**Document Type: Detail Design**

**Document Name:**

**ADSP OMX TDM**

Renesas Design Vietnam Co. Ltd

R-Car Software Solution 2 Group

Middleware 1 Team

Renesas Electronics Corporation

Table of Contents

[1. Overview 3](#_Toc529283921)

[2. Function list 4](#_Toc529283922)

[3. Detail information 5](#_Toc529283923)

[3.1 Data type and Macro definition 5](#_Toc529283924)

[3.2 Function definition 6](#_Toc529283925)

[3.2.1 OMX TDM\_Capture 6](#_Toc529283926)

[3.2.1.1 TDM\_CAPTURE\_Setup 6](#_Toc529283927)

[3.2.1.2 TDM\_CAPTURE\_RuntimeInit 7](#_Toc529283928)

[3.2.1.3 TDM\_CAPTURE\_GetParam 8](#_Toc529283929)

[3.2.1.4 TDM\_CAPTURE\_TimeStamp 9](#_Toc529283930)

[3.2.1.5 TDM\_CAPTURE\_GetParameter 10](#_Toc529283931)

[3.2.1.6 TDM\_CAPTURE\_SetParameter 11](#_Toc529283932)

[3.2.1.7 TDM\_CAPTURE\_ComponentInit 12](#_Toc529283933)

[3.2.1.8 TDM\_CAPTURE\_ComponentDeInit 13](#_Toc529283934)

[3.2.1.9 TDM\_CAPTURE\_ComponentCreate 14](#_Toc529283935)

[3.2.2 OMX TDM RENDERER 16](#_Toc529283936)

[3.2.2.1 TDM\_RENDERER\_Setup 16](#_Toc529283937)

[3.2.2.2 RENDERER\_RuntimeInit 17](#_Toc529283938)

[3.2.2.3 TDM\_RENDERER\_GetParam 18](#_Toc529283939)

[3.2.2.4 TDM\_RENDERER\_TimeStamp 19](#_Toc529283940)

[3.2.2.5 TDM\_RENDERER\_GetParameter 20](#_Toc529283941)

[3.2.2.6 TDM\_RENDERER\_SetParameter 21](#_Toc529283942)

[3.2.2.7 TDM\_RENDERER\_ComponentInit 22](#_Toc529283943)

[3.2.2.8 TDM\_RENDERER\_ComponentDeInit 23](#_Toc529283944)

[3.2.2.9 TDM\_RENDERER\_ComponentCreate 24](#_Toc529283945)

[4. Revision history 26](#_Toc529283946)

List of Figures

[Figure 1‑1 The software architecture 3](#_Toc529283947)

[Figure 3‑1 TDM\_CAPTURE\_Setup flowchart 6](#_Toc529283948)

[Figure 3‑2 TDM\_CAPTURE\_RuntimeInit flowchart 7](#_Toc529283949)

[Figure 3‑3 TDM\_CAPTURE\_GetParam flowchart 8](#_Toc529283950)

[Figure 3‑4 TDM\_CAPTURE\_TimeStamp flowchart 9](#_Toc529283951)

[Figure 3‑5 TDM\_CAPTURE\_GetParameter flowchart 10](#_Toc529283952)

[Figure 3‑6 TDM\_CAPTURE\_SetParameter flowchart 11](#_Toc529283953)

[Figure 3‑7 TDM\_CAPTURE\_ComponentInit flowchart 12](#_Toc529283954)

[Figure 3‑8 TDM\_CAPTURE\_ComponentDeInit flowchart 13](#_Toc529283955)

[Figure 3‑9 TDM\_CAPTURE\_ComponentCreate flowchart 15](#_Toc529283956)

[Figure 3‑12 TDM\_RENDERER\_Setup flowchart 16](#_Toc529283957)

[Figure 3‑13 TDM\_RENDERER\_GetParam flowchart 18](#_Toc529283958)

[Figure 3‑14 TDM\_RENDERER\_TimeStamp flowchart 19](#_Toc529283959)

[Figure 3‑15 TDM\_RENDERER\_GetParameter flowchart 20](#_Toc529283960)

[Figure 3‑16 TDM\_RENDERER\_SetParameter flowchart 21](#_Toc529283961)

[Figure 3‑17 TDM\_RENDERER\_ComponentInit flowchart 22](#_Toc529283962)

[Figure 3‑18 TDM\_RENDERER\_ComponentDeInit flowchart 23](#_Toc529283963)

[Figure 3‑19 TDM\_RENDERER\_ComponentCreate flowchart 25](#_Toc529283964)

List of Table

[Table 2‑1 Function list of OMX TDM Capture 4](#_Toc529283965)

[Table 2‑2 Function list of OMX TDM Renderer 4](#_Toc529283966)

[Table 3‑1 XAOMX\_TDM\_CAP type structure information 5](#_Toc529283967)

[Table 3‑2 XAOMX\_TDM\_RDR type structure information 5](#_Toc529283968)

[Table 3‑3 Macro definitions of OMX TDM Capture 5](#_Toc529283969)

[Table 3‑4 Macro definitions of OMX TDM Renderer 5](#_Toc529283970)

[Table 2‑1 Function list of OMX TDM Capture 4](#_Toc534286234)

[Table 2‑2 Function list of OMX TDM Renderer 4](#_Toc534286235)

[Table 3‑1 XAOMX\_TDM\_CAP type structure information 5](#_Toc534286236)

[Table 3‑2 XAOMX\_TDM\_RDR type structure information 5](#_Toc534286237)

[Table 3‑3 Macro definitions of OMX TDM Capture 5](#_Toc534286238)

[Table 3‑4 Macro definitions of OMX TDM Renderer 5](#_Toc534286239)

# Overview

In this chapter, overview of OMX TDM interface is explained.

User Application

OMX TDM

OMX Renderer/Capture

OMX Equalizer

User Space

ADSP Interface for Linux (base)

Kernel Space

ARM

ADSP Driver for Linux

Audio HW

ADSP

NC/EC

ADSP Framework

Renderer/Capture Plugin

TDM Plugin

Equalizer Plugin

Codec

This document’s target is in side of red square.

DAC/ADC

SRC/CTU/MIX/ DVC/ SSI/

ADMA

Figure 1‑1 ****The software architecture****

# Function list

The following is list of functions:

Table 2‑1 Function list of OMX TDM Capture

|  |  |  |
| --- | --- | --- |
| **No.** | **Function Name** | **Outline** |
| 1 | TDM\_CAPTURE\_Setup | TDM Capture prepare codec setup parameters. |
| 2 | TDM\_CAPTURE\_RuntimeInit | Get DMA channel 1 value to initialize output port of the component. |
| 3 | TDM\_CAPTURE\_GetParam | Get DMA channel 1 value from response structure. |
| 4 | TDM\_CAPTURE\_TimeStamp | Calculate timestamp for current output buffer and next buffer. |
| 5 | TDM\_CAPTURE\_GetParameter | Get TDM Capture parameter. |
| 6 | TDM\_CAPTURE\_SetParameter | Set TDM Capture parameter. |
| 7 | TDM\_CAPTURE\_ComponentDeInit | Destroy Capture component and private data. |
| 8 | TDM\_CAPTURE\_ComponentInit | Initialize base codec interface, parameters, PCM and TDM Capture format defaults. |
| 9 | TDM\_CAPTURE\_ComponentCreate | Create base component and initialize the TDM Capture component. |

Table 2‑2 Function list of OMX TDM Renderer

|  |  |  |
| --- | --- | --- |
| **No.** | **Function Name** | **Outline** |
| 1 | TDM\_RENDERER\_Setup | TDM Renderer prepare codec setup parameters. |
| 2 | TDM\_RENDERER\_RuntimeInit | This is a dummy function, not use in TDM Renderer runtime initialization. |
| 3 | TDM\_RENDERER\_GetParam | This is to check the message length is correct and return 0. |
| 4 | TDM\_RENDERER\_TimeStamp | Calculate timestamp for current input buffer and next buffer. |
| 5 | TDM\_RENDERER\_GetParameter | Get TDM Renderer parameter. |
| 6 | TDM\_RENDERER\_SetParameter | Set TDM Renderer parameter. |
| 7 | TDM\_RENDERER\_ComponentDeInit | Destroy TDM Renderer component and private data. |
| 8 | TDM\_RENDERER\_ComponentInit | Initialize base codec interface, parameters, PCM and TDM Renderer format. |
| 9 | TDM\_RENDERER\_ComponentCreate | Create base component and initialize the TDM Renderer component. |

# Detail information

This section describes detail information of data types, macro definitions.

## Data type and Macro definition

Table 3‑1 XAOMX\_TDM\_CAP type structure information

|  |  |
| --- | --- |
| Member name | Outline |
| XAOMXCodecBase base | Generic codec structure. |
| OMX\_AUDIO\_PARAM\_PCMMODETYPE sPCM | PCM-specific parameters (output port). |
| XAOMX\_AUDIO\_PARAM\_TDM\_CAPTURE sTDM\_CAPTURE | TDM Capture parameters. |

Note:

Structure XAOMXCodecBase is a base codec structure of OMX in omx-codec-base.h

Structure OMX\_AUDIO\_PARAM\_PCMMODETYPE is a PCM format description in OMX\_Audio.h

Structure XAOMX\_AUDIO\_PARAM\_TDM\_CAPTURE is a parametric TDM Capture settings structure.

Table 3‑2 XAOMX\_TDM\_RDR type structure information

|  |  |
| --- | --- |
| Member name | Outline |
| XAOMXCodecBase base | Generic codec structure. |
| OMX\_AUDIO\_PARAM\_PCMMODETYPE sPCM | PCM-specific parameters (input port). |
| XAOMX\_AUDIO\_PARAM\_TDM\_RENDERER sTDM\_RENDERER | TDM Renderer parameters. |

Note:

Structure XAOMX\_AUDIO\_PARAM\_TDM\_RENDERER is a parametric TDM Renderer settings structure.

Table 3‑3 Macro definitions of OMX TDM Capture

|  |  |  |
| --- | --- | --- |
| Macro | Value | Outline |
| NUM\_OUTPUT\_BUFFERS | 4 | Total amount of output buffers. |
| OUTPUT\_BUFFER\_LENGTH | 4096 | Default output buffer length. |
| BUFFER\_ALIGNMENT | 32 | Required data alignment of TDM Capture. |

Table 3‑4 Macro definitions of OMX TDM Renderer

|  |  |  |
| --- | --- | --- |
| Macro | Value | Outline |
| NUM\_INPUT\_BUFFERS | 4 | Total amount of input buffers. |
| INPUT\_BUFFER\_LENGTH | 4096 | Default input buffer length. |
| BUFFER\_ALIGNMENT | 32 | Required data alignment of TDM Renderer. |

## Function definition

### OMX TDM\_Capture

#### TDM\_CAPTURE\_Setup

DD\_API\_TDM\_01\_001

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Syntax** | static s32 TDM\_CAPTURE\_Setup(XAOMXCodecBase \*pBase, xf\_set\_param\_msg\_t \*msg) | | | |
| **Function** | TDM Capture prepare codec setup parameters. | | | |
| **Arguments** | Type | Name | I/O | Description |
| XAOMXCodecBase\* | pBase | I | Pointer to base codec structure of TDM Capture component. |
| xf\_set\_param\_msg\_t\* | msg | O | Message send to TDM Capture plugin. |
| **Return value** | length | | Number of parameters want to set for TDM Capture plugin. | |
| **Description** | * TDM\_CAPTURE\_Setup processing:   - Prepare PCM width, channels, sample rate, input1, 2; DMA channel 1, 2; out sample rate, volume rate values into command message. | | | |

[Covers: FD\_API\_TDM\_003]

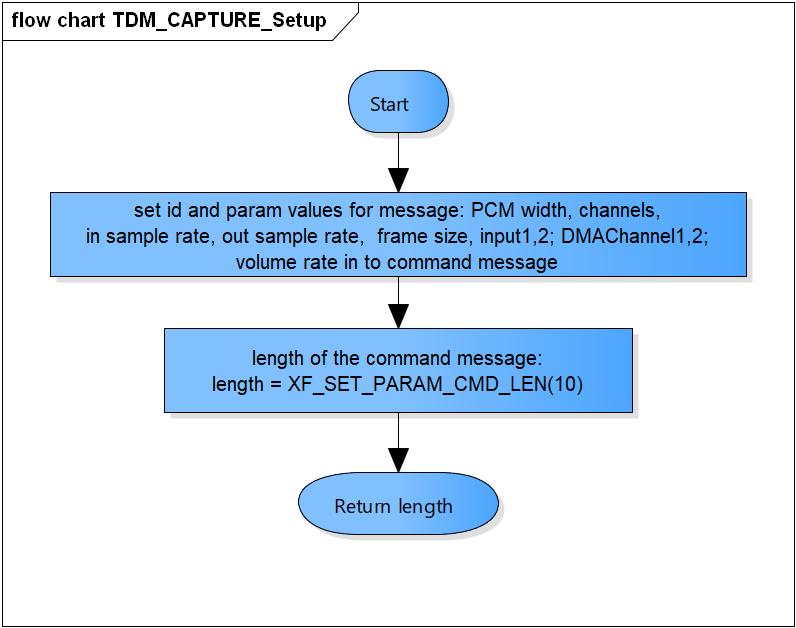


Figure 3‑1 TDM\_CAPTURE\_Setup flowchart

#### TDM\_CAPTURE\_RuntimeInit

DD\_API\_TDM\_01\_002

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Syntax** | static s32 TDM\_CAPTURE\_RuntimeInit(XAOMXCodecBase \*pBase,  xf\_start\_msg\_t \*msg) | | | |
| **Function** | Get DMA channel 1 value to initialize output port of the component. | | | |
| **Arguments** | Type | Name | I/O | Description |
| XAOMXCodecBase\* | pBase | I | Pointer to base codec structure of TDM Capture component. |
| xf\_start\_msg\_t\* | msg | O | Message send to TDM Capture plugin. |
| **Return value** | length | | Number of parameters want to set for TDM Capture plugin. | |
| **Description** | * TDM\_CAPTURE\_RuntimeInit processing:   - Save DMA channel 1 into command structure. | | | |

[Covers: FD\_API\_TDM\_003]

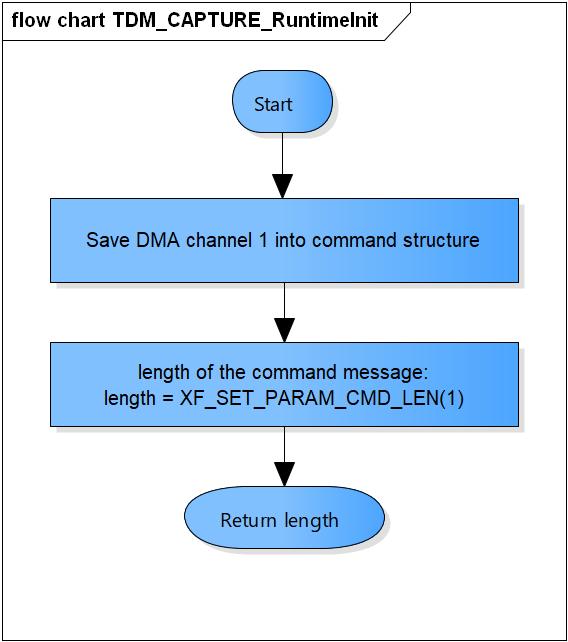


Figure 3‑2 TDM\_CAPTURE\_RuntimeInit flowchart

#### TDM\_CAPTURE\_GetParam

DD\_API\_TDM\_01\_003

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Syntax** | static s32 TDM\_CAPTURE\_GetParam(XAOMXCodecBase \*pBase,  xf\_get\_param\_msg\_t \*msg,  u32 length) | | | |
| **Function** | Get DMA channel 1 value from response structure. | | | |
| **Arguments** | Type | Name | I/O | Description |
| XAOMXCodecBase\* | pBase | I/O | Pointer to base codec structure of TDM Capture component. |
| xf\_get\_param\_msg\_t \* | msg | I | Message receive from TDM Capture plugin. |
| u32 | length | I | Length of message. |
| **Return value** | 0 | | Normal return. | |
| -EBADF | | The message length is invalid | |
| **Description** | * TDM\_CAPTURE\_GetParam processing:   - Get DMA channel 1 values from response structure. | | | |

[Covers: FD\_API\_TDM\_003]

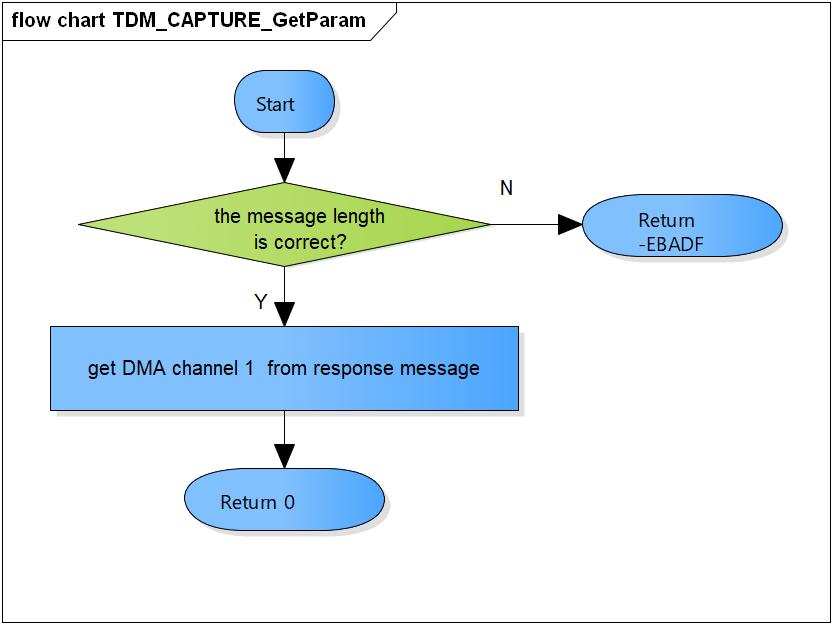


Figure 3‑3 TDM\_CAPTURE\_GetParam flowchart

#### TDM\_CAPTURE\_TimeStamp

DD\_API\_TDM\_01\_004

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Syntax** | static void TDM\_CAPTURE\_TimeStamp(XAOMXCodecBase \*pBase,  OMX\_BUFFERHEADERTYPE \*pBufHdr) | | | |
| **Function** | Calculate timestamp for current output buffer and next buffer. | | | |
| **Arguments** | Type | Name | I/O | Description |
| XAOMXCodecBase\* | pBase | I/O | Pointer to base codec structure of TDM Capture component. |
| OMX\_BUFFERHEADERTYPE\* | pBufHdr | O | Output buffer. |
| **Return value** | None. | | | |
| **Description** | * TDM\_CAPTURE\_TimeStamp processing:   - Add current timestamp to the output buffer.  - Calculate timestamp for the next buffer. | | | |

[Covers: FD\_API\_TDM\_003]

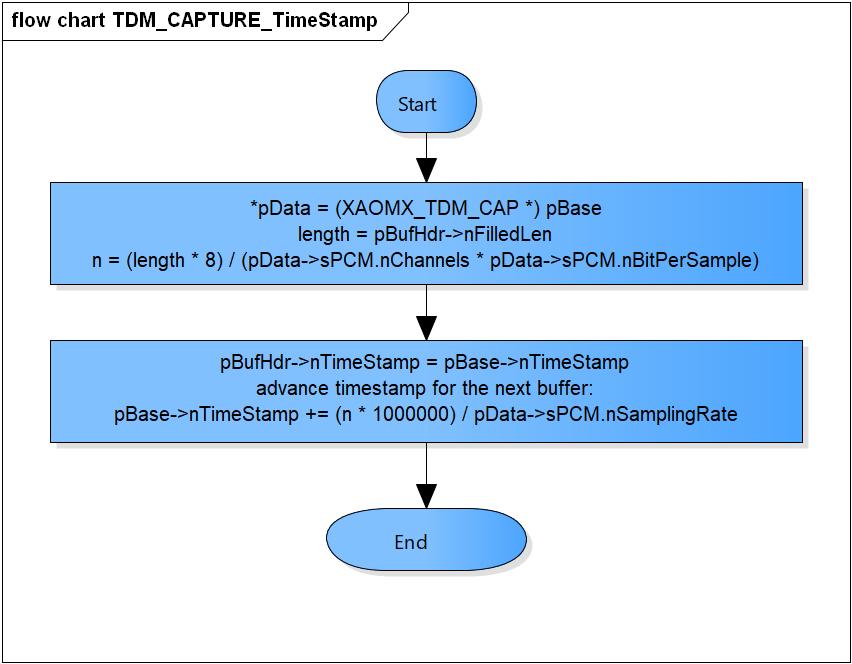


Figure 3‑4 TDM\_CAPTURE\_TimeStamp flowchart

#### TDM\_CAPTURE\_GetParameter

DD\_API\_TDM\_01\_005

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Syntax** | static OMX\_ERRORTYPE TDM\_CAPTURE\_GetParameter(XAOMXCodecBase \*pBase,  OMX\_INDEXTYPE nIndex,  OMX\_PTR pParam) | | | |
| **Function** | Get parameter and save to pParam. | | | |
| **Arguments** | Type | Name | I/O | Description |
| XAOMXCodecBase\* | pBase | I/O | Pointer to base codec structure of TDM Capture component. |
| OMX\_INDEXTYPE | nIndex | I | Index of parameter structure.  Valid values:  OMX\_IndexParamAudioPcm  XAOMX\_IndexParamAudioTDMCapture  OMX\_IndexParamCompBufferSupplier |
| OMX\_PTR | pParam | I | Pointer to parameter structure want to save |
| **Return value** | OMX\_ErrorNone | | Normal return. | |
| OMX\_ErrorUnsupportedIndex | | Wrong Index of structure | |
| **Description** | * TDM\_CAPTURE\_GetParameter processing:   - Get parameters of OMX\_AUDIO\_PARAM\_PCMMODETYPE, XAOMX\_AUDIO\_PARAM\_TDM\_CAPTURE, OMX\_PARAM\_BUFFERSUPPLIERTYPE structure base on nIndex. | | | |

[Covers: FD\_API\_TDM\_004]

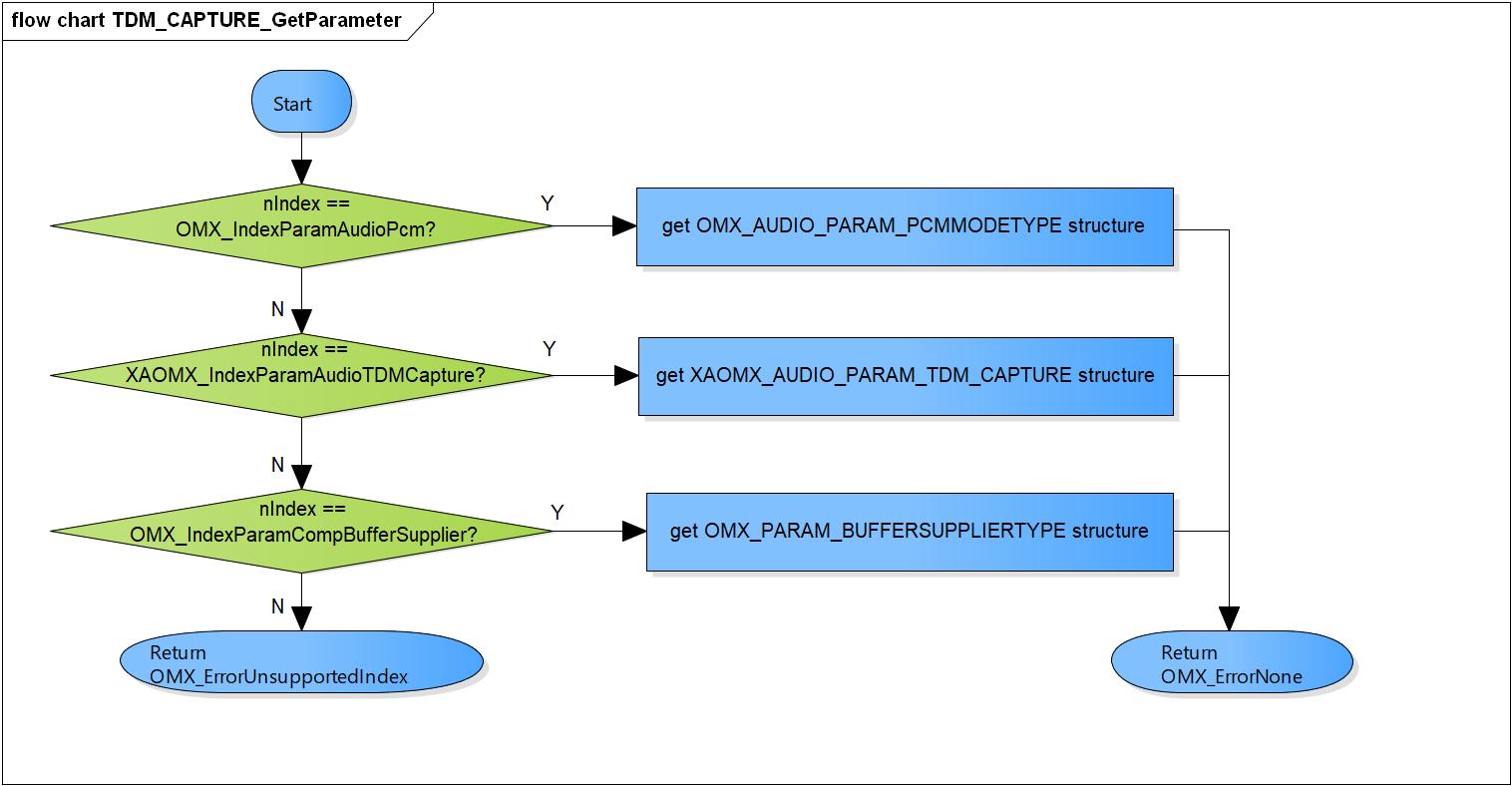


Figure 3‑5 TDM\_CAPTURE\_GetParameter flowchart

#### TDM\_CAPTURE\_SetParameter

DD\_API\_TDM\_01\_006

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Syntax** | static OMX\_ERRORTYPE TDM\_CAPTURE\_SetParameter(XAOMXCodecBase \*pBase,  OMX\_INDEXTYPE nIndex,  OMX\_PTR pParam) | | | |
| **Function** | Set parameter from pParam and save to parameter structure. | | | |
| **Arguments** | Type | Name | I/O | Description |
| XAOMXCodecBase\* | pBase | I/O | Pointer to base codec structure of TDM Capture component. |
| OMX\_INDEXTYPE | nIndex | I | Index of parameter structure.  Valid values:  OMX\_IndexParamAudioPcm  XAOMX\_IndexParamAudioTDMCapture  OMX\_IndexParamCompBufferSupplier |
| OMX\_PTR | pParam | O | Pointer to parameter structure want to set. |
| **Return value** | OMX\_ErrorNone | | Normal return. | |
| OMX\_ErrorUnsupportedIndex | | Wrong Index of structure. | |
| OMX\_ErrorBadPortIndex | | Port index is invalid. | |
| **Description** | * TDM\_CAPTURE\_SetParameter processing:   - Set parameters of OMX\_AUDIO\_PARAM\_PCMMODETYPE, XAOMX\_AUDIO\_PARAM\_TDM\_CAPTURE, OMX\_PARAM\_BUFFERSUPPLIERTYPE structure base on nIndex. | | | |

[Covers: FD\_API\_TDM\_005]

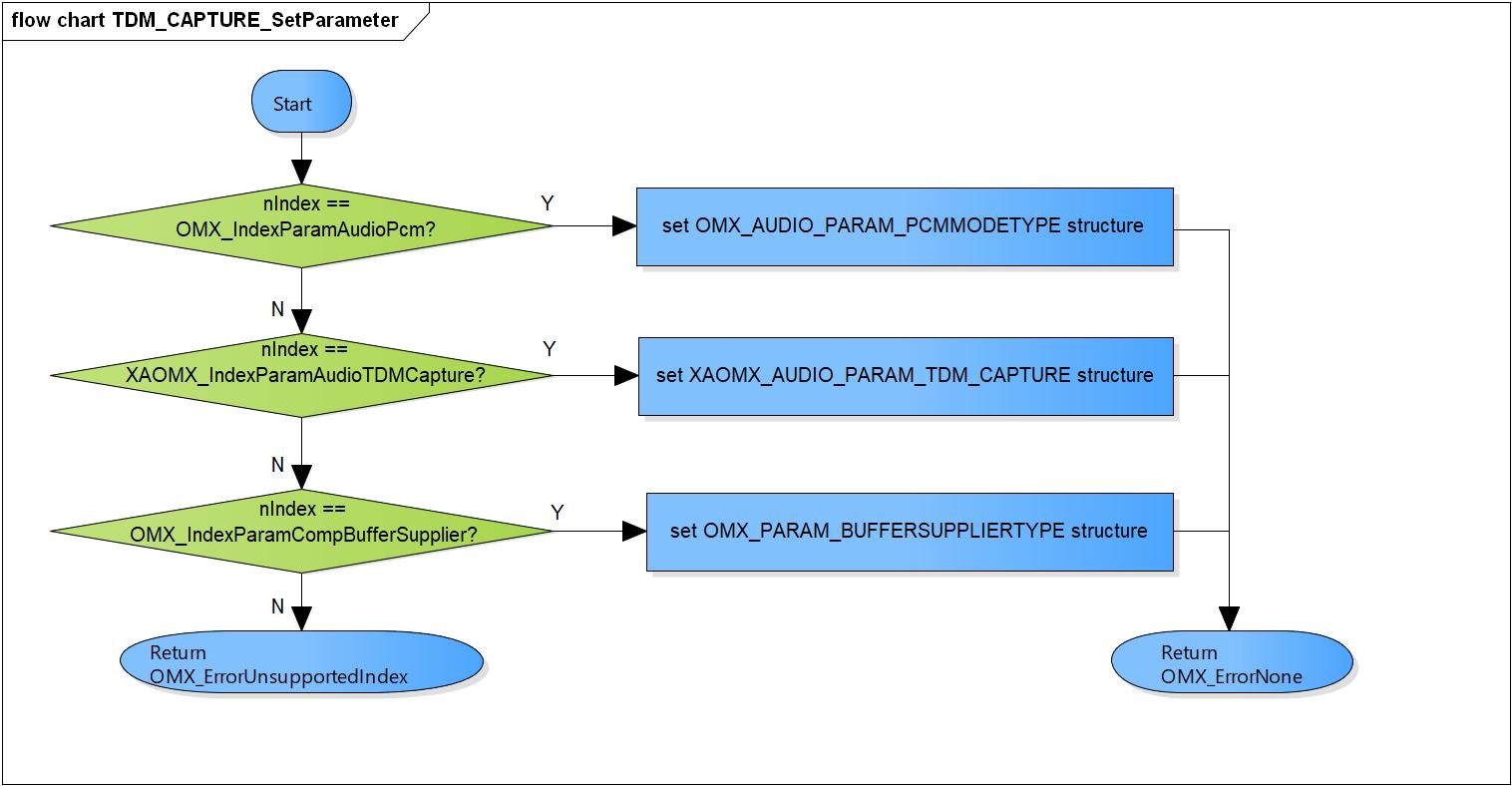


Figure 3‑6 TDM\_CAPTURE\_SetParameter flowchart

#### TDM\_CAPTURE\_ComponentInit

DD\_API\_TDM\_01\_007

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Syntax** | static OMX\_ERRORTYPE CAPTURE\_ComponentInit(xf\_proxy\_t \*proxy,  OMX\_HANDLETYPE hComponent,  OMX\_PTR pAppData,  OMX\_CALLBACKTYPE \*pCallbacks) | | | |
| **Function** | Initialize base codec interface, parameters, PCM and TDM Capture format. | | | |
| **Arguments** | Type | Name | I/O | Description |
| xf\_proxy\_t \* | proxy | O | OpenMAX interface proxy. |
| OMX\_HANDLETYPE | hComponent | O | Pointer to TDM Capture component handle. |
| OMX\_PTR | pAppData | I | Pointer to pAppData of TDM Capture component. |
| OMX\_CALLBACKTYPE | pCallbacks | I | Pointer to Callbacks of TDM Capture component. |
| **Return value** | OMX\_ErrorNone | | Normal return. | |
| OMX\_ErrorInsufficientResources | | Initialize base codec interface is fail. | |
| **Description** | * TDM\_CAPTURE\_ComponentInit processing:   - Initialize base codec interface, parameters, PCM and TDM Capture format. | | | |

[Covers: FD\_API\_TDM\_001]

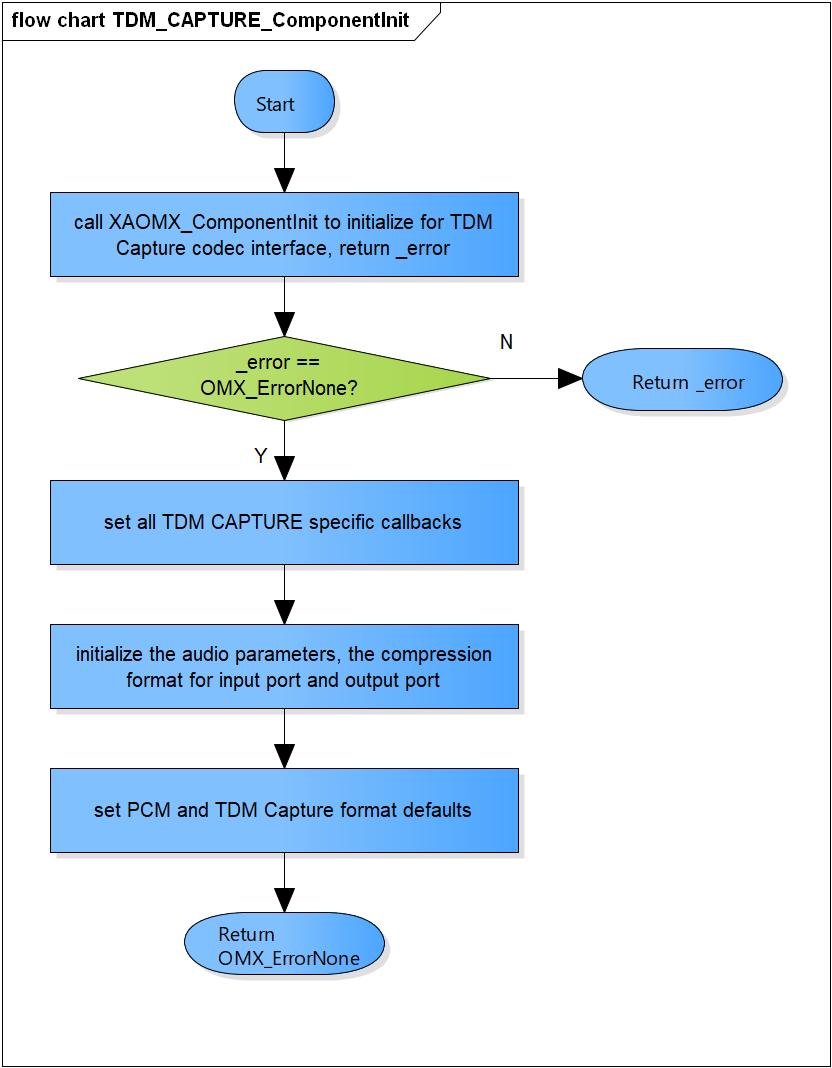


Figure 3‑7 TDM\_CAPTURE\_ComponentInit flowchart

#### TDM\_CAPTURE\_ComponentDeInit

DD\_API\_TDM\_01\_008

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Syntax** | static OMX\_ERRORTYPE TDM\_CAPTURE\_ComponentDeInit(OMX\_HANDLETYPE hComponent) | | | |
| **Function** | Destroy TDM Capture component and private data. | | | |
| **Arguments** | Type | Name | I/O | Description |
| OMX\_HANDLETYPE | hComponent | O | Pointer to TDM Capture component handle. |
| **Return value** | OMX\_ErrorNone | | Normal return. | |
| OMX\_ErrorBadParameter | | Private data is NULL. | |
| **Description** | * TDM\_CAPTURE\_ComponentDeInit processing:   - Destroy TDM Capture component and private data. | | | |

[Covers: FD\_API\_TDM\_002]

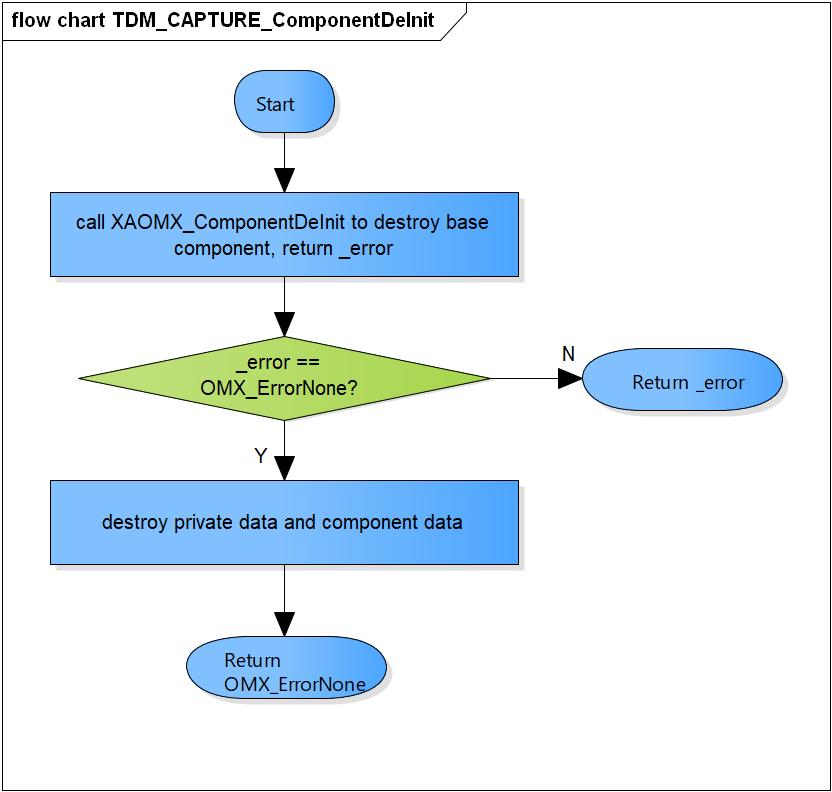


Figure 3‑8 TDM\_CAPTURE\_ComponentDeInit flowchart

#### TDM\_CAPTURE\_ComponentCreate

DD\_API\_TDM\_01\_009

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Syntax** | OMX\_ERRORTYPE TDM\_CAPTURE\_ComponentCreate(xf\_proxy\_t \*proxy, OMX\_HANDLETYPE \*hComponent, OMX\_PTR pAppData, OMX\_CALLBACKTYPE \*pCallbacks) | | | |
| **Function** | Create base component and initialize the TDM Capture component. | | | |
| **Arguments** | Type | Name | I/O | Description |
| xf\_proxy\_t\* | proxy | O | OpenMAX interface proxy. |
| OMX\_HANDLETYPE\* | hComponent | O | Pointer to TDM Capture component handle. |
| OMX\_PTR | pAppData | I | Pointer to pAppData of TDM Capture component. |
| OMX\_CALLBACKTYPE \* | pCallbacks | I | Pointer to Callbacks of TDM Capture component. |
| **Return value** | OMX\_ErrorNone | | Normal return. | |
| OMX\_ErrorInsufficientResources | | Cannot create the base component.  Cannot allocate private memory.  Initialize the TDM Capture component is fail. | |
| **Description** | * TDM\_CAPTURE\_ComponentCreate processing:   - Create the base component.  - Allocate the private memory.  - Initialize the TDM Capture component. | | | |

[Covers: FD\_API\_TDM\_001]

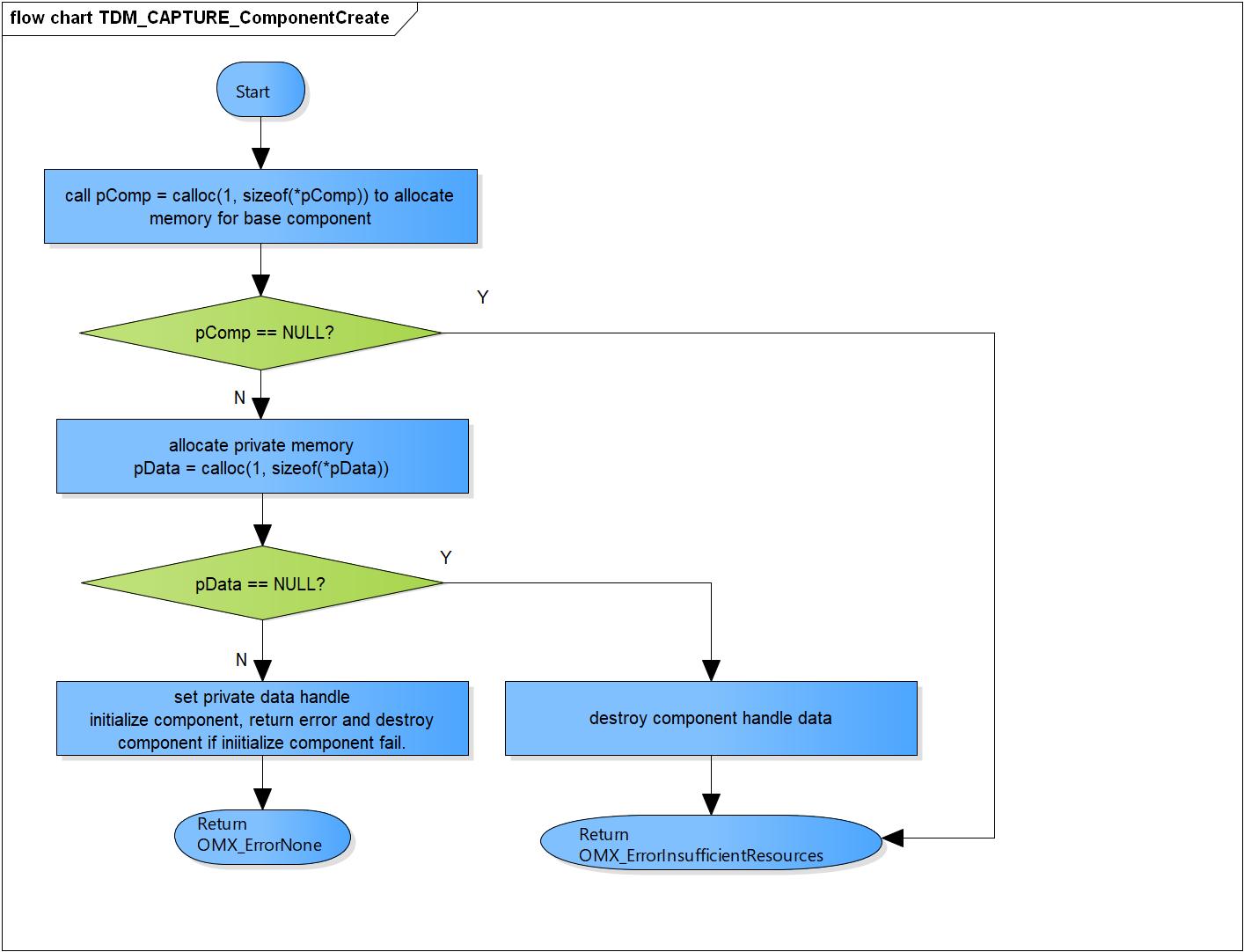


Figure 3‑9 TDM\_CAPTURE\_ComponentCreate flowchart

### OMX TDM RENDERER

#### TDM\_RENDERER\_Setup

DD\_API\_TDM\_01\_010

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Syntax** | static s32 TDM\_RENDERER\_Setup(XAOMXCodecBase \*pBase,  xf\_set\_param\_msg\_t \*msg) | | | |
| **Function** | TDM Renderer prepare codec setup parameters. | | | |
| **Arguments** | Type | Name | I/O | Description |
| XAOMXCodecBase\* | pBase | I | Pointer to base codec structure of TDM Renderer component. |
| xf\_set\_param\_msg\_t\* | msg | O | Message send to TDM Renderer plugin. |
| **Return value** | length | | Number of parameters want to set for TDM Renderer plugin. | |
| **Description** | * TDM\_Renderer\_Setup processing:   - Prepare PCM width, channels, sample rate, output1, 2; DMA channel 1, 2; out sample rate, volume rate values into command message. | | | |

[Covers: FD\_API\_TDM\_003]

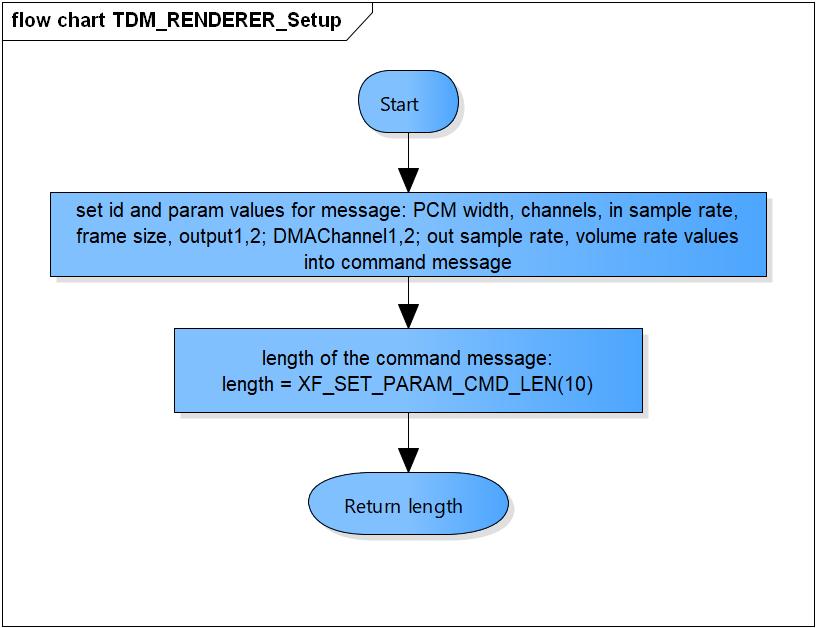


Figure 3‑10 TDM\_RENDERER\_Setup flowchart

#### RENDERER\_RuntimeInit

DD\_API\_TDM\_01\_011

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Syntax** | static s32 TDM\_RENDERER\_RuntimeInit(XAOMXCodecBase\*pBase,  xf\_start\_msg\_t \*msg) | | | |
| **Function** | This is a dummy function, not use in TDM Renderer runtime initialization. | | | |
| **Arguments** | Type | Name | I/O | Description |
| XAOMXCodecBase\* | pBase | I | Pointer to base codec structure of TDM Renderer component. |
| xf\_start\_msg\_t\* | msg | O | Message send to TDM Renderer plugin. |
| **Return value** | 0 | | No parameter need to set to plugin | |
| **Description** | * TDM\_RENDERER\_RuntimeInit processing:   - Return 0. | | | |

[Covers: FD\_API\_TDM\_003]

#### TDM\_RENDERER\_GetParam

DD\_API\_TDM\_01\_012

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Syntax** | static s32 TDM\_RENDERER\_GetParam(XAOMXCodecBase \*pBase,  xf\_get\_param\_msg\_t \*msg,  u32 length) | | | |
| **Function** | This is to check the message length is correct and return 0. | | | |
| **Arguments** | Type | Name | I/O | Description |
| XAOMXCodecBase\* | pBase | I | Pointer to base codec structure of TDM RENDERER component. |
| xf\_get\_param\_msg\_t \* | msg | I | Message receive from TDM RENDERER plugin. |
| u32 | length | I | Length of message. |
| **Return value** | 0 | | Normal return | |
| -EBADF | | The message length is invalid | |
| **Description** | * TDM\_RENDERER\_GetParam processing:   - Check the message length is correct and return 0. | | | |

[Covers: FD\_API\_TDM\_003]

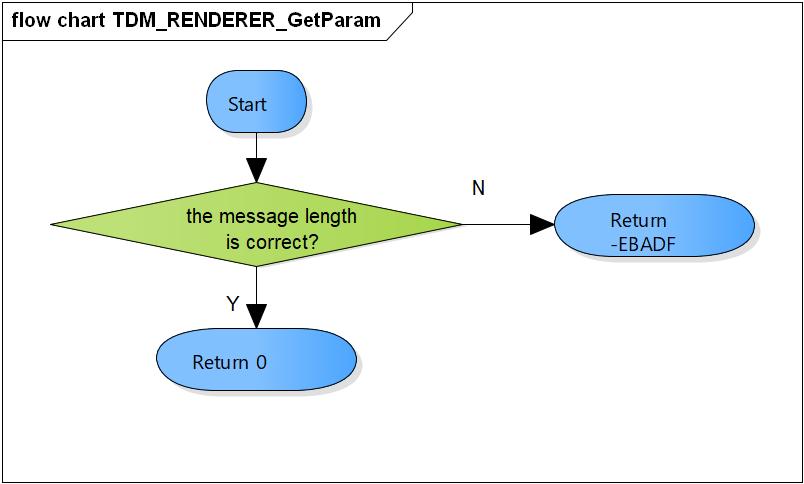


Figure 3‑11 TDM\_RENDERER\_GetParam flowchart

#### TDM\_RENDERER\_TimeStamp

DD\_API\_TDM\_01\_013

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Syntax** | static void TDM\_RENDERER\_TimeStamp(XAOMXCodecBase \*pBase,  OMX\_BUFFERHEADERTYPE \*pBufHdr) | | | |
| **Function** | Calculate timestamp for current input buffer and next buffer. | | | |
| **Arguments** | Type | Name | I/O | Description |
| XAOMXCodecBase\* | pBase | I/O | Pointer to base codec structure of TDM RENDERER component. |
| OMX\_BUFFERHEADERTYPE\* | pBufHdr | O | Input buffer. |
| **Return value** | None. | | | |
| **Description** | * TDM\_RENDERER\_TimeStamp processing:   - Add current timestamp to the Input buffer.  - Calculate timestamp for the next buffer. | | | |

[Covers: FD\_API\_TDM\_003]

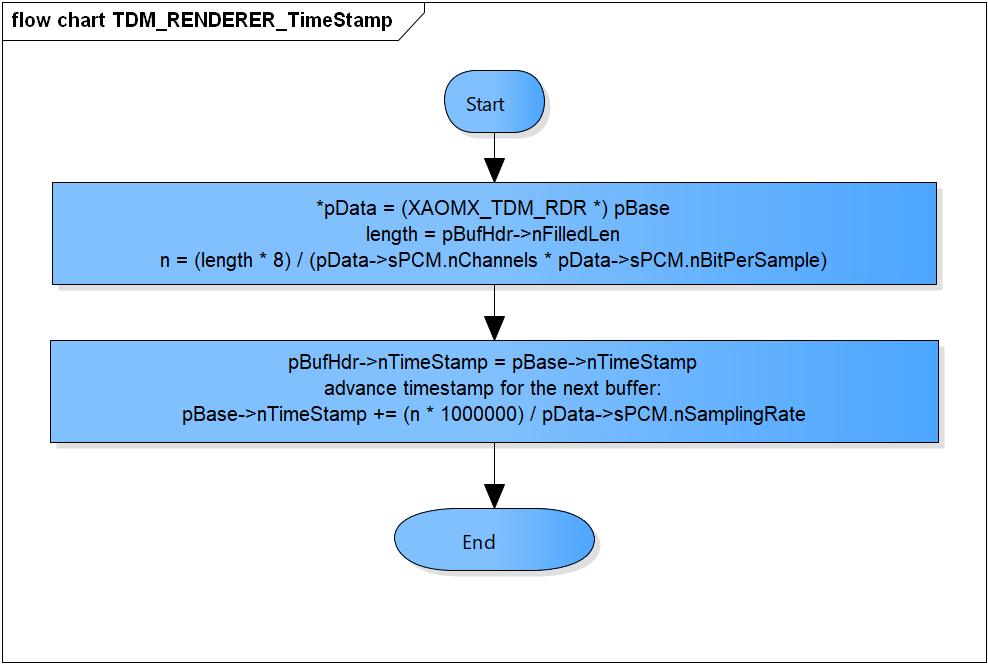


Figure 3‑12 TDM\_RENDERER\_TimeStamp flowchart

#### TDM\_RENDERER\_GetParameter

DD\_API\_TDM\_01\_014

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Syntax** | static OMX\_ERRORTYPE TDM\_RENDERER\_GetParameter(XAOMXCodecBase \*pBase,  OMX\_INDEXTYPE nIndex,  OMX\_PTR pParam) | | | |
| **Function** | Get parameter and save to pParam. | | | |
| **Arguments** | Type | Name | I/O | Description |
| XAOMXCodecBase\* | pBase | I/O | Pointer to base codec structure of TDM Renderer component. |
| OMX\_INDEXTYPE | nIndex | I | Index of parameter structure.  Valid values:  OMX\_IndexParamAudioPcm  XAOMX\_IndexParamAudioTDMRenderer  OMX\_IndexParamCompBufferSupplier |
| OMX\_PTR | pParam | O | Pointer to parameter structure want to save. |
| **Return value** | OMX\_ErrorNone | | Normal return. | |
| OMX\_ErrorUnsupportedIndex | | Wrong Index of structure. | |
| **Description** | * TDM\_RENDERER\_GetParameter pprocessing:   - Get parameters of OMX\_AUDIO\_PARAM\_PCMMODETYPE, XAOMX\_AUDIO\_PARAM\_TDM\_RENDERER, OMX\_PARAM\_BUFFERSUPPLIERTYPE structure base on nIndex. | | | |

[Covers: FD\_API\_TDM\_004]

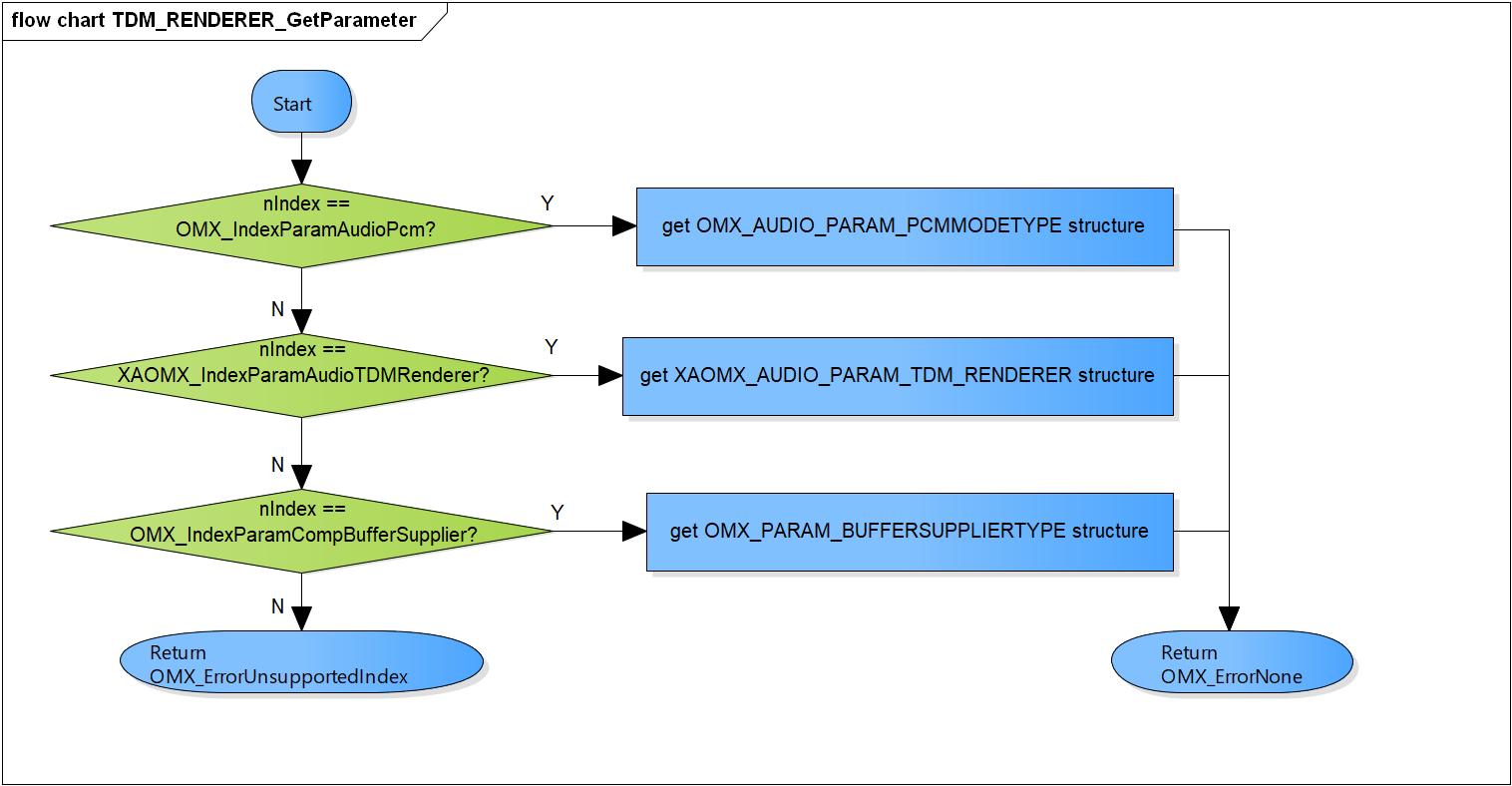


Figure 3‑13 TDM\_RENDERER\_GetParameter flowchart

#### TDM\_RENDERER\_SetParameter

DD\_API\_TDM\_01\_015

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Syntax** | static OMX\_ERRORTYPE TDM\_RENDERER\_SetParameter(XAOMXCodecBase \*pBase,  OMX\_INDEXTYPE nIndex,  OMX\_PTR pParam) | | | |
| **Function** | Set parameter from pParam and save to parameter structure. | | | |
| **Arguments** | Type | Name | I/O | Description |
| XAOMXCodecBase\* | pBase | I/O | Pointer to base codec structure of TDM Renderer component. |
| OMX\_INDEXTYPE | nIndex | I | Index of parameter structure.  Valid values:  OMX\_IndexParamAudioPcm  XAOMX\_IndexParamAudioTDMRenderer  OMX\_IndexParamCompBufferSupplier |
| OMX\_PTR | pParam | I | Pointer to parameter structure want to set. |
| **Return value** | OMX\_ErrorNone | | Normal return. | |
| OMX\_ErrorUnsupportedIndex | | Wrong Index of structure. | |
| OMX\_ErrorBadPortIndex | | Port index is invalid. | |
| **Description** | * TDM\_RENDERER\_SetParameter processing:   - Set parameters of OMX\_AUDIO\_PARAM\_PCMMODETYPE, XAOMX\_AUDIO\_PARAM\_TDM\_RENDERER, OMX\_PARAM\_BUFFERSUPPLIERTYPE structure base on nIndex. | | | |

[Covers: FD\_API\_TDM\_005]

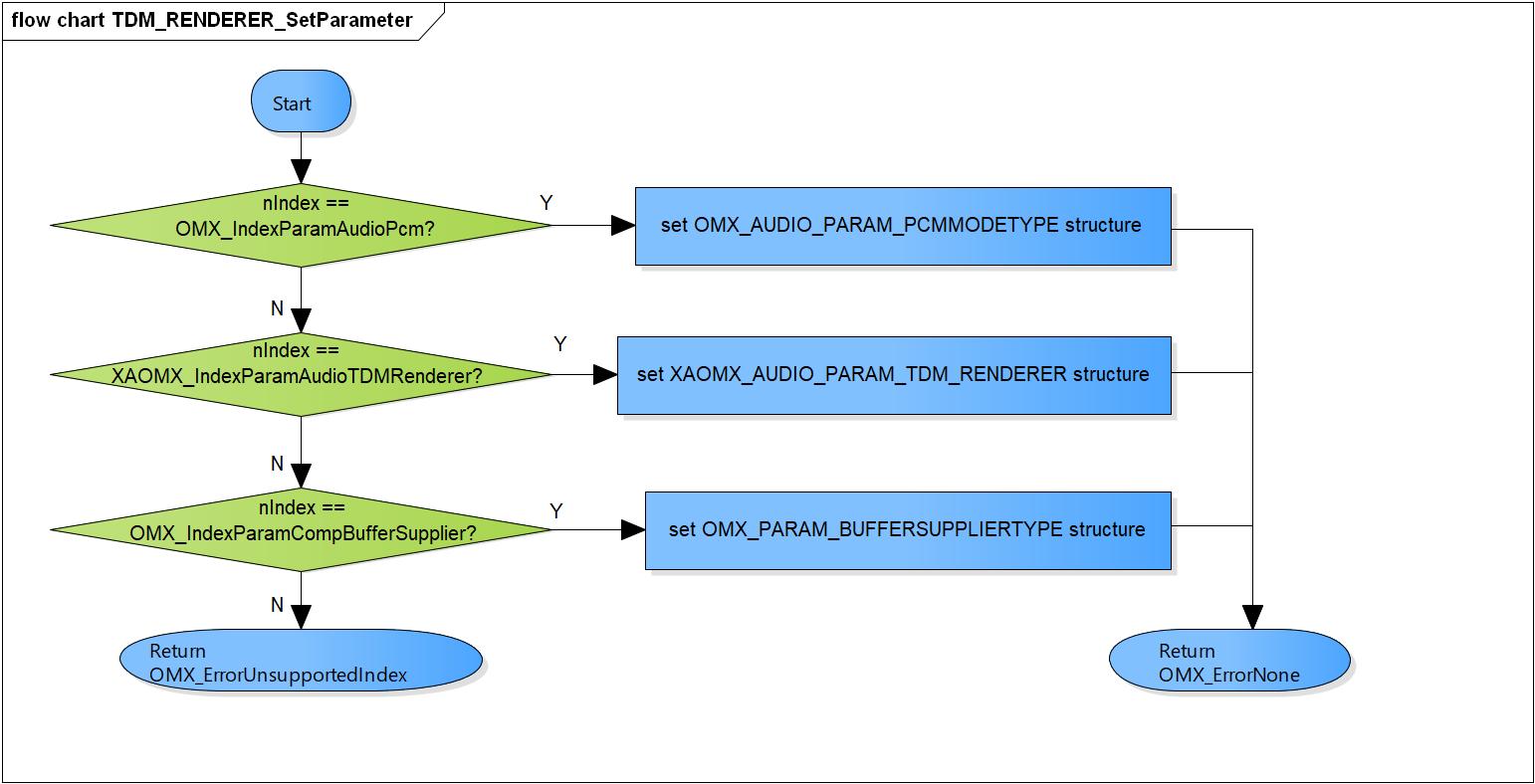


Figure 3‑14 TDM\_RENDERER\_SetParameter flowchart

#### TDM\_RENDERER\_ComponentInit

DD\_API\_TDM\_01\_016

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Syntax** | static OMX\_ERRORTYPE TDM\_RENDERER\_ComponentInit(xf\_proxy\_t \*proxy,  OMX\_HANDLETYPE hComponent,  OMX\_PTR pAppData,  OMX\_CALLBACKTYPE \*pCallbacks) | | | |
| **Function** | Initialize base codec interface, parameters, PCM and TDM Renderer format. | | | |
| **Arguments** | Type | Name | I/O | Description |
| xf\_proxy\_t \* | proxy | O | OpenMAX interface proxy. |
| OMX\_HANDLETYPE | hComponent | O | Pointer to TDM Renderer component handle. |
| OMX\_PTR | pAppData | I | Pointer to pAppData of TDM Renderer component. |
| OMX\_CALLBACKTYPE | pCallbacks | I | Pointer to Callbacks of TDM Renderer component. |
| **Return value** | OMX\_ErrorNone | | Normal return. | |
| OMX\_ErrorInsufficientResources | | Initialize base codec interface is fail. | |
| **Description** | * TDM\_RENDERER\_ComponentInit processing:   - Initialize base codec interface, parameters, PCM and TDM Renderer format. | | | |

[Covers: FD\_API\_TDM\_001]

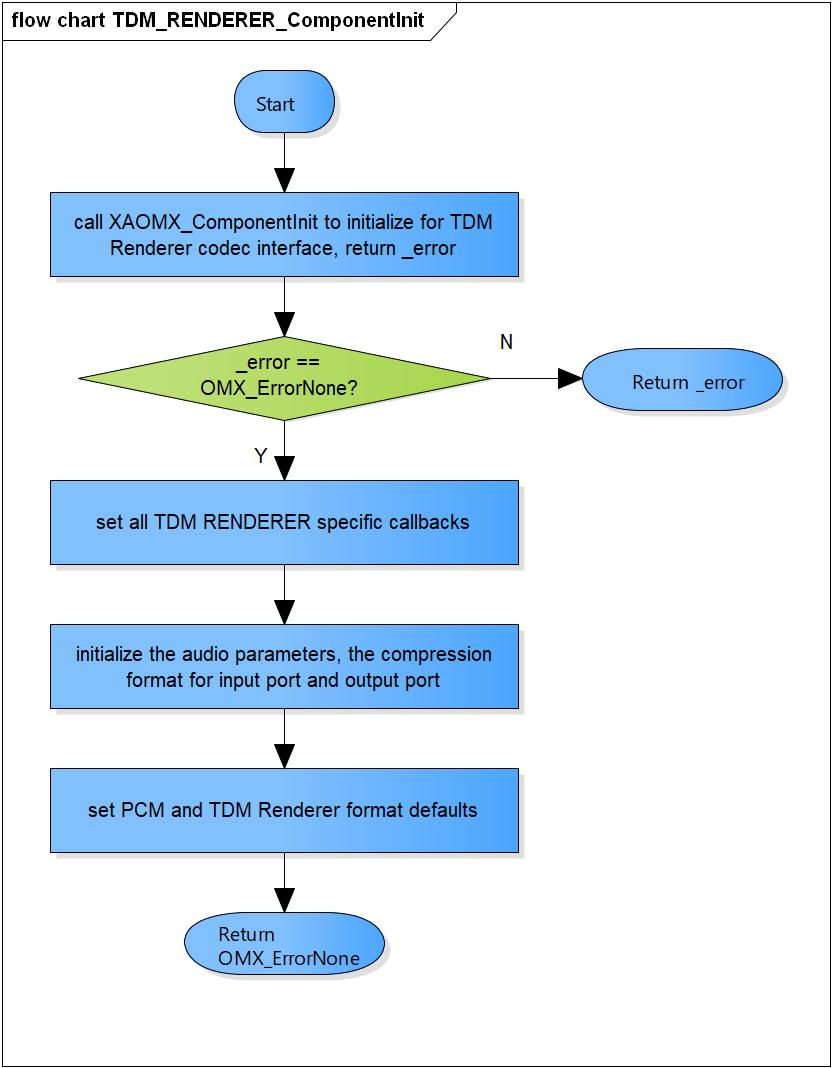


Figure 3‑15 TDM\_RENDERER\_ComponentInit flowchart

#### TDM\_RENDERER\_ComponentDeInit

DD\_API\_TDM\_01\_017

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Syntax** | static OMX\_ERRORTYPE TDM\_RENDERER\_ComponentDeInit(OMX\_HANDLETYPE hComponent) | | | |
| **Function** | Destroy TDM Renderer component and private data. | | | |
| **Arguments** | Type | Name | I/O | Description |
| OMX\_HANDLETYPE | hComponent | O | Pointer to TDM Renderer component handle. |
| **Return value** | OMX\_ErrorNone | | Normal return. | |
| OMX\_ErrorBadParameter | | Private data is NULL. | |
| **Description** | * TDM\_RENDERER\_ComponentDeInit processing:   - Destroy TDM Renderer component and private data. | | | |

[Covers: FD\_API\_TDM\_002]

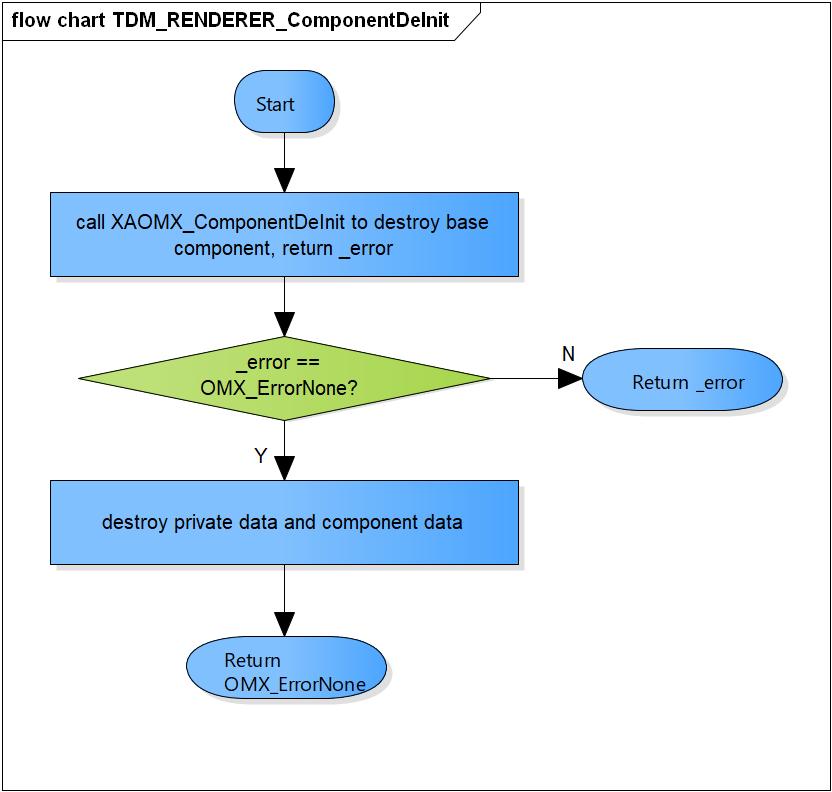


Figure 3‑16 TDM\_RENDERER\_ComponentDeInit flowchart

#### TDM\_RENDERER\_ComponentCreate

DD\_API\_TDM\_01\_018

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Syntax** | static OMX\_ERRORTYPE TDM\_RENDERER\_ComponentCreate(OMX\_HANDLETYPE hComponent) | | | |
| **Function** | Create base component and initialize the TDM Renderer component. | | | |
| **Arguments** | Type | Name | I/O | Description |
| xf\_proxy\_t\* | proxy | O | OpenMAX interface proxy. |
| OMX\_HANDLETYPE\* | hComponent | O | Pointer to TDM Renderer component handle. |
| OMX\_PTR | pAppData | I | Pointer to pAppData of TDM Renderer component. |
| OMX\_CALLBACKTYPE \* | pCallbacks | I | Pointer to Callbacks of TDM Renderer component. |
| **Return value** | OMX\_ErrorNone | | Normal return. | |
| OMX\_ErrorInsufficientResources | | Cannot create the base component.  Cannot allocate private memory.  Initialize the TDM Renderer component is fail. | |
| **Description** | * TDM\_RENDERER\_ComponentCreate processing:   - Create the base component.  - Allocate the private memory.  - Initialize the TDM Renderer component. | | | |

[Covers: FD\_API\_TDM\_001]

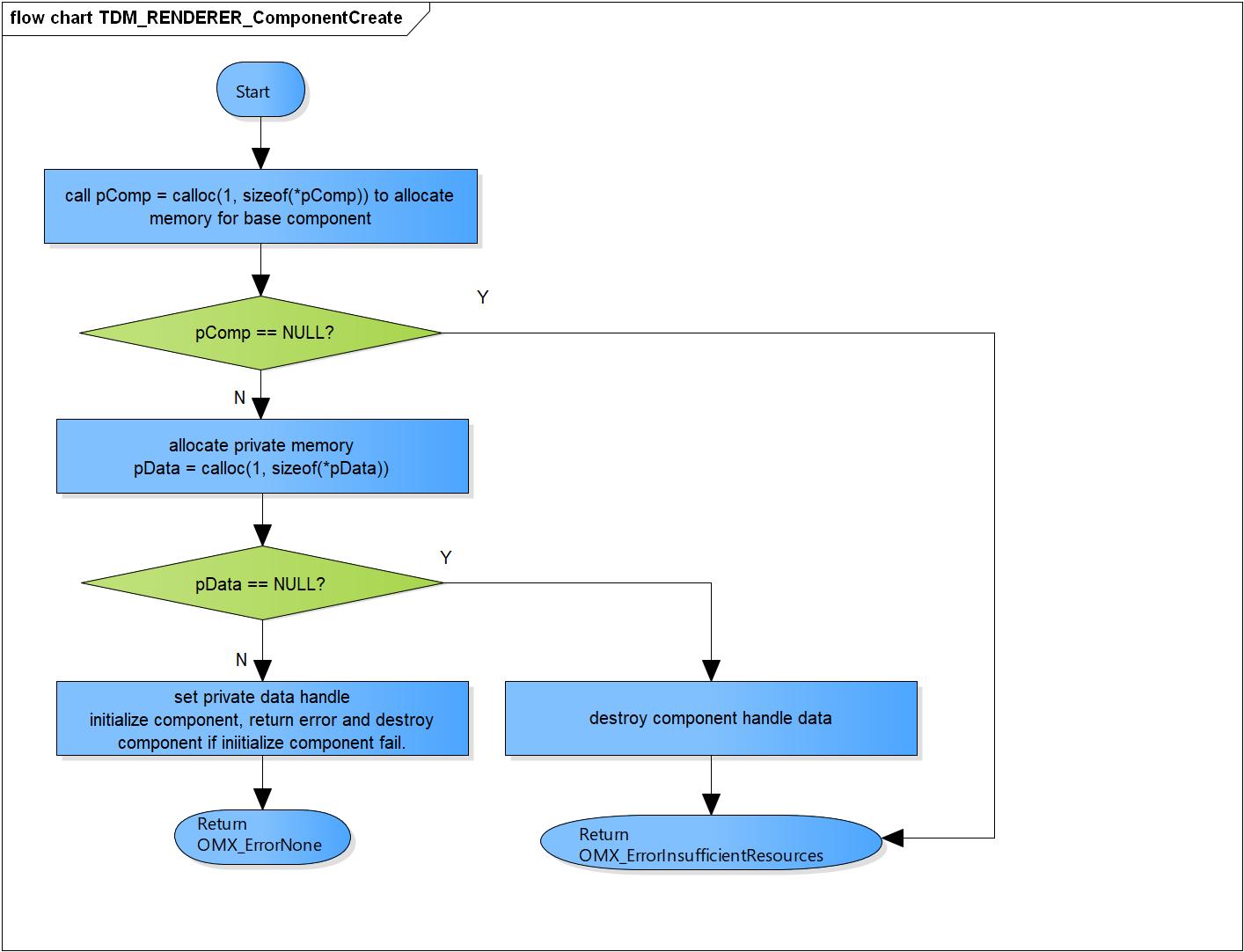


Figure 3‑17 TDM\_RENDERER\_ComponentCreate flowchart

# Revision history

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Version** | **Date** | **Page** | **Content** | **Approved** | **Changed** |
| 1.0.0 | Nov 14 2018 | - | First Edition issued | Vu Phan | Ngu Pham |
| 1.1.0 | Dec 10 2018 | - | Add traceability ID | Vu Phan | Ngu Pham |
| 1.2.0 | Jan 03, 2019 | - | Add range for input parameters | Vu Phan | Tien Tran |