nIndex;
OUTPUT:ReferIndex_o
nIndex++
ENDWHILE
INPUT:Data_i
OUTPUT:nRotation_o
ENDIF

ELSEIF ( nType == 0 )
nIndex = 0;
skipFaceRotation = 0;
OUTPUT:skipFaceRotation_o
INPUT:Data_i
nPosition_o = Data_i

IF ( ( binmode == 3 ) || ( fdmode == 1 ) )

```

```

INPUT:Data_i
facedirection = Data_i
ENDIF

IF ( facedirection == 1 ) {
skipReverseFaceDirection = 0
OUTPUT:nPosition_o
ELSE
skipReverseFaceDirection_o = 1

OUTPUT:skipReverseFaceDirection_o
INPUT:Data_i
INPUT: postP_i
Differential_o *= (postP_i == 1 ? -1 : 1 );
OUTPUT:Difference_o
WHILE ( nIndex < 3 ) {
referIndex = ( 3 * faceCount ) + nIndex;
OUTPUT:ReferIndex_o
IF ( nIndex == nPosition )
Mask_o = 1
ELSE
Mask_o = 0
OUTPUT:Mask_o
nIndex++
ENDWHILE
INPUT:Data_i
OUTPUT:nRotation_o
ENDIF

ELSE IF ( nType == 1 || nType == 3 ) {
nIndex = 0
skipFaceRotation = 1
skipReverseFaceDirection = 1
OUTPUT:skipFaceRotation_o
OUTPUT:skipReverseFaceDirection_o
Mask_o = 1
while ( nIndex < 3 ) {
OUTPUT:Mask_o
referIndex = ( faceCount * 3 ) + nIndex
OUTPUT:ReferIndex_o
INPUT:Data_i
INPUT: postP_i
Differential_o = Data_i
Differential_o *= (postP_i == 1 ? 1 : -1 );
OUTPUT:Difference_o
nIndex++
ENDWHILE
ENDIF

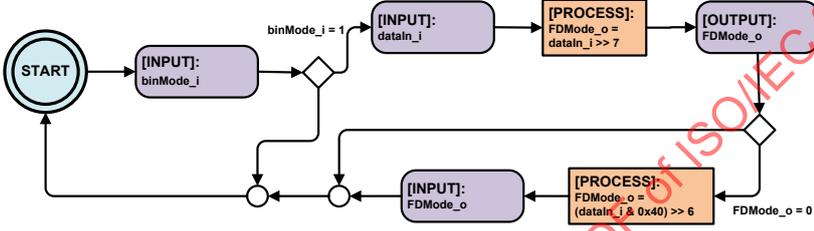
ELSE IF ( nType == 2 )
nIndex = 0
skipFaceRotation = 0
skipReverseFaceDirection = 1
OUTPUT:skipFaceRotation_o
OUTPUT:skipReverseFaceDirection_o
INPUT:Data_i
nPosition = Data_i
count = 0
WHILE ( nIndex < 3 ) {
IF ( nIndex != nPosition )
INPUT:Data_i
INPUT: postP_i
diff_i[count] = Data_i
diff_i[count] *= (postP_i == 1 ? -1 : 1 );
count++
ENDIF
nIndex++
ENDWHILE
ref[nPosition] = ( 3 * faceCount ) + nPosition;
nIndex = 1
WHILE ( nIndex < 3 )
ref[ ( ( nPosition + nIndex ) % 3 ) ] = ( 3 * faceCount ) + ( ( nPosition + nIndex ) % 3 );
diff_o[ ( ( nPosition + nIndex ) % 3 ) ] = diff_i[nIndex - 1]

```

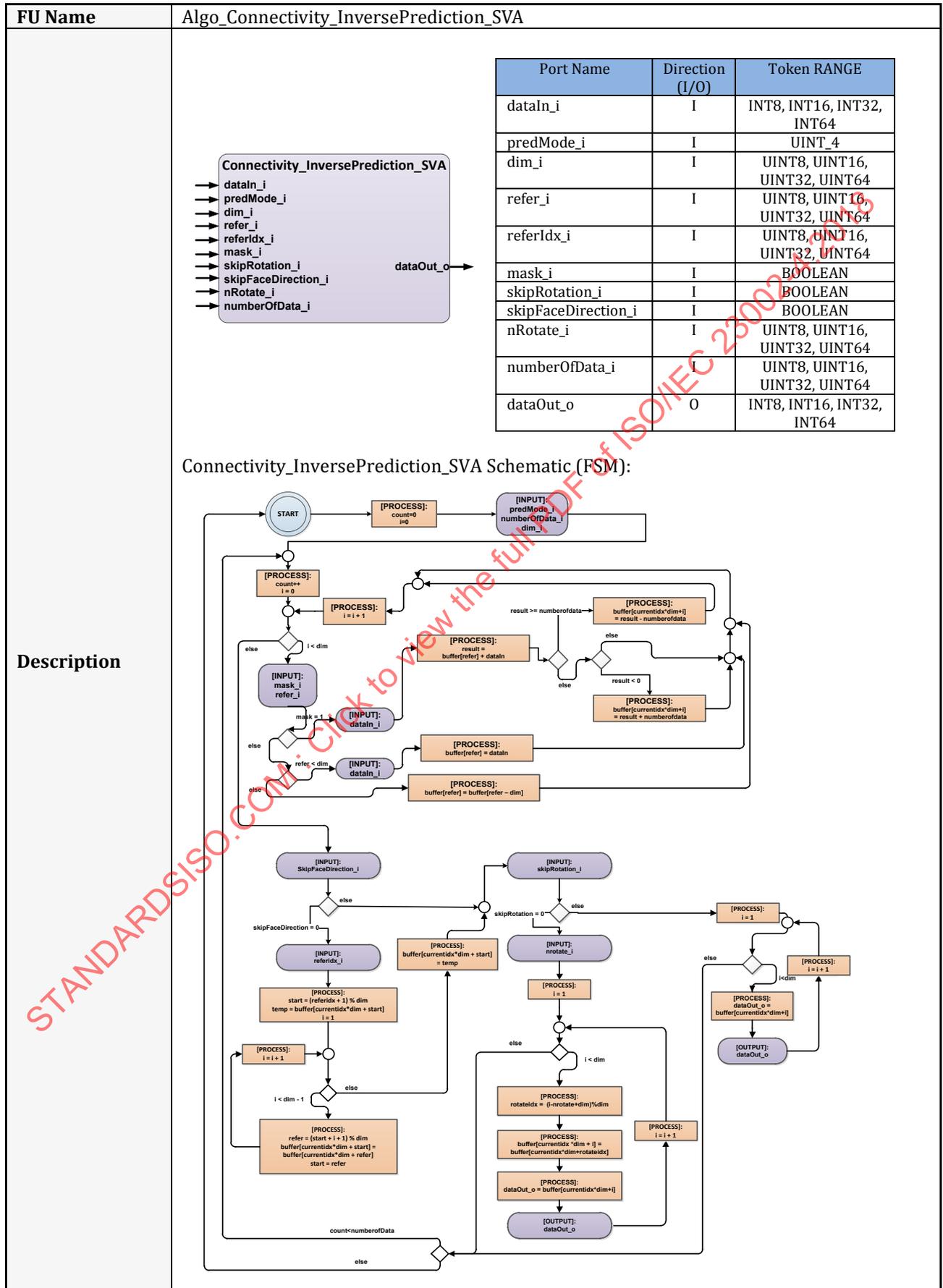
STANDARDS.PDF.COM : Click to view the full PDF of ISO/IEC 23002-4:2018

	<pre> nIndex++ ENDWHILE nIndex = 0 WHILE ( nIndex &lt; 3 ) { IF ( nIndex == nPosition ) mask = 0 ELSE mask = 1 Difference_o = diff_o[nIndex] OUTPUT:Difference_o ENDELSE ReferIndex_o = ref[nIndex] OUTPUT:Mask_o OUTPUT:ReferIndex_o nIndex++ ENDWHILE INPUT:Data_i nRotation = Data OUTPUT:nRotation_o ENDIF faceCount++ ENDWHILE </pre>	
<b>ISO Standards using the FU</b>	ISO/IEC 14496-16	
<b>Profiles@levels supported</b>	MPEG-4 Part 16 SC3DMC SVA	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
nType_i	UINT8, UINT16, UINT32, UINT64	
NumOfData_i	UINT8, UINT16, UINT32, UINT64	
Data_i	UINT8, UINT16, UINT32, UINT64	
postP_i	BOOLEAN	
BinMode_i	UINT_8	
FDMode_i	UINT8, UINT16, UINT32, UINT64	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
Mask_o	UINT8, UINT16, UINT32, UINT64	
Difference_o	UINT8, UINT16, UINT32, UINT64	
ReferIndex_o	UINT8, UINT16, UINT32, UINT64	
PredMode_o	UINT_8	
skipFaceRotation_o	BOOLEAN	
skipReverseFaceDirection_o	BOOLEAN	
nPosition_o	UINT8, UINT16, UINT32, UINT64	
nRotation_o	UINT8, UINT16, UINT32, UINT64	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part16		

9.4.5 Algo\_ExtractFaceDirection\_SVA

<b>FU Name</b>	Algo_ExtractFaceDirection_SVA													
<b>Description</b>	 <table border="1" data-bbox="820 367 1355 517"> <thead> <tr> <th>Port Name</th> <th>Direction (I/O)</th> <th>Token RANGE</th> </tr> </thead> <tbody> <tr> <td>dataIn_i</td> <td>I</td> <td>UINT_8</td> </tr> <tr> <td>binMode_i</td> <td>I</td> <td>UINT_8</td> </tr> <tr> <td>FDMode_o</td> <td>O</td> <td>UINT8, UINT16, UINT32, UINT64</td> </tr> </tbody> </table>		Port Name	Direction (I/O)	Token RANGE	dataIn_i	I	UINT_8	binMode_i	I	UINT_8	FDMode_o	O	UINT8, UINT16, UINT32, UINT64
	Port Name	Direction (I/O)	Token RANGE											
	dataIn_i	I	UINT_8											
	binMode_i	I	UINT_8											
	FDMode_o	O	UINT8, UINT16, UINT32, UINT64											
	Algo_ExtractFaceDirection_SVA Schematic (FSM):													
														
	Algo_ExtractFaceDirection_SVA Process:													
	<pre> START: INPUT:   binMode_i IF binMode_i = 1   INPUT:     dataIn_i   PROCESS:     FDMode_o = dataIn_i &gt;&gt; 7;   OUTPUT:     FDMode_o IF FDMode_o = 0   PROCESS:     FDMode_o = (dataIn_i &amp; 0x40) &gt;&gt; 6   OUTPUT:     FDMode_o GOTO START                 </pre>													
	<b>ISO Standards using the FU</b>	ISO/IEC 14496-16												
<b>Profiles@levels supported</b>	MPEG-4 Part 16 SC3DMC SVA													
<b>Input</b>														
<b>Name</b>	<b>Token</b>													
dataIn_i	UINT_8													
binMode_i	UINT_8													
<b>Output</b>														
<b>Name</b>	<b>Token</b>													
FDMode_o	UINT8, UINT16, UINT32, UINT64													
<b>Parameter</b>														
<b>Name</b>	<b>Description</b>	<b>Range</b>												
<b>Package</b>														
package org.sc29.wg11.mpeg4.part16														

9.4.6 Algo\_Connectivity\_InversePrediction\_SVA



```

Connectivity_Inverse Prediction_SVA Process:
START:

INPUT:  PredictionMode_i
        numberOfData_i
        dim_i
currentidx =0
i = 0

WHILE(currentidx < numberOfData)
  WHILE(i < dim)
    INPUT: refer_i
    INPUT: mask_i
    IF mask = 1
      INPUT : dataIn_i
      result = buffer[refer] + dataIn
      IF (result >= numberOfData)
        buffer[currentidx*dim + i] = result - numberOfData
      IF (result < 0)
        buffer[currentidx*dim + i]= result + numberOfData
    ELSE
      IF (refer < 3)
        INPUT : dataIn_i
        buffer[refer] = dataIn
      ELSE
        buffer[refer] = buffer[refer - dim]
        i = i + 1

INPUT:  skipFaceDirection_i
  IF skipFaceDirection = 0
INPUT : referIdx_i
start = (referIdx + 1) % dim
temp = buffer[currentidx*dim + start]
i = 1
WHILE i < dim - 1
  refer = (start + i + 1) % dim
  buffer[currentidx*dim + start] =
    buffer[currentidx*dim + refer]
  start = refer
  i = i + 1
buffer[currentidx*dim + start] = temp
INPUT:  skipRotation_i
  IF (skipRotation = 0)
    INPUT : nRotate_i
    i = 0
  WHILE(i < dim)
    rotateidx = (i - nRotate + dim) % dim
    buffer[currentidx*dim + i] = buffer[currentidx*dim+rotateidx]
    dataOut_o = buffer[currentidx*dim + i]
    OUTPUT: dataOut_o
    i = i + 1
  ELSE
    WHILE(i < dim)
      dataOut_o = buffer[currentidx*dim + i]
      OUTPUT: dataOut_o
      i = i + 1
    currentidx ++
  GOTO START

```

STANDARDS.PDF.COM : Click to view the full PDF of ISO/IEC 23002-4:2018

<b>ISO Standards using the FU</b>	ISO/IEC 14496-16
<b>Profiles@levels supported</b>	MPEG-4 Part 16 SC3DMC SVA
<b>Input</b>	
<b>Name</b>	<b>Token</b>
dataIn_i	INT8, INT16, INT32, INT64
predMode_i	UINT_4
dim_i	UINT8, UINT16, UINT32, UINT64
refer_i	UINT8, UINT16, UINT32, UINT64
referIdx_i	UINT8, UINT16, UINT32, UINT64

mask_i	BOOLEAN	
skipRotation_i	BOOLEAN	
skipFaceDirection_i	BOOLEAN	
nRotate_i	UINT8, UINT16, UINT32, UINT64	
numberOfData_i	UINT8, UINT16, UINT32, UINT64	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
dataOut_o	INT8, INT16, INT32, INT64	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part16		

## 10 FUs for HEVC Main Profile

### 10.1 General

FUs for building an HEVC (ISO/IEC 23008-2) Main Profile decoder are described in this clause.

### 10.2 Syntax parsing

#### 10.2.1 Algo\_SynP

<b>FU Name</b>	Algo_Parser
<b>Description</b>	FU algoParser parses syntax elements needed to decode HEVC Main profile bitstreams.
<b>Profiles@levels supported</b>	HEVC Main Profile
<b>Input</b>	
<b>Name</b>	<b>Token</b>
byte	uint(size=8) token
<b>Output</b>	
<b>Name</b>	<b>Token</b>
CUInfo	uint(size=16) token
IntraPredMode	uint(size= 6) token
SliceAddr	uint(size=16) token
PartMode	uint(size= 4) token
IsPicSlcLcu	uint(size=2) token
LFAcrossSlcTile	uint(size=2) token
TilesCoord	uint(size=16) token
LcuSizeMax	uint(size=8) token
PictSize	uint(size=16) token
Poc	uint(size=16) token
SliceType	uint(size= 2) token
SplitTransform	bool token
TUSize	int(size= 8) token
Coeff	int(size=16) token
StrongIntraSmoothing	bool token
DispCoord	uint(size=14) token
PicSizeInMb	uint(size=9) token
NumRefIdxLxActive	uint(size= 5) token
RefPicListModif	uint(size= 4) token
RefPoc	int(size=16) token

MvPredSyntaxElem	int(size=16)	
SaoSe	int(size=9) token	
SEI_MD5	uint(size=8) token	
Cbf	bool token	
DBFDisable	bool token	
DbfSe	int(size=8)	
weightedPred	int(size=16) token	
Qp	int(size=8) token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpegH.part2.main.synParser		

### 10.3 Texture decoding

#### 10.3.1 Algo\_IntraPrediction

<b>FU Name</b>	Algo_IntraPrediction	
<b>Description</b>	FU intraPrediction supports all HEVC intra prediction modes, including all 33 directional modes and the DC and planar modes.	
<b>Profiles@levels supported</b>	HEVC Main Profile	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
CUInfo	uint(size=16) token	
IntraPredMode	uint(size= 6) token	
SliceAddr	uint(size=16) token	
PartMode	uint(size= 4) token	
PictSize	uint(size=16) token	
SplitTransform	bool token	
StrongIntraSmoothing	bool token	
Sample	uint(size=8) token	
TilesCoord	uint(size=16) token	
LcuSizeMax	uint(size=8) token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
PredSample	uint(size=8) token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpegH.part2.main.intra		

#### 10.3.2 selectCU

<b>FU Name</b>	Algo_selectCU
<b>Description</b>	Actor selectCU depending on whether the current CU is predicted as intra or inter, the FU adds the residual information to inter or intra prediction.
<b>Profiles@levels supported</b>	HEVC Main Profile
<b>Input</b>	
<b>Name</b>	<b>Token</b>
CUInfo	uint(size=16) token
InterSample	uint(size=8) token
IntraSample	uint(size=8) token
PartMode	uint(size= 4) token
<b>Output</b>	

Name	Token	
Sample	uint(size=8) token	
Parameter		
Name	Description	Range
Package		
package org.sc29.wg11.mpegH.part2.main.intra		

## 10.4 Motion compensation

### 10.4.1 Algo\_InterPrediction

<b>FU Name</b>	Algo_InterPrediction	
<b>Description</b>	The FU interPrediction computes the interpolated samples for the inter prediction.	
<b>Profiles@levels supported</b>	HEVC Main Profile	
Input		
Name	Token	
CUInfo	uint(size=16) token	
IsBiPredOrLx	uint(size=2) token	
Mv	int (size=16) token	
PartMode	uint(size=4) token	
Poc	uint(size=16) token	
RefIdx	uint(size=3) token	
RefList	int(size=16) token	
Sample	uint(size=8) token	
SliceType	uint(size=2) token	
WeightedPredSe	int (size=16) token	
Output		
Name	Token	
PredSample	uint(size=8) token	
Parameter		
Name	Description	Range
Package		
package org.sc29.wg11.mpegH.part2.main.inter		

### 10.4.2 Mgnt\_DecodedPictureBuffer

<b>FU Name</b>	Mgnt_DecodedPictureBuffer	
<b>Description</b>	The Decoded Picture Buffer contains all previously decoded pictures needed for prediction or later output.	
<b>Profiles@levels supported</b>	HEVC Main Profile	
Input		
Name	Token	
IsBiPredOrLx	uint(size=2) token	
IsReadOrNewSlc	uint(size=1) token	
Mv	int(size=16) token	
PicSize	uint(size=16) token	
Poc	uint(size=16) token	
PredCuSize	uint(size=7) token	
RpsPoc	int(size=16) token	
Sample	uint(size=8) token	
Output		
Name	Token	
RefSample	uint(size=8) token	
Parameter		

Name	Description	Range
<b>Package</b>		
package org.sc29.wg11.mpeg.part2.main.inter		

### 10.4.3 Algo\_GenerateRefList

<b>FU Name</b>	Algo_GenerateRefList	
<b>Description</b>	This FU generates reference lists l0 and l1.	
<b>Profiles@levels supported</b>	HEVC Main Profile	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
RefPoc	uint(size=16) token	
RefPicListModif	uint(size=4) token	
NumRefIdxLXAct	uint(size=8) token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
RefList	int(size=16) token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg.part2.main.inter		

### 10.4.4 Algo\_MvComponentPred

<b>FU Name</b>	Algo_MvComponentPred	
<b>Description</b>	This FU reconstructs Motion Vectors for the interpolation process.	
<b>Profiles@levels supported</b>	HEVC Main Profile	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
CUInfo	uint(size=16) token	
PartMode	uint(size=4) token	
PicSize	uint(size=16) token	
Poc	uint(size=16) token	
RefList	int(size=16) token	
RpsPoc	int(size=16) token	
SliceType	uint(size=2) token	
SyntaxElem	int(size=16) token	
SliceAddr	uint(size=16) token	
TilesCoord	uint(size=16) token	
LcuSizeMax	uint(size=8) token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
IsBiPredOrLx	uint(size=2) token	
IsReadOrNewSlice	uint(size=1) token	
Mv	int(size=16) token	
PocRef	int(size=16) token	
PredCuSize	uint(size=7) token	
RefIdx	uint(size=3) token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg.part2.main.inter		

## 10.5 Filtering

### 10.5.1 Algo\_SaoFilter

<b>FU Name</b>	Algo_SaoFilter	
<b>Description</b>	This FU implements the Sample Adaptive Offset filtering stage.	
<b>Profiles@levels supported</b>	HEVC Main Profile	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
IsPicSlcLcu	uint(size=2) token	
LFAcrossSlcTile	uint(size=2) token	
PicSize	uint(size=16) token	
SampleIn	uint(size=8) token	
SaoSe	int(size=9) token	
TilesCoord	uint(size=16) token	
LcuSizeMax	uint(size=8) token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
FiltSample	uint(size=8) token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpegH.part2.main.Filters		

### 10.5.2 Algo\_GenerateBs

<b>FU Name</b>	Algo_GenerateBs	
<b>Description</b>	This FU generates boundary strength coefficients for the inloop deblocking filter.	
<b>Profiles@levels supported</b>	HEVC Main Profile	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
Cbf	bool token	
CUInfo	uint(size=16) token	
IsBiPredOrLx	uint(size=2) token	
LFAcrossSlcTile	uint(size=2) token	
Mv	int(size=16) token	
PartMode	uint(size=4) token	
RefPoc	int(size=16) token	
SliceAddr	uint(size=16) token	
SplitTransf	bool token	
TilesCoord	uint(size=16) token	
LcuSizeMax	uint(size=8) token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
BS	uint(size=2) token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpegH.part2.main.Filters		

### 10.5.3 Algo\_DeblockingFilter

<b>FU Name</b>	Algo_DeblockingFilter	
<b>Description</b>	This FU implements the deblocking filter for Coding Units.	
<b>Profiles@levels supported</b>	HEVC Main Profile	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
Bs	uint(size=2) token	
IsPicSlcLcu	uint(size=2) token	
PicSize	uint(size=16) token	
Qp	int(size=8) token	
SampleIn	uint(size=8) token	
DBFDisable	bool token	
DbfSe	int(size=8)	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
FiltSample	uint(size=8) token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpegH.part2.main.Filters		

### 10.5.4 Algo\_QpGen

<b>FU Name</b>	Algo_QpGen	
<b>Description</b>	This FU computes the Qp for 8×8, 16×16, 32×32, and 64×64 blocks.	
<b>Profiles@levels supported</b>	HEVC Main Profile	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
TuSize	int(size=7) token	
Qp	int(size=8) token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
Qp	int(size=8) token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpegH.part2.main.Filters		

## 10.6 MD5 check

### 10.6.1 Mgnt\_MD5SplitInfo

<b>FU Name</b>	Mgnt_MD5SplitInfo	
<b>Description</b>	This FU splits the file data into 512 bit units necessary for the MD5 algorithm.	
<b>Profiles@levels supported</b>	HEVC Main Profile	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
Byte	uint(size=8) token	
SEI_MD5	uint(size=8) token	
PicSize	uint(size=16) token	

Output		
Name	Token	
DataOut	uint(size=8) token	
Length	uint(size=64) token	
REF_MD5	uint(size=32) token	
Parameter		
Name	Description	Range
Package		
package org.sc29.wg11.mpeg.part2.common		

### 10.6.2 Mgnt\_padding

<b>FU Name</b>	Mgnt_padding	
<b>Description</b>	Data padder	
<b>Profiles@levels supported</b>	HEVC Main Profile	
Input		
Name	Token	
LENGTH	uint(size=64) token	
DATA	uint(size=8) token	
Output		
Name	Token	
OUT	uint(size=8) token	
NEWLENGTH	uint(size=32) token	
Parameter		
Name	Description	Range
Package		
package org.sc29.wg11.mpeg.part2.common		

### 10.6.3 Mgnt\_MD5Shifter

<b>FU Name</b>	Mgnt_MD5Shifter	
<b>Description</b>	Data shifter	
<b>Profiles@levels supported</b>	HEVC Main Profile	
Input		
Name	Token	
IN	uint(size=8) token	
Output		
Name	Token	
OUT	uint(size=32) token	
Parameter		
Name	Description	Range
Package		
package org.sc29.wg11.mpeg.part2.common		

### 10.6.4 Mgnt\_MD5Compute

<b>FU Name</b>	Mgnt_MD5Compute	
<b>Description</b>	This FU applies the MD5 algorithm and compares it with the MD5 reference key.	
<b>Profiles@levels supported</b>	HEVC Main Profile	
Input		
Name	Token	

IN	uint(size=32) token	
REF	uint(size=32) token	
NEWLENGTH	uint(size=32) token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg.part2.common		

## 10.7 Inverse transforms

### 10.7.1 Mgnt\_IT\_Splitter

<b>FU Name</b>	Mgnt_IT_Splitter	
<b>Description</b>	This FU splits the residual coefficients depending on the size of the block.	
<b>Profiles@levels supported</b>	HEVC Main Profile	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
coeff	int(size=16) token	
size	int(size=7) token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
Coeff_4x4_DST	int(size=16) token	
Coeff_4x4_IT	int(size=16) token	
Coeff_8x8	int(size=16) token	
Coeff_16x16	int(size=16) token	
Coeff_32x32	int(size=16) token	
Coeff_skip	int(size=16) token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg.part2.main.IT		

### 10.7.2 Mgnt\_IT\_Merger

<b>FU Name</b>	Mgnt_IT_Merger	
<b>Description</b>	This FU merges the decoded coefficients of the different IT CU.	
<b>Profiles@levels supported</b>	HEVC Main Profile	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
Size	int(size=8) token	
Block_skip	int(size=16) token	
Block_4x4_DST	int(size=16) token	
Block_4x4_IT	int(size=16) token	
Block_8x8	int(size=16) token	
Block_16x16	int(size=16) token	
Block_32x32	int(size=16) token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
Block	int(size=16) token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg.part2.main.IT		



10.7.6 Mgnt\_Transpose\_16x16

<b>FU Name</b>	Mgnt_Transpose_16x16	
<b>Description</b>		
<b>Profiles@levels supported</b>	HEVC Main Profile	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
Src	int(size=16) token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
Dst	int(size=16) token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg.part2.main.IT		

10.7.7 Mgnt\_Transpose\_32x32

<b>FU Name</b>	Mgnt_Transpose_32x32	
<b>Description</b>		
<b>Profiles@levels supported</b>	HEVC Main Profile	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
Src	int(size=16) token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
Dst	int(size=16) token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg.part2.main.IT		

10.7.8 Mgnt\_Transpose\_32x32

<b>FU Name</b>	Mgnt_Transpose_32x32	
<b>Description</b>		
<b>Profiles@levels supported</b>	HEVC Main Profile	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
Src	int(size=16) token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
Dst	int(size=16) token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg.part2.main.IT		

10.7.9 Algo\_IT4x4\_1d

<b>FU Name</b>	Algo_IT4x4_1d
----------------	---------------

<b>Description</b>		
<b>Profiles@levels supported</b>	HEVC Main Profile	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
Src	int(size=16) token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
Dst	int(size=16) token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpegheg.part2.main.IT		

### 10.7.10 Algo\_IT8x8\_1d

<b>FU Name</b>	Algo_IT8x8_1d	
<b>Description</b>		
<b>Profiles@levels supported</b>	HEVC Main Profile	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
Src	int(size=16) token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
Dst	int(size=16) token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpegheg.part2.main.IT		

### 10.7.11 Algo\_IT16x16\_1d

<b>FU Name</b>	Algo_IT16x16_1d	
<b>Description</b>		
<b>Profiles@levels supported</b>	HEVC Main Profile	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
Src	int(size=16) token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
Dst	int(size=16) token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpegheg.part2.main.IT		

### 10.7.12 Algo\_IT32x32\_1d

<b>FU Name</b>	Algo_IT32x32_1d	
<b>Description</b>		
<b>Profiles@levels supported</b>	HEVC Main Profile	

<b>Input</b>		
<b>Name</b>	<b>Token</b>	
Src	int(size=16) token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
Dst	int(size=16) token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part2.main.IT		

10.7.13 Algo\_invDST4x4\_1st

<b>FU Name</b>	Algo_invDST4x4_1st	
<b>Description</b>		
<b>Profiles@levels supported</b>	HEVC Main Profile	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
IN	int(size=16) token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
OUT	int(size=16) token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part2.main.IT		

10.7.14 Algo\_invDST4x4\_2nd

<b>FU Name</b>	Algo_invDST4x4_2nd	
<b>Description</b>		
<b>Profiles@levels supported</b>	HEVC Main Profile	
<b>Input</b>		
<b>Name</b>	<b>Token</b>	
IN	int(size=16) token	
<b>Output</b>		
<b>Name</b>	<b>Token</b>	
OUT	int(size=16) token	
<b>Parameter</b>		
<b>Name</b>	<b>Description</b>	<b>Range</b>
<b>Package</b>		
package org.sc29.wg11.mpeg4.part2.main.IT		

## Annex A (normative)

### Naming convention of FU

#### A.1 Functional Unit name convention

This annex details the convention used to name the Functional Units. Each Functional Unit has a unique name.

#### A.2 Simple Functional Units name convention

The skeleton of the names is in the following format:

**{Role}\_{Name}\_{property1-property2-...-propertyN}\_{size}\_{STANDARD if exists}\_{Ver\_ID}**

{...}	→ compulsory
[...]	→ optional
-	→ inside field separator
_	→ separator between fields

Some examples:

- Mgnt\_Address\_mpeg4\_16x16
- Algo\_Interpolation\_halfpel-mpeg4
- Algo\_DCRaddressing\_mpeg4\_8x8

#### A.3 Description of the fields

##### {Role}

This part is compulsory. It specifies if this Functional Unit implements a specific coding algorithm or not. If the FU is a video coding tool, the “ALGO” tag shall be used (stands for “algorithmic content”). If not, the “MGNT” tag shall be used (stands for “data management”).

**{Name}**

1. This part is compulsory. This field corresponds to the name of the Functional Unit. It should describe as much as possible the action performed by the Functional Unit. In case of an “ALGO” type it may relate to the standard subclause title or to the (shorter) common name to refer to such specific algorithm. For the other Functional Unit, one would try to put a name which can be understood by the entire MPEG community.

**[property]**

This part is optional. It provides additional information about the Functional Unit. Several properties can be mentioned in the name.

The property field can be used to specify

- a given characteristic of the algorithm. For example, ALGO\_Interpolation\_halfpel, “Halfpel” being the property field,
- luminance of chrominance appliance: “LUMA” or “CHROMA”. For example, ALGO\_Name\_Luma or ALGO\_Name\_Chroma, and
- an implementation: It provides any additional information about the implementation of the Functional Unit. If there are various algorithms for one Functional Unit, this field is use to distinguish them.

**[size]**

This part is optional. This field gives an indication concerning the amount of data the Functional Unit has been designed for. For example, the MGNT\_Address\_16x16 Functional Unit deals with blocks of 16×16 pixels.

**{STANDARD if exists}**

This tag indicates with which standard the Functional Unit is conformant with.

For example, algo\_Interpolation\_halfpel\_mpeg4 means this FU is conformant with MPEG4 Part 2. Thus, the algorithm is described in the corresponding official document.

- standard appliance: “mpeg” + {2,4,avc}. For example, “mpeg24avc”, “mpeg4avc”, “mpeg4”.

**[Ver\_ID]**

This tag is used for revision of FU description.

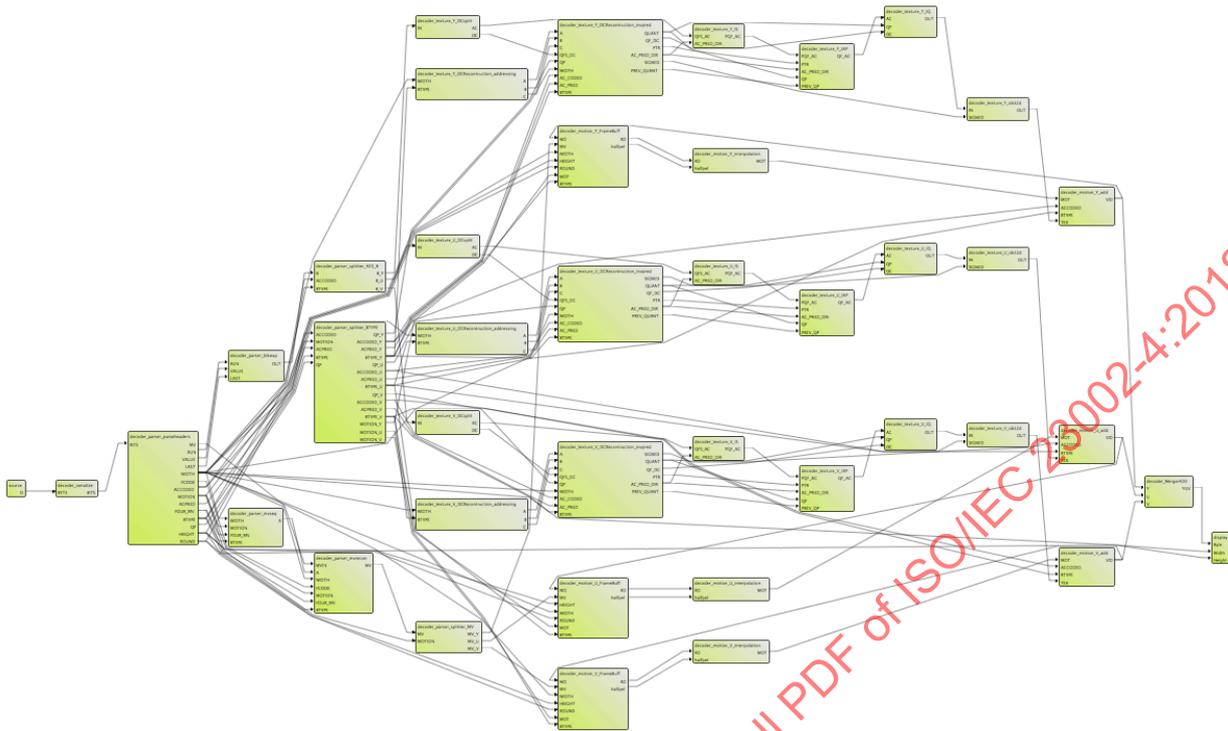
## Annex B (informative)

### FU network examples

#### B.1 Value of the RVC descriptions

Value of the RVC descriptions	Description
0x00	Unspecified
0x01	Description of MPEG-4 Visual (ISO/IEC 14496-2) Simple Profile defined as Top_mpeg4_part2_SP_decoder.xdf
0x02	Description of AVC (ISO/IEC 14496-10) Constrained Baseline Profile defined as Top_mpeg4_part10_CBP_decoder.xdf
0x03	Description of AVC (ISO/IEC 14496-10) Progressive High Profile defined as Top_mpeg4_part10_PHP_decoder.xdf
0x04	Description of HEVC (ISO/IEC 23008-2) Main Profile defined as Top_mpeg4_part2_main_decoder.xdf
0x05 ... 0xFFFF	Reserved
0xFFFF... 0xFFFFFFFF	User defined

B.2 FNL of MPEG-4 Simple Profile



Complete example of the MPEG-4 SP decoder (Refer to Top\_mpeg4\_part2\_SP\_decoder.xdf file)

```

<?xml version="1.0" encoding="UTF-8"?>
<XDF name="Top_mpeg4_part2_SP_decoder">
  <Instance id="source">
    <Class name="org.sc29.wg11.studio.Source"/>
  </Instance>
  <Instance id="display">
    <Class name="org.sc29.wg11.video.DisplayYUV"/>
  </Instance>
  <Instance id="decoder_serialize">
    <Class name="org.sc29.wg11.common.Algo_Byte2bit"/>
  </Instance>
  <Instance id="decoder_Merger420">
    <Class name="org.sc29.wg11.common.Mgnt_Merger420"/>
  </Instance>
  <Instance id="decoder_parser_parseheaders">
    <Class name="org.sc29.wg11.mpeg4.part2.sp.parser.Algo_SynP"/>
  </Instance>
  <Instance id="decoder_parser_mvseq">
    <Class name="org.sc29.wg11.mpeg4.part2.sp.parser.Mgnt_MVSequence_LeftAndTopAndTopRight"/>
  </Instance>
  <Instance id="decoder_parser_blockexpand">
    <Class name="org.sc29.wg11.mpeg4.part2.sp.parser.Mgnt_BlockExpand"/>
  </Instance>
  <Instance id="decoder_parser_mvrecon">
    <Class name="org.sc29.wg11.mpeg4.part2.sp.parser.Mgnt_MVR_MedianOfThreeLeftAndTopAndTopRight"/>
  </Instance>
  <Instance id="decoder_parser_splitter_BTYPE">
    <Class name="org.sc29.wg11.mpeg4.part2.sp.parser.Mgnt_Splitter_420_TYPE"/>
  </Instance>
  <Instance id="decoder_parser_splitter_MV">
    <Class name="org.sc29.wg11.mpeg4.part2.sp.parser.Mgnt_Splitter420MV"/>
  </Instance>
  <Instance id="decoder_parser_splitter_420_B">
    <Class name="org.sc29.wg11.mpeg4.part2.sp.parser.Mgnt_Splitter420B"/>
  </Instance>
  <Instance id="decoder_texture_Y_DCSplit">
    <Class name="org.sc29.wg11.mpeg4.part2.sp.texture.Mgnt_DCSplit"/>
  </Instance>
  <Instance id="decoder_texture_Y_IS">
    <Class name="org.sc29.wg11.mpeg4.part2.sp.texture.Algo_IS_ZigzagOrAlternateHorizontalVertical_8x8"/>
  </Instance>
  <Instance id="decoder_texture_Y_IAP">
    <Class name="org.sc29.wg11.mpeg4.part2.sp.texture.Algo_IAP_AdaptiveHorizontalOrVerticalPred_16x16"/>
  </Instance>
</XDF>

```

```

<Instance id="decoder_texture_Y_IQ">
  <Class name="org.sc29.wg11.mpeg4.part2.sp.texture.Algo_IQ_QSAbdQmatrixMp4v0rH263Scaler"/>
</Instance>
<Instance id="decoder_texture_Y_idct2d">
  <Class name="org.sc29.wg11.mpeg4.part2.sp.texture.Algo_IDCT2D_ISOIEC_23002_1"/>
</Instance>
<Instance id="decoder_texture_Y_DCReconstruction_addressing">
  <Class name="org.sc29.wg11.mpeg4.part2.sp.texture.dc_reconstruction.Algo_DCRAAddr_ThreeLeftTop_16x16"/>
</Instance>
<Instance id="decoder_texture_Y_DCReconstruction_invpred">
  <Class name="org.sc29.wg11.mpeg4.part2.sp.texture.dc_reconstruction.Algo_DCRInvPred_LUMA_16x16"/>
</Instance>
<Instance id="decoder_texture_U_DCSplit">
  <Class name="org.sc29.wg11.mpeg4.part2.sp.texture.Mgnt_DCSplit"/>
</Instance>
<Instance id="decoder_texture_U_IS">
  <Class name="org.sc29.wg11.mpeg4.part2.sp.texture.Algo_IS_ZigzagOrAlternateHorizontalVertical_8x8"/>
</Instance>
<Instance id="decoder_texture_U_IAP">
  <Class name="org.sc29.wg11.mpeg4.part2.sp.texture.Algo_IAP_AdaptiveHorizontalOrVerticalPred_8x8"/>
</Instance>
<Instance id="decoder_texture_U_IQ">
  <Class name="org.sc29.wg11.mpeg4.part2.sp.texture.Algo_IQ_QSAbdQmatrixMp4v0rH263Scaler"/>
</Instance>
<Instance id="decoder_texture_U_idct2d">
  <Class name="org.sc29.wg11.mpeg4.part2.sp.texture.Algo_IDCT2D_ISOIEC_23002_1"/>
</Instance>
<Instance id="decoder_texture_U_DCReconstruction_addressing">
  <Class name="org.sc29.wg11.mpeg4.part2.sp.texture.dc_reconstruction.Algo_DCRAAddr_ThreeLeftTop_8x8"/>
</Instance>
<Instance id="decoder_texture_U_DCReconstruction_invpred">
  <Class name="org.sc29.wg11.mpeg4.part2.sp.texture.dc_reconstruction.Algo_DCRInvPred_CHROMA_8x8"/>
</Instance>
<Instance id="decoder_texture_U_DCSplit">
  <Class name="org.sc29.wg11.mpeg4.part2.sp.texture.Mgnt_DCSplit"/>
</Instance>
<Instance id="decoder_texture_U_IS">
  <Class name="org.sc29.wg11.mpeg4.part2.sp.texture.Algo_IS_ZigzagOrAlternateHorizontalVertical_8x8"/>
</Instance>
<Instance id="decoder_texture_U_IAP">
  <Class name="org.sc29.wg11.mpeg4.part2.sp.texture.Algo_IAP_AdaptiveHorizontalOrVerticalPred_8x8"/>
</Instance>
<Instance id="decoder_texture_U_IQ">
  <Class name="org.sc29.wg11.mpeg4.part2.sp.texture.Algo_IQ_QSAbdQmatrixMp4v0rH263Scaler"/>
</Instance>
<Instance id="decoder_texture_V_idct2d">
  <Class name="org.sc29.wg11.mpeg4.part2.sp.texture.Algo_IDCT2D_ISOIEC_23002_1"/>
</Instance>
<Instance id="decoder_texture_V_DCReconstruction_addressing">
  <Class name="org.sc29.wg11.mpeg4.part2.sp.texture.dc_reconstruction.Algo_DCRAAddr_ThreeLeftTop_8x8"/>
</Instance>
<Instance id="decoder_texture_V_DCReconstruction_invpred">
  <Class name="org.sc29.wg11.mpeg4.part2.sp.texture.dc_reconstruction.Algo_DCRInvPred_CHROMA_8x8"/>
</Instance>
<Instance id="decoder_motion_Y_interpolation">
  <Class name="org.sc29.wg11.mpeg4.part2.sp.motion.Algo_Interp_HalfpelBilinearRoundingControl"/>
</Instance>
<Instance id="decoder_motion_Y_add">
  <Class name="org.sc29.wg11.mpeg4.part2.sp.motion.Algo_PictureReconstruction_Saturation"/>
</Instance>
<Instance id="decoder_motion_Y_FrameBuff">
  <Class name="org.sc29.wg11.mpeg4.part2.sp.motion.Mgnt_FB_w_Address_16X16"/>
</Instance>
<Instance id="decoder_motion_U_interpolation">
  <Class name="org.sc29.wg11.mpeg4.part2.sp.motion.Algo_Interp_HalfpelBilinearRoundingControl"/>
</Instance>
<Instance id="decoder_motion_U_add">
  <Class name="org.sc29.wg11.mpeg4.part2.sp.motion.Algo_PictureReconstruction_Saturation"/>
</Instance>
<Instance id="decoder_motion_U_FrameBuff">
  <Class name="org.sc29.wg11.mpeg4.part2.sp.motion.Mgnt_FB_w_Address_8X8"/>
</Instance>
<Instance id="decoder_motion_V_interpolation">
  <Class name="org.sc29.wg11.mpeg4.part2.sp.motion.Algo_Interp_HalfpelBilinearRoundingControl"/>
</Instance>
<Instance id="decoder_motion_V_add">
  <Class name="org.sc29.wg11.mpeg4.part2.sp.motion.Algo_PictureReconstruction_Saturation"/>
</Instance>
<Instance id="decoder_motion_V_FrameBuff">
  <Class name="org.sc29.wg11.mpeg4.part2.sp.motion.Mgnt_FB_w_Address_8X8"/>
</Instance>
<Connection dst="decoder_parser_mvrecon" dst-port="MVIN"
  src="decoder_parser_parseheaders" src-port="MV"/>
<Connection dst="decoder_parser_blkexp" dst-port="RUN"
  src="decoder_parser_parseheaders" src-port="RUN"/>
<Connection dst="decoder_parser_blkexp" dst-port="VALUE"

```

```

src="decoder_parser_parseheaders" src-port="VALUE"/>
<Connection dst="decoder_parser_blkexp" dst-port="LAST"
src="decoder_parser_parseheaders" src-port="LAST"/>
<Connection dst="decoder_parser_mvrecon" dst-port="A"
src="decoder_parser_mvseq" src-port="A"/>
<Connection dst="decoder_parser_splitter_420_B" dst-port="B"
src="decoder_parser_blkexp" src-port="OUT"/>
<Connection dst="decoder_parser_splitter_MV" dst-port="MV"
src="decoder_parser_mvrecon" src-port="MV"/>
<Connection dst="decoder_parser_mvrecon" dst-port="WIDTH"
src="decoder_parser_parseheaders" src-port="WIDTH"/>
<Connection dst="decoder_parser_mvseq" dst-port="WIDTH"
src="decoder_parser_parseheaders" src-port="WIDTH"/>
<Connection dst="decoder_parser_mvrecon" dst-port="FCODE"
src="decoder_parser_parseheaders" src-port="FCODE"/>
<Connection dst="decoder_parser_splitter_420_B" dst-port="ACCODED"
src="decoder_parser_parseheaders" src-port="ACCODED"/>
<Connection dst="decoder_parser_splitter_BTYPE" dst-port="ACCODED"
src="decoder_parser_parseheaders" src-port="ACCODED"/>
<Connection dst="decoder_parser_mvseq" dst-port="MOTION"
src="decoder_parser_parseheaders" src-port="MOTION"/>
<Connection dst="decoder_parser_splitter_MV" dst-port="MOTION"
src="decoder_parser_parseheaders" src-port="MOTION"/>
<Connection dst="decoder_parser_mvrecon" dst-port="MOTION"
src="decoder_parser_parseheaders" src-port="MOTION"/>
<Connection dst="decoder_parser_splitter_BTYPE" dst-port="MOTION"
src="decoder_parser_parseheaders" src-port="MOTION"/>
<Connection dst="decoder_parser_splitter_BTYPE" dst-port="ACRPRED"
src="decoder_parser_parseheaders" src-port="ACRPRED"/>
<Connection dst="decoder_parser_mvseq" dst-port="FOUR_MV"
src="decoder_parser_parseheaders" src-port="FOUR_MV"/>
<Connection dst="decoder_parser_mvrecon" dst-port="FOUR_MV"
src="decoder_parser_parseheaders" src-port="FOUR_MV"/>
<Connection dst="decoder_parser_splitter_BTYPE" dst-port="BTYPE"
src="decoder_parser_parseheaders" src-port="BTYPE"/>
<Connection dst="decoder_parser_mvseq" dst-port="BTYPE"
src="decoder_parser_parseheaders" src-port="BTYPE"/>
<Connection dst="decoder_parser_splitter_420_B" dst-port="BTYPE"
src="decoder_parser_parseheaders" src-port="BTYPE"/>
<Connection dst="decoder_parser_mvrecon" dst-port="BTYPE"
src="decoder_parser_parseheaders" src-port="BTYPE"/>
<Connection dst="decoder_parser_splitter_BTYPE" dst-port="QP"
src="decoder_parser_parseheaders" src-port="QP"/>
<Connection dst="decoder_parser_parseheaders" dst-port="BITS"
src="decoder_serialize" src-port="BITS"/>
<Connection dst="decoder_texture_Y_IS" dst-port="QFS_AC"
src="decoder_texture_Y_DCsplit" src-port="AC"/>
<Connection dst="decoder_texture_Y_IAP" dst-port="PQF_AC"
src="decoder_texture_Y_IS" src-port="PQF_AC"/>
<Connection dst="decoder_texture_Y_IQ" dst-port="AC"
src="decoder_texture_Y_IAP" src-port="QF_AC"/>
<Connection dst="decoder_texture_Y_idct2d" dst-port="IN"
src="decoder_texture_Y_IQ" src-port="OUT"/>
<Connection dst="decoder_texture_Y_DCReconstruction_invpred"
dst-port="A" src="decoder_texture_Y_DCReconstruction_addressing" src-port="A"/>
<Connection dst="decoder_texture_Y_DCReconstruction_invpred"
dst-port="B" src="decoder_texture_Y_DCReconstruction_addressing" src-port="B"/>
<Connection dst="decoder_texture_Y_DCReconstruction_invpred"
dst-port="C" src="decoder_texture_Y_DCReconstruction_addressing" src-port="C"/>
<Connection dst="decoder_texture_Y_IQ" dst-port="QP"
src="decoder_texture_Y_DCReconstruction_invpred" src-port="QUANT"/>
<Connection dst="decoder_texture_Y_IQ" dst-port="DC"
src="decoder_texture_Y_DCReconstruction_invpred" src-port="QF_DC"/>
<Connection dst="decoder_texture_Y_IAP" dst-port="PTR"
src="decoder_texture_Y_DCReconstruction_invpred" src-port="PTR"/>
<Connection dst="decoder_texture_Y_IAP" dst-port="AC_PRED_DIR"
src="decoder_texture_Y_DCReconstruction_invpred" src-port="AC_PRED_DIR"/>
<Connection dst="decoder_texture_Y_IS" dst-port="AC_PRED_DIR"
src="decoder_texture_Y_DCReconstruction_invpred" src-port="AC_PRED_DIR"/>
<Connection dst="decoder_texture_Y_idct2d" dst-port="SIGNED"
src="decoder_texture_Y_DCReconstruction_invpred" src-port="SIGNED"/>
<Connection dst="decoder_texture_Y_IAP" dst-port="QP"
src="decoder_texture_Y_DCReconstruction_invpred" src-port="QUANT"/>
<Connection dst="decoder_texture_Y_IAP" dst-port="PREV_QP"
src="decoder_texture_Y_DCReconstruction_invpred" src-port="PREV_QUANT"/>
<Connection dst="decoder_texture_Y_DCReconstruction_invpred"
dst-port="QFS_DC" src="decoder_texture_Y_DCsplit" src-port="DC"/>
<Connection dst="decoder_texture_Y_DCsplit" dst-port="IN"
src="decoder_parser_splitter_420_B" src-port="B_Y"/>
<Connection dst="decoder_texture_Y_DCReconstruction_invpred"
dst-port="QP" src="decoder_parser_splitter_BTYPE" src-port="QP_Y"/>
<Connection dst="decoder_texture_Y_DCReconstruction_addressing"
dst-port="WIDTH" src="decoder_parser_parseheaders" src-port="WIDTH"/>
<Connection dst="decoder_texture_Y_DCReconstruction_invpred"
dst-port="WIDTH" src="decoder_parser_parseheaders" src-port="WIDTH"/>
<Connection dst="decoder_texture_Y_DCReconstruction_invpred"

```

Click to view the full PDF of ISO/IEC 23002-4:2018

```

dst-port="AC_CODED" src="decoder_parser_splitter_BTYPE" src-port="ACCODED_Y"/>
<Connection dst="decoder_texture_Y_DCReconstruction_invpred"
dst-port="AC_PRED" src="decoder_parser_splitter_BTYPE" src-port="AC_PRED_Y"/>
<Connection dst="decoder_texture_Y_DCReconstruction_addressing"
dst-port="BTYPE" src="decoder_parser_splitter_BTYPE" src-port="BTYPE_Y"/>
<Connection dst="decoder_texture_Y_DCReconstruction_invpred"
dst-port="BTYPE" src="decoder_parser_splitter_BTYPE" src-port="BTYPE_Y"/>
<Connection dst="decoder_texture_U_IS" dst-port="QFS_AC"
src="decoder_texture_U_DCsplit" src-port="AC"/>
<Connection dst="decoder_texture_U_IAP" dst-port="PQF_AC"
src="decoder_texture_U_IS" src-port="PQF_AC"/>
<Connection dst="decoder_texture_U_IQ" dst-port="AC"
src="decoder_texture_U_IAP" src-port="QF_AC"/>
<Connection dst="decoder_texture_U_idct2d" dst-port="IN"
src="decoder_texture_U_IQ" src-port="OUT"/>
<Connection dst="decoder_texture_U_DCReconstruction_invpred"
dst-port="A" src="decoder_texture_U_DCReconstruction_addressing" src-port="A"/>
<Connection dst="decoder_texture_U_DCReconstruction_invpred"
dst-port="B" src="decoder_texture_U_DCReconstruction_addressing" src-port="B"/>
<Connection dst="decoder_texture_U_DCReconstruction_invpred"
dst-port="C" src="decoder_texture_U_DCReconstruction_addressing" src-port="C"/>
<Connection dst="decoder_texture_U_idct2d" dst-port="SIGNED"
src="decoder_texture_U_DCReconstruction_invpred" src-port="SIGNED"/>
<Connection dst="decoder_texture_U_IQ" dst-port="QP"
src="decoder_texture_U_DCReconstruction_invpred" src-port="QUANT"/>
<Connection dst="decoder_texture_U_IQ" dst-port="DC"
src="decoder_texture_U_DCReconstruction_invpred" src-port="QF_DC"/>
<Connection dst="decoder_texture_U_IAP" dst-port="PTR"
src="decoder_texture_U_DCReconstruction_invpred" src-port="PTR"/>
<Connection dst="decoder_texture_U_IAP" dst-port="AC_PRED_DIR"
src="decoder_texture_U_DCReconstruction_invpred" src-port="AC_PRED_DIR"/>
<Connection dst="decoder_texture_U_IS" dst-port="AC_PRED_DIR"
src="decoder_texture_U_DCReconstruction_invpred" src-port="AC_PRED_DIR"/>
<Connection dst="decoder_texture_U_IAP" dst-port="QP"
src="decoder_texture_U_DCReconstruction_invpred" src-port="QUANT"/>
<Connection dst="decoder_texture_U_IAP" dst-port="PREV_QP"
src="decoder_texture_U_DCReconstruction_invpred" src-port="PREV_QUANT"/>
<Connection dst="decoder_texture_U_DCReconstruction_invpred"
dst-port="QFS_DC" src="decoder_texture_U_DCsplit" src-port="DC"/>
<Connection dst="decoder_texture_U_DCsplit" dst-port="IN"
src="decoder_parser_splitter_420_B" src-port="B_U"/>
<Connection dst="decoder_texture_U_DCReconstruction_invpred"
dst-port="QP" src="decoder_parser_splitter_BTYPE" src-port="QP_U"/>
<Connection dst="decoder_texture_U_DCReconstruction_addressing"
dst-port="WIDTH" src="decoder_parser_parseheaders" src-port="WIDTH"/>
<Connection dst="decoder_texture_U_DCReconstruction_invpred"
dst-port="WIDTH" src="decoder_parser_parseheaders" src-port="WIDTH"/>
<Connection dst="decoder_texture_U_DCReconstruction_invpred"
dst-port="AC_CODED" src="decoder_parser_splitter_BTYPE" src-port="ACCODED_U"/>
<Connection dst="decoder_texture_U_DCReconstruction_invpred"
dst-port="AC_PRED" src="decoder_parser_splitter_BTYPE" src-port="AC_PRED_U"/>
<Connection dst="decoder_texture_U_DCReconstruction_addressing"
dst-port="BTYPE" src="decoder_parser_splitter_BTYPE" src-port="BTYPE_U"/>
<Connection dst="decoder_texture_U_DCReconstruction_invpred"
dst-port="BTYPE" src="decoder_parser_splitter_BTYPE" src-port="BTYPE_U"/>
<Connection dst="decoder_texture_V_IS" dst-port="QFS_AC"
src="decoder_texture_V_DCsplit" src-port="AC"/>
<Connection dst="decoder_texture_V_IAP" dst-port="PQF_AC"
src="decoder_texture_V_IS" src-port="PQF_AC"/>
<Connection dst="decoder_texture_V_IQ" dst-port="AC"
src="decoder_texture_V_IAP" src-port="QF_AC"/>
<Connection dst="decoder_texture_V_idct2d" dst-port="IN"
src="decoder_texture_V_IQ" src-port="OUT"/>
<Connection dst="decoder_texture_V_DCReconstruction_invpred"
dst-port="A" src="decoder_texture_V_DCReconstruction_addressing" src-port="A"/>
<Connection dst="decoder_texture_V_DCReconstruction_invpred"
dst-port="B" src="decoder_texture_V_DCReconstruction_addressing" src-port="B"/>
<Connection dst="decoder_texture_V_DCReconstruction_invpred"
dst-port="C" src="decoder_texture_V_DCReconstruction_addressing" src-port="C"/>
<Connection dst="decoder_texture_V_idct2d" dst-port="SIGNED"
src="decoder_texture_V_DCReconstruction_invpred" src-port="SIGNED"/>
<Connection dst="decoder_texture_V_IQ" dst-port="QP"
src="decoder_texture_V_DCReconstruction_invpred" src-port="QUANT"/>
<Connection dst="decoder_texture_V_IQ" dst-port="DC"
src="decoder_texture_V_DCReconstruction_invpred" src-port="QF_DC"/>
<Connection dst="decoder_texture_V_IAP" dst-port="PTR"
src="decoder_texture_V_DCReconstruction_invpred" src-port="PTR"/>
<Connection dst="decoder_texture_V_IAP" dst-port="AC_PRED_DIR"
src="decoder_texture_V_DCReconstruction_invpred" src-port="AC_PRED_DIR"/>
<Connection dst="decoder_texture_V_IS" dst-port="AC_PRED_DIR"
src="decoder_texture_V_DCReconstruction_invpred" src-port="AC_PRED_DIR"/>
<Connection dst="decoder_texture_V_IAP" dst-port="QP"
src="decoder_texture_V_DCReconstruction_invpred" src-port="QUANT"/>
<Connection dst="decoder_texture_V_IAP" dst-port="PREV_QP"
src="decoder_texture_V_DCReconstruction_invpred" src-port="PREV_QUANT"/>
<Connection dst="decoder_texture_V_DCReconstruction_invpred"

```

```

dst-port="QFS_DC" src="decoder_texture_V_DCSplit" src-port="DC"/>
<Connection dst="decoder_texture_V_DCSplit" dst-port="IN"
src="decoder_parser_splitter_420_B" src-port="B_V"/>
<Connection dst="decoder_texture_V_DCReconstruction_invpred"
dst-port="QP" src="decoder_parser_splitter_BTTYPE" src-port="QP_V"/>
<Connection dst="decoder_texture_V_DCReconstruction_addressing"
dst-port="WIDTH" src="decoder_parser_parseheaders" src-port="WIDTH"/>
<Connection dst="decoder_texture_V_DCReconstruction_invpred"
dst-port="WIDTH" src="decoder_parser_parseheaders" src-port="WIDTH"/>
<Connection dst="decoder_texture_V_DCReconstruction_invpred"
dst-port="AC_CODED" src="decoder_parser_splitter_BTTYPE" src-port="ACCODED_V"/>
<Connection dst="decoder_texture_V_DCReconstruction_invpred"
dst-port="AC_PRED" src="decoder_parser_splitter_BTTYPE" src-port="ACPREL_V"/>
<Connection dst="decoder_texture_V_DCReconstruction_addressing"
dst-port="BTTYPE" src="decoder_parser_splitter_BTTYPE" src-port="BTTYPE_V"/>
<Connection dst="decoder_texture_V_DCReconstruction_invpred"
dst-port="BTTYPE" src="decoder_parser_splitter_BTTYPE" src-port="BTTYPE_V"/>
<Connection dst="decoder_motion_Y_add" dst-port="MOT"
src="decoder_motion_Y_interpolation" src-port="MOT"/>
<Connection dst="decoder_motion_Y_interpolation" dst-port="RD"
src="decoder_motion_Y_FrameBuff" src-port="RD"/>
<Connection dst="decoder_motion_Y_interpolation" dst-port="halfpel"
src="decoder_motion_Y_FrameBuff" src-port="halfpel"/>
<Connection dst="decoder_motion_Y_FrameBuff" dst-port="WD"
src="decoder_motion_Y_add" src-port="VID"/>
<Connection dst="decoder_Merger420" dst-port="Y"
src="decoder_motion_Y_add" src-port="VID"/>
<Connection dst="decoder_motion_Y_FrameBuff" dst-port="MV"
src="decoder_parser_splitter_MV" src-port="MV_Y"/>
<Connection dst="decoder_motion_Y_FrameBuff" dst-port="WIDTH"
src="decoder_parser_parseheaders" src-port="WIDTH"/>
<Connection dst="decoder_motion_Y_FrameBuff" dst-port="HEIGHT"
src="decoder_parser_parseheaders" src-port="HEIGHT"/>
<Connection dst="decoder_motion_Y_FrameBuff" dst-port="ROUND"
src="decoder_parser_parseheaders" src-port="ROUND"/>
<Connection dst="decoder_motion_Y_add" dst-port="ACCODED"
src="decoder_parser_splitter_BTTYPE" src-port="ACCODED_Y"/>
<Connection dst="decoder_motion_Y_FrameBuff" dst-port="MOT"
src="decoder_parser_splitter_BTTYPE" src-port="MOTION_Y"/>
<Connection dst="decoder_motion_Y_add" dst-port="BTTYPE"
src="decoder_parser_splitter_BTTYPE" src-port="BTTYPE_Y"/>
<Connection dst="decoder_motion_Y_FrameBuff" dst-port="BTTYPE"
src="decoder_parser_splitter_BTTYPE" src-port="BTTYPE_Y"/>
<Connection dst="decoder_motion_Y_add" dst-port="TEX"
src="decoder_texture_Y_idct2d" src-port="OUT"/>
<Connection dst="decoder_motion_U_add" dst-port="MOT"
src="decoder_motion_U_interpolation" src-port="MOT"/>
<Connection dst="decoder_motion_U_interpolation" dst-port="RD"
src="decoder_motion_U_FrameBuff" src-port="RD"/>
<Connection dst="decoder_motion_U_interpolation" dst-port="halfpel"
src="decoder_motion_U_FrameBuff" src-port="halfpel"/>
<Connection dst="decoder_motion_U_FrameBuff" dst-port="WD"
src="decoder_motion_U_add" src-port="VID"/>
<Connection dst="decoder_Merger420" dst-port="U"
src="decoder_motion_U_add" src-port="VID"/>
<Connection dst="decoder_motion_U_FrameBuff" dst-port="MV"
src="decoder_parser_splitter_MV" src-port="MV_U"/>
<Connection dst="decoder_motion_U_FrameBuff" dst-port="HEIGHT"
src="decoder_parser_parseheaders" src-port="HEIGHT"/>
<Connection dst="decoder_motion_U_FrameBuff" dst-port="WIDTH"
src="decoder_parser_parseheaders" src-port="WIDTH"/>
<Connection dst="decoder_motion_U_FrameBuff" dst-port="ROUND"
src="decoder_parser_parseheaders" src-port="ROUND"/>
<Connection dst="decoder_motion_U_add" dst-port="ACCODED"
src="decoder_parser_splitter_BTTYPE" src-port="ACCODED_U"/>
<Connection dst="decoder_motion_U_FrameBuff" dst-port="MOT"
src="decoder_parser_splitter_BTTYPE" src-port="MOTION_U"/>
<Connection dst="decoder_motion_U_add" dst-port="BTTYPE"
src="decoder_parser_splitter_BTTYPE" src-port="BTTYPE_U"/>
<Connection dst="decoder_motion_U_FrameBuff" dst-port="BTTYPE"
src="decoder_parser_splitter_BTTYPE" src-port="BTTYPE_U"/>
<Connection dst="decoder_motion_U_add" dst-port="TEX"
src="decoder_texture_U_idct2d" src-port="OUT"/>
<Connection dst="decoder_motion_V_add" dst-port="MOT"
src="decoder_motion_V_interpolation" src-port="MOT"/>
<Connection dst="decoder_motion_V_interpolation" dst-port="RD"
src="decoder_motion_V_FrameBuff" src-port="RD"/>
<Connection dst="decoder_motion_V_interpolation" dst-port="halfpel"
src="decoder_motion_V_FrameBuff" src-port="halfpel"/>
<Connection dst="decoder_motion_V_FrameBuff" dst-port="WD"
src="decoder_motion_V_add" src-port="VID"/>
<Connection dst="decoder_Merger420" dst-port="V"
src="decoder_motion_V_add" src-port="VID"/>
<Connection dst="decoder_motion_V_FrameBuff" dst-port="MV"
src="decoder_parser_splitter_MV" src-port="MV_V"/>
<Connection dst="decoder_motion_V_FrameBuff" dst-port="WIDTH"

```

Click to view the full PDF of ISO/IEC 23002-4:2018

```

src="decoder_parser_parseheaders" src-port="WIDTH"/>
<Connection dst="decoder_motion_V_FrameBuff" dst-port="HEIGHT"
src="decoder_parser_parseheaders" src-port="HEIGHT"/>
<Connection dst="decoder_motion_V_FrameBuff" dst-port="ROUND"
src="decoder_parser_parseheaders" src-port="ROUND"/>
<Connection dst="decoder_motion_V_add" dst-port="ACCODED"
src="decoder_parser_splitter_BTYPE" src-port="ACCODED_V"/>
<Connection dst="decoder_motion_V_FrameBuff" dst-port="MOT"
src="decoder_parser_splitter_BTYPE" src-port="MOTION_V"/>
<Connection dst="decoder_motion_V_add" dst-port="BTYPE"
src="decoder_parser_splitter_BTYPE" src-port="BTYPE_V"/>
<Connection dst="decoder_motion_V_FrameBuff" dst-port="BTYPE"
src="decoder_parser_splitter_BTYPE" src-port="BTYPE_V"/>
<Connection dst="decoder_motion_V_add" dst-port="TEX"
src="decoder_texture_V_idct2d" src-port="OUT"/>
<Connection dst="display" dst-port="B" src="decoder_Merger420" src-port="YUV"/>
<Connection dst="display" dst-port="WIDTH"
src="decoder_parser_parseheaders" src-port="WIDTH"/>
<Connection dst="display" dst-port="HEIGHT"
src="decoder_parser_parseheaders" src-port="HEIGHT"/>
<Connection dst="decoder_serialize" dst-port="BYTE" src="source" src-port="0"/>
</XDF>

```

### B.3 FNL of MPEG-4 AVC Constrained Baseline Profile decoder



Complete example of the MPEG-4 AVC Constrained Baseline Profile decoder (refer to Top\_mpeg4\_part10\_CBP\_decoder.xdf file).

```

<?xml version="1.0" encoding="UTF-8"?>
<XDF name="Top_mpeg4_part10_CBP_decoder">
  <Instance id="source">
    <Class name="org.sc29.wg11.studio.Source"/>
  </Instance>
  <Instance id="Merger">
    <Class name="std.image.format.Merger420"/>
  </Instance>
  <Instance id="display">
    <Class name="org.sc29.wg11.video.DisplayYUV"/>
  </Instance>
  <Instance id="AVCDecoder_Syn_Parser_Algo_Synp">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.synParser.Algo_SynP"/>
  </Instance>
  <Instance id="AVCDecoder_Syn_Parser_Algo_Parser_IPCM">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.synParser.Algo_Parser_IPCM"/>
  </Instance>
  <Instance id="AVCDecoder_Syn_Parser_IntraPredSplit">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.synParser.Algo_IntraPred_Split"/>
  </Instance>
  <Instance id="AVCDecoder_Syn_Parser_Generate_Inter_Info_MMCO">

```

```

<Class name="org.sc29.wg11.mpeg4.part10.cbp.interPred.Algo_MMCO"/>
</Instance>
<Instance id="AVCDecoder_Syn_Parser_Generate_Inter_Info_RefList0">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.interPred.Algo_RefList"/>
</Instance>
<Instance id="AVCDecoder_Syn_Parser_Generate_Inter_Info_Mgnt_InterPred">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.interPred.Mgnt_InterPred"/>
</Instance>
<Instance id="AVCDecoder_Syn_Parser_Generate_Inter_Info_MgntDeblockingFilter">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.deblockingFilter.Mgnt_DBF_AdaptiveFilter "/>
</Instance>
<Instance id="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL0_Reconstr">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.interPred.Algo_MvLXReconstr"/>
  <Parameter name="IsPredL0">
    <Expr kind="Literal" literal-kind="Boolean" value="true"/>
  </Parameter>
</Instance>
<Instance id="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_RefIdxL0ToFrameNum">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.interPred.RefIdxToFrameNum"/>
</Instance>
<Instance id="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvComponentReord">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.deblockingFilter.Algo_MvComponentReorder"/>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_DecodedPictureBuffer">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.interPred.Mgnt_DPB"/>
  <Parameter name="MB_WIDTH">
    <Expr kind="Literal" literal-kind="Integer" value="16"/>
  </Parameter>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_PredictionY_Select">
  <Class name="org.sc29.wg11.common.Algo_SelectMB_4"/>
  <Parameter name="WIDTH">
    <Expr kind="Literal" literal-kind="Integer" value="256"/>
  </Parameter>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_PredictionY_Deblocking_Filter">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.deblockingFilter.Algo_DBF_AdaptiveFilter "/>
  <Parameter name="ChromaEdgeFlag">
    <Expr kind="Literal" literal-kind="Boolean" value="false"/>
  </Parameter>
  <Parameter name="MbWidth">
    <Expr kind="Literal" literal-kind="Integer" value="16"/>
  </Parameter>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_PredictionY_DemuxParserInfos">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.selecMacroblock.Algo_DemuxParserInfoForBlocks_Luma"/>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Mgnt_Intra_16x16">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.intraPred.Mgnt_Intra_16x16"/>
  <Parameter name="MB_WIDTH">
    <Expr kind="Literal" literal-kind="Integer" value="16"/>
  </Parameter>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Algo_IntraPred_LUMA_16x16">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.intraPred.Algo_IntraPred_LUMA_16x16"/>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Add_Clip">
  <Class name="org.sc29.wg11.common.Algo_AddPixSat"/>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Buffer_Neighbour_FullMb">

```

STANDARDSISO.COM Click to view the full PDF of ISO/IEC 23002-4:2018

```

<Class name="org.sc29.wg11.mpeg4.part10.cbp.intraPred.Mgnt_Buffer_Neighbour_FullMb"/>
<Parameter name="IsChroma">
  <Expr kind="Literal" literal-kind="Boolean" value="false"/>
</Parameter>
<Parameter name="MB_WIDTH">
  <Expr kind="Literal" literal-kind="Integer" value="16"/>
</Parameter>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Mgnt_Intra4x4">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.intraPred.Mgnt_Intra_4x4"/>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Algo_IntraPred_LUMA_4x4">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.intraPred.Algo_IntraPred_LUMA_4x4"/>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Algo_Add_4x4">
  <Class name="org.sc29.wg11.common.Algo_AddPixSat"/>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Algo_Merge_4x4_to_16x16_norasterscan">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.intraPred.Algo_Merge_4x4_to_16x16_norasterscan"/>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Algo_Split_16x16_to_4x4_norasterscan">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.intraPred.Algo_Split_16x16_to_4x4_norasterscan"/>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Buffer_Neighbour_4x4">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.intraPred.Mgnt_Buffer_Neighbour_YxY"/>
  <Parameter name="IsIntra4x4">
    <Expr kind="Literal" literal-kind="Boolean" value="true"/>
  </Parameter>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_PredictionY_Inter_Algo_Interp_SeparableSixTapQuarterPel">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.interPred.Algo_Interp_SeparableSixTapQuarterPel"/>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_PredictionY_Inter_Add_Clip">
  <Class name="org.sc29.wg11.common.Algo_AddPixSat"/>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_PredictionY_Inter_Algo_Interp_Reord">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.interPred.Algo_Interp_Reord"/>
  <Parameter name="SzSidePerPart">
    <Expr kind="Literal" literal-kind="Integer" value="4"/>
  </Parameter>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_ResidualY_Algo_Merge_4x4_to_16x16">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_Merge_4x4_to_16x16"/>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_IS_Zigzag_4x4_DC">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_IS_Zigzag_4x4"/>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_DCR_Hadamard_LUMA_IHT1d_0">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_DCR_Hadamard_LUMA_IHT1d"/>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_DC_Transpose4x4_0">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_Transpose4x4"/>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_DCR_Hadamard_LUMA_IHT1d_1">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_DCR_Hadamard_LUMA_IHT1d"/>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_DCR_Hadamard_LUMA_Scaling">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_DCR_Hadamard_LUMA_Scaling"/>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_DC_Transpose4x4_1">

```

```

<Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_Transpose4x4"/>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_DCR_Hadamard_LUMA_Reordering">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_DCR_Hadamard_LUMA_Reordering"/>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_Algo_IS_Zigzag_4x4_AC">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_IS_Zigzag_4x4"/>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_Mgnt_IQ_INTRA16x16">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Mgnt_IQ_INTRA16x16"/>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_Algo_IQ_QSAndSLAndIDCTScaler_4x4">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_IQ_QSAndSLAndIDCTScaler_4x4"/>
  <Parameter name="NB4x4">
    <Expr kind="Literal" literal-kind="Integer" value="16"/>
  </Parameter>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_IT4x4_Algo_IT4x4_1d_0">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_IT4x4_1d"/>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_IT4x4_Algo_Transpose4x4_0">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_Transpose4x4"/>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_IT4x4_Algo_IT4x4_1d_1">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_IT4x4_1d"/>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_IT4x4_Algo_Transpose4x4_1">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_Transpose4x4"/>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_IT4x4_Algo_IT4x4_Addshift">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_IT4x4_Addshift"/>
</Instance>
<Instance id="AVCDecoder_Decode_U_Buffer_inter">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.interPred.Mgnt_DPB"/>
  <Parameter name="MB_WIDTH">
    <Expr kind="Literal" literal-kind="Integer" value="8"/>
  </Parameter>
</Instance>
<Instance id="AVCDecoder_Decode_U_PredictionC_Select_chroma">
  <Class name="org.sc29.wg11.common.Algo_SelectMB_4"/>
  <Parameter name="WIDTH">
    <Expr kind="Literal" literal-kind="Integer" value="64"/>
  </Parameter>
</Instance>
<Instance id="AVCDecoder_Decode_U_PredictionC_Deblocking_Filter">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.deblockingFilter.Algo_DBF_AdaptiveFilter"/>
  <Parameter name="ChromaEdgeFlag">
    <Expr kind="Literal" literal-kind="Boolean" value="true"/>
  </Parameter>
  <Parameter name="MbWidth">
    <Expr kind="Literal" literal-kind="Integer" value="8"/>
  </Parameter>
</Instance>
<Instance id="AVCDecoder_Decode_U_PredictionC_DemuxParserInfos">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.selectMacroblock.Algo_DemuxParserInfoForBlocks_Chroma"/>
</Instance>
<Instance id="AVCDecoder_Decode_U_PredictionC_Intra8x8_Algo_IntraPred_CHROMA">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.intraPred.Algo_IntraPred_CHROMA"/>
</Instance>
<Instance id="AVCDecoder_Decode_U_PredictionC_Intra8x8_Algo_Add">

```

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23002-4:2018

```

<Class name="org.sc29.wg11.common.Algo_AddPixSat"/>
</Instance>
<Instance id="AVCDecoder_Decode_U_PredictionC_Intra8x8_Buffer_Neighbour_FullMb">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.intraPred.Mgmt_Buffer_Neighbour_FullMb"/>
  <Parameter name="IsChroma">
    <Expr kind="Literal" literal-kind="Boolean" value="true"/>
  </Parameter>
  <Parameter name="MB_WIDTH">
    <Expr kind="Literal" literal-kind="Integer" value="8"/>
  </Parameter>
</Instance>
<Instance id="AVCDecoder_Decode_U_PredictionC_Intra8x8_Mgmt_Intra_8x8">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.intraPred.Mgmt_Intra_16x16"/>
  <Parameter name="MB_WIDTH">
    <Expr kind="Literal" literal-kind="Integer" value="8"/>
  </Parameter>
</Instance>
<Instance id="AVCDecoder_Decode_U_PredictionC_Inter_Algo_Interp_EighthPelBilinear">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.interPred.Algo_Interp_Bilinear"/>
</Instance>
<Instance id="AVCDecoder_Decode_U_PredictionC_Inter_Algo_Add">
  <Class name="org.sc29.wg11.common.Algo_AddPixSat"/>
</Instance>
<Instance id="AVCDecoder_Decode_U_PredictionC_Inter_Algo_Interp_Reord">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.interPred.Algo_Interp_Reord"/>
  <Parameter name="SzSidePerPart">
    <Expr kind="Literal" literal-kind="Integer" value="2"/>
  </Parameter>
</Instance>
<Instance id="AVCDecoder_Decode_U_ResidualC_Algo_DCR_Hadamard_CHROMA">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_DCR_Hadamard_CHROMA"/>
</Instance>
<Instance id="AVCDecoder_Decode_U_ResidualC_Algo_Merge_4x4_to_8x8">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_Merge_4x4_to_8x8"/>
</Instance>
<Instance id="AVCDecoder_Decode_U_ResidualC_IS_IQ_IT_C_Algo_IS_Zigzag_4x4">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_IS_Zigzag_4x4"/>
</Instance>
<Instance id="AVCDecoder_Decode_U_ResidualC_IS_IQ_IT_C_Mgmt_IQ_Chroma">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Mgmt_IQ_Chroma"/>
</Instance>
<Instance id="AVCDecoder_Decode_U_ResidualC_IS_IQ_IT_C_Algo_IQ_QSAndSLAndIDCTScaler_4x4">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_IQ_QSAndSLAndIDCTScaler_4x4"/>
  <Parameter name="NB4x4">
    <Expr kind="Literal" literal-kind="Integer" value="4"/>
  </Parameter>
</Instance>
<Instance id="AVCDecoder_Decode_U_ResidualC_IS_IQ_IT_C_IT4x4_Algo_IT4x4_1d_0">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_IT4x4_1d"/>
</Instance>
<Instance id="AVCDecoder_Decode_U_ResidualC_IS_IQ_IT_C_IT4x4_Algo_Transpose4x4_0">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_Transpose4x4"/>
</Instance>
<Instance id="AVCDecoder_Decode_U_ResidualC_IS_IQ_IT_C_IT4x4_Algo_IT4x4_1d_1">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_IT4x4_1d"/>
</Instance>
<Instance id="AVCDecoder_Decode_U_ResidualC_IS_IQ_IT_C_IT4x4_Algo_Transpose4x4_1">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_Transpose4x4"/>
</Instance>
<Instance id="AVCDecoder_Decode_U_ResidualC_IS_IQ_IT_C_IT4x4_Algo_IT4x4_Addshift">

```

```

<Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_IT4x4_Addshift"/>
</Instance>
<Instance id="AVCDecoder_Decode_V_Buffer_inter">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.interPred.Mgmt_DPB"/>
  <Parameter name="MB_WIDTH">
    <Expr kind="Literal" literal-kind="Integer" value="8"/>
  </Parameter>
</Instance>
<Instance id="AVCDecoder_Decode_V_PredictionC_Select_chroma">
  <Class name="org.sc29.wg11.common.Algo_SelectMB_4"/>
  <Parameter name="WIDTH">
    <Expr kind="Literal" literal-kind="Integer" value="64"/>
  </Parameter>
</Instance>
<Instance id="AVCDecoder_Decode_V_PredictionC_Deblocking_Filter">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.deblockingFilter.Algo_DBF_AdaptiveFilter"/>
  <Parameter name="ChromaEdgeFlag">
    <Expr kind="Literal" literal-kind="Boolean" value="true"/>
  </Parameter>
  <Parameter name="MbWidth">
    <Expr kind="Literal" literal-kind="Integer" value="8"/>
  </Parameter>
</Instance>
<Instance id="AVCDecoder_Decode_V_PredictionC_DemuxParserInfos">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.selectMacroblock.Algo_DemuxParserInfoForBlocks_Chroma"/>
</Instance>
<Instance id="AVCDecoder_Decode_V_PredictionC_Intra8x8_Algo_IntraPred_CHROMA">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.intraPred.Algo_IntraPred_CHROMA"/>
</Instance>
<Instance id="AVCDecoder_Decode_V_PredictionC_Intra8x8_Algo_Add">
  <Class name="org.sc29.wg11.common.Algo_AddPixSat"/>
</Instance>
<Instance id="AVCDecoder_Decode_V_PredictionC_Intra8x8_Buffer_Neighbour_FullMb">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.intraPred.Mgmt_Buffer_Neighbour_FullMb"/>
  <Parameter name="IsChroma">
    <Expr kind="Literal" literal-kind="Boolean" value="true"/>
  </Parameter>
  <Parameter name="MB_WIDTH">
    <Expr kind="Literal" literal-kind="Integer" value="8"/>
  </Parameter>
</Instance>
<Instance id="AVCDecoder_Decode_V_PredictionC_Intra8x8_Mgmt_Intra_8x8">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.intraPred.Mgmt_Intra_16x16"/>
  <Parameter name="MB_WIDTH">
    <Expr kind="Literal" literal-kind="Integer" value="8"/>
  </Parameter>
</Instance>
<Instance id="AVCDecoder_Decode_V_PredictionC_Inter_Algo_Interp_EighthPelBilinear">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.interPred.Algo_Interp_Bilinear"/>
</Instance>
<Instance id="AVCDecoder_Decode_V_PredictionC_Inter_Algo_Add">
  <Class name="org.sc29.wg11.common.Algo_AddPixSat"/>
</Instance>
<Instance id="AVCDecoder_Decode_V_PredictionC_Inter_Algo_Interp_Reord">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.interPred.Algo_Interp_Reord"/>
  <Parameter name="SzSidePerPart">
    <Expr kind="Literal" literal-kind="Integer" value="2"/>
  </Parameter>
</Instance>
<Instance id="AVCDecoder_Decode_V_ResidualC_Algo_DCR_Hadamard_CHROMA">

```

STANDARDISO.COM : Click to view the full PDF of ISO/IEC 23002-4:2018

```

    <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_DCR_Hadamard_CHROMA"/>
  </Instance>
  <Instance id="AVCDecoder_Decode_V_ResidualC_Algo_Merge_4x4_to_8x8">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_Merge_4x4_to_8x8"/>
  </Instance>
  <Instance id="AVCDecoder_Decode_V_ResidualC_IS_IQ_IT_C_Algo_IS_Zigzag_4x4">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_IS_Zigzag_4x4"/>
  </Instance>
  <Instance id="AVCDecoder_Decode_V_ResidualC_IS_IQ_IT_C_Mgmt_IQ_Chroma">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Mgmt_IQ_Chroma"/>
  </Instance>
  <Instance id="AVCDecoder_Decode_V_ResidualC_IS_IQ_IT_C_Algo_IQ_QSAndSLAndIDCTScaler_4x4">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_IQ_QSAndSLAndIDCTScaler_4x4"/>
    <Parameter name="NB4x4">
      <Expr kind="Literal" literal-kind="Integer" value="4"/>
    </Parameter>
  </Instance>
  <Instance id="AVCDecoder_Decode_V_ResidualC_IS_IQ_IT_C_IT4x4_Algo_IT4x4_1d_0">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_IT4x4_1d"/>
  </Instance>
  <Instance id="AVCDecoder_Decode_V_ResidualC_IS_IQ_IT_C_IT4x4_Algo_Transpose4x4_0">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_Transpose4x4"/>
  </Instance>
  <Instance id="AVCDecoder_Decode_V_ResidualC_IS_IQ_IT_C_IT4x4_Algo_IT4x4_1d_1">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_IT4x4_1d"/>
  </Instance>
  <Instance id="AVCDecoder_Decode_V_ResidualC_IS_IQ_IT_C_IT4x4_Algo_Transpose4x4_1">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_Transpose4x4"/>
  </Instance>
  <Instance id="AVCDecoder_Decode_V_ResidualC_IS_IQ_IT_C_IT4x4_Algo_IT4x4_Addshift">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_IT4x4_Addshift"/>
  </Instance>
  <Instance id="AVCDecoder_CavlcExpand_BlockExpand">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.synParser.Algo_BlockExpand"/>
  </Instance>
  <Instance id="AVCDecoder_CavlcExpand_BlockSplit">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.synParser.Algo_BlockSplit"/>
  </Instance>
  <Instance id="AVCDecoder_Display_Render_DisplayRendererY">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.display.Mgmt_BufferRenderer"/>
    <Parameter name="MB_WIDTH">
      <Expr kind="Literal" literal-kind="Integer" value="16"/>
    </Parameter>
  </Instance>
  <Instance id="AVCDecoder_Display_Render_DisplayRendererU">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.display.Mgmt_BufferRenderer"/>
    <Parameter name="MB_WIDTH">
      <Expr kind="Literal" literal-kind="Integer" value="8"/>
    </Parameter>
  </Instance>
  <Instance id="AVCDecoder_Display_Render_DisplayRendererV">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.display.Mgmt_BufferRenderer"/>
    <Parameter name="MB_WIDTH">
      <Expr kind="Literal" literal-kind="Integer" value="8"/>
    </Parameter>
  </Instance>
  <Instance id="AVCDecoder_Display_Render_Mgmt_Display_Poc">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.display.Mgmt_POC"/>
  </Instance>
  <Connection dst="display" dst-port="B" src="Merger" src-port="YUV"/>

```

```

<Connection dst="AVCDecoder_Syn_Parser_Algo_Parser_IPCM"
  dst-port="I_PCM" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="I_PCM"/>
<Connection dst="AVCDecoder_Syn_Parser_IntraPredSplit"
  dst-port="MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_TYPE"/>
<Connection dst="AVCDecoder_Syn_Parser_IntraPredSplit"
  dst-port="INTRA_PRED_MODE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="INTRA_PRED_MODE"/>
<Connection dst="AVCDecoder_Syn_Parser_IntraPredSplit"
  dst-port="MB_LOCATION" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_LOCATION"/>
<Connection dst="AVCDecoder_Syn_Parser_IntraPredSplit"
  dst-port="CONSTRAINED_IFLAG"
  src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="CONSTRAINED_IFLAG"/>
<Connection dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_RefList0"
  dst-port="EMPTY_DPB"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MMCO" src-port="EMPTY_DPB"/>
<Connection dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_RefList0"
  dst-port="ST_LIST"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MMCO" src-port="ST_LIST"/>
<Connection dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_RefList0"
  dst-port="ST_LIST_SZ"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MMCO" src-port="ST_LIST_SZ"/>
<Connection dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_RefList0"
  dst-port="LT_LIST"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MMCO" src-port="LT_LIST"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_RefIdxL0ToFrameNum"
  dst-port="REF_IDX"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL0_Reconstr" src-port="REF_IDX_OUT"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvComponentReord"
  dst-port="MV"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL0_Reconstr" src-port="MV_OUT"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvComponentReord"
  dst-port="READ_FRAME"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_RefIdxL0ToFrameNum" src-port="READ_FRAME"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_RefIdxL0ToFrameNum"
  dst-port="IS_NEW_MB"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL0_Reconstr" src-port="IS_NEW_MB_SLICE"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_Mgnt_InterPred"
  dst-port="MV"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL0_Reconstr" src-port="MV_OUT"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_Mgnt_InterPred"
  dst-port="PARTSZ"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvComponentReord" src-port="PARTSZ"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_Mgnt_InterPred"
  dst-port="LOCATION"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL0_Reconstr" src-port="LOCATION"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MgntDeblockingFilter"
  dst-port="READ_FRAME"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvComponentReord" src-port="READ_FRAME_DBF"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MgntDeblockingFilter"
  dst-port="MV"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvComponentReord" src-port="MV_OUT_DBF"/>
<Connection

```

STANDARDSISO.COM · Click to view the full PDF of ISO/IEC 23002-4:2018

```

dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_RefIdxL0ToFrameNum"
dst-port="REF_LIST"
src="AVCDecoder_Syn_Parser_Generate_Inter_Info_RefList0" src-port="REF_LIST"/>
<Connection dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MMCO"
dst-port="MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_TYPE"/>
<Connection
dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MgntDeblockingFilter"
dst-port="MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_TYPE"/>
<Connection
dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL0_Reconstr"
dst-port="MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_TYPE"/>
<Connection
dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvComponentReord"
dst-port="MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_TYPE"/>
<Connection
dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL0_Reconstr"
dst-port="SUB_MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="SUB_MB_TYPE"/>
<Connection
dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvComponentReord"
dst-port="SUB_MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="SUB_MB_TYPE"/>
<Connection
dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL0_Reconstr"
dst-port="MB_PRED_MVD" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_PRED_MVD_I0"/>
<Connection
dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL0_Reconstr"
dst-port="SUB_MB_PRED_MVD" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="SUB_MB_PRED_MVD_I0"/>
<Connection
dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL0_Reconstr"
dst-port="SUB_MB_PRED_REF_IDX"
src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="SUB_MB_PRED_REF_IDX_I0"/>
<Connection
dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL0_Reconstr"
dst-port="MB_PRED_REF_IDX" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_PRED_REF_IDX_I0"/>
<Connection dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_RefList0"
dst-port="REF_TYPE_LIST" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="REF_TYPE_I0"/>
<Connection dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MMCO"
dst-port="MAX_FRAME_NUM" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MAX_FRAME_NUM"/>
<Connection dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_RefList0"
dst-port="MAX_FRAME_NUM" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MAX_FRAME_NUM"/>
<Connection dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MMCO"
dst-port="MMCO" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MMCO"/>
<Connection dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_RefList0"
dst-port="NB_REF_FRAME" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="NB_REF_FRAME_I0"/>
<Connection dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_RefList0"
dst-port="REF_REORDERING" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="REF_REORDERING"/>
<Connection dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MMCO"
dst-port="FRAME_NUM" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="FRAME_NUM"/>
<Connection dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_RefList0"
dst-port="FRAME_NUM" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="FRAME_NUM"/>
<Connection dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MMCO"
dst-port="POC" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="POC"/>
<Connection dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_RefList0"
dst-port="POC" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="POC"/>
<Connection
dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MgntDeblockingFilter"
dst-port="MB_LOCATION" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_LOCATION"/>
<Connection
dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL0_Reconstr"
dst-port="MB_LOCATION" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_LOCATION"/>
<Connection dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MMCO"

```

```

    dst-port="NUM_REF_FRAME" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="NUM_REF_FRAMES"/>
<Connection
    dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MgmtDeblockingFilter"
    dst-port="QP_Y" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="QP"/>
<Connection
    dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MgmtDeblockingFilter"
    dst-port="QP_Cb" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="QP_Cb"/>
<Connection
    dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MgmtDeblockingFilter"
    dst-port="QP_Cr" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="QP_Cr"/>
<Connection
    dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MgmtDeblockingFilter"
    dst-port="SLICE_DBF_PARAM" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="SLICE_DBF_PARAM"/>
<Connection
    dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MgmtDeblockingFilter"
    dst-port="CBP_BLK" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="CBP_BLK"/>
<Connection
    dst="AVCDecoder_Decoding_Y_PredictionY_Deblocking_Filter"
    dst-port="MB_IN" src="AVCDecoder_Decoding_Y_PredictionY_Select" src-port="OUT"/>
<Connection dst="AVCDecoder_Decoding_Y_PredictionY_Select"
    dst-port="SELECT"
    src="AVCDecoder_Decoding_Y_PredictionY_DemuxParserInfos" src-port="SELECT"/>
<Connection
    dst="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Add_Clip"
    dst-port="X"
    src="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Algo_IntraPred_LUMA_16x16" src-port="MPR"/>
<Connection
    dst="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Algo_IntraPred_LUMA_16x16"
    dst-port="AVAIL"
    src="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Mgmt_Intra_16x16" src-port="AVAIL"/>
<Connection
    dst="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Algo_IntraPred_LUMA_16x16"
    dst-port="Y_LEFT"
    src="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Mgmt_Intra_16x16" src-port="Y_LEFT"/>
<Connection
    dst="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Algo_IntraPred_LUMA_16x16"
    dst-port="Y_UP"
    src="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Mgmt_Intra_16x16" src-port="Y_UP"/>
<Connection
    dst="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Algo_IntraPred_LUMA_16x16"
    dst-port="Y_UP_LEFT"
    src="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Mgmt_Intra_16x16" src-port="Y_UP_LEFT"/>
<Connection
    dst="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Mgmt_Intra_16x16"
    dst-port="EDGE"
    src="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Buffer_Neighbour_FullMb" src-port="EDGE"/>
<Connection dst="AVCDecoder_Decoding_Y_PredictionY_Select"
    dst-port="IN_Z"
    src="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Add_Clip" src-port="Z"/>
<Connection
    dst="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Add_Clip"
    dst-port="Y"
    src="AVCDecoder_Decoding_Y_PredictionY_DemuxParserInfos" src-port="COEF_ACR_INTRA16"/>
<Connection
    dst="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Algo_IntraPred_LUMA_16x16"
    dst-port="PRED_MODE"
    src="AVCDecoder_Decoding_Y_PredictionY_DemuxParserInfos" src-port="PRED_MODE16"/>
<Connection
    dst="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Buffer_Neighbour_FullMb"
    dst-port="MB_IN" src="AVCDecoder_Decoding_Y_PredictionY_Select" src-port="OUT"/>

```

```

<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Mgnt_Intra_16x16"
  dst-port="NEIGHBOUR"
  src="AVCDecoder_Decoding_Y_PredictionY_DemuxParserInfos" src-port="NEIGHBOUR16"/>
<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Algo_IntraPred_LUMA_4x4"
  dst-port="Y_LEFT"
  src="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Mgnt_Intra4x4" src-port="Y_LEFT4"/>
<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Algo_IntraPred_LUMA_4x4"
  dst-port="Y_UP"
  src="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Mgnt_Intra4x4" src-port="Y_UP4"/>
<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Algo_IntraPred_LUMA_4x4"
  dst-port="Y_UP_LEFT"
  src="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Mgnt_Intra4x4" src-port="Y_UP_LEFT4"/>
<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Algo_Add_4x4"
  dst-port="X"
  src="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Algo_IntraPred_LUMA_4x4" src-port="MPR"/>
<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Algo_IntraPred_LUMA_4x4"
  dst-port="AVAIL"
  src="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Mgnt_Intra4x4" src-port="AVAIL"/>
<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Algo_Merge_4x4_to_16x16_norasterscan"
  dst-port="X"
  src="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Algo_Add_4x4" src-port="Z"/>
<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Algo_Add_4x4"
  dst-port="Y"
  src="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Algo_Split_16x16_to_4x4_norasterscan" src-port="Y"/>
<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Algo_IntraPred_LUMA_4x4"
  dst-port="PRED_MODE"
  src="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Mgnt_Intra4x4" src-port="PRED_MODE4"/>
<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Mgnt_Intra4x4"
  dst-port="EDGE"
  src="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Buffer_Neighbour_4x4" src-port="EDGE"/>
<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Mgnt_Intra4x4"
  dst-port="MB_4X4"
  src="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Algo_Add_4x4" src-port="Z"/>
<Connection dst="AVCDecoder_Decoding_Y_PredictionY_Select"
  dst-port="IN_1"
  src="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Algo_Merge_4x4_to_16x16_norasterscan" src-port="Y"/>
<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Algo_Split_16x16_to_4x4_norasterscan"
  dst-port="X"
  src="AVCDecoder_Decoding_Y_PredictionY_DemuxParserInfos" src-port="COEF_ACR_INTRA4"/>
<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Mgnt_Intra4x4"
  dst-port="PRED_MODE"
  src="AVCDecoder_Decoding_Y_PredictionY_DemuxParserInfos" src-port="PRED_MODE4"/>
<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Buffer_Neighbour_4x4"
  dst-port="MB_IN" src="AVCDecoder_Decoding_Y_PredictionY_Select" src-port="OUT"/>
<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Mgnt_Intra4x4"

```

```

dst-port="NEIGHBOUR"
src="AVCDecoder_Decoding_Y_PredictionY_DemuxParserInfos" src-port="NEIGHBOUR4"/>
<Connection dst="AVCDecoder_Decoding_Y_PredictionY_Inter_Add_Clip"
dst-port="X"
src="AVCDecoder_Decoding_Y_PredictionY_Inter_Algo_Interp_Reord" src-port="MBPRED"/>
<Connection
dst="AVCDecoder_Decoding_Y_PredictionY_Inter_Algo_Interp_Reord"
dst-port="INTERP"
src="AVCDecoder_Decoding_Y_PredictionY_Inter_Algo_Interp_SeparableSixTapQuarterPel" src-port="INTERP"/>
<Connection dst="AVCDecoder_Decoding_Y_PredictionY_Select"
dst-port="IN_3"
src="AVCDecoder_Decoding_Y_PredictionY_Inter_Add_Clip" src-port="Z"/>
<Connection dst="AVCDecoder_Decoding_Y_PredictionY_Inter_Add_Clip"
dst-port="Y"
src="AVCDecoder_Decoding_Y_PredictionY_DemuxParserInfos" src-port="COEF_ACR_INTER"/>
<Connection dst="AVCDecoder_Decoding_Y_DecodedPictureBuffer"
dst-port="WD"
src="AVCDecoder_Decoding_Y_PredictionY_Deblocking_Filter" src-port="MB_OUT"/>
<Connection
dst="AVCDecoder_Decoding_Y_PredictionY_Inter_Algo_Interp_SeparableSixTapQuarterPel"
dst-port="RD" src="AVCDecoder_Decoding_Y_DecodedPictureBuffer" src-port="RD"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_DCR_Hadamard_LUMA_IHT1d_0"
dst-port="X"
src="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_IS_Zigzag_4x4_DC" src-port="Y"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_DC_Transpose4x4_0"
dst-port="X"
src="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_DCR_Hadamard_LUMA_IHT1d_0" src-port="Y"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_DCR_Hadamard_LUMA_IHT1d_1"
dst-port="X"
src="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_DC_Transpose4x4_0" src-port="Y"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_DCR_Hadamard_LUMA_Scaling"
dst-port="I"
src="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_DCR_Hadamard_LUMA_IHT1d_1" src-port="Y"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_DC_Transpose4x4_1"
dst-port="X"
src="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_DCR_Hadamard_LUMA_Scaling" src-port="0"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_DCR_Hadamard_LUMA_Reordering"
dst-port="X"
src="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_DC_Transpose4x4_1" src-port="Y"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_Mgnt_IQ_INTRA16x16"
dst-port="COEF_AC"
src="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_Algo_IS_Zigzag_4x4_AC" src-port="Y"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_Algo_IQ_QSAndSLAndIDCTScaler_4x4"
dst-port="IN"
src="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_Mgnt_IQ_INTRA16x16" src-port="0"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_Algo_IQ_QSAndSLAndIDCTScaler_4x4"
dst-port="IS_DC"
src="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_Mgnt_IQ_INTRA16x16" src-port="IS_DC"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_IT4x4_Algo_Transpose4x4_0"
dst-port="X"

```

```

src="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_IT4x4_Algo_IT4x4_1d_0" src-port="Y"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_IT4x4_Algo_IT4x4_1d_1"
dst-port="X"
src="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_IT4x4_Algo_Transpose4x4_0" src-port="Y"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_IT4x4_Algo_Transpose4x4_1"
dst-port="X"
src="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_IT4x4_Algo_IT4x4_1d_1" src-port="Y"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_IT4x4_Algo_IT4x4_Addshift"
dst-port="X"
src="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_IT4x4_Algo_Transpose4x4_1" src-port="Y"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_IT4x4_Algo_IT4x4_1d_0"
dst-port="X"
src="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_Algo_IQ_QSAndSLAndIDCTScaler_4x4" src-port="OUT"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_Algo_Merge_4x4_to_16x16"
dst-port="X"
src="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_IT4x4_Algo_IT4x4_Addshift" src-port="Y"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_Mgmt_IQ_INTRA16x16"
dst-port="COEF_DC"
src="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_DCR_Hadamard_LUMA_Reordering" src-port="Y"/>
<Connection dst="AVCDecoder_Decoding_Y_PredictionY_DemuxParserInfos"
dst-port="COEF_ACR"
src="AVCDecoder_Decoding_Y_ResidualY_Algo_Merge_4x4_to_16x16" src-port="Y"/>
<Connection dst="AVCDecoder_Decoding_Y_DecodedPictureBuffer"
dst-port="PIC_SIZE_IN_MB" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="PIC_SIZE_IN_MB"/>
<Connection
dst="AVCDecoder_Decoding_Y_PredictionY_Deblocking_Filter"
dst-port="PIC_SIZE_IN_MB" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="PIC_SIZE_IN_MB"/>
<Connection dst="AVCDecoder_Decoding_Y_PredictionY_DemuxParserInfos"
dst-port="CONSTRAINED_IFLAG"
src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="CONSTRAINED_IFLAG"/>
<Connection
dst="AVCDecoder_Decoding_Y_PredictionY_Inter_Algo_Interp_Reord"
dst-port="SUB_MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="SUB_MB_TYPE"/>
<Connection dst="AVCDecoder_Decoding_Y_DecodedPictureBuffer"
dst-port="FRAME_NUM" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="FRAME_NUM"/>
<Connection dst="AVCDecoder_Decoding_Y_DecodedPictureBuffer"
dst-port="DEL_LIST"
src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MMCO" src-port="DEL_LIST"/>
<Connection
dst="AVCDecoder_Decoding_Y_PredictionY_Inter_Algo_Interp_SeparableSixTapQuarterPel"
dst-port="MV"
src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL0_Reconstr" src-port="MV_OUT"/>
<Connection
dst="AVCDecoder_Decoding_Y_PredictionY_Inter_Algo_Interp_SeparableSixTapQuarterPel"
dst-port="PARTSZ"
src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvComponentReord" src-port="PARTSZ"/>
<Connection dst="AVCDecoder_Decoding_Y_DecodedPictureBuffer"
dst-port="ENABLE_READ"
src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL0_Reconstr" src-port="IS_NEW_MB_PIC"/>
<Connection dst="AVCDecoder_Decoding_Y_PredictionY_Select"
dst-port="IN_0" src="AVCDecoder_Syn_Parser_Algo_Parser_IPCM" src-port="L_PCM_Y"/>
<Connection dst="AVCDecoder_Decoding_Y_PredictionY_DemuxParserInfos"
dst-port="MB_LOCATION" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_LOCATION"/>
<Connection

```

```

dst="AVCDecoder_Decoding_Y_PredictionY_Deblocking_Filter"
dst-port="MB_LOCATION" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_LOCATION"/>
<Connection
dst="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Buffer_Neighbour_FullMb"
dst-port="MB_LOCATION" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_LOCATION"/>
<Connection
dst="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Buffer_Neighbour_4x4"
dst-port="MB_LOCATION" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_LOCATION"/>
<Connection dst="AVCDecoder_Decoding_Y_DecodedPictureBuffer"
dst-port="FRAME_TO_READ"
src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_RefIdxL0ToFrameNum" src-port="READ_FRAME"/>
<Connection
dst="AVCDecoder_Decoding_Y_PredictionY_Deblocking_Filter"
dst-port="BS"
src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MgntDeblockingFilter" src-port="BS"/>
<Connection
dst="AVCDecoder_Decoding_Y_PredictionY_Deblocking_Filter"
dst-port="MB_DBF_PARAM"
src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MgntDeblockingFilter" src-port="MB_DBF_PARAM_Y"/>
<Connection dst="AVCDecoder_Decoding_Y_DecodedPictureBuffer"
dst-port="MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_TYPE"/>
<Connection dst="AVCDecoder_Decoding_Y_PredictionY_DemuxParserInfos"
dst-port="MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_TYPE"/>
<Connection
dst="AVCDecoder_Decoding_Y_PredictionY_Deblocking_Filter"
dst-port="MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_TYPE"/>
<Connection
dst="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Buffer_Neighbour_FullMb"
dst-port="MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_TYPE"/>
<Connection
dst="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Buffer_Neighbour_4x4"
dst-port="MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_TYPE"/>
<Connection
dst="AVCDecoder_Decoding_Y_PredictionY_Inter_Algo_Interp_Reord"
dst-port="MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_TYPE"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_DCR_Hadamard_LUMA_Scaling"
dst-port="MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_TYPE"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_Mgnt_IQ_INTRA16x16"
dst-port="MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_TYPE"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_DCR_Hadamard_LUMA_Scaling"
dst-port="QP" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="QP"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_Algo_IQ_QSAndSLAndIDCTScaler_4x4"
dst-port="QP" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="QP"/>
<Connection dst="AVCDecoder_Decoding_Y_DecodedPictureBuffer"
dst-port="RA"
src="AVCDecoder_Syn_Parser_Generate_Inter_Info_Mgnt_InterPred" src-port="R_ADDR_L"/>
<Connection dst="AVCDecoder_Decoding_Y_PredictionY_DemuxParserInfos"
dst-port="PRED_MODE" src="AVCDecoder_Syn_Parser_IntraPredSplit" src-port="PRED_MODE"/>
<Connection dst="AVCDecoder_Decode_U_PredictionC_Deblocking_Filter"
dst-port="MB_IN"
src="AVCDecoder_Decode_U_PredictionC_Select_chroma" src-port="OUT"/>
<Connection dst="AVCDecoder_Decode_U_PredictionC_Select_chroma"
dst-port="SELECT"
src="AVCDecoder_Decode_U_PredictionC_DemuxParserInfos" src-port="SELECT"/>
<Connection
dst="AVCDecoder_Decode_U_PredictionC_Intra8x8_Mgnt_Intra_8x8"

```

```

dst-port="EDGE"
src="AVCDecoder_Decode_U_PredictionC_Intra8x8_Buffer_Neighbour_FullMb" src-port="EDGE"/>
<Connection
dst="AVCDecoder_Decode_U_PredictionC_Intra8x8_Algo_IntraPred_CHROMA"
dst-port="AVAIL"
src="AVCDecoder_Decode_U_PredictionC_Intra8x8_Mgmt_Intra_8x8" src-port="AVAIL"/>
<Connection
dst="AVCDecoder_Decode_U_PredictionC_Intra8x8_Algo_IntraPred_CHROMA"
dst-port="C_LEFT"
src="AVCDecoder_Decode_U_PredictionC_Intra8x8_Mgmt_Intra_8x8" src-port="Y_LEFT"/>
<Connection
dst="AVCDecoder_Decode_U_PredictionC_Intra8x8_Algo_IntraPred_CHROMA"
dst-port="C_UP"
src="AVCDecoder_Decode_U_PredictionC_Intra8x8_Mgmt_Intra_8x8" src-port="Y_UP"/>
<Connection
dst="AVCDecoder_Decode_U_PredictionC_Intra8x8_Algo_IntraPred_CHROMA"
dst-port="C_UP_LEFT"
src="AVCDecoder_Decode_U_PredictionC_Intra8x8_Mgmt_Intra_8x8" src-port="Y_UP_LEFT"/>
<Connection dst="AVCDecoder_Decode_U_PredictionC_Intra8x8_Algo_Add"
dst-port="X"
src="AVCDecoder_Decode_U_PredictionC_Intra8x8_Algo_IntraPred_CHROMA" src-port="MPR"/>
<Connection dst="AVCDecoder_Decode_U_PredictionC_Select_chroma"
dst-port="IN_1"
src="AVCDecoder_Decode_U_PredictionC_Intra8x8_Algo_Add" src-port="Z"/>
<Connection dst="AVCDecoder_Decode_U_PredictionC_Intra8x8_Algo_Add"
dst-port="Y"
src="AVCDecoder_Decode_U_PredictionC_DemuxParserInfos" src-port="COEF_ACR_INTRA"/>
<Connection
dst="AVCDecoder_Decode_U_PredictionC_Intra8x8_Buffer_Neighbour_FullMb"
dst-port="MB_IN"
src="AVCDecoder_Decode_U_PredictionC_Select_chroma" src-port="OUT"/>
<Connection
dst="AVCDecoder_Decode_U_PredictionC_Intra8x8_Mgmt_Intra_8x8"
dst-port="NEIGHBOUR"
src="AVCDecoder_Decode_U_PredictionC_DemuxParserInfos" src-port="NEIGHBOUR_INTRA"/>
<Connection dst="AVCDecoder_Decode_U_PredictionC_Inter_Algo_Add"
dst-port="X"
src="AVCDecoder_Decode_U_PredictionC_Inter_Algo_Interp_Reord" src-port="MBPRED"/>
<Connection
dst="AVCDecoder_Decode_U_PredictionC_Inter_Algo_Interp_Reord"
dst-port="INTERP"
src="AVCDecoder_Decode_U_PredictionC_Inter_Algo_Interp_EighthPelBilinear" src-port="INTERP"/>
<Connection dst="AVCDecoder_Decode_U_PredictionC_Select_chroma"
dst-port="IN_2"
src="AVCDecoder_Decode_U_PredictionC_Inter_Algo_Add" src-port="Z"/>
<Connection dst="AVCDecoder_Decode_U_PredictionC_Inter_Algo_Add"
dst-port="Y"
src="AVCDecoder_Decode_U_PredictionC_DemuxParserInfos" src-port="COEF_ACR_INTER"/>
<Connection dst="AVCDecoder_Decode_U_Buffer_inter" dst-port="WD"
src="AVCDecoder_Decode_U_PredictionC_Deblocking_Filter" src-port="MB_OUT"/>
<Connection
dst="AVCDecoder_Decode_U_PredictionC_Inter_Algo_Interp_EighthPelBilinear"
dst-port="RD" src="AVCDecoder_Decode_U_Buffer_inter" src-port="RD"/>
<Connection
dst="AVCDecoder_Decode_U_ResidualC_IS_IQ_IT_C_Mgmt_IQ_Chroma"
dst-port="COEF_AC"
src="AVCDecoder_Decode_U_ResidualC_IS_IQ_IT_C_Algo_IS_Zigzag_4x4" src-port="Y"/>
<Connection
dst="AVCDecoder_Decode_U_ResidualC_IS_IQ_IT_C_Algo_IQ_QSAndSLAndIDCTScaler_4x4"
dst-port="IN"

```

```

src="AVCDecoder_Decompose_U_ResidualC_IS_IQ_IT_C_Mgmt_IQ_Chroma" src-port="0"/>
<Connection
dst="AVCDecoder_Decompose_U_ResidualC_IS_IQ_IT_C_Algo_IQ_QSAndSLAndIDCTScaler_4x4"
dst-port="IS_DC"
src="AVCDecoder_Decompose_U_ResidualC_IS_IQ_IT_C_Mgmt_IQ_Chroma" src-port="IS_DC"/>
<Connection
dst="AVCDecoder_Decompose_U_ResidualC_IS_IQ_IT_C_IT4x4_Algo_Transpose4x4_0"
dst-port="X"
src="AVCDecoder_Decompose_U_ResidualC_IS_IQ_IT_C_IT4x4_Algo_IT4x4_1d_0" src-port="Y"/>
<Connection
dst="AVCDecoder_Decompose_U_ResidualC_IS_IQ_IT_C_IT4x4_Algo_IT4x4_1d_1"
dst-port="X"
src="AVCDecoder_Decompose_U_ResidualC_IS_IQ_IT_C_IT4x4_Algo_Transpose4x4_0" src-port="Y"/>
<Connection
dst="AVCDecoder_Decompose_U_ResidualC_IS_IQ_IT_C_IT4x4_Algo_Transpose4x4_1"
dst-port="X"
src="AVCDecoder_Decompose_U_ResidualC_IS_IQ_IT_C_IT4x4_Algo_IT4x4_1d_1" src-port="Y"/>
<Connection
dst="AVCDecoder_Decompose_U_ResidualC_IS_IQ_IT_C_IT4x4_Algo_IT4x4_Addshift"
dst-port="X"
src="AVCDecoder_Decompose_U_ResidualC_IS_IQ_IT_C_IT4x4_Algo_Transpose4x4_1" src-port="Y"/>
<Connection
dst="AVCDecoder_Decompose_U_ResidualC_IS_IQ_IT_C_IT4x4_Algo_IT4x4_1d_0"
dst-port="X"
src="AVCDecoder_Decompose_U_ResidualC_IS_IQ_IT_C_Algo_IQ_QSAndSLAndIDCTScaler_4x4" src-port="OUT"/>
<Connection
dst="AVCDecoder_Decompose_U_ResidualC_Algo_Merge_4x4_to_8x8"
dst-port="X"
src="AVCDecoder_Decompose_U_ResidualC_IS_IQ_IT_C_IT4x4_Algo_IT4x4_Addshift" src-port="Y"/>
<Connection
dst="AVCDecoder_Decompose_U_ResidualC_IS_IQ_IT_C_Mgmt_IQ_Chroma"
dst-port="COEF_DC"
src="AVCDecoder_Decompose_U_ResidualC_Algo_DCR_Hadamard_CHROMA" src-port="OUT"/>
<Connection dst="AVCDecoder_Decompose_U_PredictionC_DemuxParserInfos"
dst-port="COEF_ACR"
src="AVCDecoder_Decompose_U_ResidualC_Algo_Merge_4x4_to_8x8" src-port="Y"/>
<Connection dst="AVCDecoder_Decompose_U_Buffer_inter"
dst-port="PIC_SIZE_IN_MB" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="PIC_SIZE_IN_MB"/>
<Connection dst="AVCDecoder_Decompose_U_PredictionC_Deblocking_Filter"
dst-port="PIC_SIZE_IN_MB" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="PIC_SIZE_IN_MB"/>
<Connection
dst="AVCDecoder_Decompose_U_PredictionC_Intra8x8_Algo_IntraPred_CHROMA"
dst-port="PredMode" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="INTRA_PRED_MODEC"/>
<Connection dst="AVCDecoder_Decompose_U_PredictionC_DemuxParserInfos"
dst-port="CONSTRAINED_IFLAG"
src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="CONSTRAINED_IFLAG"/>
<Connection
dst="AVCDecoder_Decompose_U_PredictionC_Inter_Algo_Interp_Reord"
dst-port="SUB_MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="SUB_MB_TYPE"/>
<Connection dst="AVCDecoder_Decompose_U_Buffer_inter"
dst-port="FRAME_NUM" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="FRAME_NUM"/>
<Connection dst="AVCDecoder_Decompose_U_Buffer_inter"
dst-port="DEL_LIST"
src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MMC0" src-port="DEL_LIST"/>
<Connection
dst="AVCDecoder_Decompose_U_PredictionC_Inter_Algo_Interp_EighthPelBilinear"
dst-port="MV"
src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL0_Reconstr" src-port="MV_OUT"/>
<Connection
dst="AVCDecoder_Decompose_U_PredictionC_Inter_Algo_Interp_EighthPelBilinear"

```

STANDARDISO.COM · Click to view the full PDF of ISO/IEC 23002-4:2018

```

dst-port="PARTSZ"
src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvComponentReord" src-port="PARTSZ"/>
<Connection dst="AVCDecoder_Decode_U_Buffer_inter" dst-port="RA"
src="AVCDecoder_Syn_Parser_Generate_Inter_Info_Mgnt_InterPred" src-port="R_ADDR_C"/>
<Connection dst="AVCDecoder_Decode_U_Buffer_inter"
dst-port="ENABLE_READ"
src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL0_Reconstr" src-port="IS_NEW_MB_PIC"/>
<Connection dst="AVCDecoder_Decode_U_PredictionC_Select_chroma"
dst-port="IN_0" src="AVCDecoder_Syn_Parser_Algo_Parser_IPCM" src-port="L_PCM_U"/>
<Connection dst="AVCDecoder_Decode_U_PredictionC_DemuxParserInfos"
dst-port="MB_LOCATION" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_LOCATION"/>
<Connection dst="AVCDecoder_Decode_U_PredictionC_Deblocking_Filter"
dst-port="MB_LOCATION" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_LOCATION"/>
<Connection
dst="AVCDecoder_Decode_U_PredictionC_Intra8x8_Buffer_Neighbour_FullMb"
dst-port="MB_LOCATION" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_LOCATION"/>
<Connection dst="AVCDecoder_Decode_U_Buffer_inter"
dst-port="FRAME_TO_READ"
src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_RefIdxL0ToFrameNum" src-port="READ_FRAME"/>
<Connection dst="AVCDecoder_Decode_U_PredictionC_Deblocking_Filter"
dst-port="BS"
src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MgntDeblockingFilter" src-port="BS"/>
<Connection dst="AVCDecoder_Decode_U_PredictionC_Deblocking_Filter"
dst-port="MB_DBF_PARAM"
src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MgntDeblockingFilter" src-port="MB_DBF_PARAM_U"/>
<Connection dst="AVCDecoder_Decode_U_Buffer_inter"
dst-port="MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_TYPE"/>
<Connection dst="AVCDecoder_Decode_U_PredictionC_Deblocking_Filter"
dst-port="MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_TYPE"/>
<Connection dst="AVCDecoder_Decode_U_PredictionC_DemuxParserInfos"
dst-port="MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_TYPE"/>
<Connection
dst="AVCDecoder_Decode_U_PredictionC_Intra8x8_Buffer_Neighbour_FullMb"
dst-port="MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_TYPE"/>
<Connection
dst="AVCDecoder_Decode_U_PredictionC_Inter_Algo_Interp_Reord"
dst-port="MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_TYPE"/>
<Connection
dst="AVCDecoder_Decode_U_ResidualC_Algo_DCR_Hadamard_CHROMA"
dst-port="QP" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="QP_Cb"/>
<Connection
dst="AVCDecoder_Decode_U_ResidualC_IS_IQ_IT_C_Algo_IQ_QSAndSIAndIDCTScaler_4x4"
dst-port="QP" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="QP_Cb"/>
<Connection dst="AVCDecoder_Decode_V_PredictionC_Deblocking_Filter"
dst-port="MB_IN"
src="AVCDecoder_Decode_V_PredictionC_Select_chroma" src-port="OUT"/>
<Connection dst="AVCDecoder_Decode_V_PredictionC_Select_chroma"
dst-port="SELECT"
src="AVCDecoder_Decode_V_PredictionC_DemuxParserInfos" src-port="SELECT"/>
<Connection
dst="AVCDecoder_Decode_V_PredictionC_Intra8x8_Mgnt_Intra_8x8"
dst-port="EDGE"
src="AVCDecoder_Decode_V_PredictionC_Intra8x8_Buffer_Neighbour_FullMb" src-port="EDGE"/>
<Connection
dst="AVCDecoder_Decode_V_PredictionC_Intra8x8_Algo_IntraPred_CHROMA"
dst-port="AVAIL"
src="AVCDecoder_Decode_V_PredictionC_Intra8x8_Mgnt_Intra_8x8" src-port="AVAIL"/>
<Connection
dst="AVCDecoder_Decode_V_PredictionC_Intra8x8_Algo_IntraPred_CHROMA"
dst-port="C_LEFT"

```

```

src="AVCDecoder_Decode_V_PredictionC_Intra8x8_Mgnt_Intra_8x8" src-port="Y_LEFT"/>
<Connection
dst="AVCDecoder_Decode_V_PredictionC_Intra8x8_Algo_IntraPred_CHROMA"
dst-port="C_UP"
src="AVCDecoder_Decode_V_PredictionC_Intra8x8_Mgnt_Intra_8x8" src-port="Y_UP"/>
<Connection
dst="AVCDecoder_Decode_V_PredictionC_Intra8x8_Algo_IntraPred_CHROMA"
dst-port="C_UP_LEFT"
src="AVCDecoder_Decode_V_PredictionC_Intra8x8_Mgnt_Intra_8x8" src-port="Y_UP_LEFT"/>
<Connection dst="AVCDecoder_Decode_V_PredictionC_Intra8x8_Algo_Add"
dst-port="X"
src="AVCDecoder_Decode_V_PredictionC_Intra8x8_Algo_IntraPred_CHROMA" src-port="MPR"/>
<Connection dst="AVCDecoder_Decode_V_PredictionC_Select_chroma"
dst-port="IN_1"
src="AVCDecoder_Decode_V_PredictionC_Intra8x8_Algo_Add" src-port="Z"/>
<Connection dst="AVCDecoder_Decode_V_PredictionC_Intra8x8_Algo_Add"
dst-port="Y"
src="AVCDecoder_Decode_V_PredictionC_DemuxParserInfos" src-port="COEF_ACR_INTRA"/>
<Connection
dst="AVCDecoder_Decode_V_PredictionC_Intra8x8_Buffer_Neighbour_FullMb"
dst-port="MB_IN"
src="AVCDecoder_Decode_V_PredictionC_Select_chroma" src-port="OUT"/>
<Connection
dst="AVCDecoder_Decode_V_PredictionC_Intra8x8_Mgnt_Intra_8x8"
dst-port="NEIGHBOUR"
src="AVCDecoder_Decode_V_PredictionC_DemuxParserInfos" src-port="NEIGHBOUR_INTRA"/>
<Connection dst="AVCDecoder_Decode_V_PredictionC_Inter_Algo_Add"
dst-port="X"
src="AVCDecoder_Decode_V_PredictionC_Inter_Algo_Interp_Reord" src-port="MBPRED"/>
<Connection
dst="AVCDecoder_Decode_V_PredictionC_Inter_Algo_Interp_Reord"
dst-port="INTERP"
src="AVCDecoder_Decode_V_PredictionC_Inter_Algo_Interp_EighthPelBilinear" src-port="INTERP"/>
<Connection dst="AVCDecoder_Decode_V_PredictionC_Select_chroma"
dst-port="IN_2"
src="AVCDecoder_Decode_V_PredictionC_Inter_Algo_Add" src-port="Z"/>
<Connection dst="AVCDecoder_Decode_V_PredictionC_Inter_Algo_Add"
dst-port="Y"
src="AVCDecoder_Decode_V_PredictionC_DemuxParserInfos" src-port="COEF_ACR_INTER"/>
<Connection dst="AVCDecoder_Decode_V_Buffer_inter" dst-port="WD"
src="AVCDecoder_Decode_V_PredictionC_Deblocking_Filter" src-port="MB_OUT"/>
<Connection
dst="AVCDecoder_Decode_V_PredictionC_Inter_Algo_Interp_EighthPelBilinear"
dst-port="RD" src="AVCDecoder_Decode_V_Buffer_inter" src-port="RD"/>
<Connection
dst="AVCDecoder_Decode_V_ResidualC_IS_IQ_IT_C_Mgnt_IQ_Chroma"
dst-port="COEF_AC"
src="AVCDecoder_Decode_V_ResidualC_IS_IQ_IT_C_Algo_IS_Zigzag_4x4" src-port="Y"/>
<Connection
dst="AVCDecoder_Decode_V_ResidualC_IS_IQ_IT_C_Algo_IQ_QSAndSLAndIDCTScaler_4x4"
dst-port="IN"
src="AVCDecoder_Decode_V_ResidualC_IS_IQ_IT_C_Mgnt_IQ_Chroma" src-port="O"/>
<Connection
dst="AVCDecoder_Decode_V_ResidualC_IS_IQ_IT_C_Algo_IQ_QSAndSLAndIDCTScaler_4x4"
dst-port="IS_DC"
src="AVCDecoder_Decode_V_ResidualC_IS_IQ_IT_C_Mgnt_IQ_Chroma" src-port="IS_DC"/>
<Connection
dst="AVCDecoder_Decode_V_ResidualC_IS_IQ_IT_C_IT4x4_Algo_Transpose4x4_0"
dst-port="X"
src="AVCDecoder_Decode_V_ResidualC_IS_IQ_IT_C_IT4x4_Algo_IT4x4_1d_0" src-port="Y"/>

```

```

<Connection
  dst="AVCDecoder_Decode_V_ResidualC_IS_IQ_IT_C_IT4x4_Algo_IT4x4_1d_1"
  dst-port="X"
  src="AVCDecoder_Decode_V_ResidualC_IS_IQ_IT_C_IT4x4_Algo_Transpose4x4_0" src-port="Y"/>
<Connection
  dst="AVCDecoder_Decode_V_ResidualC_IS_IQ_IT_C_IT4x4_Algo_Transpose4x4_1"
  dst-port="X"
  src="AVCDecoder_Decode_V_ResidualC_IS_IQ_IT_C_IT4x4_Algo_IT4x4_1d_1" src-port="Y"/>
<Connection
  dst="AVCDecoder_Decode_V_ResidualC_IS_IQ_IT_C_IT4x4_Algo_IT4x4_Addshift"
  dst-port="X"
  src="AVCDecoder_Decode_V_ResidualC_IS_IQ_IT_C_IT4x4_Algo_Transpose4x4_1" src-port="Y"/>
<Connection
  dst="AVCDecoder_Decode_V_ResidualC_IS_IQ_IT_C_IT4x4_Algo_IT4x4_1d_0"
  dst-port="X"
  src="AVCDecoder_Decode_V_ResidualC_IS_IQ_IT_C_Algo_IQ_QSAndSLAndIDCTScaler_4x4" src-port="OUT"/>
<Connection
  dst="AVCDecoder_Decode_V_ResidualC_Algo_Merge_4x4_to_8x8"
  dst-port="X"
  src="AVCDecoder_Decode_V_ResidualC_IS_IQ_IT_C_IT4x4_Algo_IT4x4_Addshift" src-port="Y"/>
<Connection
  dst="AVCDecoder_Decode_V_ResidualC_IS_IQ_IT_C_Mgmt_IQ_Chroma"
  dst-port="COEF_DC"
  src="AVCDecoder_Decode_V_ResidualC_Algo_DCR_Hadamard_CHROMA" src-port="OUT"/>
<Connection dst="AVCDecoder_Decode_V_PredictionC_DemuxParserInfos"
  dst-port="COEF_ACR"
  src="AVCDecoder_Decode_V_ResidualC_Algo_Merge_4x4_to_8x8" src-port="Y"/>
<Connection dst="AVCDecoder_Decode_V_Buffer_inter"
  dst-port="PIC_SIZE_IN_MB" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="PIC_SIZE_IN_MB"/>
<Connection dst="AVCDecoder_Decode_V_PredictionC_Deblocking_Filter"
  dst-port="PIC_SIZE_IN_MB" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="PIC_SIZE_IN_MB"/>
<Connection
  dst="AVCDecoder_Decode_V_PredictionC_Intra8x8_Algo_IntraPred_CHROMA"
  dst-port="PredMode" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="INTRA_PRED_MODEC"/>
<Connection dst="AVCDecoder_Decode_V_PredictionC_DemuxParserInfos"
  dst-port="CONSTRAINED_IFLAG"
  src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="CONSTRAINED_IFLAG"/>
<Connection
  dst="AVCDecoder_Decode_V_PredictionC_Inter_Algo_Interp_Reord"
  dst-port="SUB_MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="SUB_MB_TYPE"/>
<Connection dst="AVCDecoder_Decode_V_Buffer_inter"
  dst-port="FRAME_NUM" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="FRAME_NUM"/>
<Connection dst="AVCDecoder_Decode_V_Buffer_inter"
  dst-port="DEL_LIST"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MMCO" src-port="DEL_LIST"/>
<Connection
  dst="AVCDecoder_Decode_V_PredictionC_Inter_Algo_Interp_EighthPelBilinear"
  dst-port="MV"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL0_Reconstr" src-port="MV_OUT"/>
<Connection
  dst="AVCDecoder_Decode_V_PredictionC_Inter_Algo_Interp_EighthPelBilinear"
  dst-port="PARTSZ"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvComponentReord" src-port="PARTSZ"/>
<Connection dst="AVCDecoder_Decode_V_Buffer_inter"
  dst-port="ENABLE_READ"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL0_Reconstr" src-port="IS_NEW_MB_PIC"/>
<Connection dst="AVCDecoder_Decode_V_PredictionC_Select_chroma"
  dst-port="IN_0" src="AVCDecoder_Syn_Parser_Algo_Parser_IPCM" src-port="L_PCM_V"/>
<Connection dst="AVCDecoder_Decode_V_PredictionC_DemuxParserInfos"
  dst-port="MB_LOCATION" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_LOCATION"/>

```

```

<Connection dst="AVCDecoder_Decode_V_PredictionC_Deblocking_Filter"
  dst-port="MB_LOCATION" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_LOCATION"/>
<Connection
  dst="AVCDecoder_Decode_V_PredictionC_Intra8x8_Buffer_Neighbour_FullMb"
  dst-port="MB_LOCATION" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_LOCATION"/>
<Connection dst="AVCDecoder_Decode_V_Buffer_inter"
  dst-port="FRAME_TO_READ"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_RefIdxL0ToFrameNum" src-port="READ_FRAME"/>
<Connection dst="AVCDecoder_Decode_V_PredictionC_Deblocking_Filter"
  dst-port="BS"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MgntDeblockingFilter" src-port="BS"/>
<Connection dst="AVCDecoder_Decode_V_Buffer_inter"
  dst-port="MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_TYPE"/>
<Connection dst="AVCDecoder_Decode_V_PredictionC_Deblocking_Filter"
  dst-port="MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_TYPE"/>
<Connection dst="AVCDecoder_Decode_V_PredictionC_DemuxParserInfos"
  dst-port="MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_TYPE"/>
<Connection
  dst="AVCDecoder_Decode_V_PredictionC_Intra8x8_Buffer_Neighbour_FullMb"
  dst-port="MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_TYPE"/>
<Connection
  dst="AVCDecoder_Decode_V_PredictionC_Inter_Algo_Interp_Reord"
  dst-port="MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_TYPE"/>
<Connection
  dst="AVCDecoder_Decode_V_ResidualC_Algo_DCR_Hadamard_CHROMA"
  dst-port="QP" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="QP_Cr"/>
<Connection
  dst="AVCDecoder_Decode_V_ResidualC_IS_IQ_IT_C_Algo_IQ_QSAndSLAndIDCTScaler_4x4"
  dst-port="QP" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="QP_Cr"/>
<Connection dst="AVCDecoder_Decode_V_Buffer_inter" dst-port="RA"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_Mgnt_InterPred" src-port="R_ADDR_C"/>
<Connection dst="AVCDecoder_Decode_V_PredictionC_Deblocking_Filter"
  dst-port="MB_DBF_PARAM"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MgntDeblockingFilter" src-port="MB_DBF_PARAM_V"/>
<Connection dst="AVCDecoder_CavlcExpand_BlockSplit" dst-port="BLOCK"
  src="AVCDecoder_CavlcExpand_BlockExpand" src-port="BLOCK"/>
<Connection
  dst="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_Algo_IS_Zigzag_4x4_AC"
  dst-port="X" src="AVCDecoder_CavlcExpand_BlockSplit" src-port="BLOCK_Y_AC"/>
<Connection
  dst="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_IS_Zigzag_4x4_DC"
  dst-port="X" src="AVCDecoder_CavlcExpand_BlockSplit" src-port="BLOCK_Y_DC"/>
<Connection
  dst="AVCDecoder_Decode_U_ResidualC_IS_IQ_IT_C_Algo_IS_Zigzag_4x4"
  dst-port="X" src="AVCDecoder_CavlcExpand_BlockSplit" src-port="BLOCK_U_AC"/>
<Connection
  dst="AVCDecoder_Decode_U_ResidualC_Algo_DCR_Hadamard_CHROMA"
  dst-port="IN" src="AVCDecoder_CavlcExpand_BlockSplit" src-port="BLOCK_U_DC"/>
<Connection
  dst="AVCDecoder_Decode_V_ResidualC_IS_IQ_IT_C_Algo_IS_Zigzag_4x4"
  dst-port="X" src="AVCDecoder_CavlcExpand_BlockSplit" src-port="BLOCK_V_AC"/>
<Connection
  dst="AVCDecoder_Decode_V_ResidualC_Algo_DCR_Hadamard_CHROMA"
  dst-port="IN" src="AVCDecoder_CavlcExpand_BlockSplit" src-port="BLOCK_V_DC"/>
<Connection dst="AVCDecoder_CavlcExpand_BlockExpand"
  dst-port="MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_TYPE"/>
<Connection dst="AVCDecoder_CavlcExpand_BlockSplit"
  dst-port="MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_TYPE"/>
<Connection dst="AVCDecoder_CavlcExpand_BlockExpand" dst-port="LAST"
  src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="LAST"/>

```

STANDARDSISO.COM · Click to view the full PDF of ISO/IEC 23002-4:2018

```

<Connection dst="AVCDecoder_CavlcExpand_BlockExpand" dst-port="RUN"
  src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="RUN"/>
<Connection dst="AVCDecoder_CavlcExpand_BlockExpand"
  dst-port="VALUE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="VALUE"/>
<Connection dst="AVCDecoder_Display_Render_DisplayRendererY"
  dst-port="POC_DISPLAY"
  src="AVCDecoder_Display_Render_Mgmt_Display_Poc" src-port="POC_DISPLAY"/>
<Connection dst="AVCDecoder_Display_Render_DisplayRendererV"
  dst-port="POC_DISPLAY"
  src="AVCDecoder_Display_Render_Mgmt_Display_Poc" src-port="POC_DISPLAY"/>
<Connection dst="AVCDecoder_Display_Render_DisplayRendererU"
  dst-port="POC_DISPLAY"
  src="AVCDecoder_Display_Render_Mgmt_Display_Poc" src-port="POC_DISPLAY"/>
<Connection dst="AVCDecoder_Display_Render_DisplayRendererY"
  dst-port="POC" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="POC"/>
<Connection dst="AVCDecoder_Display_Render_DisplayRendererU"
  dst-port="POC" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="POC"/>
<Connection dst="AVCDecoder_Display_Render_DisplayRendererV"
  dst-port="POC" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="POC"/>
<Connection dst="AVCDecoder_Display_Render_Mgmt_Display_Poc"
  dst-port="POC" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="POC"/>
<Connection dst="AVCDecoder_Display_Render_DisplayRendererY"
  dst-port="MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_TYPE"/>
<Connection dst="AVCDecoder_Display_Render_DisplayRendererU"
  dst-port="MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_TYPE"/>
<Connection dst="AVCDecoder_Display_Render_DisplayRendererV"
  dst-port="MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_TYPE"/>
<Connection dst="AVCDecoder_Display_Render_Mgmt_Display_Poc"
  dst-port="MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_TYPE"/>
<Connection dst="AVCDecoder_Display_Render_Mgmt_Display_Poc"
  dst-port="EMPTY_DPB"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MMCO" src-port="EMPTY_DPB"/>
<Connection dst="AVCDecoder_Display_Render_Mgmt_Display_Poc"
  dst-port="DBP_MAX_SZ" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="DBP_MAX_SZ"/>
<Connection dst="AVCDecoder_Display_Render_DisplayRendererY"
  dst-port="WD"
  src="AVCDecoder_Decoding_Y_PredictionY_Deblocking_Filter" src-port="MB_OUT"/>
<Connection dst="AVCDecoder_Display_Render_DisplayRendererU"
  dst-port="WD"
  src="AVCDecoder_Decode_U_PredictionC_Deblocking_Filter" src-port="MB_OUT"/>
<Connection dst="AVCDecoder_Display_Render_DisplayRendererV"
  dst-port="WD"
  src="AVCDecoder_Decode_V_PredictionC_Deblocking_Filter" src-port="MB_OUT"/>
<Connection dst="Merger" dst-port="U"
  src="AVCDecoder_Display_Render_DisplayRendererU" src-port="0"/>
<Connection dst="Merger" dst-port="V"
  src="AVCDecoder_Display_Render_DisplayRendererV" src-port="0"/>
<Connection dst="Merger" dst-port="Y"
  src="AVCDecoder_Display_Render_DisplayRendererY" src-port="0"/>
<Connection dst="display" dst-port="WIDTH"
  src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="WIDTH"/>
<Connection dst="display" dst-port="HEIGHT"
  src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="HEIGHT"/>
<Connection dst="AVCDecoder_Syn_Parser_Algo_Synp" dst-port="BYTE"
  src="source" src-port="0"/>
</XDF>

```

B.4 FNL of HEVC Main Profile



FNL specification of the HEVC Main Profile decoder (refer to Top\_mpeg\_h\_part2\_main\_decoder.xdf file).

```

<?xml version="1.0" encoding="UTF-8"?>
<XDF name="Top_mpeg_h_part2_main">
  <Instance id="Source">
    <Class name="org.sc29.wg11.common.SourceTest"/>
  </Instance>
  <Instance id="display">
    <Class name="org.sc29.wg11.common.DisplayYUVWithCrop"/>
    <Parameter name="BLK_SIDE">
      <Expr kind="Literal" literal-kind="Integer" value="16"/>
    </Parameter>
  </Instance>
  <Instance id="HevcDecoder_Algo_Parser">
    <Class name="devel.org.sc29.wg11.mpeg_h_part2_main.synParser.Algo_Parser"/>
  </Instance>
  <Instance id="HevcDecoder_IntraPrediction">
    <Class name="devel.org.sc29.wg11.mpeg_h_part2_main.intra.IntraPrediction"/>
    <Parameter name="EnabSkipInPix">
      <Expr kind="Literal" literal-kind="Boolean" value="false"/>
    </Parameter>
  </Instance>
  <Instance id="HevcDecoder_SelectCU">
    <Class name="devel.org.sc29.wg11.mpeg_h_part2.SelectCu"/>
  </Instance>

```

```

<Instance id="HevcDecoder_DecodedPictureBuffer">
  <Class name="devel.org.sc29.wg11.mpeg.part2.main.inter.DecodedPictureBuffer"/>
</Instance>
<Instance id="HevcDecoder_InterPrediction">
  <Class name="devel.org.sc29.wg11.mpeg.part2.main.inter.InterPrediction"/>
</Instance>
<Instance id="HevcDecoder_SAO">
  <Class name="devel.org.sc29.wg11.mpeg.part2.main.Filters.SaoFilter"/>
  <Parameter name="DEBUG">
    <Expr kind="Literal" literal-kind="Boolean" value="true"/>
  </Parameter>
</Instance>
<Instance id="HevcDecoder_QpGen">
  <Class name="devel.org.sc29.wg11.mpeg.part2.main.Filters.QpGen"/>
</Instance>
<Instance id="HevcDecoder_xIT_IT_Splitter">
  <Class name="devel.org.sc29.wg11.mpeg.part2.main.IT.IT_Splitter"/>
</Instance>
<Instance id="HevcDecoder_xIT_IT_Merger">
  <Class name="devel.org.sc29.wg11.mpeg.part2.main.IT.IT_Merger"/>
</Instance>
<Instance id="HevcDecoder_xIT_invDST4x4_1st">
  <Class name="devel.org.sc29.wg11.mpeg.part2.main.IT.invDST4x4_1st"/>
</Instance>
<Instance id="HevcDecoder_xIT_invDST4x4_2nd">
  <Class name="devel.org.sc29.wg11.mpeg.part2.main.IT.invDST4x4_2nd"/>
</Instance>
<Instance id="HevcDecoder_xIT_Block_Merger">
  <Class name="devel.org.sc29.wg11.mpeg.part2.main.IT.Block_Merge"/>
</Instance>
<Instance id="HevcDecoder_xIT_IT4x4_IT4x4_1d_0">
  <Class name="devel.org.sc29.wg11.mpeg.part2.main.IT.IT4x4_1d"/>
  <Parameter name="shift">
    <Expr kind="Literal" literal-kind="Integer" value="7"/>
  </Parameter>
</Instance>
<Instance id="HevcDecoder_xIT_IT4x4_Transpose4x4_0">
  <Class name="devel.org.sc29.wg11.mpeg.part2.main.IT.Transpose4x4"/>
</Instance>
<Instance id="HevcDecoder_xIT_IT4x4_IT4x4_1d_1">
  <Class name="devel.org.sc29.wg11.mpeg.part2.main.IT.IT4x4_1d"/>
  <Parameter name="shift">
    <Expr kind="Literal" literal-kind="Integer" value="12"/>
  </Parameter>
</Instance>
<Instance id="HevcDecoder_xIT_IT4x4_Transpose4x4_1">
  <Class name="devel.org.sc29.wg11.mpeg.part2.main.IT.Transpose4x4"/>
</Instance>
<Instance id="HevcDecoder_xIT_IT8x8_IT8x8_1d_0">
  <Class name="devel.org.sc29.wg11.mpeg.part2.main.IT.IT8x8_1d"/>
  <Parameter name="shift">
    <Expr kind="Literal" literal-kind="Integer" value="7"/>
  </Parameter>
</Instance>
<Instance id="HevcDecoder_xIT_IT8x8_Transpose8x8_0">
  <Class name="devel.org.sc29.wg11.mpeg.part2.main.IT.Transpose8x8"/>
</Instance>
<Instance id="HevcDecoder_xIT_IT8x8_IT8x8_1d_1">
  <Class name="devel.org.sc29.wg11.mpeg.part2.main.IT.IT8x8_1d"/>
  <Parameter name="shift">

```

```

    <Expr kind="Literal" literal-kind="Integer" value="12"/>
  </Parameter>
</Instance>
<Instance id="HevcDecoder_xiT_IT8x8_Transpose8x8_1">
  <Class name="devel.org.sc29.wg11.mpeg.part2.main.IT.Transpose8x8"/>
</Instance>
<Instance id="HevcDecoder_xiT_IT16x16_IT16x16_1d_0">
  <Class name="devel.org.sc29.wg11.mpeg.part2.main.IT.IT16x16_1d"/>
  <Parameter name="shift">
    <Expr kind="Literal" literal-kind="Integer" value="7"/>
  </Parameter>
</Instance>
<Instance id="HevcDecoder_xiT_IT16x16_Transpose16x16_0">
  <Class name="devel.org.sc29.wg11.mpeg.part2.main.IT.Transpose16x16"/>
</Instance>
<Instance id="HevcDecoder_xiT_IT16x16_IT16x16_1d_1">
  <Class name="devel.org.sc29.wg11.mpeg.part2.main.IT.IT16x16_1d"/>
  <Parameter name="shift">
    <Expr kind="Literal" literal-kind="Integer" value="12"/>
  </Parameter>
</Instance>
<Instance id="HevcDecoder_xiT_IT16x16_Transpose16x16_1">
  <Class name="devel.org.sc29.wg11.mpeg.part2.main.IT.Transpose16x16"/>
</Instance>
<Instance id="HevcDecoder_xiT_IT32x32_IT32x32_1d_0">
  <Class name="devel.org.sc29.wg11.mpeg.part2.main.IT.IT32x32_1d"/>
  <Parameter name="shift">
    <Expr kind="Literal" literal-kind="Integer" value="7"/>
  </Parameter>
</Instance>
<Instance id="HevcDecoder_xiT_IT32x32_Transpose32x32_0">
  <Class name="devel.org.sc29.wg11.mpeg.part2.main.IT.Transpose32x32"/>
</Instance>
<Instance id="HevcDecoder_xiT_IT32x32_IT32x32_1d_1">
  <Class name="devel.org.sc29.wg11.mpeg.part2.main.IT.IT32x32_1d"/>
  <Parameter name="shift">
    <Expr kind="Literal" literal-kind="Integer" value="12"/>
  </Parameter>
</Instance>
<Instance id="HevcDecoder_xiT_IT32x32_Transpose32x32_1">
  <Class name="devel.org.sc29.wg11.mpeg.part2.main.IT.Transpose32x32"/>
</Instance>
<Instance id="HevcDecoder_DBFilter_DeblockFilt">
  <Class name="devel.org.sc29.wg11.mpeg.part2.main.Filters.DeblockingFilter"/>
  <Parameter name="FILT_HOR_EDGES">
    <Expr kind="Literal" literal-kind="Boolean" value="true"/>
  </Parameter>
  <Parameter name="FILT_VERT_EDGES">
    <Expr kind="Literal" literal-kind="Boolean" value="true"/>
  </Parameter>
  <Parameter name="DEBUG">
    <Expr kind="Literal" literal-kind="Boolean" value="false"/>
  </Parameter>
</Instance>
<Instance id="HevcDecoder_DBFilter_GenerateBs">
  <Class name="devel.org.sc29.wg11.mpeg.part2.main.Filters.GenerateBs"/>
</Instance>
<Instance id="HevcDecoder_generateInfo_GenerateRefList">
  <Class name="devel.org.sc29.wg11.mpeg.part2.main.inter.GenerateRefList"/>
</Instance>

```

STANDARDS19.COM : Click to view the full PDF of ISO/IEC 23002-4:2018

```

<Instance id="HevcDecoder_generateInfo_MvComponentPred">
  <Class name="devel.org.sc29.wg11.mpeg.part2.main.inter.MvComponentPred"/>
  <Parameter name="FIX_TMVP_REFIDX0">
    <Expr kind="Literal" literal-kind="Boolean" value="true"/>
  </Parameter>
</Instance>
<Instance id="MD5_padding">
  <Class name="devel.org.sc29.wg11.mpeg.part2.common.padding"/>
</Instance>
<Instance id="MD5_shifter">
  <Class name="devel.org.sc29.wg11.mpeg.part2.common.MD5Shifter"/>
</Instance>
<Instance id="MD5_compute">
  <Class name="devel.org.sc29.wg11.mpeg.part2.common.MD5Compute"/>
</Instance>
<Instance id="MD5_MD5SplitInfo">
  <Class name="devel.org.sc29.wg11.mpeg.part2.common.MD5SplitInfo"/>
  <Parameter name="BLK_SIDE">
    <Expr kind="Literal" literal-kind="Integer" value="16"/>
  </Parameter>
</Instance>
<Connection dst="HevcDecoder_IntraPrediction" dst-port="CUInfo"
  src="HevcDecoder_Algo_Parser" src-port="CUInfo"/>
<Connection dst="HevcDecoder_IntraPrediction"
  dst-port="IntraPredMode" src="HevcDecoder_Algo_Parser" src-port="IntraPredMode"/>
<Connection dst="HevcDecoder_IntraPrediction" dst-port="PictSize"
  src="HevcDecoder_Algo_Parser" src-port="PictSize"/>
<Connection dst="HevcDecoder_IntraPrediction" dst-port="PartMode"
  src="HevcDecoder_Algo_Parser" src-port="PartMode"/>
<Connection dst="HevcDecoder_IntraPrediction" dst-port="SplitTransform"
  src="HevcDecoder_Algo_Parser" src-port="SplitTransform"/>
<Connection dst="HevcDecoder_IntraPrediction" dst-port="SliceAddr"
  src="HevcDecoder_Algo_Parser" src-port="SliceAddr"/>
<Connection dst="HevcDecoder_SelectCU" dst-port="IntraSample"
  src="HevcDecoder_IntraPrediction" src-port="PredSample"/>
<Connection dst="HevcDecoder_SelectCU" dst-port="PartMode"
  src="HevcDecoder_Algo_Parser" src-port="PartMode"/>
<Connection dst="HevcDecoder_SelectCU" dst-port="CUInfo"
  src="HevcDecoder_Algo_Parser" src-port="CUInfo"/>
<Connection dst="HevcDecoder_IntraPrediction" dst-port="StrongIntraSmoothing"
  src="HevcDecoder_Algo_Parser" src-port="StrongIntraSmoothing"/>
<Connection dst="HevcDecoder_DecodedPictureBuffer" dst-port="Poc"
  src="HevcDecoder_Algo_Parser" src-port="Poc"/>
<Connection dst="HevcDecoder_DecodedPictureBuffer"
  dst-port="PictSize" src="HevcDecoder_Algo_Parser" src-port="PictSize"/>
<Connection dst="HevcDecoder_DecodedPictureBuffer"
  dst-port="RpsPoc" src="HevcDecoder_Algo_Parser" src-port="RefPoc"/>
<Connection dst="HevcDecoder_InterPrediction" dst-port="CUInfo"
  src="HevcDecoder_Algo_Parser" src-port="CUInfo"/>
<Connection dst="HevcDecoder_InterPrediction" dst-port="PartMode"
  src="HevcDecoder_Algo_Parser" src-port="PartMode"/>
<Connection dst="HevcDecoder_InterPrediction" dst-port="Poc"
  src="HevcDecoder_Algo_Parser" src-port="Poc"/>
<Connection dst="HevcDecoder_InterPrediction" dst-port="Sample"
  src="HevcDecoder_DecodedPictureBuffer" src-port="RefSample"/>
<Connection dst="HevcDecoder_InterPrediction" dst-port="SliceType"
  src="HevcDecoder_Algo_Parser" src-port="SliceType"/>
<Connection dst="HevcDecoder_SelectCU" dst-port="InterSample"
  src="HevcDecoder_InterPrediction" src-port="PredSample"/>
<Connection dst="HevcDecoder_IntraPrediction" dst-port="Sample"

```

```

src="HevcDecoder_SelectCU" src-port="Sample"/>
<Connection dst="HevcDecoder_SAO" dst-port="SaoSe"
src="HevcDecoder_Algo_Parser" src-port="SaoSe">
  <Attribute kind="Value" name="bufferSize">
    <Expr kind="Literal" literal-kind="Integer" value="16384"/>
  </Attribute>
</Connection>
<Connection dst="HevcDecoder_SAO" dst-port="IsPicSlcLcu"
src="HevcDecoder_Algo_Parser" src-port="IsPicSlcLcu"/>
<Connection dst="HevcDecoder_SAO" dst-port="PicSize"
src="HevcDecoder_Algo_Parser" src-port="PictSize"/>
<Connection dst="HevcDecoder_DecodedPictureBuffer"
dst-port="Sample" src="HevcDecoder_SAO" src-port="FiltSample"/>
<Connection dst="HevcDecoder_QpGen" dst-port="TuSize"
src="HevcDecoder_Algo_Parser" src-port="TUSize" />
<Connection dst="HevcDecoder_QpGen" dst-port="QP_IN"
src="HevcDecoder_Algo_Parser" src-port="Qp"/>
<Connection dst="HevcDecoder_InterPrediction"
dst-port="WeightedPredSe" src="HevcDecoder_Algo_Parser" src-port="weightedPred">
  <Attribute kind="Value" name="bufferSize">
    <Expr kind="Literal" literal-kind="Integer" value="16384"/>
  </Attribute>
</Connection>
<Connection dst="HevcDecoder_SAO" dst-port="LFAcrossSlcTile"
src="HevcDecoder_Algo_Parser" src-port="LFAcrossSlcTile"/>
<Connection dst="HevcDecoder_IntraPrediction" dst-port="LcuSizeMax"
src="HevcDecoder_Algo_Parser" src-port="LcuSizeMax"/>
<Connection dst="HevcDecoder_SAO" dst-port="LcuSizeMax"
src="HevcDecoder_Algo_Parser" src-port="LcuSizeMax"/>
<Connection dst="HevcDecoder_IntraPrediction" dst-port="TilesCoord"
src="HevcDecoder_Algo_Parser" src-port="TilesCoord"/>
<Connection dst="HevcDecoder_SAO" dst-port="TilesCoord"
src="HevcDecoder_Algo_Parser" src-port="TilesCoord"/>
<Connection dst="HevcDecoder_xIT_invDST4x4_1st" dst-port="IN"
src="HevcDecoder_xIT_IT_Splitter" src-port="Coeff_4x4_DST"/>
<Connection dst="HevcDecoder_xIT_invDST4x4_2nd" dst-port="IN"
src="HevcDecoder_xIT_invDST4x4_1st" src-port="OUT"/>
<Connection dst="HevcDecoder_xIT_IT_Merger" dst-port="Block_4x4_DST"
src="HevcDecoder_xIT_invDST4x4_2nd" src-port="OUT"/>
<Connection dst="HevcDecoder_xIT_IT_Merger" dst-port="Block_skip"
src="HevcDecoder_xIT_IT_Splitter" src-port="Coeff_skip"/>
<Connection dst="HevcDecoder_xIT_Block_Merger" dst-port="Block_in"
src="HevcDecoder_xIT_IT_Merger" src-port="Block"/>
<Connection dst="HevcDecoder_xIT_IT4x4_Transpose4x4_1"
dst-port="Src" src="HevcDecoder_xIT_IT4x4_IT4x4_1d_0" src-port="Dst"/>
<Connection dst="HevcDecoder_xIT_IT4x4_Transpose4x4_1" dst-port="Src"
src="HevcDecoder_xIT_IT4x4_Transpose4x4_1" src-port="Dst"/>
<Connection dst="HevcDecoder_xIT_IT4x4_Transpose4x4_0" dst-port="Src"
src="HevcDecoder_xIT_IT4x4_Transpose4x4_0" src-port="Dst"/>
<Connection dst="HevcDecoder_xIT_IT_Merger" dst-port="Block_4x4_IT"
src="HevcDecoder_xIT_IT4x4_IT4x4_1d_1" src-port="Dst"/>
<Connection dst="HevcDecoder_xIT_IT4x4_Transpose4x4_0"
dst-port="Src" src="HevcDecoder_xIT_IT_Splitter" src-port="Coeff_4x4_IT"/>
<Connection dst="HevcDecoder_xIT_IT8x8_Transpose8x8_1"
dst-port="Src" src="HevcDecoder_xIT_IT8x8_IT8x8_1d_0" src-port="Dst"/>
<Connection dst="HevcDecoder_xIT_IT8x8_Transpose8x8_1" dst-port="Src"
src="HevcDecoder_xIT_IT8x8_Transpose8x8_1" src-port="Dst"/>
<Connection dst="HevcDecoder_xIT_IT8x8_Transpose8x8_0" dst-port="Src"
src="HevcDecoder_xIT_IT8x8_Transpose8x8_0" src-port="Dst"/>
<Connection dst="HevcDecoder_xIT_IT_Merger" dst-port="Block_8x8"

```

```

src="HevcDecoder_xIT_IT8x8_IT8x8_1d_1" src-port="Dst"/>
<Connection dst="HevcDecoder_xIT_IT8x8_Transpose8x8_0"
dst-port="Src" src="HevcDecoder_xIT_IT_Splitter" src-port="Coeff_8x8"/>
<Connection dst="HevcDecoder_xIT_IT16x16_Transpose16x16_1"
dst-port="Src" src="HevcDecoder_xIT_IT16x16_IT16x16_1d_0" src-port="Dst"/>
<Connection dst="HevcDecoder_xIT_IT16x16_IT16x16_1d_1"
dst-port="Src" src="HevcDecoder_xIT_IT16x16_Transpose16x16_1" src-port="Dst"/>
<Connection dst="HevcDecoder_xIT_IT16x16_IT16x16_1d_0"
dst-port="Src" src="HevcDecoder_xIT_IT16x16_Transpose16x16_0" src-port="Dst"/>
<Connection dst="HevcDecoder_xIT_IT_Merger" dst-port="Block_16x16"
src="HevcDecoder_xIT_IT16x16_IT16x16_1d_1" src-port="Dst"/>
<Connection dst="HevcDecoder_xIT_IT16x16_Transpose16x16_0"
dst-port="Src" src="HevcDecoder_xIT_IT_Splitter" src-port="Coeff_16x16"/>
<Connection dst="HevcDecoder_xIT_IT32x32_IT32x32_1d_1"
dst-port="Src" src="HevcDecoder_xIT_IT32x32_Transpose32x32_1" src-port="Dst"/>
<Connection dst="HevcDecoder_xIT_IT32x32_Transpose32x32_1"
dst-port="Src" src="HevcDecoder_xIT_IT32x32_IT32x32_1d_0" src-port="Dst"/>
<Connection dst="HevcDecoder_xIT_IT32x32_IT32x32_1d_0"
dst-port="Src" src="HevcDecoder_xIT_IT32x32_Transpose32x32_0" src-port="Dst"/>
<Connection dst="HevcDecoder_xIT_IT_Merger" dst-port="Block_32x32"
src="HevcDecoder_xIT_IT32x32_IT32x32_1d_1" src-port="Dst"/>
<Connection dst="HevcDecoder_xIT_IT32x32_Transpose32x32_0"
dst-port="Src" src="HevcDecoder_xIT_IT_Splitter" src-port="Coeff_32x32"/>
<Connection dst="HevcDecoder_SelectCU" dst-port="ResidualSample"
src="HevcDecoder_xIT_Block_Merger" src-port="Block"/>
<Connection dst="HevcDecoder_xIT_IT_Splitter" dst-port="Size"
src="HevcDecoder_Algo_Parser" src-port="TUSize"/>
<Connection dst="HevcDecoder_xIT_IT_Merger" dst-port="Size"
src="HevcDecoder_Algo_Parser" src-port="TUSize"/>
<Connection dst="HevcDecoder_xIT_Block_Merger" dst-port="Size"
src="HevcDecoder_Algo_Parser" src-port="TUSize"/>
<Connection dst="HevcDecoder_xIT_IT_Splitter" dst-port="Coeff"
src="HevcDecoder_Algo_Parser" src-port="Coeff"/>
<Connection dst="HevcDecoder_DBFilter_DeblockFilt" dst-port="Bs"
src="HevcDecoder_DBFilter_GenerateBs" src-port="Bs"/>
<Connection dst="HevcDecoder_SAO" dst-port="SampleIn"
src="HevcDecoder_DBFilter_DeblockFilt" src-port="FiltSample"/>
<Connection dst="HevcDecoder_DBFilter_DeblockFilt" dst-port="Qp"
src="HevcDecoder_QpGen" src-port="Qp"/>
<Connection dst="HevcDecoder_DBFilter_GenerateBs" dst-port="CUInfo"
src="HevcDecoder_Algo_Parser" src-port="CUInfo"/>
<Connection dst="HevcDecoder_DBFilter_GenerateBs" dst-port="Cbf"
src="HevcDecoder_Algo_Parser" src-port="Cbf"/>
<Connection dst="HevcDecoder_DBFilter_GenerateBs"
dst-port="SplitTrans" src="HevcDecoder_Algo_Parser" src-port="SplitTransform"/>
<Connection dst="HevcDecoder_DBFilter_GenerateBs"
dst-port="PartMode" src="HevcDecoder_Algo_Parser" src-port="PartMode"/>
<Connection dst="HevcDecoder_DBFilter_GenerateBs"
dst-port="SliceAddr" src="HevcDecoder_Algo_Parser" src-port="SliceAddr"/>
<Connection dst="HevcDecoder_DBFilter_DeblockFilt"
dst-port="PicSize" src="HevcDecoder_Algo_Parser" src-port="PictSize"/>
<Connection dst="HevcDecoder_DBFilter_DeblockFilt"
dst-port="IsPicSlcLcu" src="HevcDecoder_Algo_Parser" src-port="IsPicSlcLcu"/>
<Connection dst="HevcDecoder_DBFilter_DeblockFilt"
dst-port="DBFDisable" src="HevcDecoder_Algo_Parser" src-port="DBFDisable"/>
<Connection dst="HevcDecoder_DBFilter_DeblockFilt"
dst-port="SampleIn" src="HevcDecoder_SelectCU" src-port="Sample"/>
<Connection dst="HevcDecoder_DBFilter_GenerateBs" dst-port="LFAcrossSlcTile"
src="HevcDecoder_Algo_Parser" src-port="LFAcrossSlcTile"/>
<Connection dst="HevcDecoder_DBFilter_GenerateBs"

```

```

    dst-port="LcuSizeMax" src="HevcDecoder_Algo_Parser" src-port="LcuSizeMax"/>
<Connection dst="HevcDecoder_DBFilter_DeblockFilt"
    dst-port="LcuSizeMax" src="HevcDecoder_Algo_Parser" src-port="LcuSizeMax"/>
<Connection dst="HevcDecoder_DBFilter_GenerateBs"
    dst-port="TilesCoord" src="HevcDecoder_Algo_Parser" src-port="TilesCoord"/>
<Connection dst="HevcDecoder_DBFilter_DeblockFilt"
    dst-port="TilesCoord" src="HevcDecoder_Algo_Parser" src-port="TilesCoord"/>
<Connection dst="HevcDecoder_DBFilter_DeblockFilt"
    dst-port="SyntaxElem" src="HevcDecoder_Algo_Parser" src-port="DbfSe"/>
<Connection dst="HevcDecoder_generateInfo_MvComponentPred"
    dst-port="RefList"
    src="HevcDecoder_generateInfo_GenerateRefList" src-port="RefList"/>
<Connection dst="HevcDecoder_InterPrediction" dst-port="RefIdx"
    src="HevcDecoder_generateInfo_MvComponentPred" src-port="RefIdx"/>
<Connection dst="HevcDecoder_InterPrediction" dst-port="RefList"
    src="HevcDecoder_generateInfo_GenerateRefList" src-port="RefList"/>
<Connection dst="HevcDecoder_InterPrediction"
    dst-port="IsBiPredOrLx"
    src="HevcDecoder_generateInfo_MvComponentPred" src-port="IsBiPredOrLx"/>
<Connection dst="HevcDecoder_InterPrediction" dst-port="Mv"
    src="HevcDecoder_generateInfo_MvComponentPred" src-port="Mv"/>
<Connection dst="HevcDecoder_DecodedPictureBuffer"
    dst-port="IsBiPredOrLx"
    src="HevcDecoder_generateInfo_MvComponentPred" src-port="IsBiPredOrLx"/>
<Connection dst="HevcDecoder_DecodedPictureBuffer"
    dst-port="IsReadOrNewSlc"
    src="HevcDecoder_generateInfo_MvComponentPred" src-port="IsReadOrNewSlice"/>
<Connection dst="HevcDecoder_DecodedPictureBuffer" dst-port="Mv"
    src="HevcDecoder_generateInfo_MvComponentPred" src-port="Mv"/>
<Connection dst="HevcDecoder_DecodedPictureBuffer"
    dst-port="PredCuSize"
    src="HevcDecoder_generateInfo_MvComponentPred" src-port="PredCuSize"/>
<Connection dst="HevcDecoder_DecodedPictureBuffer" dst-port="PocRef"
    src="HevcDecoder_generateInfo_MvComponentPred" src-port="PocRef"/>
<Connection dst="HevcDecoder_DBFilter_GenerateBs" dst-port="Mv"
    src="HevcDecoder_generateInfo_MvComponentPred" src-port="Mv"/>
<Connection dst="HevcDecoder_DBFilter_GenerateBs"
    dst-port="IsBiPredOrLx"
    src="HevcDecoder_generateInfo_MvComponentPred" src-port="IsBiPredOrLx"/>
<Connection dst="HevcDecoder_DBFilter_GenerateBs" dst-port="RefPoc"
    src="HevcDecoder_generateInfo_MvComponentPred" src-port="PocRef"/>
<Connection dst="HevcDecoder_generateInfo_MvComponentPred"
    dst-port="PartMode" src="HevcDecoder_Algo_Parser" src-port="PartMode"/>
<Connection dst="HevcDecoder_generateInfo_MvComponentPred"
    dst-port="PicSize" src="HevcDecoder_Algo_Parser" src-port="PictSize"/>
<Connection dst="HevcDecoder_generateInfo_GenerateRefList"
    dst-port="RefPoc" src="HevcDecoder_Algo_Parser" src-port="RefPoc"/>
<Connection dst="HevcDecoder_generateInfo_MvComponentPred"
    dst-port="RpsPoc" src="HevcDecoder_Algo_Parser" src-port="RefPoc"/>
<Connection dst="HevcDecoder_generateInfo_MvComponentPred"
    dst-port="SliceType" src="HevcDecoder_Algo_Parser" src-port="SliceType"/>
<Connection dst="HevcDecoder_generateInfo_MvComponentPred"
    dst-port="SliceAddr" src="HevcDecoder_Algo_Parser" src-port="SliceAddr"/>
<Connection dst="HevcDecoder_generateInfo_MvComponentPred"
    dst-port="Poc" src="HevcDecoder_Algo_Parser" src-port="Poc"/>
<Connection dst="HevcDecoder_generateInfo_MvComponentPred"
    dst-port="CUInfo" src="HevcDecoder_Algo_Parser" src-port="CUInfo"/>
<Connection dst="HevcDecoder_generateInfo_MvComponentPred"
    dst-port="SyntaxElem" src="HevcDecoder_Algo_Parser" src-port="MvPredSyntaxElem"/>
<Connection dst="HevcDecoder_generateInfo_GenerateRefList" dst-port="NumRefIdxLXAct"

```

STANDARDS1.COM . Click to view the full PDF of ISO/IEC 23002-4:2018

```

src="HevcDecoder_Algo_Parser" src-port="NumRefIdxLxActive"/>
<Connection dst="HevcDecoder_generateInfo_GenerateRefList" dst-port="RefPicListModif"
src="HevcDecoder_Algo_Parser" src-port="RefPicListModif"/>
<Connection dst="HevcDecoder_generateInfo_MvComponentPred"
dst-port="LcuSizeMax" src="HevcDecoder_Algo_Parser" src-port="LcuSizeMax"/>
<Connection dst="HevcDecoder_generateInfo_MvComponentPred"
dst-port="TilesCoord" src="HevcDecoder_Algo_Parser" src-port="TilesCoord"/>
<Connection dst="display" dst-port="DispCoord"
src="HevcDecoder_Algo_Parser" src-port="DispCoord"/>
<Connection dst="display" dst-port="PicSizeInMb"
src="HevcDecoder_Algo_Parser" src-port="PicSizeInMb"/>
<Connection dst="display" dst-port="Byte" src="HevcDecoder_SAO"
src-port="FiltSample"/>
<Connection dst="HevcDecoder_Algo_Parser" dst-port="byte"
src="Source" src-port="0"/>
<Connection dst="MD5_shifter" dst-port="IN" src="MD5_padding" src-port="OUT"/>
<Connection dst="MD5_padding" dst-port="DATA" src="MD5_MD5SplitInfo"
src-port="DataOut"/>
<Connection dst="MD5_padding" dst-port="LENGTH"
src="MD5_MD5SplitInfo" src-port="Length"/>
<Connection dst="MD5_compute" dst-port="IN" src="MD5_shifter" src-port="OUT"/>
<Connection dst="MD5_compute" dst-port="NEWLENGTH" src="MD5_padding"
src-port="NEWLENGTH"/>
<Connection dst="MD5_compute" dst-port="REF" src="MD5_MD5SplitInfo"
src-port="REF_MD5"/>
<Connection dst="MD5_MD5SplitInfo" dst-port="Byte"
src="HevcDecoder_SAO" src-port="FiltSample"/>
<Connection dst="MD5_MD5SplitInfo" dst-port="SEI_MD5"
src="HevcDecoder_Algo_Parser" src-port="SEI_MD5"/>
<Connection dst="MD5_MD5SplitInfo" dst-port="PictSize"
src="HevcDecoder_Algo_Parser" src-port="PictSize"/>
</XDF>

```

**Annex C**  
(informative)

**FNL of MPEG-4 AVC Progressive High Profile decoder**

STANDARDSISO.COM : Click to view the full PDF of ISO/IEC 23002-4:2018

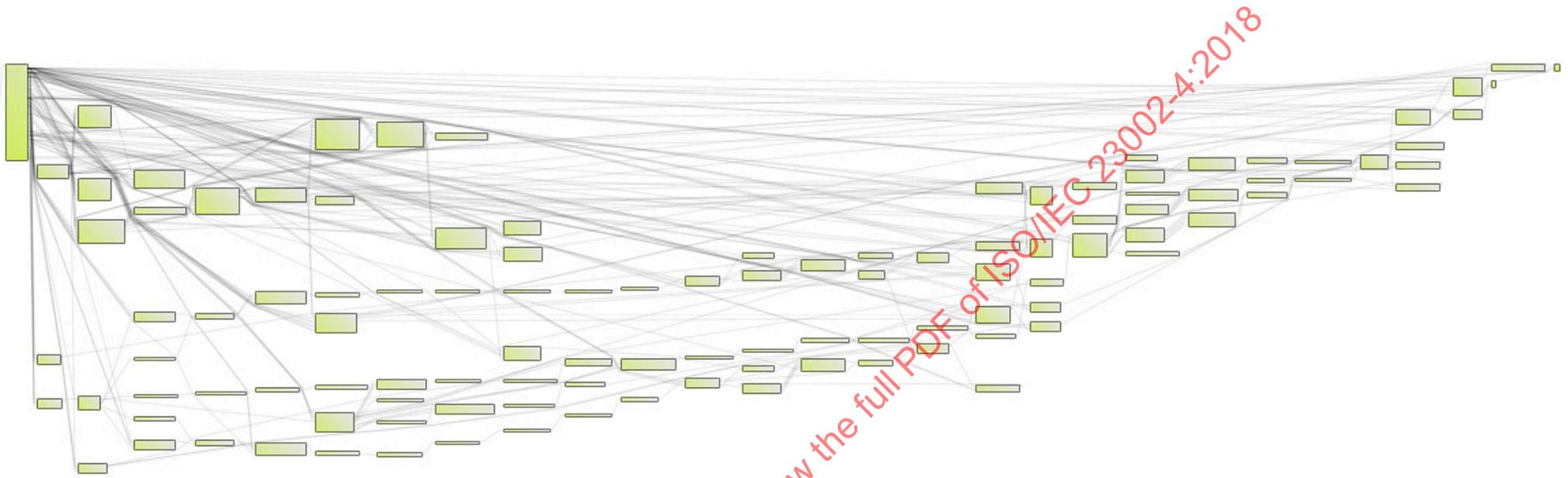


Figure C.2 — Complete example of the FU network description of the MPEG-4 AVC Progressive High Profile decoder (refer to Top\_mpeg4\_part10\_PHP\_decoder.xdf file)

Complete example of a graphical representation of the MPEG-4 AVC Progressive High Profile decoder (refer to Top\_mpeg4\_part10\_PHP\_decoder.xml).

```

<?xml version="1.0" encoding="UTF-8"?>
<XDF name="Top_mpeg4_part10_PHP_decoder">
  <Instance id="source">
    <Class name="org.sc29.wg11.stdio.Source"/>
  </Instance>
  <Instance id="Merger">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.display.Mgnt_Merger420_AVC"/>
  </Instance>
  <Instance id="display">
    <Class name="org.sc29.wg11.video.DisplayYUV"/>
  </Instance>
  <Instance id="AVCDecoder_Syn_Parser_Algo_Synp">
    <Class name="org.sc29.wg11.mpeg4.part10.php.synParser.Algo_SynP"/>
  </Instance>
  <Instance id="AVCDecoder_Syn_Parser_Algo_Parser_IPCM">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.synParser.Algo_Parser_IPCM"/>
  </Instance>
  <Instance id="AVCDecoder_Syn_Parser_IntraPredSplit">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.synParser.Algo_IntraPred_Split"/>
  </Instance>
  <Instance id="AVCDecoder_Syn_Parser_Generate_Inter_Info_MMCO">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.interPred.Algo_MMCO"/>
  </Instance>
  <Instance id="AVCDecoder_Syn_Parser_Generate_Inter_Info_RefList0">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.interPred.Algo_RefList"/>
  </Instance>
  <Instance id="AVCDecoder_Syn_Parser_Generate_Inter_Info_Mgnt_InterPred">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.interPred.Mgnt_InterPred"/>
  </Instance>
  <Instance id="AVCDecoder_Syn_Parser_Generate_Inter_Info_MgntDeblockingFilter">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.deblockingFilter.Mgnt_DBF_AdaptiveFilter"/>
  </Instance>
  <Instance id="AVCDecoder_Syn_Parser_Generate_Inter_Info_RefList1">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.interPred.Algo_RefList"/>
  </Instance>
  <Instance id="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL0_Reconstr">
    <Class name="org.sc29.wg11.mpeg4.part10.php.interPred.Algo_MvLXReconstr"/>
    <Parameter name="IsPredL0">
      <Expr kind="Literal" literal-kind="Boolean" value="true"/>
    </Parameter>
  </Instance>
  <Instance id="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_RefldxL0ToFrameNum">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.interPred.Algo_RefldxtoFrameNum"/>
  </Instance>
  <Instance id="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvComponentReord">
    <Class name="org.sc29.wg11.mpeg4.part10.php.deblockingFilter.Algo_MvComponentReorder"/>
  </Instance>
  <Instance id="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL1_Reconstr">
    <Class name="org.sc29.wg11.mpeg4.part10.php.interPred.Algo_MvLXReconstr"/>
    <Parameter name="IsPredL0">
      <Expr kind="Literal" literal-kind="Boolean" value="false"/>
    </Parameter>
  </Instance>
  <Instance id="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_RefldxL1ToFrameNum">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.interPred.Algo_RefldxtoFrameNum"/>
  </Instance>
  <Instance id="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvBuffer">

```

```

    <Class name="org.sc29.wg11.mpeg4.part10.php.interPred.Algo_MvBuffer"/>
  </Instance>
  <Instance id="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_SelectMvLX">
    <Class name="org.sc29.wg11.mpeg4.part10.php.interPred.Mgnt_SelectMvpLX"/>
  </Instance>
  <Instance id="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_SelectReflDx">
    <Class name="org.sc29.wg11.mpeg4.part10.php.interPred.Mgnt_SelectReflDx"/>
  </Instance>
  <Instance id="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_FrameNumToPocList">
    <Class name="org.sc29.wg11.mpeg4.part10.php.interPred.Algo_FrameNumToPocList"/>
  </Instance>
  <Instance id="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_GeneratePredWeight">
    <Class name="org.sc29.wg11.mpeg4.part10.php.interPred.Algo_GeneratePredWeight"/>
  </Instance>
  <Instance id="AVCDecoder_Decoding_Y_DecodedPictureBuffer">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.interPred.Mgnt_DPB"/>
    <Parameter name="MB_WIDTH">
      <Expr kind="Literal" literal-kind="Integer" value="16"/>
    </Parameter>
  </Instance>
  <Instance id="AVCDecoder_Decoding_Y_PredictionY_Select">
    <Class name="org.sc29.wg11.common.Algo_SelectMB_8"/>
    <Parameter name="WIDTH">
      <Expr kind="Literal" literal-kind="Integer" value="256"/>
    </Parameter>
  </Instance>
  <Instance id="AVCDecoder_Decoding_Y_PredictionY_Deblocking_Filter">
    <Class name="org.sc29.wg11.mpeg4.part10.php.deblockingFilter.Algo_DBF_AdaptiveFilter"/>
    <Parameter name="ChromaEdgeFlag">
      <Expr kind="Literal" literal-kind="Boolean" value="false"/>
    </Parameter>
    <Parameter name="MbWidth">
      <Expr kind="Literal" literal-kind="Integer" value="16"/>
    </Parameter>
  </Instance>
  <Instance id="AVCDecoder_Decoding_Y_PredictionY_DemuxParserInfos">
    <Class name="org.sc29.wg11.mpeg4.part10.php.selecMacroblock.Algo_DemuxParserInfoForBlocks_Luma"/>
  </Instance>
  <Instance id="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Mgnt_Intra_16x16">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.intraPred.Mgnt_Intra_16x16"/>
    <Parameter name="MB_WIDTH">
      <Expr kind="Literal" literal-kind="Integer" value="16"/>
    </Parameter>
  </Instance>
  <Instance id="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Algo_IntraPred_LUMA_16x16">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.intraPred.Algo_IntraPred_LUMA_16x16"/>
  </Instance>
  <Instance id="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Add_Clip">
    <Class name="org.sc29.wg11.common.Algo_AddPixSat"/>
  </Instance>
  <Instance id="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Buffer_Neighbour_FullMb">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.intraPred.Mgnt_Buffer_Neighbour_FullMb"/>
    <Parameter name="IsChroma">
      <Expr kind="Literal" literal-kind="Boolean" value="false"/>
    </Parameter>
    <Parameter name="MB_WIDTH">
      <Expr kind="Literal" literal-kind="Integer" value="16"/>
    </Parameter>
  </Instance>
  <Instance id="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Mgnt_Intra4x4">

```

```

<Class name="org.sc29.wg11.mpeg4.part10.cbp.intraPred.Mgnt_Intra_4x4"/>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Algo_IntraPred_LUMA_4x4">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.intraPred.Algo_IntraPred_LUMA_4x4"/>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Algo_Add_4x4">
  <Class name="org.sc29.wg11.common.Algo_AddPixSat"/>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Algo_Merge_4x4_to_16x16_norasterscan">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.intraPred.Algo_Merge_4x4_to_16x16_norasterscan"/>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Algo_Split_16x16_to_4x4_norasterscan">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.intraPred.Algo_Split_16x16_to_4x4_norasterscan"/>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Buffer_Neighbour_4x4">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.intraPred.Mgnt_Buffer_Neighbour_YxY"/>
  <Parameter name="IsIntra4x4">
    <Expr kind="Literal" literal-kind="Boolean" value="true"/>
  </Parameter>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_PredictionY_Inter_Algo_Interp_SeparableSixTapQuarterPelAVC">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.interPred.Algo_Interp_SeparableSixTapQuarterPel"/>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_PredictionY_Inter_Add_Clip">
  <Class name="org.sc29.wg11.common.Algo_AddPixSat"/>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_PredictionY_Inter_Algo_Interp_Reord">
  <Class name="org.sc29.wg11.mpeg4.part10.php.interPred.Algo_Interp_Reord"/>
  <Parameter name="SzSidePerPart">
    <Expr kind="Literal" literal-kind="Integer" value="4"/>
  </Parameter>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_PredictionY_Intra_8x8_Mgnt_Intra8x8">
  <Class name="org.sc29.wg11.mpeg4.part10.php.intraPred.Mgnt_Intra_8x8"/>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_PredictionY_Intra_8x8_Algo_IntraPred_LUMA_8x8">
  <Class name="org.sc29.wg11.mpeg4.part10.php.intraPred.Algo_IntraPred_LUMA_8x8"/>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_PredictionY_Intra_8x8_Algo_Add_8x8">
  <Class name="org.sc29.wg11.common.Algo_AddPixSat"/>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_PredictionY_Intra_8x8_Algo_Merge_8x8_to_16x16_norasterscan">
  <Class name="org.sc29.wg11.mpeg4.part10.php.intraPred.Algo_Merge_8x8_to_16x16_norasterscan"/>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_PredictionY_Intra_8x8_Buffer_Neighbour_8x8">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.intraPred.Mgnt_Buffer_Neighbour_YxY"/>
  <Parameter name="IsIntra4x4">
    <Expr kind="Literal" literal-kind="Boolean" value="false"/>
  </Parameter>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_PredictionY_Intra_8x8_Algo_Split_16x16_to_8x8_norasterscan">
  <Class name="org.sc29.wg11.mpeg4.part10.php.intraPred.Algo_Split_16x16_to_8x8_norasterscan"/>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_ResidualY_Algo_Merge_4x4_to_16x16">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_Merge_4x4_to_16x16"/>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_ResidualY_I4x4_I8x8_demux">
  <Class name="org.sc29.wg11.mpeg4.part10.php.Residual.Mgnt_I4x4_I8x8_demux"/>
</Instance>
<Instance id="AVCDecoder_Decoding_Y_ResidualY_I4x4_I8x8_mux">

```

```

    <Class name="org.sc29.wg11.mpeg4.part10.php.Residual.Mgnt_I4x4_I8x8_mux"/>
  </Instance>
  <Instance id="AVCDecoder_Decoding_Y_ResidualY_Algo_Merge_8x8_to_16x16">
    <Class name="org.sc29.wg11.mpeg4.part10.php.Residual.Algo_Merge_8x8_to_16x16"/>
  </Instance>
  <Instance id="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_IS_Zigzag_4x4_DC">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_IS_Zigzag_4x4"/>
  </Instance>
  <Instance id="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_DCR_Hadamard_LUMA_IHT1d_0">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_DCR_Hadamard_LUMA_IHT1d"/>
  </Instance>
  <Instance id="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_DC_Transpose4x4_0">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_Transpose4x4"/>
  </Instance>
  <Instance id="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_DCR_Hadamard_LUMA_IHT1d_1">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_DCR_Hadamard_LUMA_IHT1d"/>
  </Instance>
  <Instance id="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_DCR_Hadamard_LUMA_Scaling">
    <Class name="org.sc29.wg11.mpeg4.part10.php.Residual.Algo_DCR_Hadamard_LUMA_Scaling"/>
  </Instance>
  <Instance id="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_DC_Transpose4x4_1">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_Transpose4x4"/>
  </Instance>
  <Instance id="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_DCR_Hadamard_LUMA_Reordering">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_DCR_Hadamard_LUMA_Reordering"/>
  </Instance>
  <Instance id="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_Algo_IS_Zigzag_4x4_AC">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_IS_Zigzag_4x4"/>
  </Instance>
  <Instance id="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_Mgnt_IQ_INTRA16x16">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Mgnt_IQ_INTRA16x16"/>
  </Instance>
  <Instance id="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_Algo_IQ_QSAndSLAndIDCTScaler_4x4">
    <Class name="org.sc29.wg11.mpeg4.part10.php.Residual.Algo_IQ_QSAndSLAndIDCTScaler_4x4"/>
    <Parameter name="NB4x4">
      <Expr kind="Literal" literal-kind="Integer" value="16"/>
    </Parameter>
  </Instance>
  <Instance id="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_IT4x4_Algo_IT4x4_1d_0">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_IT4x4_1d"/>
  </Instance>
  <Instance id="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_IT4x4_Algo_Transpose4x4_0">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_Transpose4x4"/>
  </Instance>
  <Instance id="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_IT4x4_Algo_IT4x4_1d_1">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_IT4x4_1d"/>
  </Instance>
  <Instance id="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_IT4x4_Algo_Transpose4x4_1">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_Transpose4x4"/>
  </Instance>
  <Instance id="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_IT4x4_Algo_IT4x4_Addshift">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_IT4x4_Addshift"/>
  </Instance>
  <Instance id="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_8x8_Algo_IS_Zigzag_8x8_AC">
    <Class name="org.sc29.wg11.mpeg4.part10.php.Residual.Algo_IS_Zigzag_8x8"/>
  </Instance>
  <Instance id="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_8x8_Algo_ICT_IntraPred_8x8">
    <Class name="org.sc29.wg11.mpeg4.part10.php.Residual.Algo_IIT_8x8"/>
  </Instance>
  <Instance id="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_8x8_Algo_IQ_QSAndSLAndIDCTScaler_8x8">

```

```

<Class name="org.sc29.wg11.mpeg4.part10.php.Residual.Algo_IQ_QSAndSLAndIDCTScaler_8x8"/>
</Instance>
<Instance id="AVCDecoder_Decode_U_Buffer_inter">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.interPred.Mgnt_DPB"/>
  <Parameter name="MB_WIDTH">
    <Expr kind="Literal" literal-kind="Integer" value="8"/>
  </Parameter>
</Instance>
<Instance id="AVCDecoder_Decode_U_PredictionC_Select_chroma">
  <Class name="org.sc29.wg11.common.Algo_SelectMB_4"/>
  <Parameter name="WIDTH">
    <Expr kind="Literal" literal-kind="Integer" value="64"/>
  </Parameter>
</Instance>
<Instance id="AVCDecoder_Decode_U_PredictionC_Deblocking_Filter">
  <Class name="org.sc29.wg11.mpeg4.part10.php.deblockingFilter.Algo_DBF_AdaptiveFilter"/>
  <Parameter name="ChromaEdgeFlag">
    <Expr kind="Literal" literal-kind="Boolean" value="true"/>
  </Parameter>
  <Parameter name="MbWidth">
    <Expr kind="Literal" literal-kind="Integer" value="8"/>
  </Parameter>
</Instance>
<Instance id="AVCDecoder_Decode_U_PredictionC_DemuxParserInfos">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.selecMacroblock.Algo_DemuxParserInfoForBlocks_Chroma"/>
</Instance>
<Instance id="AVCDecoder_Decode_U_PredictionC_Intra8x8_Algo_IntraPred_CHROMA">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.intraPred.Algo_IntraPred_CHROMA"/>
</Instance>
<Instance id="AVCDecoder_Decode_U_PredictionC_Intra8x8_Algo_Add">
  <Class name="org.sc29.wg11.common.Algo_AddPixSat"/>
</Instance>
<Instance id="AVCDecoder_Decode_U_PredictionC_Intra8x8_Buffer_Neighbour_FullMb">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.intraPred.Mgnt_Buffer_Neighbour_FullMb"/>
  <Parameter name="IsChroma">
    <Expr kind="Literal" literal-kind="Boolean" value="true"/>
  </Parameter>
  <Parameter name="MB_WIDTH">
    <Expr kind="Literal" literal-kind="Integer" value="8"/>
  </Parameter>
</Instance>
<Instance id="AVCDecoder_Decode_U_PredictionC_Intra8x8_Mgnt_Intra_8x8">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.intraPred.Mgnt_Intra_16x16"/>
  <Parameter name="MB_WIDTH">
    <Expr kind="Literal" literal-kind="Integer" value="8"/>
  </Parameter>
</Instance>
<Instance id="AVCDecoder_Decode_U_PredictionC_Inter_Algo_Interp_EighthPelBilinear">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.interPred.Algo_Interp_Bilinear"/>
</Instance>
<Instance id="AVCDecoder_Decode_U_PredictionC_Inter_Algo_Add">
  <Class name="org.sc29.wg11.common.Algo_AddPixSat"/>
</Instance>
<Instance id="AVCDecoder_Decode_U_PredictionC_Inter_Algo_Interp_Reord">
  <Class name="org.sc29.wg11.mpeg4.part10.php.interPred.Algo_Interp_Reord"/>
  <Parameter name="SzSidePerPart">
    <Expr kind="Literal" literal-kind="Integer" value="2"/>
  </Parameter>
</Instance>
<Instance id="AVCDecoder_Decode_U_ResidualC_Algo_DCR_Hadamard_CHROMA">

```

```

    <Class name="org.sc29.wg11.mpeg4.part10.php.Residual.Algo_DCR_Hadamard_CHROMA"/>
  </Instance>
  <Instance id="AVCDecoder_Decode_U_ResidualC_Algo_Merge_4x4_to_8x8">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_Merge_4x4_to_8x8"/>
  </Instance>
  <Instance id="AVCDecoder_Decode_U_ResidualC_IS_IQ_IT_C_Algo_IS_Zigzag_4x4">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_IS_Zigzag_4x4"/>
  </Instance>
  <Instance id="AVCDecoder_Decode_U_ResidualC_IS_IQ_IT_C_Mgnt_IQ_Chroma">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Mgnt_IQ_Chroma"/>
  </Instance>
  <Instance id="AVCDecoder_Decode_U_ResidualC_IS_IQ_IT_C_Algo_IQ_QSAndSLAndIDCTScaler_4x4">
    <Class name="org.sc29.wg11.mpeg4.part10.php.Residual.Algo_IQ_QSAndSLAndIDCTScaler_4x4"/>
    <Parameter name="NB4x4">
      <Expr kind="Literal" literal-kind="Integer" value="4"/>
    </Parameter>
  </Instance>
  <Instance id="AVCDecoder_Decode_U_ResidualC_IS_IQ_IT_C_IT4x4_Algo_IT4x4_1d_0">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_IT4x4_1d"/>
  </Instance>
  <Instance id="AVCDecoder_Decode_U_ResidualC_IS_IQ_IT_C_IT4x4_Algo_Transpose4x4_0">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_Transpose4x4"/>
  </Instance>
  <Instance id="AVCDecoder_Decode_U_ResidualC_IS_IQ_IT_C_IT4x4_Algo_IT4x4_1d_1">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_IT4x4_1d"/>
  </Instance>
  <Instance id="AVCDecoder_Decode_U_ResidualC_IS_IQ_IT_C_IT4x4_Algo_Transpose4x4_1">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_Transpose4x4"/>
  </Instance>
  <Instance id="AVCDecoder_Decode_U_ResidualC_IS_IQ_IT_C_IT4x4_Algo_IT4x4_Addshift">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_IT4x4_Addshift"/>
  </Instance>
  <Instance id="AVCDecoder_Decode_V_Buffer_inter">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.interPred.Mgnt_DPB"/>
    <Parameter name="MB_WIDTH">
      <Expr kind="Literal" literal-kind="Integer" value="8"/>
    </Parameter>
  </Instance>
  <Instance id="AVCDecoder_Decode_V_PredictionC_Select_chroma">
    <Class name="org.sc29.wg11.common.Algo_SelectMB_4"/>
    <Parameter name="WIDTH">
      <Expr kind="Literal" literal-kind="Integer" value="64"/>
    </Parameter>
  </Instance>
  <Instance id="AVCDecoder_Decode_V_PredictionC_Deblocking_Filter">
    <Class name="org.sc29.wg11.mpeg4.part10.php.deblockingFilter.Algo_DBF_AdaptiveFilter"/>
    <Parameter name="ChromaEdgeFlag">
      <Expr kind="Literal" literal-kind="Boolean" value="true"/>
    </Parameter>
    <Parameter name="MbWidth">
      <Expr kind="Literal" literal-kind="Integer" value="8"/>
    </Parameter>
  </Instance>
  <Instance id="AVCDecoder_Decode_V_PredictionC_DemuxParserInfos">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.selectMacroblock.Algo_DemuxParserInfoForBlocks_Chroma"/>
  </Instance>
  <Instance id="AVCDecoder_Decode_V_PredictionC_Intra8x8_Algo_IntraPred_CHROMA">
    <Class name="org.sc29.wg11.mpeg4.part10.cbp.intraPred.Algo_IntraPred_CHROMA"/>
  </Instance>
  <Instance id="AVCDecoder_Decode_V_PredictionC_Intra8x8_Algo_Add">

```

```

<Class name="org.sc29.wg11.common.Algo_AddPixSat"/>
</Instance>
<Instance id="AVCDecoder_Decode_V_PredictionC_Intra8x8_Buffer_Neighbour_FullMb">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.intraPred.Mgnt_Buffer_Neighbour_FullMb"/>
  <Parameter name="IsChroma">
    <Expr kind="Literal" literal-kind="Boolean" value="true"/>
  </Parameter>
  <Parameter name="MB_WIDTH">
    <Expr kind="Literal" literal-kind="Integer" value="8"/>
  </Parameter>
</Instance>
<Instance id="AVCDecoder_Decode_V_PredictionC_Intra8x8_Mgnt_Intra_8x8">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.intraPred.Mgnt_Intra_16x16"/>
  <Parameter name="MB_WIDTH">
    <Expr kind="Literal" literal-kind="Integer" value="8"/>
  </Parameter>
</Instance>
<Instance id="AVCDecoder_Decode_V_PredictionC_Inter_Algo_Interp_EighthPelBilinear">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.interPred.Algo_Interp_Bilinear"/>
</Instance>
<Instance id="AVCDecoder_Decode_V_PredictionC_Inter_Algo_Add">
  <Class name="org.sc29.wg11.common.Algo_AddPixSat"/>
</Instance>
<Instance id="AVCDecoder_Decode_V_PredictionC_Inter_Algo_Interp_Reord">
  <Class name="org.sc29.wg11.mpeg4.part10.php.interPred.Algo_Interp_Reord"/>
  <Parameter name="SzSidePerPart">
    <Expr kind="Literal" literal-kind="Integer" value="2"/>
  </Parameter>
</Instance>
<Instance id="AVCDecoder_Decode_V_ResidualC_Algo_DCR_Hadamard_CHROMA">
  <Class name="org.sc29.wg11.mpeg4.part10.php.Residual.Algo_DCR_Hadamard_CHROMA"/>
</Instance>
<Instance id="AVCDecoder_Decode_V_ResidualC_Algo_Merge_4x4_to_8x8">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_Merge_4x4_to_8x8"/>
</Instance>
<Instance id="AVCDecoder_Decode_V_ResidualC_IS_IQ_IT_C_Algo_IS_Zigzag_4x4">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_IS_Zigzag_4x4"/>
</Instance>
<Instance id="AVCDecoder_Decode_V_ResidualC_IS_IQ_IT_C_Mgnt_IQ_Chroma">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Mgnt_IQ_Chroma"/>
</Instance>
<Instance id="AVCDecoder_Decode_V_ResidualC_IS_IQ_IT_C_Algo_IQ_QSAndSLAndIDCTScaler_4x4">
  <Class name="org.sc29.wg11.mpeg4.part10.php.Residual.Algo_IQ_QSAndSLAndIDCTScaler_4x4"/>
  <Parameter name="NB4x4">
    <Expr kind="Literal" literal-kind="Integer" value="4"/>
  </Parameter>
</Instance>
<Instance id="AVCDecoder_Decode_V_ResidualC_IS_IQ_IT_C_IT4x4_Algo_IT4x4_1d_0">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_IT4x4_1d"/>
</Instance>
<Instance id="AVCDecoder_Decode_V_ResidualC_IS_IQ_IT_C_IT4x4_Algo_Transpose4x4_0">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_Transpose4x4"/>
</Instance>
<Instance id="AVCDecoder_Decode_V_ResidualC_IS_IQ_IT_C_IT4x4_Algo_IT4x4_1d_1">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_IT4x4_1d"/>
</Instance>
<Instance id="AVCDecoder_Decode_V_ResidualC_IS_IQ_IT_C_IT4x4_Algo_Transpose4x4_1">
  <Class name="org.sc29.wg11.mpeg4.part10.cbp.Residual.Algo_Transpose4x4"/>
</Instance>
<Instance id="AVCDecoder_Decode_V_ResidualC_IS_IQ_IT_C_IT4x4_Algo_IT4x4_Addshift">

```



```

dst-port="LT_LIST"
src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MMCO" src-port="LT_LIST"/>
<Connection dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_RefList1"
dst-port="ST_LIST_SZ"
src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MMCO" src-port="ST_LIST_SZ"/>
<Connection
dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_RefIdxL0ToFrameNum"
dst-port="REF_IDX"
src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL0_Reconstr" src-port="REF_IDX_OUT"/>
<Connection
dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_RefIdxL0ToFrameNum"
dst-port="IS_NEW_MB"
src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL0_Reconstr"
src-port="IS_NEW_MB_SLICE"/>
<Connection
dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_RefIdxL1ToFrameNum"
dst-port="REF_IDX"
src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL1_Reconstr" src-port="REF_IDX_OUT"/>
<Connection
dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_RefIdxL1ToFrameNum"
dst-port="IS_NEW_MB"
src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL1_Reconstr"
src-port="IS_NEW_MB_SLICE"/>
<Connection
dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL0_Reconstr"
dst-port="COL_ZERO_FLAG"
src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvBuffer" src-port="COL_ZERO_FLAG"/>
<Connection
dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL1_Reconstr"
dst-port="COL_ZERO_FLAG"
src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvBuffer" src-port="COL_ZERO_FLAG"/>
<Connection
dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL0_Reconstr"
dst-port="TEMPORAL_PRED_INFO"
src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvBuffer"
src-port="TEMPORAL_PRED_INFO"/>
<Connection
dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL0_Reconstr"
dst-port="DIRECT_PRED_LX_I"
src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL1_Reconstr"
src-port="DIRECT_PRED_LX_O"/>
<Connection
dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL1_Reconstr"
dst-port="DIRECT_PRED_LX_I"
src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL0_Reconstr"
src-port="DIRECT_PRED_LX_O"/>
<Connection
dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvBuffer"
dst-port="MV"
src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvComponentReord"
src-port="MV_OUT_DBF"/>
<Connection
dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_SelectMvLX"
dst-port="FRAME_L0"
src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_RefIdxL0ToFrameNum"
src-port="READ_FRAME"/>
<Connection
dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_SelectMvLX"
dst-port="LOCATION_L0"
src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL0_Reconstr" src-port="LOCATION"/>

```

```

<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_SelectMvLX"
  dst-port="MV_L0"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL0_Reconstr" src-port="MV_OUT"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_SelectMvLX"
  dst-port="LOCATION_L1"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL1_Reconstr" src-port="LOCATION"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_SelectMvLX"
  dst-port="MV_L1"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL1_Reconstr" src-port="MV_OUT"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_SelectMvLX"
  dst-port="FRAME_L1"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_RefIdxL1ToFrameNum"
  src-port="READ_FRAME"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_SelectMvLX"
  dst-port="REF_IDX_L0"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL0_Reconstr" src-port="REF_IDX_OUT"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_SelectMvLX"
  dst-port="REF_IDX_L1"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL1_Reconstr" src-port="REF_IDX_OUT"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_SelectMvLX"
  dst-port="DIRECT_PRED_L0"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL0_Reconstr"
  src-port="DIRECT_PRED_LX_0"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_SelectMvLX"
  dst-port="DIRECT_PRED_L1"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL1_Reconstr"
  src-port="DIRECT_PRED_LX_0"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvComponentReord"
  dst-port="READ_FRAME"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_SelectMvLX" src-port="READ_FRAME"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvComponentReord"
  dst-port="MV"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_SelectMvLX" src-port="MV"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvComponentReord"
  dst-port="REF_IDX"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_SelectMvLX" src-port="REF_IDX"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvComponentReord"
  dst-port="SELECT_LIST"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_SelectMvLX" src-port="SELECT_LIST"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvBuffer"
  dst-port="REF_IDX"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvComponentReord"
  src-port="REF_IDX_TAB"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_SelectRefIdx"
  dst-port="READ_FRAME_L0"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_RefIdxL0ToFrameNum"

```

```

    src-port="READ_FRAME"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_SelectRefIdx"
  dst-port="IS_NEW_MB_L0"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL0_Reconstr"
  src-port="IS_NEW_MB_PIC"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_SelectRefIdx"
  dst-port="READ_FRAME_L1"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_RefIdxL1ToFrameNum"
  src-port="READ_FRAME"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_SelectRefIdx"
  dst-port="IS_NEW_MB_L1"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL1_Reconstr"
  src-port="IS_NEW_MB_PIC"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_SelectRefIdx"
  dst-port="SELECT_LIST"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_SelectMvLX" src-port="SELECT_LIST"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL1_Reconstr"
  dst-port="TEMPORAL_PRED_INFO"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvBuffer"
  src-port="TEMPORAL_PRED_INFO"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_GeneratePredWeight"
  dst-port="POC_LISTX"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_FrameNumToPocList"
  src-port="POC_LISTX"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_GeneratePredWeight"
  dst-port="POC_IS_LT"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_FrameNumToPocList"
  src-port="POC_IS_LT"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_GeneratePredWeight"
  dst-port="IS_NEW_MB_L1"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL1_Reconstr"
  src-port="IS_NEW_MB_SLICE"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_GeneratePredWeight"
  dst-port="REF_IDX_L1"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL1_Reconstr" src-port="REF_IDX_OUT"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_GeneratePredWeight"
  dst-port="IS_NEW_MB_L0"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL0_Reconstr"
  src-port="IS_NEW_MB_SLICE"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_GeneratePredWeight"
  dst-port="REF_IDX_L0"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL0_Reconstr" src-port="REF_IDX_OUT"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_GeneratePredWeight"
  dst-port="SELECT_LIST"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_SelectMvLX" src-port="SELECT_LIST"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_Mgnt_InterPred"
  dst-port="MV"

```

```

src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_SelectMvLX" src-port="MV"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_Mgnt_InterPred"
  dst-port="PARTSZ"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvComponentReord" src-port="PARTSZ"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_Mgnt_InterPred"
  dst-port="LOCATION"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_SelectMvLX" src-port="LOCATION"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MgntDeblockingFilter"
  dst-port="READ_FRAME"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvComponentReord"
  src-port="READ_FRAME_DBF"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MgntDeblockingFilter"
  dst-port="MV"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvComponentReord"
  src-port="MV_OUT_DBF"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_RefIdxL0ToFrameNum"
  dst-port="REF_LIST"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_RefList0" src-port="REF_LIST"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvBuffer"
  dst-port="REF_LIST0"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_RefList0" src-port="REF_LIST"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_FrameNumToPocList"
  dst-port="REF_LIST0"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_RefList0" src-port="REF_LIST"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_RefIdxL1ToFrameNum"
  dst-port="REF_LIST"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_RefList1" src-port="REF_LIST"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvBuffer"
  dst-port="REF_LIST1"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_RefList1" src-port="REF_LIST"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_FrameNumToPocList"
  dst-port="REF_LIST1"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_RefList1" src-port="REF_LIST"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvBuffer"
  dst-port="DEL_LIST"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MMCO" src-port="DEL_LIST"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_FrameNumToPocList"
  dst-port="DEL_LIST"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MMCO" src-port="DEL_LIST"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvBuffer"
  dst-port="LT"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MMCO" src-port="LT_LIST"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_FrameNumToPocList"
  dst-port="LT"
  src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MMCO" src-port="LT_LIST"/>
<Connection

```

```

dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvBuffer"
dst-port="EMPTY_DPB"
src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MMCO" src-port="EMPTY_DPB"/>
<Connection
dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_FrameNumToPocList"
dst-port="EMPTY_DPB"
src="AVCDecoder_Syn_Parser_Generate_Inter_Info_MMCO" src-port="EMPTY_DPB"/>
<Connection dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MMCO"
dst-port="MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_TYPE"/>
<Connection
dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MgntDeblockingFilter"
dst-port="MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_TYPE"/>
<Connection
dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL0_Reconstr"
dst-port="MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_TYPE"/>
<Connection
dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvComponentReord"
dst-port="MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_TYPE"/>
<Connection
dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL1_Reconstr"
dst-port="MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_TYPE"/>
<Connection
dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvBuffer"
dst-port="MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_TYPE"/>
<Connection
dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_SelectMvLX"
dst-port="MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_TYPE"/>
<Connection
dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_FrameNumToPocList"
dst-port="MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_TYPE"/>
<Connection
dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL0_Reconstr"
dst-port="SUB_MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="SUB_MB_TYPE"/>
<Connection
dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvComponentReord"
dst-port="SUB_MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="SUB_MB_TYPE"/>
<Connection
dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL1_Reconstr"
dst-port="SUB_MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="SUB_MB_TYPE"/>
<Connection
dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvBuffer"
dst-port="SUB_MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="SUB_MB_TYPE"/>
<Connection
dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_SelectMvLX"
dst-port="SUB_MB_TYPE" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="SUB_MB_TYPE"/>
<Connection
dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL0_Reconstr"
dst-port="MB_PRED_MVD" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_PRED_MVD_I0"/>
<Connection
dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL0_Reconstr"
dst-port="SUB_MB_PRED_MVD" src="AVCDecoder_Syn_Parser_Algo_Synp"
src-port="SUB_MB_PRED_MVD_I0"/>
<Connection
dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL0_Reconstr"
dst-port="SUB_MB_PRED_REF_IDX"
src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="SUB_MB_PRED_REF_IDX_I0"/>
<Connection
dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL0_Reconstr"
dst-port="MB_PRED_REF_IDX" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_PRED_REF_IDX_I0"/>
<Connection dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_RefList0"

```

```

dst-port="REF_TYPE_LIST" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="REF_TYPE_10"/>
<Connection dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MMCO"
  dst-port="MAX_FRAME_NUM" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MAX_FRAME_NUM"/>
<Connection dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_RefList0"
  dst-port="MAX_FRAME_NUM" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MAX_FRAME_NUM"/>
<Connection dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_RefList1"
  dst-port="MAX_FRAME_NUM" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MAX_FRAME_NUM"/>
<Connection dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MMCO"
  dst-port="MMCO" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MMCO"/>
<Connection dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_RefList0"
  dst-port="NB_REF_FRAME" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="NB_REF_FRAME_10"/>
<Connection dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_RefList0"
  dst-port="REF_REORDERING" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="REF_REORDERING_10"/>
<Connection dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MMCO"
  dst-port="FRAME_NUM" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="FRAME_NUM"/>
<Connection dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_RefList0"
  dst-port="FRAME_NUM" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="FRAME_NUM"/>
<Connection dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_RefList1"
  dst-port="FRAME_NUM" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="FRAME_NUM"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvBuffer"
  dst-port="FRAME_NUM" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="FRAME_NUM"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_FrameNumToPocList"
  dst-port="FRAME_NUM" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="FRAME_NUM"/>
<Connection dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MMCO"
  dst-port="POC" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="POC"/>
<Connection dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_RefList0"
  dst-port="POC" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="POC"/>
<Connection dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_RefList1"
  dst-port="POC" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="POC"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvBuffer"
  dst-port="POC" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="POC"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_FrameNumToPocList"
  dst-port="POC" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="POC"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MgntDeblockingFilter"
  dst-port="MB_LOCATION" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_LOCATION"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL0_Reconstr"
  dst-port="MB_LOCATION" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_LOCATION"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL1_Reconstr"
  dst-port="MB_LOCATION" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_LOCATION"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvBuffer"
  dst-port="MB_LOCATION" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_LOCATION"/>
<Connection dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MMCO"
  dst-port="NUM_REF_FRAME" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="NUM_REF_FRAMES"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MgntDeblockingFilter"
  dst-port="QP_Y" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="QP"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MgntDeblockingFilter"
  dst-port="QP_Cb" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="QP_Cb"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MgntDeblockingFilter"
  dst-port="QP_Cr" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="QP_Cr"/>

```

```

<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MgntDeblockingFilter"
  dst-port="SLICE_DBF_PARAM" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="SLICE_DBF_PARAM"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MgntDeblockingFilter"
  dst-port="CBP_BLK" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="CBP_BLK"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL0_Reconstr"
  dst-port="DIRECT_SPATIAL_MV_FLAG"
  src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="DIRECT_SPATIAL_MV_PRED_FLAG"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL1_Reconstr"
  dst-port="DIRECT_SPATIAL_MV_FLAG"
  src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="DIRECT_SPATIAL_MV_PRED_FLAG"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvBuffer"
  dst-port="DIRECT_SPATIAL_MV_FLAG"
  src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="DIRECT_SPATIAL_MV_PRED_FLAG"/>
<Connection dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_RefList1"
  dst-port="REF_REORDERING" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="REF_REORDERING_I1"/>
<Connection dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_RefList1"
  dst-port="REF_TYPE_LIST" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="REF_TYPE_I1"/>
<Connection dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_RefList1"
  dst-port="NB_REF_FRAME" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="NB_REF_FRAME_I1"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL1_Reconstr"
  dst-port="MB_PRED_REF_IDX" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_PRED_REF_IDX_I1"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL1_Reconstr"
  dst-port="MB_PRED_MVD" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="MB_PRED_MVD_I1"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL1_Reconstr"
  dst-port="SUB_MB_PRED_REF_IDX"
  src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="SUB_MB_PRED_REF_IDX_I1"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvL1_Reconstr"
  dst-port="SUB_MB_PRED_MVD" src="AVCDecoder_Syn_Parser_Algo_Synp"
  src-port="SUB_MB_PRED_MVD_I1"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_MvBuffer"
  dst-port="DIRECT8x8_INFERENCE_FLAG"
  src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="DIRECT_8X8_INTERFERENCE_FLAG"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_GeneratePredWeight"
  dst-port="WP_PRED_IDC" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="WEIGHTED_PRED_IDC"/>
<Connection
  dst="AVCDecoder_Syn_Parser_Generate_Inter_Info_MVReconstruct_GeneratePredWeight"
  dst-port="WP_PARAM" src="AVCDecoder_Syn_Parser_Algo_Synp" src-port="WP_PARAM"/>
<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Deblocking_Filter"
  dst-port="MB_IN" src="AVCDecoder_Decoding_Y_PredictionY_Select" src-port="OUT"/>
<Connection dst="AVCDecoder_Decoding_Y_PredictionY_Select"
  dst-port="SELECT"
  src="AVCDecoder_Decoding_Y_PredictionY_DemuxParserInfos" src-port="SELECT"/>
<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Add_Clip"
  dst-port="X"
  src="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Algo_IntraPred_LUMA_16x16" src-port="MPR"/>
<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Algo_IntraPred_LUMA_16x16"

```

```

dst-port="AVAIL"
src="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Mgnt_Intra_16x16" src-port="AVAIL"/>
<Connection
dst="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Algo_IntraPred_LUMA_16x16"
dst-port="Y_LEFT"
src="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Mgnt_Intra_16x16" src-port="Y_LEFT"/>
<Connection
dst="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Algo_IntraPred_LUMA_16x16"
dst-port="Y_UP"
src="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Mgnt_Intra_16x16" src-port="Y_UP"/>
<Connection
dst="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Algo_IntraPred_LUMA_16x16"
dst-port="Y_UP_LEFT"
src="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Mgnt_Intra_16x16" src-port="Y_UP_LEFT"/>
<Connection
dst="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Mgnt_Intra_16x16"
dst-port="EDGE"
src="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Buffer_Neighbour_FullMb" src-port="EDGE"/>
<Connection dst="AVCDecoder_Decoding_Y_PredictionY_Select"
dst-port="IN_2"
src="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Add_Clip" src-port="Z"/>
<Connection
dst="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Add_Clip"
dst-port="Y"
src="AVCDecoder_Decoding_Y_PredictionY_DemuxParserInfos" src-port="COEF_ACR_INTRA16"/>
<Connection
dst="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Algo_IntraPred_LUMA_16x16"
dst-port="PRED_MODE"
src="AVCDecoder_Decoding_Y_PredictionY_DemuxParserInfos" src-port="PRED_MODE16"/>
<Connection
dst="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Buffer_Neighbour_FullMb"
dst-port="MB_IN" src="AVCDecoder_Decoding_Y_PredictionY_Select" src-port="OUT"/>
<Connection
dst="AVCDecoder_Decoding_Y_PredictionY_Intra_16x16_Mgnt_Intra_16x16"
dst-port="NEIGHBOUR"
src="AVCDecoder_Decoding_Y_PredictionY_DemuxParserInfos" src-port="NEIGHBOUR16"/>
<Connection
dst="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Algo_IntraPred_LUMA_4x4"
dst-port="Y_LEFT"
src="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Mgnt_Intra4x4" src-port="Y_LEFT4"/>
<Connection
dst="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Algo_IntraPred_LUMA_4x4"
dst-port="Y_UP"
src="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Mgnt_Intra4x4" src-port="Y_UP4"/>
<Connection
dst="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Algo_IntraPred_LUMA_4x4"
dst-port="Y_UP_LEFT"
src="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Mgnt_Intra4x4" src-port="Y_UP_LEFT4"/>
<Connection
dst="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Algo_Add_4x4"
dst-port="X"
src="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Algo_IntraPred_LUMA_4x4" src-port="MPR"/>
<Connection
dst="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Algo_IntraPred_LUMA_4x4"
dst-port="AVAIL"
src="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Mgnt_Intra4x4" src-port="AVAIL"/>
<Connection
dst="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Algo_Merge_4x4_to_16x16_norasterscan"
dst-port="X"
src="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Algo_Add_4x4" src-port="Z"/>

```

```

<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Algo_Add_4x4"
  dst-port="Y"
  src="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Algo_Split_16x16_to_4x4_norasterscan" src-port="Y"/>
<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Algo_IntraPred_LUMA_4x4"
  dst-port="PRED_MODE"
  src="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Mgnt_Intra4x4" src-port="PRED_MODE4"/>
<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Mgnt_Intra4x4"
  dst-port="EDGE"
  src="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Buffer_Neighbour_4x4" src-port="EDGE"/>
<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Mgnt_Intra4x4"
  dst-port="MB_4X4"
  src="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Algo_Add_4x4" src-port="Z"/>
<Connection dst="AVCDecoder_Decoding_Y_PredictionY_Select"
  dst-port="IN_1"
  src="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Algo_Merge_4x4_to_16x16_norasterscan" src-port="Y"/>
<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Algo_Split_16x16_to_4x4_norasterscan"
  dst-port="X"
  src="AVCDecoder_Decoding_Y_PredictionY_DemuxParserInfos" src-port="COEF_ACR_INTRA4"/>
<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Mgnt_Intra4x4"
  dst-port="PRED_MODE"
  src="AVCDecoder_Decoding_Y_PredictionY_DemuxParserInfos" src-port="PRED_MODE4"/>
<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Buffer_Neighbour_4x4"
  dst-port="MB_IN" src="AVCDecoder_Decoding_Y_PredictionY_Select" src-port="OUT"/>
<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Intra_4x4_Mgnt_Intra4x4"
  dst-port="NEIGHBOUR"
  src="AVCDecoder_Decoding_Y_PredictionY_DemuxParserInfos" src-port="NEIGHBOUR4"/>
<Connection dst="AVCDecoder_Decoding_Y_PredictionY_Inter_Add_Clip"
  dst-port="X"
  src="AVCDecoder_Decoding_Y_PredictionY_Inter_Algo_Interp_Reord" src-port="MBPRED"/>
<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Inter_Algo_Interp_Reord"
  dst-port="INTERP"
  src="AVCDecoder_Decoding_Y_PredictionY_Inter_Algo_Interp_SeparableSixTapQuarterPelAVC"
  src-port="INTERP"/>
<Connection dst="AVCDecoder_Decoding_Y_PredictionY_Select"
  dst-port="IN_3"
  src="AVCDecoder_Decoding_Y_PredictionY_Inter_Add_Clip" src-port="Z"/>
<Connection dst="AVCDecoder_Decoding_Y_PredictionY_Inter_Add_Clip"
  dst-port="Y"
  src="AVCDecoder_Decoding_Y_PredictionY_DemuxParserInfos" src-port="COEF_ACR_INTER"/>
<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Intra_8x8_Algo_Add_8x8"
  dst-port="X"
  src="AVCDecoder_Decoding_Y_PredictionY_Intra_8x8_Algo_IntraPred_LUMA_8x8" src-port="MPR"/>
<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Intra_8x8_Algo_IntraPred_LUMA_8x8"
  dst-port="AVAIL"
  src="AVCDecoder_Decoding_Y_PredictionY_Intra_8x8_Mgnt_Intra8x8" src-port="AVAIL"/>
<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Intra_8x8_Algo_Merge_8x8_to_16x16_norasterscan"
  dst-port="X"
  src="AVCDecoder_Decoding_Y_PredictionY_Intra_8x8_Algo_Add_8x8" src-port="Z"/>

```

```

<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Intra_8x8_Mgnt_Intra8x8"
  dst-port="EDGE"
  src="AVCDecoder_Decoding_Y_PredictionY_Intra_8x8_Buffer_Neighbour_8x8" src-port="EDGE"/>
<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Intra_8x8_Mgnt_Intra8x8"
  dst-port="MB_8X8"
  src="AVCDecoder_Decoding_Y_PredictionY_Intra_8x8_Algo_Add_8x8" src-port="Z"/>
<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Intra_8x8_Algo_IntraPred_LUMA_8x8"
  dst-port="Y_LEFT"
  src="AVCDecoder_Decoding_Y_PredictionY_Intra_8x8_Mgnt_Intra8x8" src-port="Y_LEFT"/>
<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Intra_8x8_Algo_IntraPred_LUMA_8x8"
  dst-port="Y_UP"
  src="AVCDecoder_Decoding_Y_PredictionY_Intra_8x8_Mgnt_Intra8x8" src-port="Y_UP"/>
<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Intra_8x8_Algo_IntraPred_LUMA_8x8"
  dst-port="Y_UP_LEFT"
  src="AVCDecoder_Decoding_Y_PredictionY_Intra_8x8_Mgnt_Intra8x8" src-port="Y_UP_LEFT"/>
<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Intra_8x8_Algo_IntraPred_LUMA_8x8"
  dst-port="Y_UP_RIGHT"
  src="AVCDecoder_Decoding_Y_PredictionY_Intra_8x8_Mgnt_Intra8x8" src-port="Y_UP_RIGHT"/>
<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Intra_8x8_Algo_IntraPred_LUMA_8x8"
  dst-port="PRED_MODE"
  src="AVCDecoder_Decoding_Y_PredictionY_Intra_8x8_Mgnt_Intra8x8" src-port="PRED_MODE8"/>
<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Intra_8x8_Algo_Add_8x8"
  dst-port="Y"
  src="AVCDecoder_Decoding_Y_PredictionY_Intra_8x8_Algo_Split_16x16_to_8x8_norasterscan" src-port="Y"/>
<Connection dst="AVCDecoder_Decoding_Y_PredictionY_Select"
  dst-port="IN_4"
  src="AVCDecoder_Decoding_Y_PredictionY_Intra_8x8_Algo_Merge_8x8_to_16x16_norasterscan" src-port="Y"/>
<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Intra_8x8_Algo_Split_16x16_to_8x8_norasterscan"
  dst-port="X"
  src="AVCDecoder_Decoding_Y_PredictionY_DemuxParserInfos" src-port="COEF_ACR_INTRA8"/>
<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Intra_8x8_Mgnt_Intra8x8"
  dst-port="PRED_MODE"
  src="AVCDecoder_Decoding_Y_PredictionY_DemuxParserInfos" src-port="PRED_MODE8"/>
<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Intra_8x8_Buffer_Neighbour_8x8"
  dst-port="MB_IN" src="AVCDecoder_Decoding_Y_PredictionY_Select" src-port="OUT"/>
<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Intra_8x8_Mgnt_Intra8x8"
  dst-port="NEIGHBOUR"
  src="AVCDecoder_Decoding_Y_PredictionY_DemuxParserInfos" src-port="NEIGHBOUR8"/>
<Connection dst="AVCDecoder_Decoding_Y_DecodedPictureBuffer"
  dst-port="WD"
  src="AVCDecoder_Decoding_Y_PredictionY_Deblocking_Filter" src-port="MB_OUT"/>
<Connection
  dst="AVCDecoder_Decoding_Y_PredictionY_Inter_Algo_Interp_SeparableSixTapQuarterPelAVC"
  dst-port="RD" src="AVCDecoder_Decoding_Y_DecodedPictureBuffer" src-port="RD"/>
<Connection dst="AVCDecoder_Decoding_Y_ResidualY_I4x4_I8x8_mux"
  dst-port="COEF_AC_4x4"
  src="AVCDecoder_Decoding_Y_ResidualY_Algo_Merge_4x4_to_16x16" src-port="Y"/>
<Connection dst="AVCDecoder_Decoding_Y_ResidualY_I4x4_I8x8_mux"

```

```

dst-port="COEF_AC_8x8"
src="AVCDecoder_Decoding_Y_ResidualY_Algo_Merge_8x8_to_16x16" src-port="Y"/>
<Connection dst="AVCDecoder_Decoding_Y_ResidualY_I4x4_I8x8_mux"
dst-port="TRANSFORM_SIZE_8x8_FLAG"
src="AVCDecoder_Decoding_Y_ResidualY_I4x4_I8x8_demux" src-port="TRANSFORM_SIZE_8x8_FLAG_0"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_DCR_Hadamard_LUMA_IHT1d_0"
dst-port="X"
src="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_IS_Zigzag_4x4_DC" src-port="Y"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_DC_Transpose4x4_0"
dst-port="X"
src="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_DCR_Hadamard_LUMA_IHT1d_0" src-port="Y"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_DCR_Hadamard_LUMA_IHT1d_1"
dst-port="X"
src="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_DC_Transpose4x4_0" src-port="Y"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_DCR_Hadamard_LUMA_Scaling"
dst-port="I"
src="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_DCR_Hadamard_LUMA_IHT1d_1" src-port="Y"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_DC_Transpose4x4_1"
dst-port="X"
src="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_DCR_Hadamard_LUMA_Scaling" src-port="0"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_DCR_Hadamard_LUMA_Reordering"
dst-port="X"
src="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_DC_Transpose4x4_1" src-port="Y"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_DCR_Hadamard_LUMA_Scaling"
dst-port="QP"
src="AVCDecoder_Decoding_Y_ResidualY_I4x4_I8x8_demux" src-port="QP_4x4"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_DCR_Hadamard_LUMA_Scaling"
dst-port="MB_TYPE"
src="AVCDecoder_Decoding_Y_ResidualY_I4x4_I8x8_demux" src-port="MB_TYPE_4x4"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_DCR_Hadamard_LUMA_Scaling"
dst-port="SCALING_LIST"
src="AVCDecoder_Decoding_Y_ResidualY_I4x4_I8x8_demux" src-port="SCALING_LIST_4x4"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_Mgnt_IQ_INTRA16x16"
dst-port="COEF_AC"
src="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_Algo_IS_Zigzag_4x4_AC" src-port="Y"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_Algo_IQ_QSAndSLAndIDCTScaler_4x4"
dst-port="IN"
src="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_Mgnt_IQ_INTRA16x16" src-port="0"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_Algo_IQ_QSAndSLAndIDCTScaler_4x4"
dst-port="IS_DC"
src="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_Mgnt_IQ_INTRA16x16" src-port="IS_DC"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_IT4x4_Algo_Transpose4x4_0"
dst-port="X"
src="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_IT4x4_Algo_IT4x4_1d_0" src-port="Y"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_IT4x4_Algo_IT4x4_1d_1"
dst-port="X"

```

```

src="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_IT4x4_Algo_Transpose4x4_0" src-port="Y"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_IT4x4_Algo_Transpose4x4_1"
dst-port="X"
src="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_IT4x4_Algo_IT4x4_1d_1" src-port="Y"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_IT4x4_Algo_IT4x4_Addshift"
dst-port="X"
src="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_IT4x4_Algo_Transpose4x4_1" src-port="Y"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_IT4x4_Algo_IT4x4_1d_0"
dst-port="X"
src="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_Algo_IQ_QSAndSLAndIDCTScaler_4x4"
src-port="OUT"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_Algo_Merge_4x4_to_16x16"
dst-port="X"
src="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_IT4x4_Algo_IT4x4_Addshift" src-port="Y"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_Algo_IS_Zigzag_4x4_AC"
dst-port="X"
src="AVCDecoder_Decoding_Y_ResidualY_I4x4_I8x8_demux" src-port="COEF_AC_4x4"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_Algo_IQ_QSAndSLAndIDCTScaler_4x4"
dst-port="QP"
src="AVCDecoder_Decoding_Y_ResidualY_I4x4_I8x8_demux" src-port="QP_4x4"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_Mgnt_IQ_INTRA16x16"
dst-port="MB_TYPE"
src="AVCDecoder_Decoding_Y_ResidualY_I4x4_I8x8_demux" src-port="MB_TYPE_4x4"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_Algo_IQ_QSAndSLAndIDCTScaler_4x4"
dst-port="MB_TYPE"
src="AVCDecoder_Decoding_Y_ResidualY_I4x4_I8x8_demux" src-port="MB_TYPE_4x4"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_Algo_IQ_QSAndSLAndIDCTScaler_4x4"
dst-port="SCALING_LIST"
src="AVCDecoder_Decoding_Y_ResidualY_I4x4_I8x8_demux" src-port="SCALING_LIST_4x4"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_Mgnt_IQ_INTRA16x16"
dst-port="COEF_DC"
src="AVCDecoder_Decoding_Y_ResidualY_FUN_dcr_Algo_DCR_Hadamard_LUMA_Reordering" src-port="Y"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_8x8_Algo_IQ_QSAndSLAndIDCTScaler_8x8"
dst-port="IN"
src="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_8x8_Algo_IS_Zigzag_8x8_AC" src-port="Y"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_8x8_Algo_ICT_IntraPred_8x8"
dst-port="X"
src="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_8x8_Algo_IQ_QSAndSLAndIDCTScaler_8x8"
src-port="OUT"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_Algo_Merge_8x8_to_16x16"
dst-port="X"
src="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_8x8_Algo_ICT_IntraPred_8x8" src-port="Y"/>
<Connection
dst="AVCDecoder_Decoding_Y_ResidualY_FUN_IS_IQ_IT_L_8x8_Algo_IS_Zigzag_8x8_AC"
dst-port="X"
src="AVCDecoder_Decoding_Y_ResidualY_I4x4_I8x8_demux" src-port="COEF_AC_8x8"/>
<Connection

```