

# OMX Media Component

User's Manual: Video Decoder Common Part

32

— Preliminary —

All information contained in these materials, including products and product specifications, represents information on the product at the time of publication and is subject to change by Renesas Electronics Corp. without notice. Please review the latest information published by Renesas Electronics Corp. through various means, including the Renesas Electronics Corp. website (<http://www.renesas.com>).

## Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation of these circuits, software, and information in the design of your equipment. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics has used reasonable care in preparing the information included in this document, but Renesas Electronics does not warrant that such information is error free. Renesas Electronics assumes no liability whatsoever for any damages incurred by you resulting from errors in or omissions from the information included herein.
3. Renesas Electronics does not assume any liability for infringement of patents, copyrights, or other intellectual property rights of third parties by or arising from the use of Renesas Electronics products or technical information described in this document. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You should not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part. Renesas Electronics assumes no responsibility for any losses incurred by you or third parties arising from such alteration, modification, copy or otherwise misappropriation of Renesas Electronics product.
5. Renesas Electronics products are classified according to the following two quality grades: "Standard" and "High Quality". The recommended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below.  
"Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots etc.  
"High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control systems; anti-disaster systems; anti-crime systems; and safety equipment etc.

Renesas Electronics products are neither intended nor authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems, surgical implantations etc.), or may cause serious property damages (nuclear reactor control systems, military equipment etc.). You must check the quality grade of each Renesas Electronics product before using it in a particular application. You may not use any Renesas Electronics product for any application for which it is not intended. Renesas Electronics shall not be in any way liable for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for which the product is not intended by Renesas Electronics.

6. You should use the Renesas Electronics products described in this document within the range specified by Renesas Electronics, especially with respect to the maximum rating, operating supply voltage range, movement power voltage range, heat radiation characteristics, installation and other product characteristics. Renesas Electronics shall have no liability for malfunctions or damages arising out of the use of Renesas Electronics products beyond such specified ranges.
7. Although Renesas Electronics endeavors to improve the quality and reliability of its products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please be sure to implement safety measures to guard them against the possibility of physical injury, and injury or damage caused by fire in the event of the failure of a Renesas Electronics product, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult, please evaluate the safety of the final products or systems manufactured by you.
8. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please use Renesas Electronics products in compliance with all applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive. Renesas Electronics assumes no liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
9. Renesas Electronics products and technology may not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You should not use Renesas Electronics products or technology described in this document for any purpose relating to military applications or use by the military, including but not limited to the development of weapons of mass destruction. When exporting the Renesas Electronics products or technology described in this document, you should comply with the applicable export control laws and regulations and follow the procedures required by such laws and regulations.
10. It is the responsibility of the buyer or distributor of Renesas Electronics products, who distributes, disposes of, or otherwise places the product with a third party, to notify such third party in advance of the contents and conditions set forth in this document, Renesas Electronics assumes no responsibility for any losses incurred by you or third parties as a result of unauthorized use of Renesas Electronics products.
11. This document may not be reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products, or if you have any other inquiries.

(Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

(Note 2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

## Table of Contents

<b>1. OVERVIEW.....</b>	<b>5</b>
1.1. About This Document .....	5
1.2. Video Decoder Media Component Overview and Scope .....	5
1.3. Required Header Files .....	6
1.4. Related Documents .....	6
1.5. Terminology .....	7
<b>2. FUNCTIONS.....</b>	<b>8</b>
2.1. Function Details .....	8
2.1.1. Video Decode Post Processing .....	8
2.1.2. Output Reordering .....	8
2.1.3. Multi Stream Decoding .....	8
2.1.4. Error Handling .....	8
2.2. Port .....	9
2.2.1. Port .....	9
2.3. Data Flow .....	10
<b>3. I/O DATA FORMAT.....</b>	<b>11</b>
3.1. Buffer Payload .....	11
3.1.1. Input Buffer Payload .....	11
3.1.2. Output Buffer Payload .....	11
3.2. Input Stream Data Format .....	12
3.3. Output Picture Data Format .....	13
<b>4. API REFERENCE.....</b>	<b>14</b>
4.1. OpenMAX IL Macro Functions .....	14
4.1.1. OMX_UseBuffer .....	14
4.2. Callback Functions .....	15
4.2.1. (*EventHandler)( ) .....	15
<b>5. INDEXES.....</b>	<b>18</b>
5.1. Standard Indexes of Video Decoder Media Component .....	18
5.1.1. OMX_IndexParamPortDefinition .....	18
5.1.2. OMX_IndexParamVideoPortFormat .....	18
5.1.3. OMX_IndexConfigCommonOutputCrop .....	19
5.1.4. OMX_IndexConfigCommonScale .....	19
5.1.5. OMX_IndexParamVideoProfileLevelQuerySupported .....	19
5.1.6. OMX_IndexParamVideoProfileLevelCurrent .....	19
5.2. Extended Indexes of Video Decoder Media Component .....	20
5.2.1. OMXR_MC_IndexParamVideoReorder .....	20
5.2.2. OMXR_MC_IndexParamVideoDeinterlaceMode .....	20
5.3. Valid Indexes for OpenMAX IL Macro Functions .....	21
<b>6. STRUCTURES.....</b>	<b>22</b>
6.1. OMX_VIDEO_PORTDEFINITIONTYPE .....	23
6.1.1. OMX_VIDEO_PORTDEFINITIONTYPE (Input Port) .....	23
6.1.2. OMX_VIDEO_PORTDEFINITIONTYPE (Output Port) .....	25
6.2. OMX_VIDEO_PARAM_PORTFORMATTYPE .....	34
6.2.1. OMX_VIDEO_PARAM_PORTFORMATTYPE (Input Port) .....	34
6.2.2. OMX_VIDEO_PARAM_PORTFORMATTYPE (Output Port) .....	36
6.3. OMX_VIDEO_PARAM_PROFILELEVELTYPE .....	38
6.4. OMXR_MC_VIDEO_PARAM_REORDERTYPE .....	39
6.5. OMXR_MC_VIDEO_DECODERRESULTTYPE .....	44

6.6.	OMX_CONFIG_RECTTYPE .....	39
6.7.	OMX_CONFIG_SCALEFACTORTYPE.....	41
6.8.	OMXR_MC_VIDEO_PARAM_DEINTERLACE_MODETYPE.....	46
6.9.	Specific Usage on Common Structure Members.....	48
6.9.1.	Buffer Flags ( <i>nFlags</i> ) .....	49
<b>7.</b>	<b>APPENDIX .....</b>	<b>52</b>
7.1.	Errors and Error Handling.....	52
7.2.	Attentions on Decoding for Corrupted Streams.....	53
7.3.	Attentions on Port Flush Operation .....	54
7.4.	Parameter Relation between PortDefinition and Crop .....	55
7.5.	OMX_EventPortSettingsChanged Event Conditions .....	56
7.6.	The output delay and the means of reduction.....	57

## Figures

Figure 1-1 Software Stacks and Scope.....	5
Figure 2-1 Data Flow of Video Decoder Media Component.....	10
Figure 3-1 Output Buffer Sequence .....	11
Figure 4-1 OMX_EventPortSettingsChanged Event Sequence (1).....	16
Figure 4-2 OMX_EventPortSettingsChanged Event Sequence (2).....	17
Figure 6-1 Byte and Plane Order: YUV420 Planar .....	29
Figure 6-2 Byte and Plane Order: YUV420 Semi Planar .....	29
Figure 6-3 Picture Format and Parameters: YUV420 Planar - Frame Arrangement.....	30
Figure 6-4 Picture Format and Parameters: YUV420 Planar - Field Arrangement .....	31
Figure 6-5 Picture Format and Parameters: YUV420 Semi Planar - Frame Arrangement .....	32
Figure 6-6 Picture Format and Parameters: YUV420 Semi Planar – Field Arrangement.....	33
Figure 6-7 Input Buffer Sequence: OMX_BUFFERFLAG_EOS Flag Usage .....	50
Figure 6-8 Output Buffer Sequence: OMX_BUFFERFLAG_EOS Flag Usage .....	51
Figure 6-9 Input Buffer Sequence: OMX_BUFFERFLAG_ENDOFFRAME Flag Usage.....	51
Figure 7-1 Parameter Relation between PortDefinition and Crop .....	55
Figure 7-2 Definition of the output delay.....	57
Figure 7-2 Output delay reduction .....	58

## Tables

Table 1-1 Required Header Files .....	6
Table 1-2 List of Related Documents .....	6
Table 1-3 Terminology .....	7
Table 2-1 Ports of Video Decoder Media Component .....	9
Table 5-1 Available Standard Indexes for Video Decoder Media Component .....	18
Table 5-2 Available extended indexes for Video Decoder Media Component .....	20
Table 5-3 Valid Indexes and OpenMAX IL Macro Function .....	21
Table 6-1 Video Decoder Media Component Specific Structures .....	22
Table 6-2 Notation for the access attribute of a structure member .....	22
Table 6-3 Combination of Picture Format and Memory Arrangement .....	28
Table 6-4 Specific Usage on Common Structure Members .....	48
Table 6-5 Specific Usage on Buffer Flags .....	49
Table 7-1 Errors and Error Handling .....	52

## OMX Media Component Video Decoder Common Part

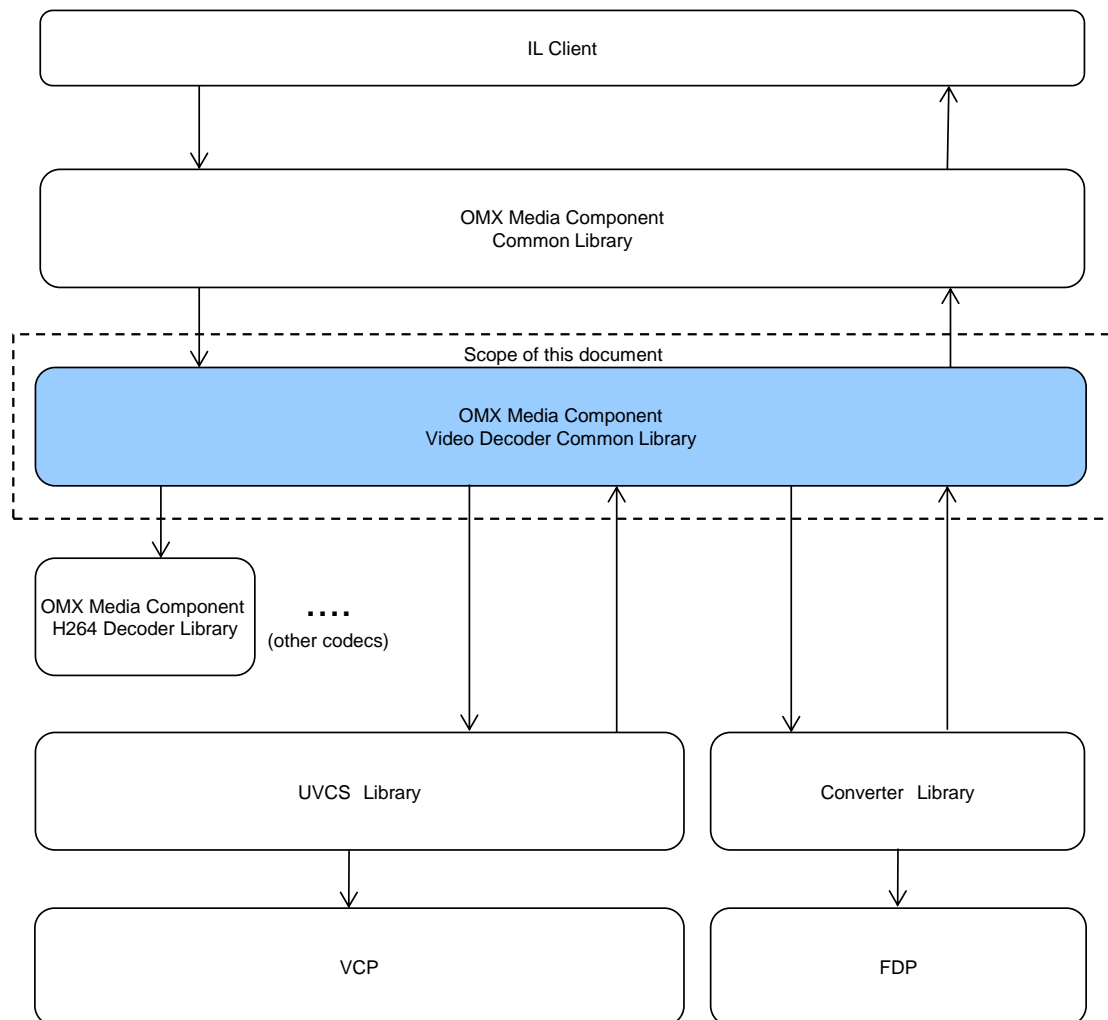
### 1. Overview

#### 1.1. About This Document

This document is the User's Manual for OMX Media Component. It describes the specifications that are common to Video Decoder Media Components. For the specifications that are common to all the OMX Media Components, see related documents [1].

#### 1.2. Video Decoder Media Component Overview and Scope

Figure 1-1 illustrates the software stacks for the Video Decoder Media Component and shows the scope of this document. OMX Media Component Video Decoder Library is a library that provides functions that are common to all video decoders.



**Figure 1-1 Software Stacks and Scope**

This document describes the specifications of the OMX Media Component Video Decoder Library part. For the specifications of the OMX Media Component Common Library, see related documents [1]. For the specifications of individual Video Decoder Media Component Libraries, see the corresponding User's Manuals.

### 1.3. Required Header Files

Table 1-1 lists the header files that are required to use the OMX extended indexes and structures that are described in this document. Regarding the other header files, see related documents [1].

**Table 1-1 Required Header Files**

File name	Remarks
OMXR_Extension_video.h	-
OMXR_Extension_vdcmn.h	-

### 1.4. Related Documents

Table 1-2 lists the related documents.

**Table 1-2 List of Related Documents**

No.	Document Name	Remarks
[1]	OMX Media Component User's Manual Common Part	The common specifications for OMX Media Component
[2]	OpenMAX Integration Layer Application Programming Interface Specification Version 1.1.2, September 1, 2008	<a href="http://www.khronos.org/registry/omxil/specs/OpenMAX_IL_1_1_2_Specification.pdf">http://www.khronos.org/registry/omxil/specs/OpenMAX_IL_1_1_2_Specification.pdf</a>
[3]	OMX Integration Guide for <OS >	Integration guide for OMX Media Component. Substitute <OS> with your target operating system name.



## 1.5. Terminology

Table 1-3 lists the terms that are used in this document.

**Table 1-3 Terminology**

Term	Abbreviation	Description
Video Port Base	VPB	The base value of the port index of the Video Media Component. The port index values of the input and output ports are obtained by adding offset values to this base value.
UVCS	-	Renesas proprietary video codec software module that provides multi-processing function for video decoding and encoding. OMX Video Codec products contain UVCS libraries.
VCP	-	The abbreviation for Video Coding Processor. The VCP is the Renesas Hardware IP and provides video decoding and encoding functions for compressed video stream.
FDP	-	The abbreviation for Fine Display Processor. The FDP is the Renesas Hardware IP and provides video processing functions for deinterlacing, color space conversion and so on.
Converter Library	-	The software module controls video post processing module (ex.FDP).
Hardware Address	-	A memory address that is accessible from Hardware IPs.
Key Frame	-	A frame in which a complete image is stored with no reference to the other frames. Also known as Intra Frame.

---

## 2. Functions

Video Decoder Media Component is a media component which provides functions to decode video stream that is compressed according to each codec standard. Video Decoder Media Component receives encoded stream data on the input port and emits the decoded video frame data on the output ports.

### 2.1. Function Details

#### 2.1.1. Video Decode Post Processing

Since the UVCS Library outputs a picture data in tiled addressing, the IL client cannot read as is. Video Decoder Media Component converts a tiled picture data to linear data that is available for the IL client.

Video Decoder Media Component also provides the color space conversion and the deinterlace function as the video decode post processing. For the data flow of the video decoding, see section 2.3. For the setting of the output color format and the deinterlace mode, see section 5.1.1, section 5.1.2 and section 5.2.2.

#### 2.1.2. Output Reordering

Video Decoder Media Component supports two types of output ordering: display ordering and decode output ordering. For the details, see section 5.2.1.

#### 2.1.3. Multi Stream Decoding

OMX Video Decoder and OMX Video Encoder provide simultaneous multi stream processing. The maximum number of streams that OMX Video Decoder and OMX Video Encoder guarantee to process simultaneously is 2 per a VCP hardware instance. For example, the maximum number of streams for a LSI which has 2 VCP hardware instances is 4. When the total number of streams exceeds the maximum number, the performance must be checked by user to ensure the system requirement.

#### 2.1.4. Error Handling

For a corrupted input stream, Video Decoder Media Component continues to decode with applying error concealments as possible. When Video Decoder Media Component detects an error that affects continuous processing, Video Decoder Media Component notifies an error event (OMX\_ErrorStreamCorrupt) to IL client. Once an error event notified, Video Decoder Media Component stops the decode operation and the state transition to the OMX\_StateIdle state is required to restart the decode operation.

For the details of the error handling, see section 7.1 and section 7.2.

## 2.2. Port

### 2.2.1. Port

Table 2-1 lists the ports of Video Decoder Media Component.

**Table 2-1 Ports of Video Decoder Media Component**

Port Index	Domain	Direction
VPB+0	video	input
VPB+1	video	output

For the available indexes of each port, see section 5.3.

### 2.3. Data Flow

Figure 2-1 illustrates the data flow of Video Decoder Media Component.

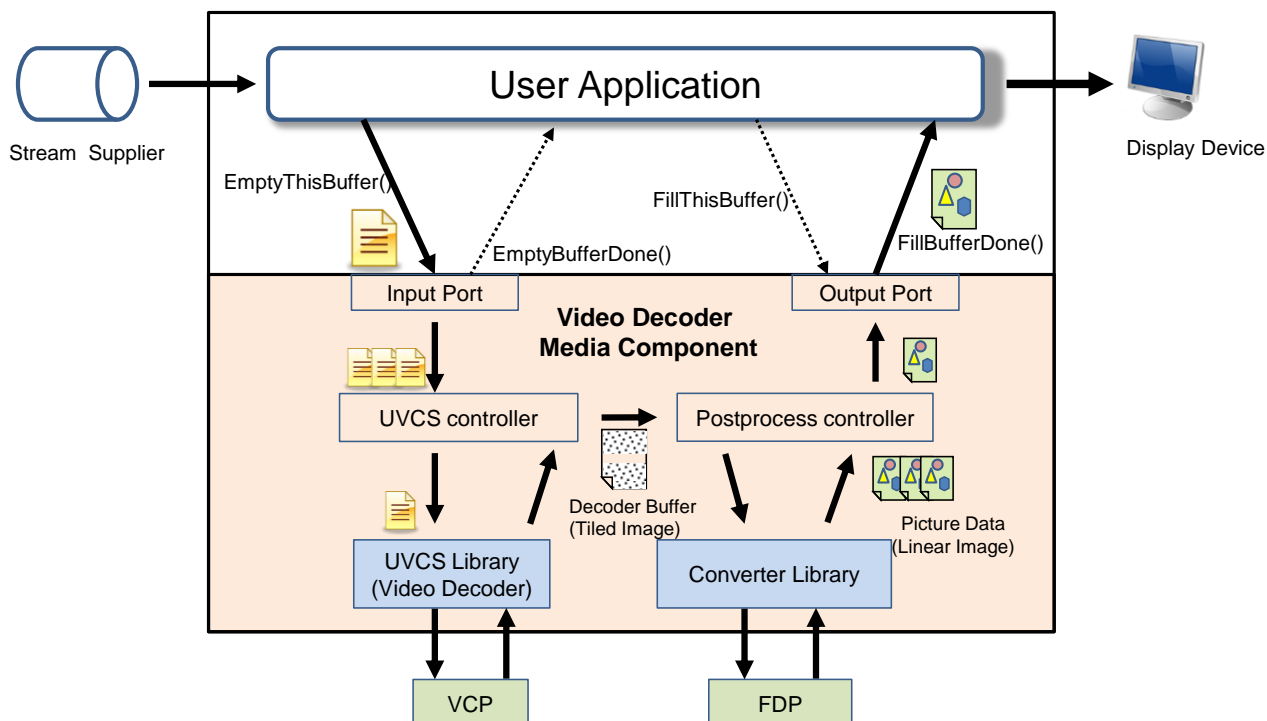


Figure 2-1 Data Flow of Video Decoder Media Component

### 3. I/O Data Format

#### 3.1. Buffer Payload

##### 3.1.1. Input Buffer Payload

The input buffer payload depends on the codec component. See Media Component User's Manual for each codec component.

##### 3.1.2. Output Buffer Payload

Figure 3-1 illustrates the output buffer sequence of Video Decoder Media Component. Video Decoder Media Component stores one decoded picture data into one output buffer.

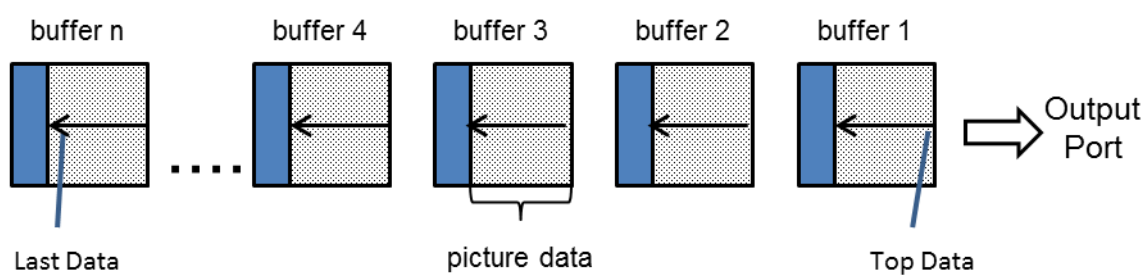


Figure 3-1 Output Buffer Sequence

### **3.2. Input Stream Data Format**

The format of input stream data depends on the codec component. See Media Component User's Manual for each codec component.

### 3.3. Output Picture Data Format

Video Decoder Media Component supports the two output format: YUV420 Planar (I420) and YUV420 Semi Planar (NV12). The output format can be specified via the `OMX_IndexParamPortDefinition` index and the `OMX_IndexParamVideoPortFormat` index. For the detail of the picture formats, see 6.1.2. For the details of the indexes, see 5.1.1 and 5.1.2.

For an interlaced input stream, the output format depends on the *eDeinterlace* member of the `OMXR_MC_VIDEO_PARAM_DEINTERLACE_MODETYPE` structure:

- (1) When the *eDeinterlace* member is not `OMXR_MC_VIDEO_DeinterlaceNone`, the output data is deinterlaced frame data.
- (2) When the *eDeinterlace* member is `OMXR_MC_VIDEO_DeinterlaceNone`, the output data is a pair of field data that is not deinterlaced.

For the details, see section 6.8.

## 4. API Reference

Video Decoder Media Component inherits the API specifications from related document [1]. This section describes only the difference that depends on Video Decoder Media Component.

### 4.1. OpenMAX IL Macro Functions

#### 4.1.1. OMX\_UseBuffer

[Reference]	Related document [1]
[Description]	<ul style="list-style-type: none"><li>- OMX_UseBuffer is supported for the output port.</li><li>- The <i>nSizeBytes</i> parameter must be set the same value with the <i>nBufferSize</i> member of the OMX_PARAM_PORTDEFINITIONTYPE structure that is obtained via OMX_GetParameter().</li><li>- The <i>pBuffer</i> parameter must be a "hardware address". For the detail of 'hardware address' see related document [3].</li></ul>
[Notes]	None



## 4.2. Callback Functions

### 4.2.1. (\*EventHandler)()

- OMX\_EventPortSettingsChanged Event

When Video Decoder Media Component needs to reallocate the buffers allocated for the output port, Video Decoder Media Component notifies the OMX\_EventPortSettingsChanged event to the IL client. The IL client can get the updated information for the port via the OMX\_IndexParamPortDefinition index (see section 5.1.1) and the OMX\_VIDEO\_PORTDEFINITIONTYPE structure (see section 6.1). For the detail conditions of the event notification, see section 7.5.

Once Video Decoder Media Component sends the OMX\_EventPortSettingsChanged event, it stops the decode operation and starts to wait the buffer reallocation for the output port. The IL client can continue the decode operation by either of the following procedure:

- (1) Disable the output port and then enable the output port (see Figure 4-1)
- (2) Change into the OMX\_StateIdle state and then resume to the OMX\_StateExecuting state (see Figure 4-2)

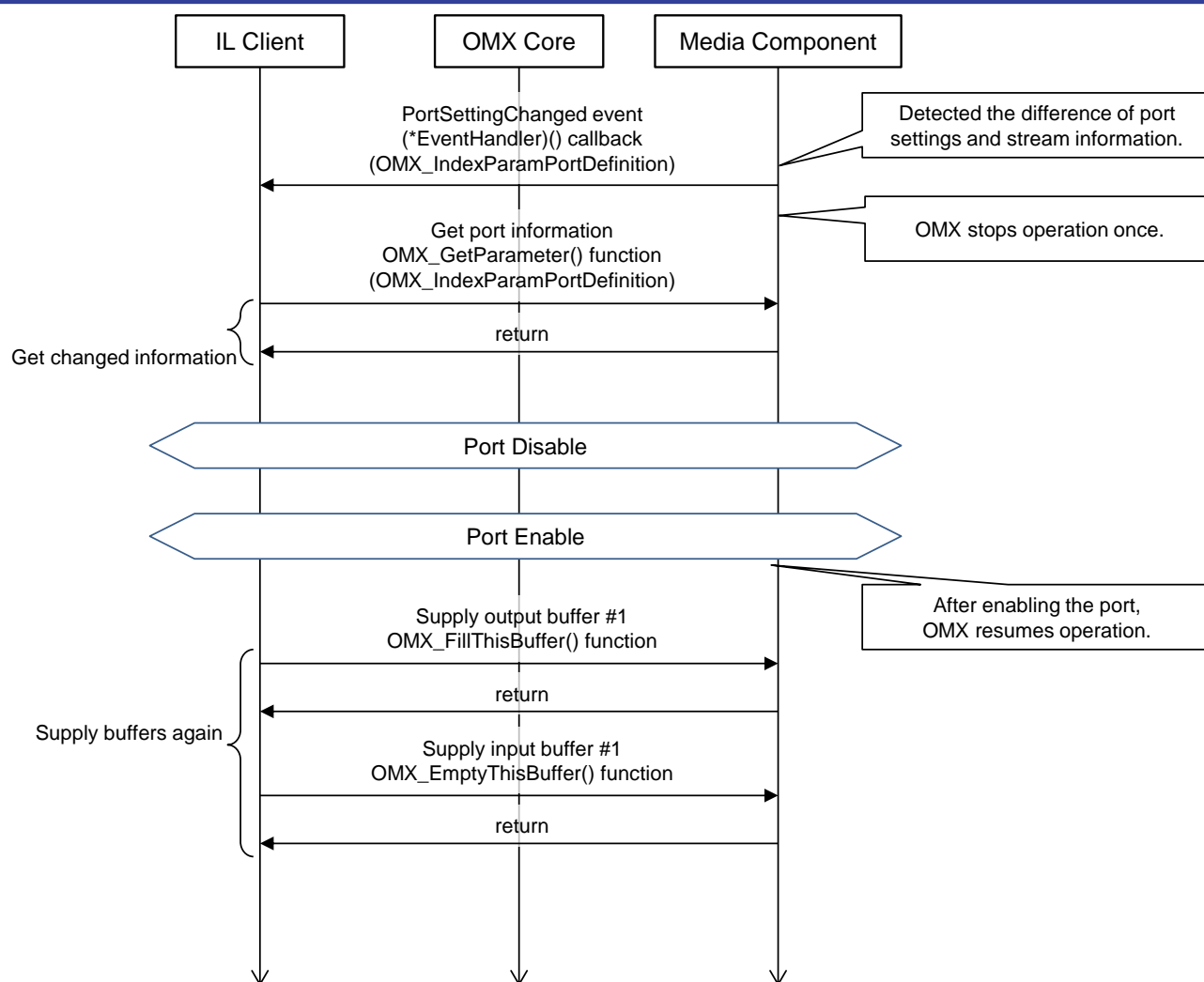


Figure 4-1 OMX\_EventPortSettingsChanged Event Sequence (1)

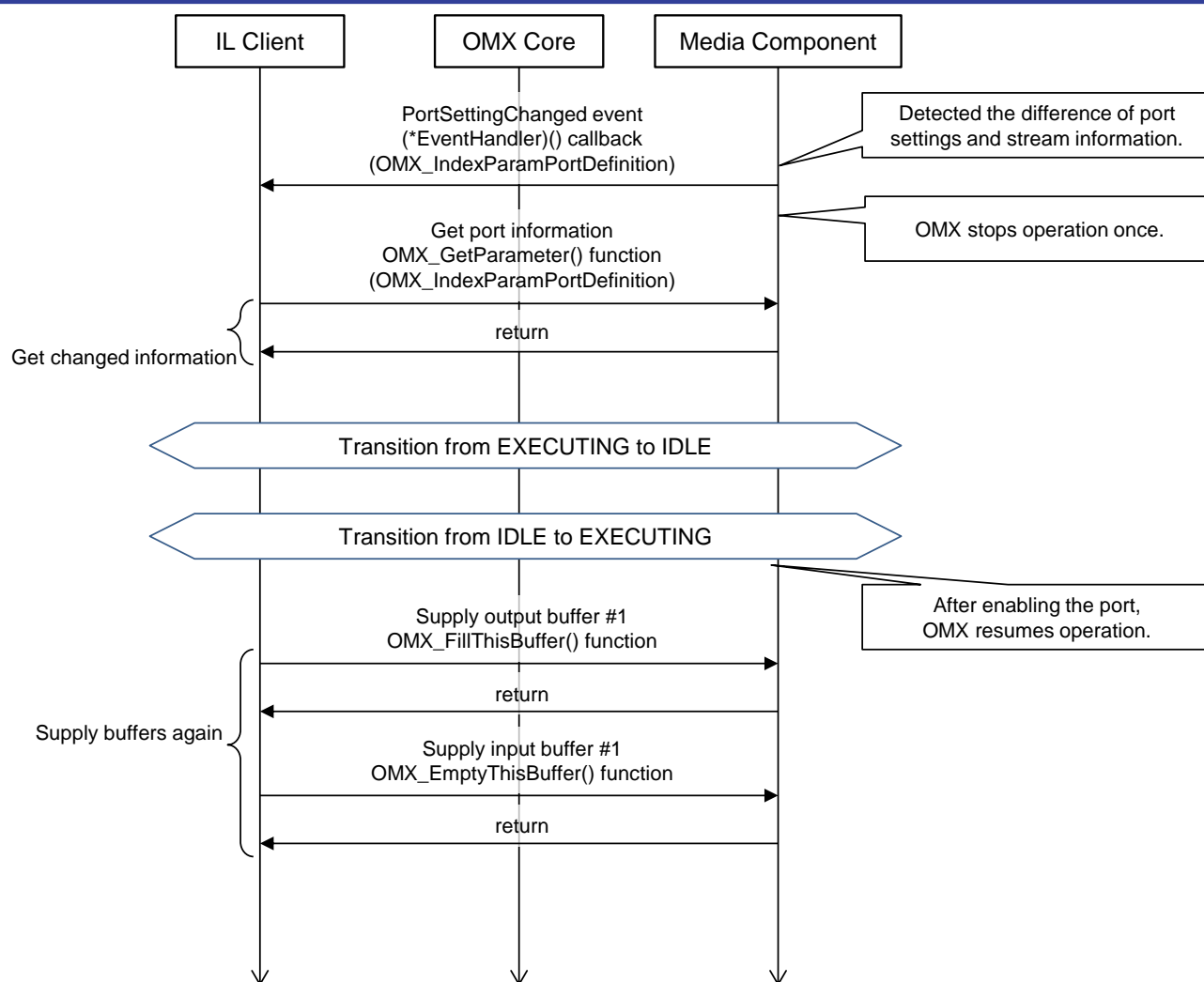


Figure 4-2 OMX\_EventPortSettingsChanged Event Sequence (2)

## 5. Indexes

### 5.1. Standard Indexes of Video Decoder Media Component

Table 5-1 lists the OpenMAX IL standard indexes that are available for Video Decoder Media Component.

**Table 5-1 Available Standard Indexes for Video Decoder Media Component**

Index	Description
OMX_IndexParamPortDefinition	See section 5.1.1
OMX_IndexParamVideoPortFormat	See section 5.1.2
OMX_IndexConfigCommonOutputCrop	See section 5.1.3
OMX_IndexConfigCommonScale	See section 5.1.4
OMX_IndexParamVideoProfileLevelQuerySupported	See section 5.1.5
OMX_IndexParamVideoProfileLevelCurrent	See section 5.1.6

#### 5.1.1. OMX\_IndexParamPortDefinition

[Description]	The index to access the video port information such as picture size and the color format for both the input port and the output port.
[Corresponding Structure]	OMX_VIDEO_PORTDEFINITIONTYPE structure
[Notes]	None

#### 5.1.2. OMX\_IndexParamVideoPortFormat

[Description]	The index to query the supported output formats.
[Corresponding Structure]	OMX_VIDEO_PARAM_PORTFORMATTYPE structure
[Notes]	None

### 5.1.3. OMX\_IndexConfigCommonOutputCrop

[Description]	The index to get the crop information of the decoded picture.
[Corresponding Structure]	OMX_CONFIG_RECTTYPE structure
[Notes]	None

### 5.1.4. OMX\_IndexConfigCommonScale

[Description]	The index to get the aspect ratio of the decoded picture.
[Corresponding Structure]	OMX_CONFIG_SCALEFACTORTYPE structure
[Notes]	None

### 5.1.5. OMX\_IndexParamVideoProfileLevelQuerySupported

[Description]	The index to query the profiles and the levels that are supported by the Video Decoder Media Component.
[Corresponding Structure]	OMX_VIDEO_PARAM_PROFILELEVELTYPE structure
[Notes]	None

### 5.1.6. OMX\_IndexParamVideoProfileLevelCurrent

[Description]	The index to get the profiles and the levels of the stream that is processing by the Video Decoder Media Component.
[Corresponding Structure]	OMX_VIDEO_PARAM_PROFILELEVELTYPE structure
[Notes]	None

## 5.2. Extended Indexes of Video Decoder Media Component

Table 5-2 lists the OMX extended indexes that are available for Video Decoder Media Component.

**Table 5-2 Available extended indexes for Video Decoder Media Component**

Index	Description
OMXR_MC_IndexParamVideoReorder	See section 5.2.1
OMXR_MC_IndexParamVideoDeinterlaceMode	See section 5.2.2

### 5.2.1. OMXR\_MC\_IndexParamVideoReorder

[Description] The index to access the output picture reordering setting.

[String] OMX.RENESAS.INDEX.PARAM.VIDEO.REORDER

[Corresponding Structure] OMX\_CONFIG\_RECTTYPE  
[Definition] See related document [2] 4.2.18.

[Index] OMX\_IndexConfigCommonOutputCrop

[Member]

Member Name
<i>nSize</i>
<i>nVersion</i>
<i>nPortIndex</i>
<i>nLeft</i>
<i>nTop</i>
<i>nWidth</i>
<i>nHeight</i>

[Notes] None.

[Details]

#### ***nSize***

<b>Write Value</b>	The size of the structure in bytes.
<b>Read Value</b>	-
<b>Initial Value</b>	-
<b>Notes</b>	-

#### ***nVersion***

<b>Write Value</b>	The version number of OpenMAX IL specifications
<b>Read Value</b>	-
<b>Initial Value</b>	-
<b>Notes</b>	-

#### ***nPortIndex***

<b>Write Value</b>	VPB + 1
<b>Read Value</b>	-
<b>Initial Value</b>	-
<b>Notes</b>	-

#### ***nLeft***

<b>Write Value</b>	-
<b>Read Value</b>	The leftmost coordinate of the active video

<b>Initial Value</b>	0
<b>Notes</b>	<ul style="list-style-type: none"> <li>– This member is updated based on the input stream.</li> <li>– For the relationship between this member and the other information, see Figure 7-1.</li> </ul>

#### ***nTop***

<b>Write Value</b>	-
<b>Read Value</b>	The topmost coordinate of the active video
<b>Initial Value</b>	0
<b>Notes</b>	<ul style="list-style-type: none"> <li>– This member is updated based on the input stream.</li> <li>– For the relationship between this member and the other information, see Figure 7-1.</li> </ul>

#### ***nWidth***

<b>Write Value</b>	-
<b>Read Value</b>	The width of the active video
<b>Initial Value</b>	176
<b>Notes</b>	<ul style="list-style-type: none"> <li>– This member is updated in accordance with the input stream.</li> <li>– This member follows the <i>nFrameWidth</i> member of the OMX_VIDEO_PORTDEFINITIONTYPE structure between this member and the other size-related information.</li> </ul>

#### ***nHeight***

<b>Write Value</b>	-
<b>Read Value</b>	The height of the active video
<b>Initial Value</b>	144
<b>Notes</b>	<ul style="list-style-type: none"> <li>– This member is updated in accordance with the input stream.</li> <li>– This member follows the <i>nFrameHeight</i> member of the OMX_VIDEO_PORTDEFINITIONTYPE structure between this member and the other size-related information, see Figure 7-1.</li> </ul>

### **5.1. OMX\_CONFIG\_SCALEFACTORTYPE**

[Definition] See related document [2] 4.2.9.

[Index] OMX\_IndexConfigCommonScale

[Member]

Member Name
<i>nSize</i>
<i>nVersion</i>
<i>nPortIndex</i>
<i>xWidth</i>
<i>xHeight</i>

[Notes] None.

[Details]

#### ***nSize***

<b>Write Value</b>	The size of the structure in bytes.
<b>Read Value</b>	-
<b>Initial Value</b>	-
<b>Notes</b>	-

***nVersion***

<b>Write Value</b>	The version number of OpenMAX IL specifications 1.1.2
<b>Read Value</b>	-
<b>Initial Value</b>	-
<b>Notes</b>	-

***nPortIndex***

<b>Write Value</b>	VPB + 1
<b>Read Value</b>	-
<b>Initial Value</b>	-
<b>Notes</b>	-

***xWidth***

<b>Write Value</b>	-
<b>Read Value</b>	The scaling ratio in the horizontal direction in Q16 format.
<b>Initial Value</b>	0x00010000
<b>Notes</b>	<p>The lesser of <i>xWidth</i> and <i>xHeight</i> is set to 0x00010000 normalized. For example, in the case where the source image is 4:3, <i>xHeight</i> is set to 0x00010000 because the vertical side is the shorter side. Then <i>xWidth</i> is calculated by the aspect ratio.</p> $4 : 3 = xWidth : 0x00010000$ $3 * xWidth = 4 * 0x00010000$ $xWidth = (4 * 0x00010000) / 3$ $xWidth = 0x00015555$ <p>Thus <i>xWidth</i> and <i>xHeight</i> are determined to 0x00015555 and 0x00010000 respectively.</p>

***xHeight***

<b>Write Value</b>	-
<b>Read Value</b>	The scaling ratio in the vertical direction in Q16 format.
<b>Initial Value</b>	0x00010000
<b>Notes</b>	- See Notes of the <i>xWidth</i> member.

OMXR\_MC\_VIDEO\_PARAM\_REORDERTYPE structure

[Notes] None

**5.1.1. OMXR\_MC\_IndexParamVideoDeinterlaceMode**

[Description] The index to access the deinterlace mode setting.

[String] OMX.RENESAS.INDEX.PARAM.VIDEO.DEINTERLACEMODE

[Corresponding Structure] OMXR\_MC\_VIDEO\_PARAM\_DEINTERLACE\_MODETYPE structure

[Notes] None





## 5.2. Valid Indexes for OpenMAX IL Macro Functions

Table 5-3 shows which index is available for each port and which OpenMAX IL Macro function can be called to access the index.

**Table 5-3 Valid Indexes and OpenMAX IL Macro Function**

PortIndex	Index	Get/SetParameter		Get/SetConfig	
		Get	Set	Get	Set
VPB+0	OMX_IndexParamPortDefinition	X	X	-	-
	OMX_IndexParamVideoPortFormat	X	X	-	-
	OMX_IndexParamVideoProfileLevelQuerySupported	X	-	-	-
	OMX_IndexParamVideoProfileLevelCurrent	X	X	-	-
VPB+1	OMX_IndexParamPortDefinition	X	X	-	-
	OMX_IndexParamVideoPortFormat	X	X	-	-
	OMX_IndexConfigCommonOutputCrop	-	-	X	-
	OMX_IndexConfigCommonScale	-	-	X	-
	OMXR_MC_IndexParamVideoReorder	X	X	-	-
	OMXR_MC_IndexParamVideoDeinterlaceMode	X	X	-	-

X : Valid  
- : Invalid

## 6. Structures

Table 6-1 lists Video Decoder Media Component specific structures.

**Table 6-1 Video Decoder Media Component Specific Structures**

Structure Name		Description
OMX_VIDEO_PORTDEFINITIONTYPE		See section 6.1
OMX_VIDEO_PARAM_PORTFORMATTYPE		See section 6.2
OMX_VIDEO_PARAM_PROFILELEVELTYPE		See section 6.3
OMX_CONFIG_RECTTYPE		See section 6.5
[Definition]	See related document [2] 4.2.18.	
[Index]	OMX_IndexConfigCommonOutputCrop	
[Member]	<b>Member Name</b>	
	<i>nSize</i>	
	<i>nVersion</i>	
	<i>nPortIndex</i>	
	<i>nLeft</i>	
	<i>nTop</i>	
	<i>nWidth</i>	
	<i>nHeight</i>	
[Notes]	None.	
[Details]	<b><i>nSize</i></b>	
	<b>Write Value</b>	The size of the structure in byte
	<b>Read Value</b>	-
	<b>Initial Value</b>	-
	<b>Notes</b>	-
	<b><i>nVersion</i></b>	
	<b>Write Value</b>	The version number of OpenMA
	<b>Read Value</b>	-
	<b>Initial Value</b>	-
	<b>Notes</b>	-
	<b><i>nPortIndex</i></b>	
	<b>Write Value</b>	VPB + 1
	<b>Read Value</b>	-
	<b>Initial Value</b>	-
	<b>Notes</b>	-
	<b><i>nLeft</i></b>	
	<b>Write Value</b>	-
	<b>Read Value</b>	The leftmost coordinate of the a
	<b>Initial Value</b>	0
	<b>Notes</b>	<ul style="list-style-type: none"> <li>- This member is updated ba</li> <li>- For the relationship between information, see Figure 7-1.</li> </ul>
	<b><i>nTop</i></b>	
	<b>Write Value</b>	-
	<b>Read Value</b>	The topmost coordinate of the a

	<b>Initial Value</b>	0																								
	<b>Notes</b>	- This member is updated based on the input stream. - For the relationship between this member and the other size-related information, see Figure 7-1.																								
	<b><i>nWidth</i></b>																									
	<b>Write Value</b>	-																								
	<b>Read Value</b>	The width of the active video																								
	<b>Initial Value</b>	176																								
	<b>Notes</b>	- This member is updated in accordance with the picture size information of the input stream. - This member follows the <i>nFrameWidth</i> member of the OMX_VIDEO_PORTDEFINITIONTYPE structure. For the relationship between this member and the other size-related information, see Figure 7-1.																								
	<b><i>nHeight</i></b>																									
	<b>Write Value</b>	-																								
	<b>Read Value</b>	The height of the active video																								
	<b>Initial Value</b>	144																								
	<b>Notes</b>	- This member is updated in accordance with the picture size information of the input stream. - This member follows the <i>nFrameHeight</i> member of the OMX_VIDEO_PORTDEFINITIONTYPE structure. - For the relationship between this member and the other size-related information, see Figure 7-1.																								
<b>6.1. OMX_CONFIG_SCALEFACTORTYPE</b>																										
[Definition]	See related document [2] 4.2.9.																									
[Index]	OMX_IndexConfigCommonScale																									
[Member]	<table><tr><th colspan="2">Member Name</th></tr><tr><td><i>nSize</i></td><td></td></tr><tr><td><i>nVersion</i></td><td></td></tr><tr><td><i>nPortIndex</i></td><td></td></tr><tr><td><i>xWidth</i></td><td></td></tr><tr><td><i>xHeight</i></td><td></td></tr></table>				Member Name		<i>nSize</i>		<i>nVersion</i>		<i>nPortIndex</i>		<i>xWidth</i>		<i>xHeight</i>											
Member Name																										
<i>nSize</i>																										
<i>nVersion</i>																										
<i>nPortIndex</i>																										
<i>xWidth</i>																										
<i>xHeight</i>																										
[Notes]	None.																									
[Details]	<table><tr><td colspan="2"><b><i>nSize</i></b></td></tr><tr><td><b>Write Value</b></td><td>The size of the structure in byte</td></tr><tr><td><b>Read Value</b></td><td>-</td></tr><tr><td><b>Initial Value</b></td><td>-</td></tr><tr><td><b>Notes</b></td><td>-</td></tr><tr><td colspan="2"><b><i>nVersion</i></b></td></tr><tr><td><b>Write Value</b></td><td>The version number of OpenMA</td></tr><tr><td><b>Read Value</b></td><td>-</td></tr><tr><td><b>Initial Value</b></td><td>-</td></tr><tr><td><b>Notes</b></td><td>-</td></tr><tr><td colspan="2"><b><i>nPortIndex</i></b></td></tr></table>				<b><i>nSize</i></b>		<b>Write Value</b>	The size of the structure in byte	<b>Read Value</b>	-	<b>Initial Value</b>	-	<b>Notes</b>	-	<b><i>nVersion</i></b>		<b>Write Value</b>	The version number of OpenMA	<b>Read Value</b>	-	<b>Initial Value</b>	-	<b>Notes</b>	-	<b><i>nPortIndex</i></b>	
<b><i>nSize</i></b>																										
<b>Write Value</b>	The size of the structure in byte																									
<b>Read Value</b>	-																									
<b>Initial Value</b>	-																									
<b>Notes</b>	-																									
<b><i>nVersion</i></b>																										
<b>Write Value</b>	The version number of OpenMA																									
<b>Read Value</b>	-																									
<b>Initial Value</b>	-																									
<b>Notes</b>	-																									
<b><i>nPortIndex</i></b>																										

	<b>Write Value</b>	VPB + 1		
	<b>Read Value</b>	-		
	<b>Initial Value</b>	-		
	<b>Notes</b>	-		
	<b><i>xWidth</i></b>			
	<b>Write Value</b>	-		
	<b>Read Value</b>	The scaling ratio in the horizontal direction in Q16 format.		
	<b>Initial Value</b>	0x00010000		
	<b>Notes</b>	The lesser of <i>xWidth</i> and <i>xHeight</i> is set to 0x00010000 and the other is normalized. For example, in the case where the source aspect ratio is 4:3, <i>xHeight</i> is set to 0x00010000 because the vertical side is lesser than the horizontal side. Then <i>xWidth</i> is calculated by the aspect ratio as follows:  $4 : 3 = xWidth : 0x00010000$ $3 * xWidth = 4 * 0x00010000$ $xWidth = (4 * 0x00010000) / 3$ $xWidth = 0x00015555$  Thus <i>xWidth</i> and <i>xHeight</i> are determined to 0x00015555 and 0x00010000 respectively.		
	<b><i>xHeight</i></b>			
	<b>Write Value</b>	-		
	<b>Read Value</b>	The scaling ratio in the vertical direction in Q16 format.		
	<b>Initial Value</b>	0x00010000		
	<b>Notes</b>	– See Notes of the <i>xWidth</i> member.		
	OMXR_MC_VIDEO_PARAM_REORDERTYPE			
	OMXR_MC_VIDEO_DECODERESULTTYPE		See section 6.8	
	OMX_CONFIG_RECTTYPE		See section 6.5	
	OMX_CONFIG_SCALEFACTORTYPE		See section 6.6	
	OMXR_MC_VIDEO_PARAM_DEINTERLACE_MODETYPE		See section 6.9	

Table 6-2 shows the notation for the access attribute of a structure member described in this section.

**Table 6-2 Notation for the access attribute of a structure member**

Member Name	Get	Set
Indicates the member name	<p>Indicates the access attribute of the member in the OMX_GetParameter() or OMX_GetConfig().</p> <p>“R” means IL client can get a value from the member.</p> <p>“W” means IL client must specify a value for the member.</p>	<p>Indicates the access attribute of the member in the OMX_SetParameter() or OMX_SetConfig().</p> <p>“W” means IL client must/can specify a value for the member.</p> <p>“-” means a specified value is ignored and not reflected.</p>

## 6.2. OMX\_VIDEO\_PORTDEFINITIONTYPE

### 6.2.1. OMX\_VIDEO\_PORTDEFINITIONTYPE (Input Port)

[Definition] See related document [2] 4.3.4.

[Index] OMX\_IndexParamPortDefinition

Member Name	Get	Set
<i>cMIMETYPE</i>	R	-
<i>pNativeRender</i>	R	-
<i>nFrameWidth</i>	R	W
<i>nFrameHeight</i>	R	W
<i>nStride</i>	R	-
<i>nSliceHeight</i>	R	-
<i>nBitrate</i>	R	-
<i>xFramerate</i>	R	-
<i>bFlagErrorConcealment</i>	R	-
<i>eCompressionFormat</i>	R	-
<i>eColorFormat</i>	R	-
<i>pNativeWindow</i>	R	-

[Notes] None.

[Details]

#### ***cMIMETYPE***

<b>Write Value</b>	-
<b>Read Value</b>	NULL
<b>Initial Value</b>	NULL
<b>Notes</b>	-

#### ***pNativeRender***

<b>Write Value</b>	-
<b>Read Value</b>	NULL
<b>Initial Value</b>	NULL
<b>Notes</b>	-

#### ***nFrameWidth***

- See each Media Component User's Manual.

#### ***nFrameHeight***

- See each Media Component User's Manual.

#### ***nStride***

<b>Write Value</b>	-
<b>Read Value</b>	0
<b>Initial Value</b>	0
<b>Notes</b>	-

#### ***nSliceHeight***

<b>Write Value</b>	-
<b>Read Value</b>	0
<b>Initial Value</b>	0
<b>Notes</b>	-

***nBitrate***

<b>Write Value</b>	-
<b>Read Value</b>	64000
<b>Initial Value</b>	64000
<b>Notes</b>	-

***xFramerate***

<b>Write Value</b>	-
<b>Read Value</b>	0x000F0000 : 15fps
<b>Initial Value</b>	0x000F0000
<b>Notes</b>	-

***bFlagErrorConcealment***

<b>Write Value</b>	-
<b>Read Value</b>	OMX_FALSE
<b>Initial Value</b>	OMX_FALSE
<b>Notes</b>	-

***eCompressionFormat***

- See each Media Component User's Manual.

***eColorFormat***

<b>Write Value</b>	-
<b>Read Value</b>	OMX_COLOR_FormatUnused
<b>Initial Value</b>	OMX_COLOR_FormatUnused
<b>Notes</b>	-

***pNativeWindow***

<b>Write Value</b>	-
<b>Read Value</b>	NULL
<b>Initial Value</b>	NULL
<b>Notes</b>	-

### 6.2.2. OMX\_VIDEO\_PORTDEFINITIONTYPE (Output Port)

[Definition] See related document [2] 4.3.4.

[Index] OMX\_IndexParamPortDefinition

Member Name	Get	Set
<i>cMIMEType</i>	R	-
<i>pNativeRender</i>	R	-
<i>nFrameWidth</i>	R	W
<i>nFrameHeight</i>	R	W
<i>nStride</i>	R	W
<i>nSliceHeight</i>	R	W
<i>nBitrate</i>	R	-
<i>xFramerate</i>	R	-
<i>bFlagErrorConcealment</i>	R	-
<i>eCompressionFormat</i>	R	-
<i>eColorFormat</i>	R	W
<i>pNativeWindow</i>	R	-

[Notes] None.

[Details]

#### *cMIMEType*

<b>Write Value</b>	-
<b>Read Value</b>	NULL
<b>Initial Value</b>	NULL
<b>Notes</b>	-

#### *pNativeRender*

<b>Write Value</b>	-
<b>Read Value</b>	NULL
<b>Initial Value</b>	NULL
<b>Notes</b>	-

#### *nFrameWidth*

<b>Write Value</b>	80 - 1920
<b>Read Value</b>	(Current setting)
<b>Initial Value</b>	176
<b>Notes</b>	<ul style="list-style-type: none"> <li>- An odd value is rounded down to the closest even value.</li> <li>- No effects on the decode processing.</li> <li>- This member is updated in accordance with the picture size information in the input stream. For the details, see section 7.4 and Figure 7-1.</li> </ul>

#### *nFrameHeight*

<b>Write Value</b>	80 - 1920
<b>Read Value</b>	(Current setting)
<b>Initial Value</b>	144
<b>Notes</b>	<ul style="list-style-type: none"> <li>- An odd value is rounded down to the closest even value.</li> <li>- No effects on the decode processing.</li> <li>- This member is updated in accordance with the picture size information in the input stream. For the details, see section 7.4 and Figure 7-1.</li> </ul>



### ***nStride***

<b>Write Value</b>	0 and 80 - 1920
<b>Read Value</b>	(Current setting)
<b>Initial Value</b>	0
<b>Notes</b>	<ul style="list-style-type: none"> <li>- An odd value is rounded down to the closest even value.</li> <li>- No effects on the decode processing.</li> <li>- If the <i>nStride</i> member is set to zero, the <i>nStride</i> is automatically updated in accordance with the picture size information in the input stream. In this case, buffers on the output port are allocated at the maximum size.</li> <li>- If the <i>nStride</i> member is set to zero, the <i>nSliceHeight</i> member also must be zero.</li> <li>- For the relationship between the <i>nStride</i> member and the other picture size-related information, see section 7.4.</li> <li>- If the stream requires larger buffer size than already allocated, the OMX_EventPortSettingsChanged event is generated. For the detail conditions, see section 7.5.</li> </ul>

### ***nSliceHeight***

<b>Write Value</b>	0 and 80 - 1920
<b>Read Value</b>	(Current setting)
<b>Initial Value</b>	0
<b>Notes</b>	<ul style="list-style-type: none"> <li>- An odd value is rounded down to the closest even value.</li> <li>- No effects on the decode processing.</li> <li>- If the <i>nSliceHeight</i> member is set to zero, the <i>nSliceHeight</i> is automatically updated in accordance with the picture size information in the input stream. In this case, buffers on the output port are allocated at the maximum size.</li> <li>- If the <i>nSliceHeight</i> member is set to zero, the <i>nStride</i> member also must be zero.</li> <li>- For the relationship between the <i>nSliceHeight</i> member and the other picture size-related information, see section 7.4.</li> <li>- If the stream requires larger buffer size than already allocated, the OMX_EventPortSettingsChanged event is generated. For the detail conditions, see section 7.5.</li> </ul>

### ***nBitrate***

<b>Write Value</b>	-
<b>Read Value</b>	0
<b>Initial Value</b>	0
<b>Notes</b>	-

### ***xFramerate***

<b>Write Value</b>	-
<b>Read Value</b>	0
<b>Initial Value</b>	0
<b>Notes</b>	-

### ***bFlagErrorConcealment***

<b>Write Value</b>	-
<b>Read Value</b>	OMX_FALSE
<b>Initial Value</b>	OMX_FALSE
<b>Notes</b>	-

### ***eCompressionFormat***

<b>Write Value</b>	-
<b>Read Value</b>	OMX_VIDEO_CodingUnused
<b>Initial Value</b>	OMX_VIDEO_CodingUnused
<b>Notes</b>	-

### ***eColorFormat***

<b>Write Value</b>	OMX_COLOR_FormatYUV420Planar OMX_COLOR_FormatYUV420SemiPlanar
<b>Read Value</b>	(Current setting)
<b>Initial Value</b>	OMX_COLOR_FormatYUV420Planar
<b>Notes</b>	<ul style="list-style-type: none"> <li>- Regarding the plane and byte order for each color format, see Figure 6-1 and Figure 6-2.</li> <li>- The memory arrangement of the output picture data is determined by the <i>eColorFormat</i> member and the <i>eDeinterlace</i> member of the OMXR_MC_VIDEO_PARAM_DEINTERLACE_MODETYPE structure. For the combination, see Table 6-3.</li> </ul>

### ***pNativeWindow***

<b>Write Value</b>	-
<b>Read Value</b>	NULL
<b>Initial Value</b>	NULL
<b>Notes</b>	-

**Table 6-3 Combination of Picture Format and Memory Arrangement**

<i>eColorFormat</i>	<i>eDeinterlace</i>	
	except OMXR_MC_VIDEO_ DeinterlaceNone	OMXR_MC_VIDEO_ DeinterlaceNone
OMX_COLOR_FormatYUV420Planar	Figure 6-3	Figure 6-4
OMX_COLOR_FormatYUV420SemiPlanar	Figure 6-5	Figure 6-6

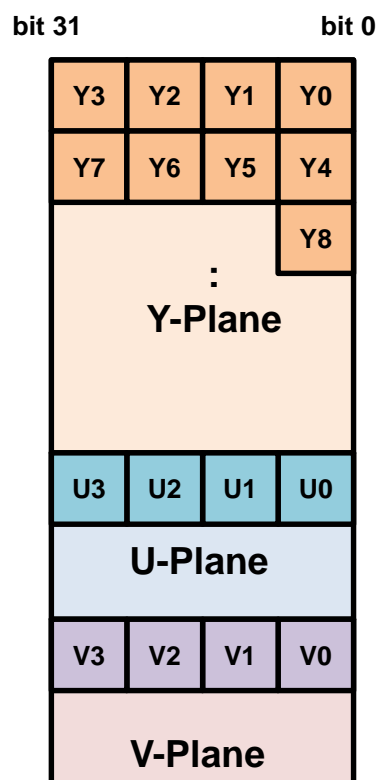


Figure 6-1 Byte and Plane Order: YUV420 Planar

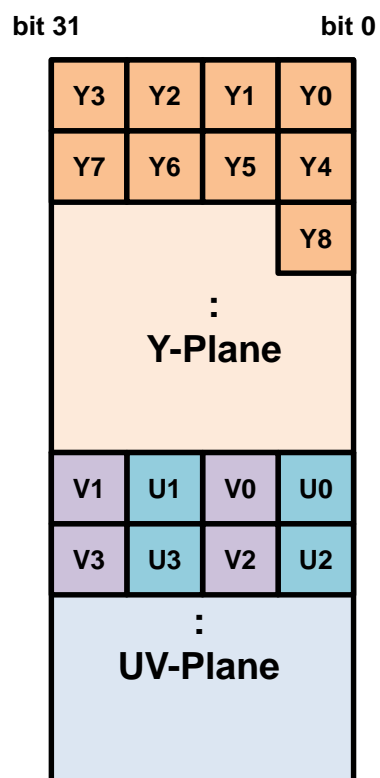
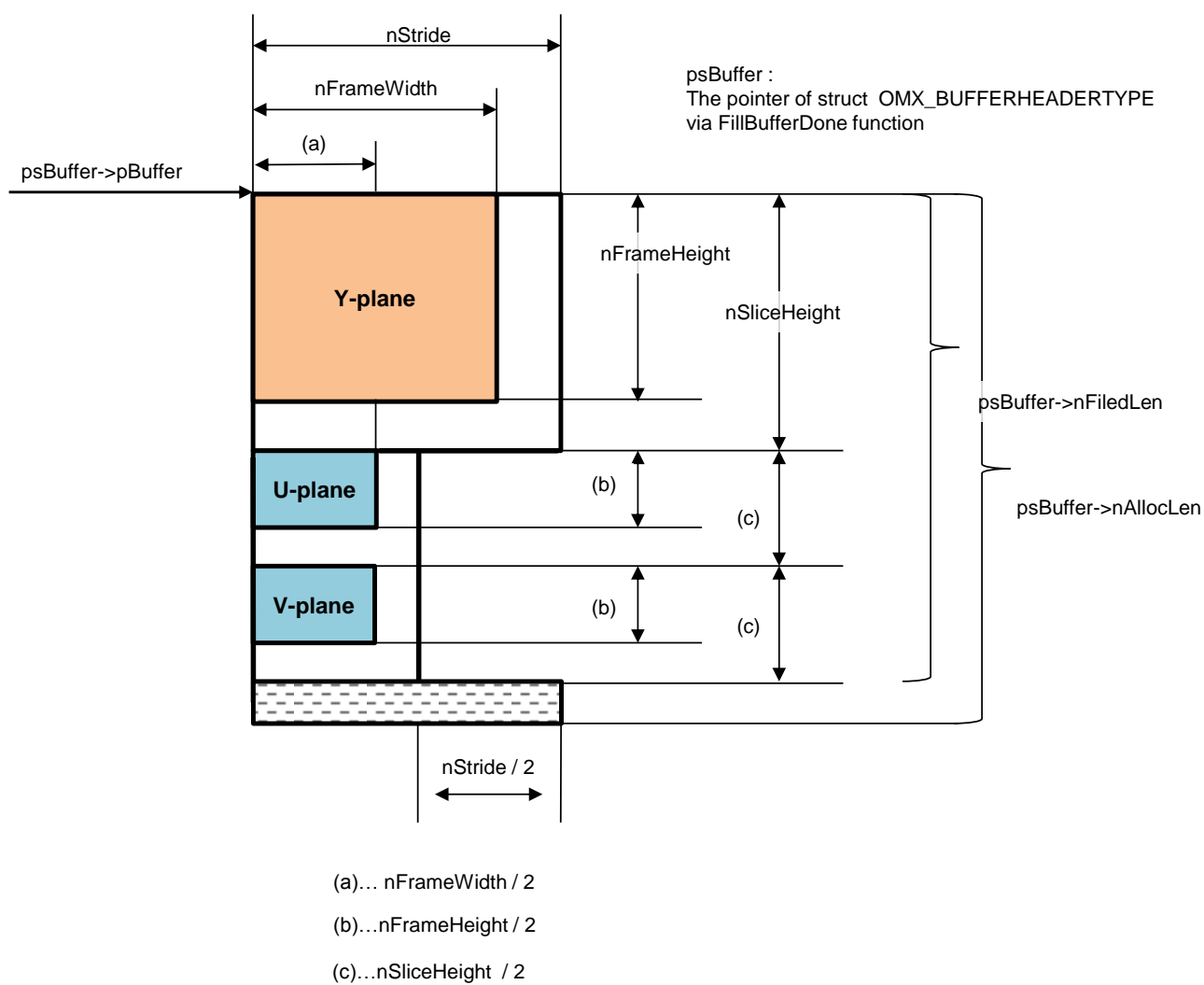
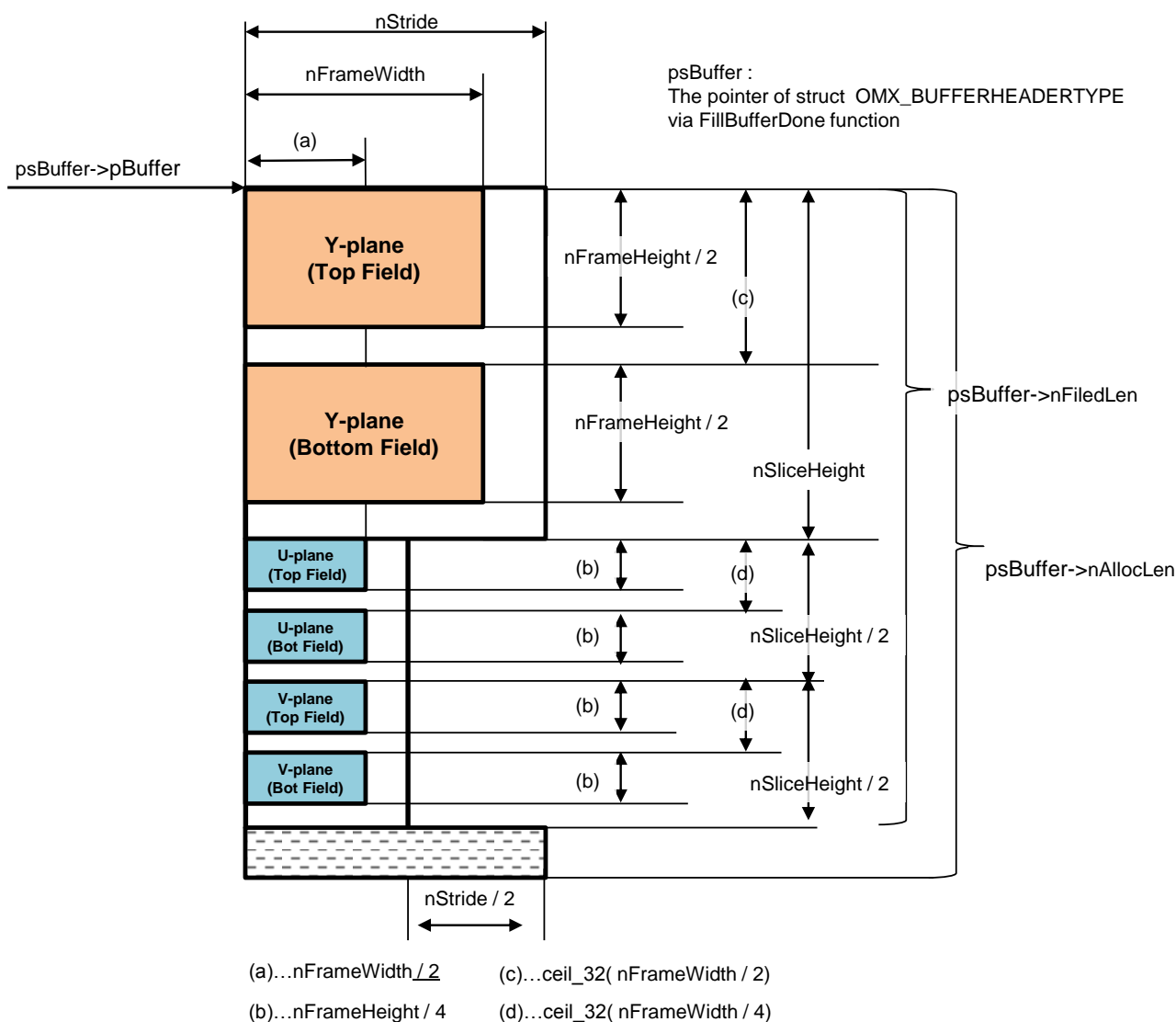


Figure 6-2 Byte and Plane Order: YUV420 Semi Planar



**Figure 6-3 Picture Format and Parameters: YUV420 Planar - Frame Arrangement**



**Figure 6-4 Picture Format and Parameters: YUV420 Planar - Field Arrangement**

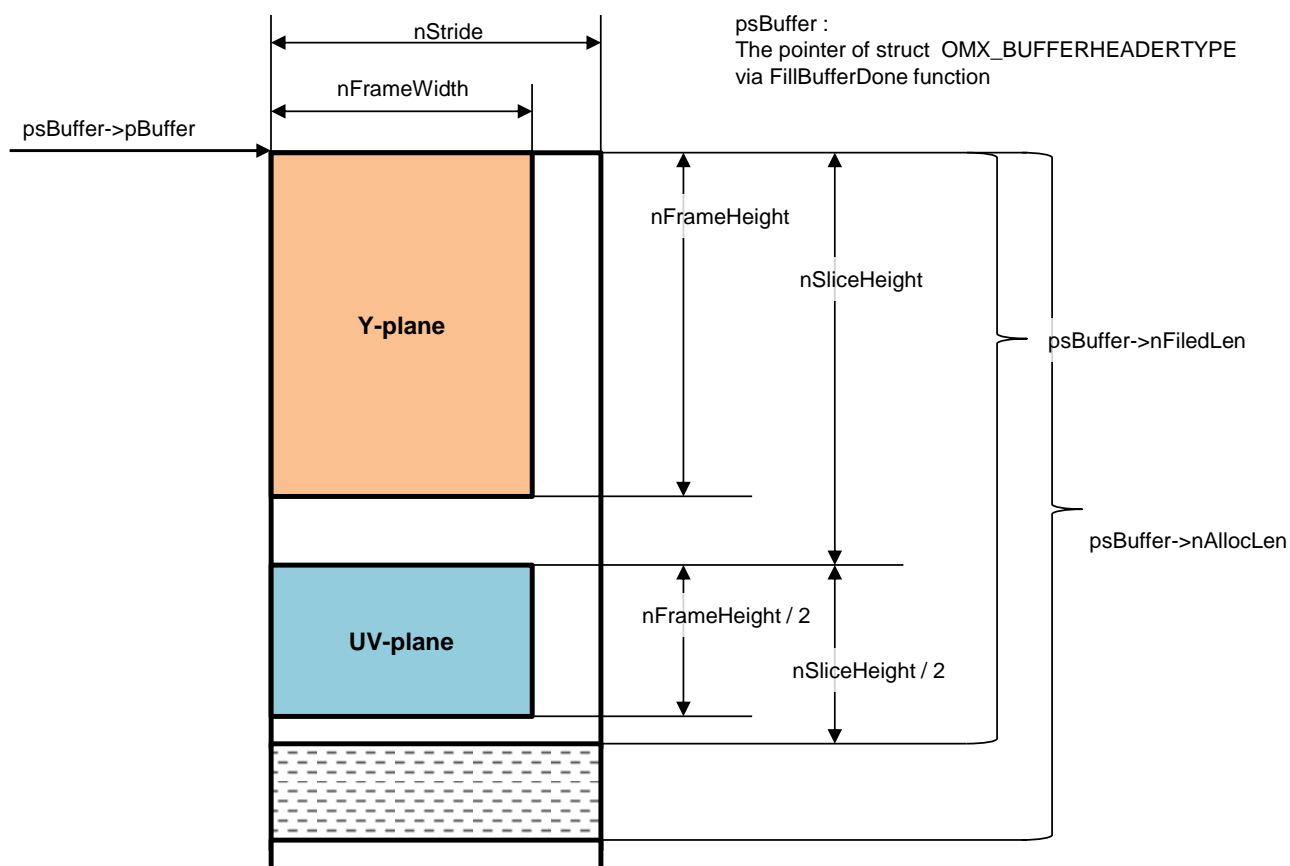


Figure 6-5 Picture Format and Parameters: YUV420 Semi Planar - Frame Arrangement

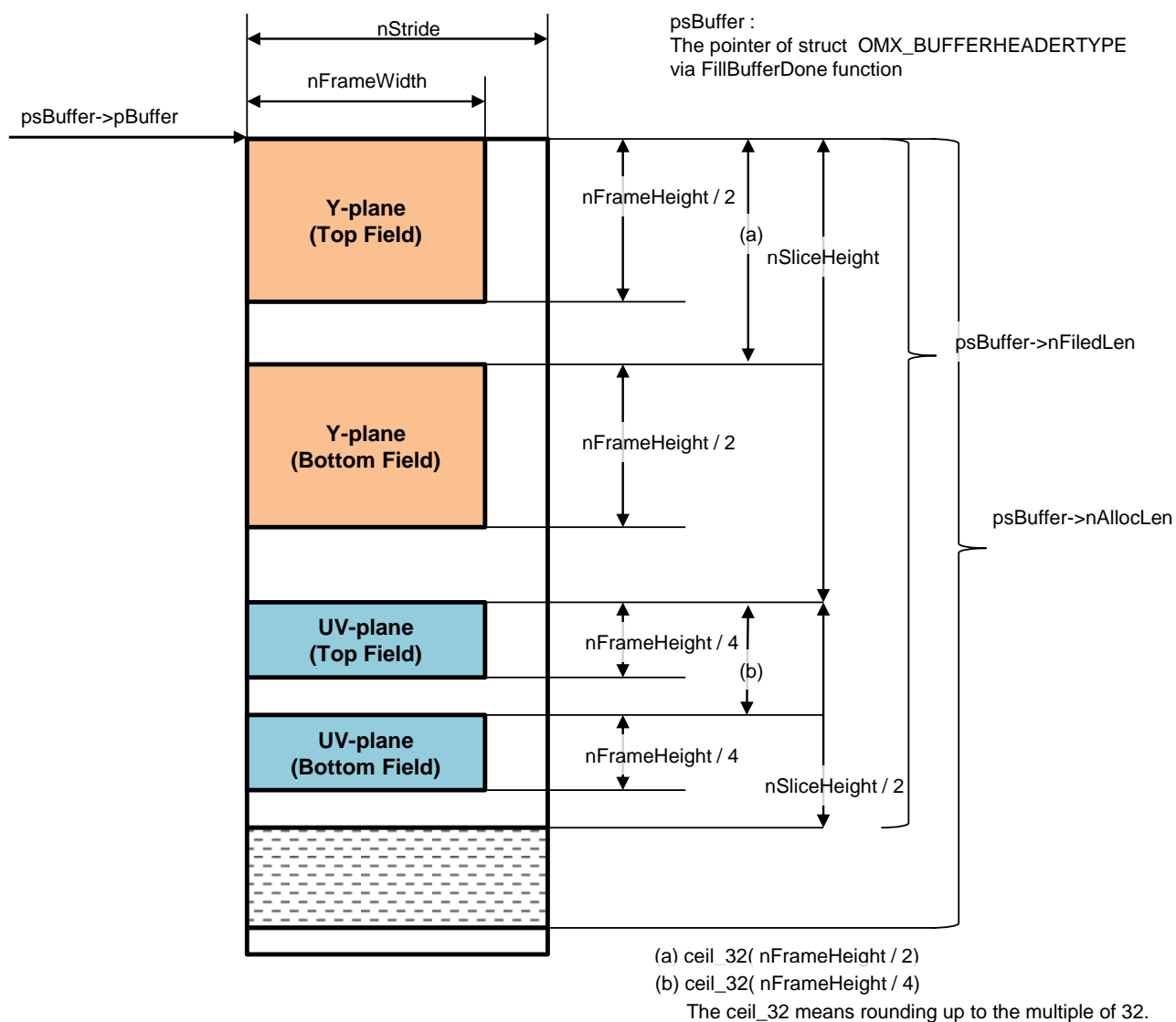


Figure 6-6 Picture Format and Parameters: YUV420 Semi Planar – Field Arrangement



## 6.3. OMX\_VIDEO\_PARAM\_PORTFORMATTYPE

### 6.3.1. OMX\_VIDEO\_PARAM\_PORTFORMATTYPE (Input Port)

[Definition] See related document [2] 4.3.5.

[Index] OMX\_IndexParamVideoPortFormat

Member Name	Get	Set
<i>nSize</i>	W	W
<i>nVersion</i>	W	W
<i>nPortIndex</i>	W	W
<i>nIndex</i>	W	W
<i>eCompressionFormat</i>	R	-
<i>eColorFormat</i>	R	-
<i>xFramerate</i>	R	-

[Notes] None.

[Details]

#### *nSize*

<b>Write Value</b>	The size of the structure in bytes.
<b>Read Value</b>	-
<b>Initial Value</b>	-
<b>Notes</b>	-

#### *nVersion*

<b>Write Value</b>	The version number of OpenMAX IL specifications 1.1.2
<b>Read Value</b>	-
<b>Initial Value</b>	-
<b>Notes</b>	-

#### *nPortIndex*

<b>Write Value</b>	VPB + 0
<b>Read Value</b>	-
<b>Initial Value</b>	-
<b>Notes</b>	-

#### *nIndex*

<b>Write Value</b>	0
<b>Read Value</b>	-
<b>Initial Value</b>	-
<b>Notes</b>	-

#### *eCompressionFormat*

– See each Media Component User's Manual.

#### *eColorFormat*

<b>Write Value</b>	-
<b>Read Value</b>	OMX_COLOR_FormatUnused
<b>Initial Value</b>	OMX_COLOR_FormatUnused
<b>Notes</b>	-

---

***xFramerate***

<b>Write Value</b>	-
<b>Read Value</b>	0x000F0000 : 15fps
<b>Initial Value</b>	0x000F0000 : 15fps
<b>Notes</b>	-

### 6.3.2. OMX\_VIDEO\_PARAM\_PORTFORMATTYPE (Output Port)

[Definition] See related document [2] 4.3.5.

[Index] OMX\_IndexParamVideoPortFormat

Member Name	Get	Set
<i>nSize</i>	W	W
<i>nVersion</i>	W	W
<i>nPortIndex</i>	W	W
<i>nIndex</i>	W	W
<i>eCompressionFormat</i>	R	-
<i>eColorFormat</i>	R	W
<i>xFramerate</i>	R	-

[Notes] None.

[Details]

#### *nSize*

<b>Write Value</b>	The size of the structure in bytes.
<b>Read Value</b>	-
<b>Initial Value</b>	-
<b>Notes</b>	-

#### *nVersion*

<b>Write Value</b>	The version number of OpenMAX IL specifications 1.1.2
<b>Read Value</b>	-
<b>Initial Value</b>	-
<b>Notes</b>	-

#### *nPortIndex*

<b>Write Value</b>	VPB + 1
<b>Read Value</b>	-
<b>Initial Value</b>	-
<b>Notes</b>	-

#### *nIndex*

<b>Write Value</b>	0, 1
<b>Read Value</b>	-
<b>Initial Value</b>	-
<b>Notes</b>	-

#### *eCompressionFormat*

<b>Write Value</b>	-
<b>Read Value</b>	OMX_VIDEO_CodingUnused
<b>Initial Value</b>	OMX_VIDEO_CodingUnused
<b>Notes</b>	-

### ***eColorFormat***

<b>Write Value</b>	OMX_COLOR_FormatYUV420Planar OMX_COLOR_FormatYUV420SemiPlanar
<b>Read Value</b>	OMX_COLOR_FormatYUV420Planar ( <i>nIndex</i> = 0) OMX_COLOR_FormatYUV420SemiPlanar ( <i>nIndex</i> = 1)
<b>Initial Value</b>	OMX_COLOR_FormatYUV420Planar
<b>Notes</b>	– On OMX_SetParameter, the value of the <i>eColorFormat</i> member is reflected in the <i>eColorFormat</i> member of the OMX_VIDEO_PORTDEFINITIONTYPE structure.

### ***xFramerate***

<b>Write Value</b>	-
<b>Read Value</b>	0
<b>Initial Value</b>	0
<b>Notes</b>	-

#### 6.4. OMX\_VIDEO\_PARAM\_PROFILELEVELTYPE

[Definition] See related document [2] 4.3.25.

[Index] OMX\_IndexParamVideoProfileLevelQuerySupported  
OMX\_IndexParamVideoProfileLevelCurrent

Member Name	Get	Set
<i>nSize</i>	W	W
<i>nVersion</i>	W	W
<i>nPortIndex</i>	W	W
<i>eProfile</i>	R	-
<i>eLevel</i>	R	-
<i>nProfileIndex</i>	W	-

[Notes] None.

[Details]

##### ***nSize***

<b>Write Value</b>	The size of the structure in bytes.
<b>Read Value</b>	-
<b>Initial Value</b>	-
<b>Notes</b>	-

##### ***nVersion***

<b>Write Value</b>	The version number of OpenMAX IL specifications 1.1.2
<b>Read Value</b>	-
<b>Initial Value</b>	-
<b>Notes</b>	-

##### ***nPortIndex***

<b>Write Value</b>	VPB + 0
<b>Read Value</b>	-
<b>Initial Value</b>	-
<b>Notes</b>	-

##### ***eProfile***

- See each Media Component User's Manual.

##### ***eLevel***

- See each Media Component User's Manual.

##### ***nProfileIndex***

- See each Media Component User's Manual.

## 6.5. OMX\_CONFIG\_RECTTYPE

[Definition] See related document [2] 4.2.18.

[Index] OMX\_IndexConfigCommonOutputCrop

[Member]	Member Name	Get
	<i>nSize</i>	W
	<i>nVersion</i>	W
	<i>nPortIndex</i>	W
	<i>nLeft</i>	R
	<i>nTop</i>	R
	<i>nWidth</i>	R
	<i>nHeight</i>	R

[Notes] None.

[Details]

### *nSize*

<b>Write Value</b>	The size of the structure in bytes.
<b>Read Value</b>	-
<b>Initial Value</b>	-
<b>Notes</b>	-

### *nVersion*

<b>Write Value</b>	The version number of OpenMAX IL specifications 1.1.2
<b>Read Value</b>	-
<b>Initial Value</b>	-
<b>Notes</b>	-

### *nPortIndex*

<b>Write Value</b>	VPB + 1
<b>Read Value</b>	-
<b>Initial Value</b>	-
<b>Notes</b>	-

### *nLeft*

<b>Write Value</b>	-
<b>Read Value</b>	The leftmost coordinate of the active video
<b>Initial Value</b>	0
<b>Notes</b>	<ul style="list-style-type: none"> <li>- This member is updated based on the input stream.</li> <li>- For the relationship between this member and the other size-related information, see Figure 7-1.</li> </ul>

### ***nTop***

<b>Write Value</b>	-
<b>Read Value</b>	The topmost coordinate of the active video
<b>Initial Value</b>	0
<b>Notes</b>	<ul style="list-style-type: none"> <li>- This member is updated based on the input stream.</li> <li>- For the relationship between this member and the other size-related information, see Figure 7-1.</li> </ul>

### ***nWidth***

<b>Write Value</b>	-
<b>Read Value</b>	The width of the active video
<b>Initial Value</b>	176
<b>Notes</b>	<ul style="list-style-type: none"> <li>- This member is updated in accordance with the picture size information of the input stream.</li> <li>- This member follows the <i>nFrameWidth</i> member of the OMX_VIDEO_PORTDEFINITIONTYPE structure. For the relationship between this member and the other size-related information, see Figure 7-1.</li> </ul>

### ***nHeight***

<b>Write Value</b>	-
<b>Read Value</b>	The height of the active video
<b>Initial Value</b>	144
<b>Notes</b>	<ul style="list-style-type: none"> <li>- This member is updated in accordance with the picture size information of the input stream.</li> <li>- This member follows the <i>nFrameHeight</i> member of the OMX_VIDEO_PORTDEFINITIONTYPE structure.</li> <li>- For the relationship between this member and the other size-related information, see Figure 7-1.</li> </ul>

## 6.6. OMX\_CONFIG\_SCALEFACTORTYPE

[Definition] See related document [2] 4.2.9.

[Index] OMX\_IndexConfigCommonScale

Member Name	Get
<i>nSize</i>	W
<i>nVersion</i>	W
<i>nPortIndex</i>	W
<i>xWidth</i>	R
<i>xHeight</i>	R

[Notes] None.

[Details]

### *nSize*

<b>Write Value</b>	The size of the structure in bytes.
<b>Read Value</b>	-
<b>Initial Value</b>	-
<b>Notes</b>	-

### *nVersion*

<b>Write Value</b>	The version number of OpenMAX IL specifications 1.1.2
<b>Read Value</b>	-
<b>Initial Value</b>	-
<b>Notes</b>	-

### *nPortIndex*

<b>Write Value</b>	VPB + 1
<b>Read Value</b>	-
<b>Initial Value</b>	-
<b>Notes</b>	-

### *xWidth*

<b>Write Value</b>	-
<b>Read Value</b>	The scaling ratio in the horizontal direction in Q16 format.
<b>Initial Value</b>	0x00010000
<b>Notes</b>	<p>The lesser of <i>xWidth</i> and <i>xHeight</i> is set to 0x00010000 and the other is normalized. For example, in the case where the source aspect ratio is 4:3, <i>xHeight</i> is set to 0x00010000 because the vertical side is lesser than the horizontal side. Then <i>xWidth</i> is calculated by the aspect ratio as follows:</p> $4 : 3 = xWidth : 0x00010000$ $3 * xWidth = 4 * 0x00010000$ $xWidth = (4 * 0x00010000) / 3$ $xWidth = 0x00015555$ <p>Thus <i>xWidth</i> and <i>xHeight</i> are determined to 0x00015555 and 0x00010000 respectively.</p>

### *xHeight*

<b>Write Value</b>	-
<b>Read Value</b>	The scaling ratio in the vertical direction in Q16 format.
<b>Initial Value</b>	0x00010000
<b>Notes</b>	- See Notes of the <i>xWidth</i> member.





## 6.7. OMXR\_MC\_VIDEO\_PARAM\_REORDERTYPE

[Definition]     typedef struct tagOMXR\_MC\_VIDEO\_PARAM\_REORDERTYPE {  
                       OMX\_U32                        nSize;  
                       OMX\_VERSIONTYPE            nVersion;  
                       OMX\_U32                        nPortIndex;  
                       OMX\_BOOL                      bReorder;  
                   } OMXR\_MC\_VIDEO\_PARAM\_REORDERTYPE;

[Index]            OMXR\_MC\_IndexParamVideoReorder

Member Name	Get	Set
nSize	W	W
nVersion	W	W
nPortIndex	W	W
bReorder	R	W

[Notes]            None.

[Details]

### nSize

<b>Write Value</b>	The size of the structure in bytes.
<b>Read Value</b>	-
<b>Initial Value</b>	-
<b>Notes</b>	-

### nVersion

<b>Write Value</b>	The version number of OpenMAX IL specifications 1.1.2
<b>Read Value</b>	-
<b>Initial Value</b>	-
<b>Notes</b>	-

### nPortIndex

<b>Write Value</b>	VPB + 1
<b>Read Value</b>	-
<b>Initial Value</b>	-
<b>Notes</b>	-

### bReorder

<b>Write Value</b>	OMX_TRUE : Output picture data in display order OMX_FALSE : Output picture data in decoding order
<b>Read Value</b>	(current setting)
<b>Initial Value</b>	OMX_TRUE
<b>Notes</b>	When a port of the component is disabled by OMX_CommandPortDisable command, the setting value is not reflected after OMX_CommandPortEnable command.

## 6.8. OMXR\_MC\_VIDEO\_DECODERESULTTYPE

[Definition]    typedef struct tagOMXR\_MC\_VIDEO\_DECODERESULTTYPE {  
                       OMX\_U32                nSize;  
                       OMX\_PTR                pvPhysImageAddressY;  
                       OMX\_U32                u32PictWidth;  
                       OMX\_U32                u32PictHeight;  
                       OMX\_U32                u32DecodeError;  
                       OMX\_U32                u32PhyAddr;  
                       OMX\_U32                u32Stride;  
                       OMX\_U32                u32Area;  
                       OMX\_PTR                pvReserved;  
                   } OMXR\_MC\_VIDEO\_DECODERESULTTYPE;

[Index]        N/A

[Notes]        A pointer to this structure is set to the *pOutputPortPrivate* member of the OMX\_BUFFERHEADERTYPE structure that is returned via the FillBufferDone callback. When the *nFilledLen* member of the OMX\_BUFFERHEADERTYPE structure is equal to 0x0 at FillBufferDone callback, the *pOutputPortPrivate* is set to NULL.

[Details]

### ***nSize***

<b>Write Value</b>	-
<b>Read Value</b>	The size of the structure in bytes.
<b>Initial Value</b>	-
<b>Notes</b>	-

### ***pvPhysImageAddressY***

<b>Write Value</b>	-
<b>Read Value</b>	The top address of the YUV picture data
<b>Initial Value</b>	-
<b>Notes</b>	<ul style="list-style-type: none"> <li>- For the memory arrangement of the YUV picture data, see Table 6-3.</li> <li>- When the buffer has no output data, this member is set to NULL.</li> <li>- The address is 'hardware address' that is accessible from hardware IPs. For the detail of 'hardware address' see related document [3].</li> </ul>

### ***u32PictWidth***

- See each Media Component User's Manual.

### ***u32PictHeight***

- See each Media Component User's Manual.

#### ***u32DecodeError***

<b>Write Value</b>	-
<b>Read Value</b>	The decode error flags. The notes of the flags are as follows: <ul style="list-style-type: none"> <li>OMXR_MC_VIDEO_DECODE_ERROR_FLAG_CAUTION : Detects minor errors that have no effects on output picture data.</li> <li>OMXR_MC_VIDEO_DECODE_ERROR_FLAG_CONCEAL : Detects major errors that need error concealment.</li> </ul>
<b>Initial Value</b>	-
<b>Notes</b>	When the OMXR_MC_VIDEO_DECODE_ERROR_FLAG_CONCEAL flag is set, the decoded picture data may be corrupted.

#### ***u32PhyAddr***

<b>Write Value</b>	-
<b>Read Value</b>	0
<b>Initial Value</b>	0
<b>Notes</b>	This member is not used.

#### ***u32Stride***

<b>Write Value</b>	-
<b>Read Value</b>	0
<b>Initial Value</b>	0
<b>Notes</b>	This member is not used.

#### ***u32Area***

<b>Write Value</b>	-
<b>Read Value</b>	0
<b>Initial Value</b>	0
<b>Notes</b>	This member is not used.

#### ***pvReserved***

<b>Write Value</b>	-
<b>Read Value</b>	- ( Reserved )
<b>Initial Value</b>	-
<b>Notes</b>	This member is not used.

## 6.9. OMXR\_MC\_VIDEO\_PARAM\_DEINTERLACE\_MODETYPE

[Definition] typedef struct tagOMXR\_MC\_VIDEO\_PARAM\_DEINTERLACE\_MODETYPE {  
                     OMX\_U32 *nSize*;  
                     OMX\_VERSIONTYPE *nVersion*;  
                     OMX\_U32 *nPortIndex*;  
                     OMXR\_MC\_VIDEO\_DEINTERLACE\_MODETYPE *eDeinterlace*;  
 } OMXR\_MC\_VIDEO\_PARAM\_DEINTERLACE\_MODETYPE;

[Index] OMXR\_MC\_IndexParamVideoDeinterlaceMode

Member Name	Get	Set
<i>nSize</i>	W	W
<i>nVersion</i>	W	W
<i>nPortIndex</i>	W	W
<i>eDeinterlace</i>	R	W

[Notes] None.

[Details]

### *nSize*

<b>Write Value</b>	The size of the structure in bytes.
<b>Read Value</b>	-
<b>Initial Value</b>	-
<b>Notes</b>	-

### *nVersion*

<b>Write Value</b>	The version number of OpenMAX IL specifications 1.1.2
<b>Read Value</b>	-
<b>Initial Value</b>	-
<b>Notes</b>	-

### *nPortIndex*

<b>Write Value</b>	VPB + 1
<b>Read Value</b>	-
<b>Initial Value</b>	-
<b>Notes</b>	-

**eDeinterlace**

<b>Write Value</b>	OMXR_MC_VIDEO_DeinterlaceNone : Not applying IP(Interlace-Progressive) Conversion OMXR_MC_VIDEO_Deinterlace2DHalf: 2D IP Conversion with half rate output OMXR_MC_VIDEO_Deinterlace2DFull : 2D IP Conversion with full rate output OMXR_MC_VIDEO_Deinterlace3DHalf : 3D IP Conversion with half rate output OMXR_MC_VIDEO_Deinterlace3DFull : 3D IP Conversion with full rate output
<b>Read Value</b>	(Current setting)
<b>Initial Value</b>	OMXR_MC_VIDEO_Deinterlace3DHalf
<b>Notes</b>	<p>eDeinterlace provides two methods of deinterlacing.</p> <ul style="list-style-type: none"> <li>•2D IP Conversion: conversion from one field to a progressive image. This method reduces combing noise but makes line flickers and blurred image.</li> <li>•3D IP Conversion: conversion from multiple fields to a progressive image. This method reduces line flickers and makes clear image when decoding static image.</li> </ul> <p>Also, eDeinterlace affects on the numbers of output pictures. In case of half rate output, there is one-to-one correspondence between input and output frames. On the other hand, in case of full rate output, there is one-to-one correspondence between input and output fields. As the component' acceptable format is unit of frame, nTimestamp of OMX_BUFFERHEADERTYPE is interpolated when it is full rate.</p> <p>In 3D IP conversion mode, there are some exceptions for applying 3D IP conversion to the image.</p> <ul style="list-style-type: none"> <li>- the first image in the sequence</li> <li>- the image after flush operation selected.</li> <li>- the image after EOS input (OMXR_MC_VIDEO_Deinterlace3DFull only)</li> </ul> <p>In this case 2D IP Conversion is applied in stead of 3D IP Conversion.</p> <p>When input stream is progressive, this member is ignored.</p>

## 6.10. Specific Usage on Common Structure Members

Table 6-4 describes Video Decoder Media Component specific usage of the structures that are described in related document [1].

**Table 6-4 Specific Usage on Common Structure Members**

Structure	Member	Description
OMX_BUFFERHEADERTYPE	<i>nTickCount</i>	The <i>nTickCount</i> member of the input buffer can be set to any value. Video Decoder Media Component propagates the value to its associated output buffer. When a frame data is stored into multiple input buffers, the value which is set to the first input buffer is propagated to its associated output buffer.
	<i>nTimeStamp</i>	The <i>nTimeStamp</i> member of the input buffer can be set to any value. Video Decoder Media Component propagates the value to its associated output buffer. When a frame data is stored into multiple input buffers, the value which is set to the first input buffer is propagated to its associated output buffer.
	<i>pOutputPortPrivate</i>	This member must not be set by the IL client. Video Decoder Media Component sends the OMXR_MC_VIDEO_DECODERESULTTYPE structure with this member via FillBufferDone callback.
	<i>nFlags</i>	For the details, see section 6.10.1.
	<i>nOffset</i>	The <i>nOffset</i> member of the input buffer can be set to any value.
OMX_PARAM_PORTDEFINITIONTYPE	<i>nBufferCountActual</i>	The <i>nBufferCountActual</i> member can be set in the range <i>nBufferCountMin</i> to 8. The default value is the same as the <i>nBufferCountMin</i> member of each port.
	<i>nBufferCountMin</i>	[input port] 2 (read only)  [output port] 3 (read only)
	<i>nBufferSize</i>	[input port] <i>nBufferSize</i> is fixed to 1,572,864.  [output port] <i>nBufferSize</i> is calculated by using the <i>nStride</i> member and the <i>nSliceHeight</i> member of the OMX_VIDEO_PORTDEFINITIONTYPE structure as follows: ➤ $nBufferSize = (nStride \times nSliceHeight \times 3) / 2$  When both <i>nStride</i> and <i>nSliceHeight</i> are set to zero, <i>nBufferSize</i> is set to 3,133,440 (maximum).
	<i>format</i>	See section 6.2.1 and 6.2.2.

### 6.10.1. Buffer Flags (*nFlags*)

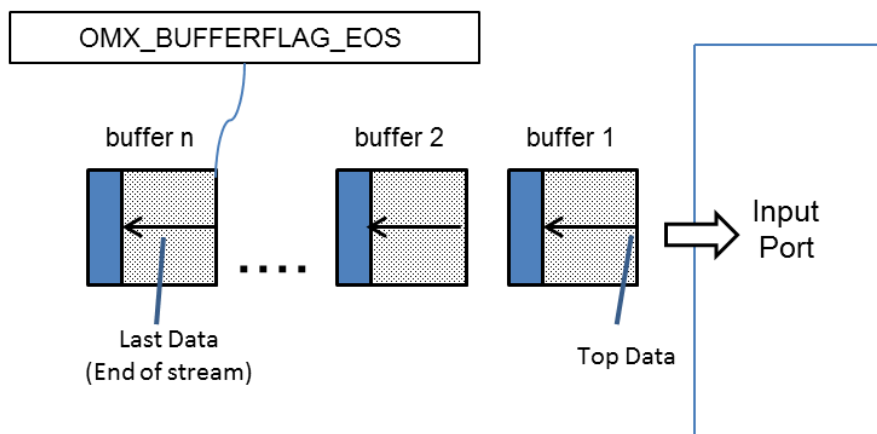
Table 6-5 shows the usage of the *nFlags* member of the OMX\_BUFFERHEADERTYPE structure. For the basic definition of these flags, see related document [2].

**Table 6-5 Specific Usage on Buffer Flags**

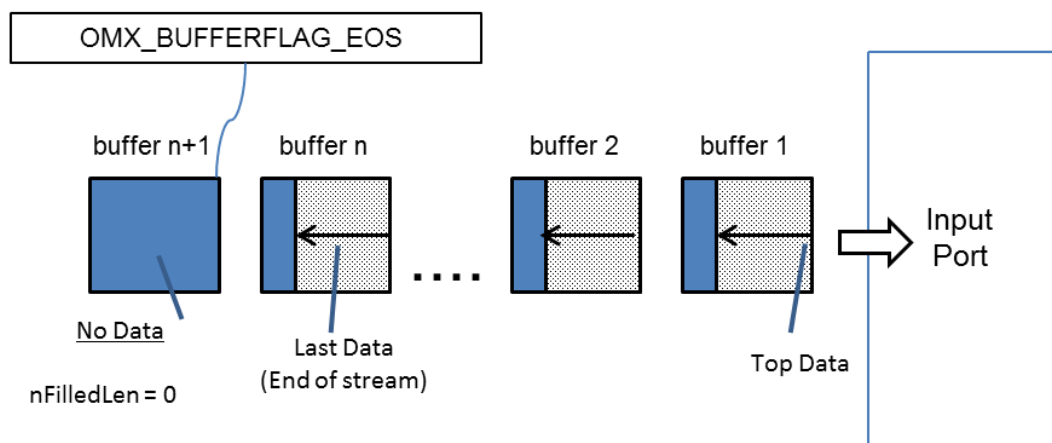
flag	Description
OMX_BUFFERFLAG_EOS	<p>[Input Buffer] The IL client can select the two patterns of EOS buffer sequence as illustrated in Figure 6-7.</p> <p>[Output Buffer] The Video Decoder Media Component returns the buffer with this flag after the buffer contains the last output picture data as illustrated in Figure 6-8.</p>
OMX_BUFFERFLAG_STARTTIME	Video Decoder Media Component propagates the flag of an input buffer to the output buffer without any operations.
OMX_BUFFERFLAG_DECODEONLY	
OMX_BUFFERFLAG_DATACORRUPT	
OMX_BUFFERFLAG_ENDOFFRAME	The IL client can store one input picture data into multiple buffers. The last buffer for the input picture data must be sent with this flag. For the buffer sequence, see Figure 6-9. For the details of this flags, see each Media Component User's Manual.
OMX_BUFFERFLAG_SYNCFRAME	Video Decoder Media Component propagates the flag of an input buffer to the output buffer without any operations.
OMX_BUFFERFLAG_EXTRADATA	
OMX_BUFFERFLAG_CODECCONFIG	See each Media Component User's Manual.



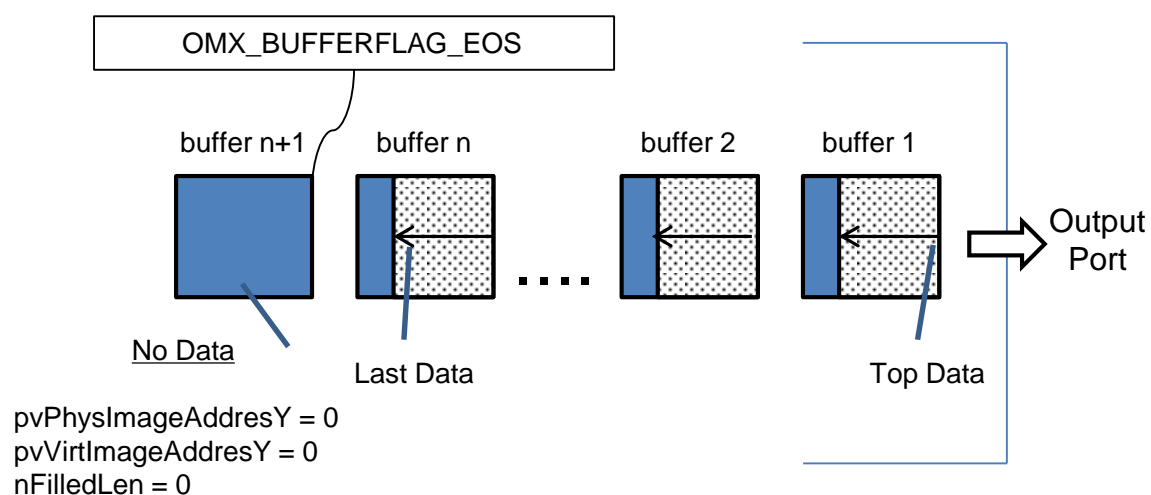
DATA+EOS case: Set EOS flag to the last data buffer.



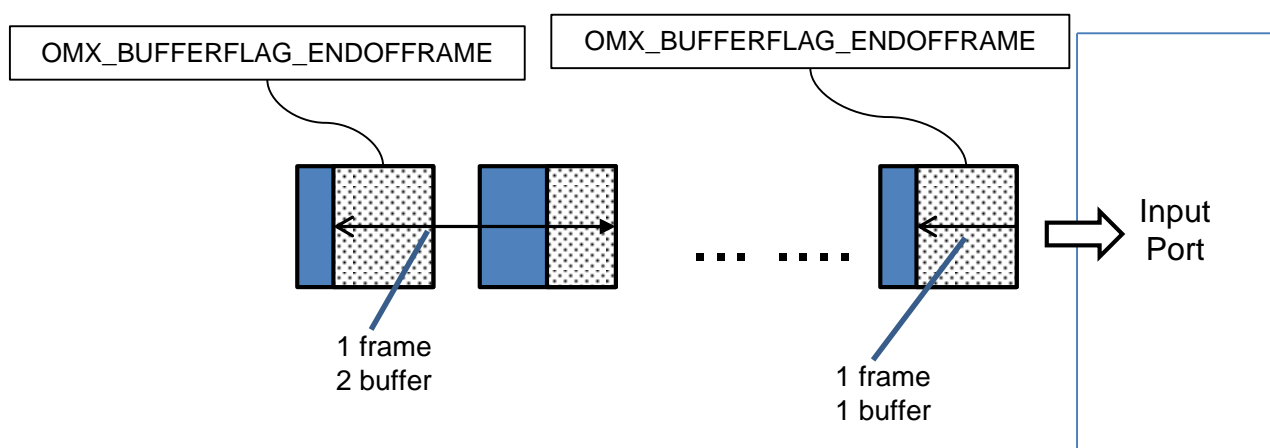
0+EOS case: Set EOS flag to the next buffer(with no data) after last data buffer.



**Figure 6-7 Input Buffer Sequence: OMX\_BUFFERFLAG\_EOS Flag Usage**



**Figure 6-8 Output Buffer Sequence: OMX\_BUFFERFLAG\_EOS Flag Usage**



**Figure 6-9 Input Buffer Sequence: OMX\_BUFFERFLAG\_ENDOFFRAME Flag Usage**

## 7. Appendix

### 7.1. Errors and Error Handling

Table 7-1 shows the description for the flow control and stream errors and how to handle the errors.

**Table 7-1 Errors and Error Handling**

Error Code	Description
OMX_ErrorStreamCorrupt	<p>[Reason] This error is reported via event callback when the Video Decoder Media Component cannot handle the input stream by the following reasons:</p> <ul style="list-style-type: none"> <li>– The picture size of the input stream is out of support.</li> <li>– The input stream contains unsupported profile or tools.</li> <li>– The input stream requires the large memory size that Video Decoder Media Component cannot allocate.</li> </ul> <p>This error is reported via event callback.</p> <p>[Error Handling] To resume the decode operation it requires the state transition to OMX_StateIdle state.</p>
OMX_ErrorOverflow	<p>[Reason] The Video Decoder Media Component receives a buffer that is already received from the IL client via OMX_EmptyThisBuffer or OMX_FillThisBuffer.</p> <p>[Error Handling] Although it is possible to continue the decode operation, make sure that the buffer management and the API sequence are correct in the IL client side.</p>
OMX_ErrorUnderflow	<p>[Reason] Video Decoder Media Component does not return this error code.</p> <p>[Error Handling] None.</p>

## 7.2. Attentions on Decoding for Corrupted Streams

For a corrupted input stream, Video Decoder Media Component continues to decode with applying error concealments as possible. If the corrupted data can be concealed, the Video Decoder Media Component returns the buffer that contains corrupted picture via FillBufferDone callback. The IL client can check whether the error concealment was applied for the output picture data by referring the *u32DecodeError* member of the OMXR\_MC\_VIDEO\_DECODERESULTTYPE structure. If the input data is severely corrupted and cannot be concealed, Video Decoder Media Component discards the input data and does not send the corresponding FillBufferDone callback. Therefore, the IL client should implement the timeout operation against continuous output missing by stream corruption.

### **7.3. Attentions on Port Flush Operation**

Regarding the port flush operation of Video Decoder Media Component, the following items should be noticed.

- When the IL client flushes the port of the Video Decoder Media Component, all the ports should be flushed to continue the decode processing.
- After the port flush operation, the IL client should input the stream from a key-frame data.

#### 7.4. Parameter Relation between PortDefinition and Crop

Figure 7-1 illustrates the relation of the picture size information that is accessible via the OMX\_IndexParamPortDefinition index and the OMX\_IndexConfigCommonOutputCrop index.

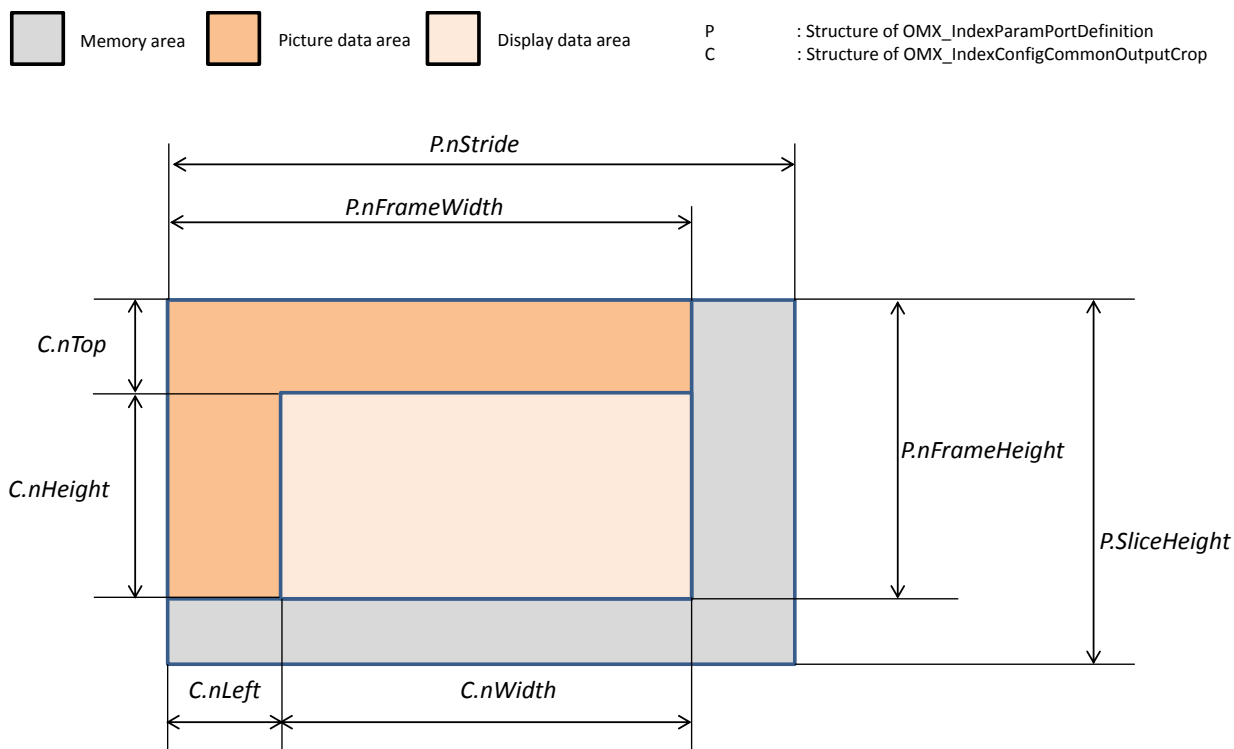


Figure 7-1 Parameter Relation between PortDefinition and Crop

---

## 7.5. OMX\_EventPortSettingsChanged Event Conditions

When the following conditions are satisfied, Video Decoder Media Component notifies the OMX\_EventPortSettingsChanged event to the IL client.

[Definition]

*P.nStride* The *nStride* member of the OMX\_VIDEO\_PORTDEFINITIONTYPE structure

*P.nSliceHeight* The *nSliceHeight* member of the OMX\_VIDEO\_PORTDEFINITIONTYPE structure

*S.nStride* The *nStride* value calculated with the stream header information

*S.nSliceHeight* The *nSliceHeight* value calculated with the stream header information

[Conditions]

$P.nStride < S.nStride$  or  $P.nSliceHeight < S.nSliceHeight$

The OMX\_EventPortSettingsChanged event is generated at the first stream header decoding after the transition to OMX\_ExecutingState. When the conditions are satisfied after the stream header decoding, Video Decoder Media Component stops the operation and issues the OMX\_EventError event instead of the OMX\_EventPortSettingsChanged event. Thus, Video Decoder Media Component may fail to decode multi-sequence streams that have changes of the picture size in the middle of the streams.

## 7.6. The output delay and the means of reduction

For the Video Decoder Media Component, the output delay is defined as the time from the OMX\_EmptyThisBuffer function call for the first data to the FillBufferDone event for the first output frame.

Figure 7-2 illustrates the definition of the output delay using OMX Media Component APIs.

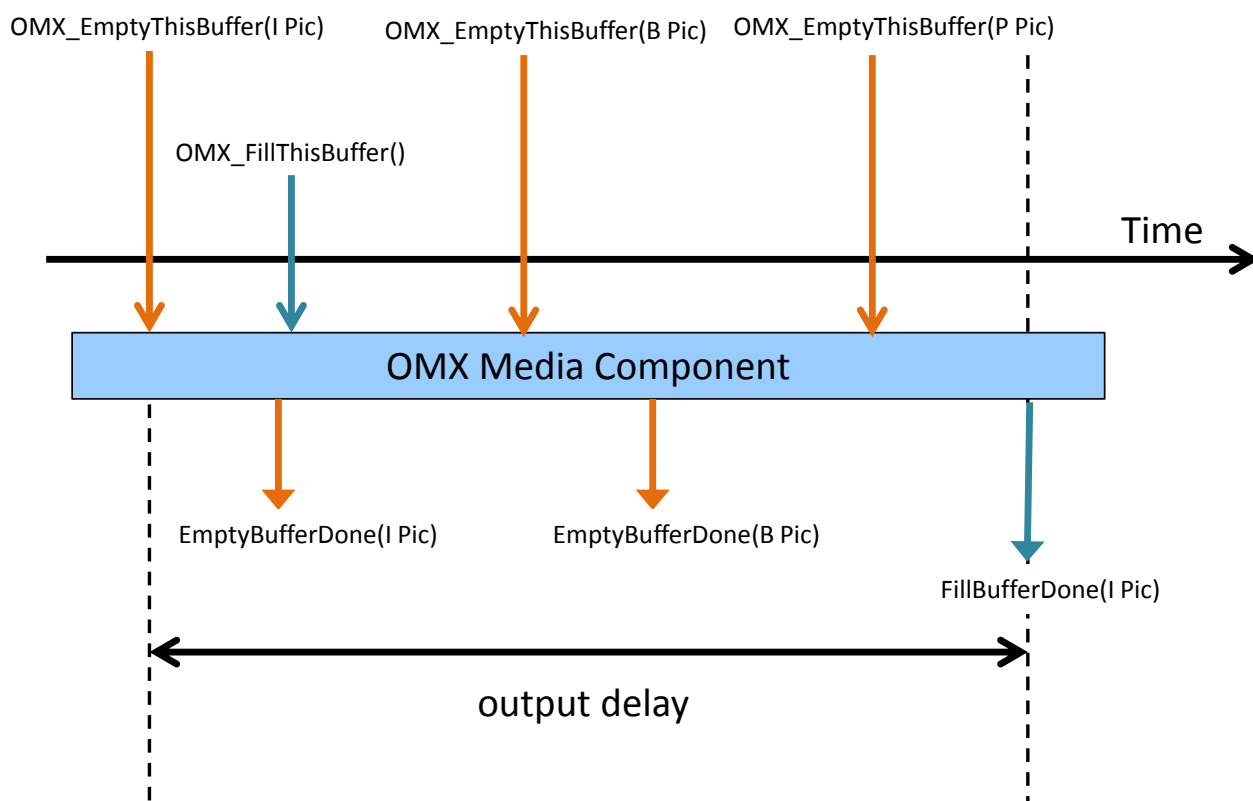


Figure 7-2 Definition of the output delay



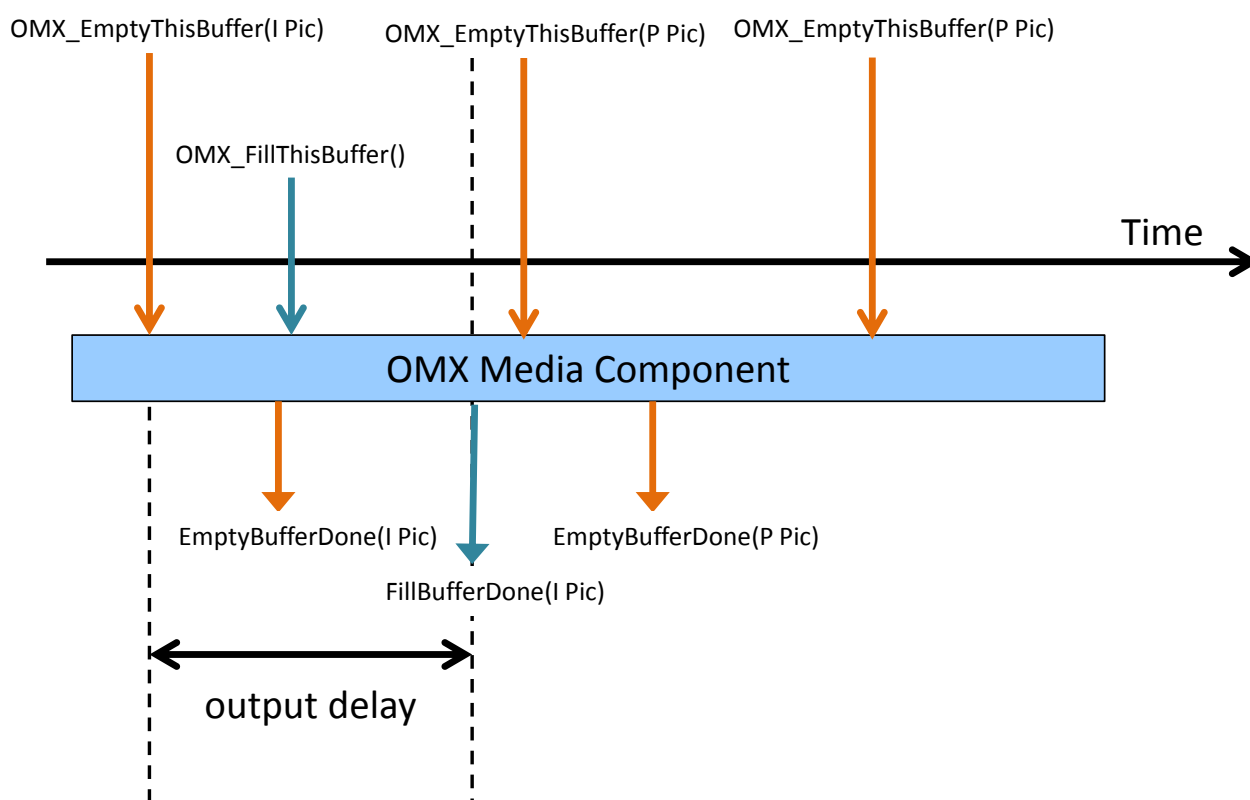
It is possible to reduce the output delay by following setting:

- Disable the output picture reordering.  
Refer to 5.2.1 for the index and 6.5 for the parameter.

When the IL client applies this setting to Video Decoder Media Component, the input stream should meet the following condition:

- The display order is the same with the decode order.

Figure 7-3 illustrates the sequence after applying output delay reduction.



**Figure 7-3 Output delay reduction**

When the IL client inputs a frame data with multiple buffers, the output delay might be increased. The IL client should input a frame data with single buffer in order not to increase the output delay.

<b>REVISION HISTORY</b>	OMX Media Component User's Manual : Video Decoder Common Part
-------------------------	--

Rev.	Date	Description	
		Page	Summary
0.06	Dec. 20, 2013	—	Draft revision based on Japanese User's Manual Rev.0.06.
0.07	Jan. 31, 2014	—	Fixed typos.
0.08	May. 26, 2014	31,33	Fixed Figure 0-1 and Figure 0-2.
0.09	May. 29, 2014	5	Fixed Figure1-1: "Video Common Library" to "Video Decoder Common Library"
	May. 30, 2014	47,48	Add description about information propagation in Table 6-4, Table 6-5
		47	Fixed typos.
	June. 4, 2014	56	Add section 7.6. in Appendix about the output delay
	June. 9, 2014	40	Add description in Notes of 6.5.
1.00	Aug. 19, 2014	39	Fixed description of bReorder parameter.
	Aug, 20, 2014	40	Fixed section 6.5.OMXR_MC_VIDEO_DECODERESULTTYPE.
	Aug, 21, 2014	47	Fixed section 6.8 OMXR_MC_VIDEO_PARAM_DEINTERLACE_MODETYPE.
	Aug, 21, 2014	57	Fixed section 7.6 The output delay and the means of reduction.
	Aug, 21, 2014	40	Fixed section 6.5 OMXR_MC_VIDEO_DECODERESULTTYPE.
	Aug, 21, 2014	14	Fixed section 4.1.1.OMX_UseBuffer.
	Aug. 26, 2014	8	Change VCP decoder to UVCS Library.
	Aug. 26, 2014	44	Change pvPhysImageAddressY to pvPhysImageAddressY;.
	Aug. 27. 2014	39,41	Fixed column of "Definition"

**SALES OFFICES****Renesas Electronics Corporation**<http://www.renesas.com>Refer to "<http://www.renesas.com/>" for the latest and detailed information.**Renesas Electronics America Inc.**  
2880 Scott Boulevard Santa Clara, CA 95050-2554, U.S.A.  
Tel: +1-408-588-6000, Fax: +1-408-588-6130**Renesas Electronics Canada Limited**  
1101 Nicholson Road, Newmarket, Ontario L3Y 9C3, Canada  
Tel: +1-905-898-5441, Fax: +1-905-898-3220**Renesas Electronics Europe Limited**  
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K  
Tel: +44-1628-651-700, Fax: +44-1628-651-804**Renesas Electronics Europe GmbH**  
Arcadiastrasse 10, 40472 Düsseldorf, Germany  
Tel: +49-211-65030, Fax: +49-211-6503-1327**Renesas Electronics (China) Co., Ltd.**  
7th Floor, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100083, P.R.China  
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679**Renesas Electronics (Shanghai) Co., Ltd.**  
Unit 204, 205, AZIA Center, No.1233 Lujiazui Ring Rd., Pudong District, Shanghai 200120, China  
Tel: +86-21-5877-1818, Fax: +86-21-6887-7858 / -7898**Renesas Electronics Hong Kong Limited**  
Unit 1601-1613, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong  
Tel: +852-2886-9318, Fax: +852 2886-9022/9044**Renesas Electronics Taiwan Co., Ltd.**  
13F, No. 363, Fu Shing North Road, Taipei, Taiwan  
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670**Renesas Electronics Singapore Pte. Ltd.**  
80 Bendemeer Road, Unit #06-02 Hyflux Innovation Centre Singapore 339949  
Tel: +65-6213-0200, Fax: +65-6213-0300**Renesas Electronics Malaysia Sdn.Bhd.**  
Unit 906, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia  
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510**Renesas Electronics Korea Co., Ltd.**  
11F, Samik Lavied' or Bldg. 720-2 Yeoksam-Dong, Kangnam-Ku, Seoul 135-080, Korea  
Tel: +82-2-558-3737, Fax: +82-2-558-5141

---

OMX Media Component  
User's Manual: Video Decoder Common Part

Publication Date: Rev. 1.00 Aug. 29, 2014

Published by: Renesas Electronics Corporation  
© 2014 Renesas Electronics Corporation. All rights reserved.

---

# OMX Media Component

User's Manual: Video Decoder Common Part



Renesas Electronics Corporation