CONFIDENTIAL

RH850/D1x Device Family

Renesas Graphics Library
Video Data Controller E (VDCE) Driver

User's Manual: Software

CONFIDENTIAL

CONFIDENTIAL

CONFIDENTIAL

How to Use This Manual

1. Purpose and Target Readers

This manual is designed to provide the user with an understanding the functions of VDCE driver. This manual is written for engineers who use VDCE driver.

> Particular attention should be paid to the precautionary notes when using the manual. These notes occur within the body of the text, at the end of each section, and in the Usage Notes section.

> The revision history summarizes the locations of revisions and additions. It does not list all revisions. Refer to the text of the manual for details.

Please refer to documents of drivers and hardware for a target system implementing VDCE as necessary.

The following documents are related documents. Make sure to refer to the latest versions of these documents.

Document Type	Description	Document Title	Document No.
User's manual for Hardware	Hardware specifications (pin assignments, memory maps, peripheral function specifications, electrical characteristics, timing charts) and operation description	RH850/D1L/D1M Group User's Manual: Hardware	R01UH0451EJ0220
User's manual for Software	Description of RGL overview	Renesas Graphics Library User's Manual: Software	R01US0181ED0400
	Description of WM	Renesas Graphics Library Window Manager (WM) Driver User's Manual: Software	LLWEB-10035990
	Description of SPEA	Renesas Graphics Library Sprite Engine A (SPEA) Driver User's Manual: Software	LLWEB-10035991
	Description of VDCE	Renesas Graphics Library Video Data Controller E (VDCE) Driver User's Manual: Software	LLWEB-10035992 (This manual)
	Description of VOWE	Renesas Graphics Library Video Output Warping Engine (VOWE) Driver User's Manual: Software	LLWEB-10035993
	Description of JCUA	Renesas Graphics Library JPEG Codec Unit A (JCUA) Driver User's Manual: Software	LLWEB-10035994
	Description of SFMA	Renesas Graphics Library Serial Flash Memory Interface A (SFMA) Driver User's Manual: Software	LLWEB-10064753
	Description of HYPB	Renesas Graphics Library HyperBus Controller (HYPB) Driver User's Manual: Software	LLWEB-10064754
	Description of OCTA	Renesas Graphics Library OctaBus Controller (OCTA) Driver User's Manual: Software	LLWEB-10064755
	Description of VOCA	Renesas Graphics Library Video Output Checker (VOCA) Driver User's Manual: Software	LLWEB-10063801

	Description of DISCOM	Renesas Graphics Library Display Output Comparator (DISCOM) Driver User's Manual: Software	LLWEB-10063802
	Description of DRW2D	Renesas Graphics Library 2D Graphics (DRW2D) Driver User's Manual: Software	LLWEB-10059472
Porting Layer Guide	Description of porting layer of RGL	Renesas Graphics Library Porting Layer Guide	LLWEB-10035995

2. Notation of Numbers and Symbols

This manual uses the following notation.

Binary	0bXXXXXXXX	(X=0 or 1)
Decimal	XXX	(X=0-9)
Hex	0xXXXXXXXX	(X=0-9,A-F)

3. List of Abbreviations and Acronyms

Abbreviation	Full Form
API	Application Programming Interface
bpp	bit per pixel
CLUT	Color Look Up Table
CPU	Central Processing Unit. The microprocessor core of the LSI.
Frame buffer	A region in the memory attached to a layer that can be shown on the screen.
H/W	Hardware
Layer	A H/W concept of the stackable visual area on the display
OIR	Output Image Rendering.
Stride	Distance in pixels between two adjacent pixel rows of the frame buffer in the memory
Screen	A physical display surface; a S/W abstraction of the attached physical display
S/W	Software
VDCE	Video Data Controller E. This is H/W, which controls video input, image synthesis and video output.
WM	Window Manager. This is a driver stack, which enables an abstract access to VDCE driver.

All trademarks and registered trademarks are the property of their respective owners.

Table of Contents

1. Overview

1.1 Feature and Scope

The VDCE driver is a driver stack, which enables an abstract access to the device's video input and video output hardware. The abstraction shall simplify the usage by the application developer and also make it possible to use the same API for different hardware.

1.2 Component Structure

The component structure of VDCE is shown in *Figure 1-1.*
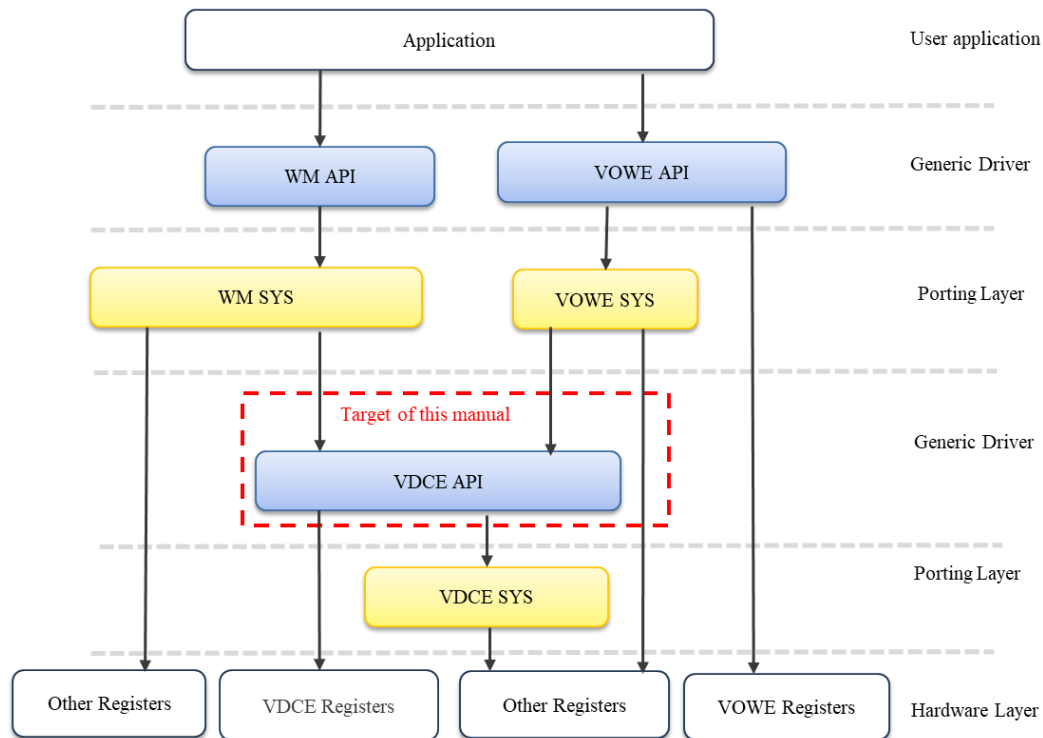
Figure 1-1 : Component structure

For the details of the API, please refer to *chapter 4.*

2. Basic Specification

2.1 Summary Specification

The summary of specification is described in *Table 2-1.*

Table 2-1 Summary Specifications

Items	Description
Target LSI	RH850/D1L2(H), RH850/D1M1(H), RH850/D1M1-V2, RH850/D1M1A, RH850/D1M2(H)

Main Feature	• Operating frequency
	  ○ Video input clock: 27 MHz (for video image), 48 MHz or less (for RGB/YCbCr video image)
	  ○ Panel clock: 48 MHz or less
	• Input video image specification
	  ○ 8-bit input conforming to ITU-R BT.656 standard (27 MHz, interlace signal)
	  ○ Digital pin input: YCbCr422, RGB666, and RGB565 video image
	  ○ Maximum input video image size to be set: 1024 pixels × 1024 lines (horizontal × vertical).
	• Video image scaling processing
	  ○ Vertical: x1/8 to x8, linear/hold interpolation
	  ○ Horizontal: x1/8 to x8, linear/hold interpolation
	• Video image rotation function
	  ○ Horizontal and Vertical mirroring.
	• Graphics
	  ○ Number of graphic layers: Four layers
	  ○ Supported pixel formats:
	    ■ RGB565 format (A(α): none, R: 5 bits, G: 6 bits, B: 5 bits; 16 bits in total)
	    ■ RGB888 format (A(α): none, R: 8 bits, G: 8 bits, B: 8 bits; 24 bits in total)
	    ■ ARGB1555 format (A(α): 1 bit, R: 5 bits, G: 5 bits, B: 5 bits; 16 bits in total)
	    ■ ARGB4444 format (A(α): 4 bits, R: 4 bits, G: 4 bits, B: 4 bits; 16 bits in total)
	    ■ ARGB8888 format (A(α): 8 bits, R: 8 bits, G: 8 bits, B: 8 bits; 32 bits in total)
	    ■ RGBA5551 format (R: 5 bits, G: 5 bits, B: 5 bits, A(α): 1 bit; 16 bits in total)
	    ■ RGBA8888 format (R: 8 bits, G: 8 bits, B: 8 bits, A(α): 8 bits; 32 bits in total)
	    ■ CLUT8 format (CLUT: 8 bits)
	    ■ CLUT4 format (CLUT: 4 bits)
	    ■ CLUT1 format (CLUT: 1 bit)
	    ■ YCbCr422 format (Y: 8 bits, Cb/Cr: 8 bits; 16 bits in total)
	    ■ YCbCr444 format (Y: 8 bits, Cb/Cr: 8 bits; 16 bits in total)
	• Maximum input graphic size to be set: 1280 pixels x 1024 lines (horizontal x vertical).
	• Graphics function
	  ○ Alpha blending in rectangular area: Mixes images according to transparency rate alpha in the entire layer (fade-in and fade-out functions are available.)
	  ○ Chroma-key: Mixes images using the specified RGB color and CLUT value according to transparency rate alpha.
	  ○ Alpha blending in one-pixel units: Mixes images according to transparency rate alpha when the target graphics image is in the ARGB1555, ARGB4444, ARGB8888, RGBA5551, RGBA8888, or CLUT8/4/1 format.
	• Graphics image scaling processing
	  ○ Vertical: x1 to x8, linear/hold interpolation
	  ○ Horizontal: x1 to x8, linear/hold interpolation
	• Graphics image rotation function
	  ○ Vertical mirroring.

Renesas Graphics Library Video Data Controller E (VDCE) Driver

	• Output video image size		
	o Maximum output video image size to be set: 1280 pixels x 1024 lines (horizontal x vertical)		
	• Output video image format		
	o RGB888 progressive video output (24-bit parallel output)		
	o RGB666 progressive video output (18-bit parallel output)		
	o RGB565 progressive video output (16-bit parallel output)		
	o Serial RGB progressive output (8-bit serial output)		
	• Panel output adjustment		
	o Panel brightness/contrast adjustment, output format conversion		
Semaphore / Mutex	N/A for VDCE. This can be implemented with porting layer.		
Interrupts	Used in the VDCE driver. For more details please see *section 2.3.*		

2.2 Reserved Word

VDCE driver uses the following prefixes for avoiding confusion from other software. Prefixes of VDCE is described in *Table 2-2*.

Table 2-2 Prefixes

Prefix	Description
R_VDCE_*	Prefix for VDCE Module
r_vdce_*	

2.3 Interrupt Handler List

Table 2-3 Interrupt Handler List

No.	Interrupt Name	Interrupt Handler Name	Description
Unit 0			
(1)	INTVDCE0S0LOVSYNC	R_VDCE_Isr	VSYNC output at Scaler 0 interrupt.
(2)	INTVDCE0S1LOVSYNC	R_VDCE_Isr	VSYNC output at Scaler 1 interrupt.
(3)	INTVDCE0GR3VBLANK	R_VDCE_Isr	VBLANK detection at Graphics 3.
(4)	INTVDCE0GR3VLINE	R_VDCE_Isr	Scan Line detection of designated line at Graphics 3 interrupt.
(5)	INTVDCE0OIRVLINE	R_VDCE_Isr	Scan Line detection of designated line at Output Image Render Interrupt.
(6)	INTVDCE0OIRLOVSYNC	R_VDCE_Isr	VSYNC output at Output Image Render interrupt.
(7)	INTVDCE0OIRVIVSYNC	R_VDCE_Isr	VSYNC input at Output Image Render interrupt.
(8)	INTVDCE0S0VIVSYNC	R_VDCE_Isr	VSYNC input at Scaler 0 interrupt.
(9)	INTVDCE0S0VFIELD	R_VDCE_Isr	End of field for record function at Scaler 0 interrupt.
(10)	INTVDCE0S0LOVSYNC	R_VDCE_Isr	VSYNC output at Scaler 0 interrupt.
(11)	INTVDCE0ERR	R_VDCE_IsrError	Error interrupt.
Unit 1			
(12)	INTVDCE1S0LOVSYNC	R_VDCE_Isr	VSYNC output at Scaler 0 interrupt.
(13)	INTVDCE1S1LOVSYNC	R_VDCE_Isr	VSYNC output at Scaler 1 interrupt.
(14)	INTVDCE1GR3VBLANK	R_VDCE_Isr	VBLANK detection at Graphics 3.
(15)	INTVDCE1GR3VLINE	R_VDCE_Isr	Scan Line detection of designated line at Graphics 3 interrupt.
(16)	INTVDCE1S0VIVSYNC	R_VDCE_Isr	VSYNC input at Scaler 0 interrupt.
(17)	INTVDCE1S0VFIELD	R_VDCE_Isr	End of field for record function at Scaler 0 interrupt.
(18)	INTVDCE1S0LOVSYNC	R_VDCE_Isr	VSYNC output at Scaler 0 interrupt.
(19)	INTVDCE1ERR	R_VDCE_IsrError	Error interrupt.

2.4 Error Handling

2.4.1 Return code

VDCE driver returns 5 types of error codes.

2.4.1.1 Parameter level

Following errors occur by a cause such as abnormality of parameter. In this case, please set valid parameter again.
- R_VDCE_ERR_PARAM_INCORRECT
- R_VDCE_ERR_RANGE_UNIT
- R_VDCE_ERR_RANGE_PARAM
- R_VDCE_ERR_RANGE_LAYER
- R_VDCE_ERR_PIXEL_CLOCK
- R_VDCE_ERR_STRIDE_INCORRECT
- R_VDCE_ERR_CHROMAKEY_INCORRECT
- R_VDCE_ERR_SCALED_SIZE_INCORRECT

2.4.1.2 Timing level

Following errors occur by a cause such as abnormality of execution timing. In this case, please call again after changing to valid state or timing.
- R_VDCE_ERR_NOT_ACCEPTABLE
- R_VDCE_ERR_UNIT_LOCKED
- R_VDCE_ERR_UNIT_NOTLOCKED
- R_VDCE_ERR_DISPLAY_NO_TIMING
- R_VDCE_ERR_LAYER_NO_FORMAT
- R_VDCE_ERR_LAYER_NO_MEM_GEOMETRY
- R_VDCE_ERR_LAYER_NO_VIEW_PORT
- R_VDCE_ERR_LAYER_NO_BASE
- R_VDCE_ERR_CAP_NO_BUF_GEOMETRY
- R_VDCE_ERR_CAP_NO_MODE
- R_VDCE_ERR_INTERRUPT_ENABLED

2.4.1.3 Hardware level

Following errors occur when unexpected error occurs internally. In this case, please reset the RH850/D1x device.
- R_VDCE_ERR_NG
- R_VDCE_ERR_FATAL_HW

2.4.1.4 System level

Following errors occur by a cause such as OS dependent error (e.g. system call error, resource shortage). In this case, please do recovery processing from a system layer, because this status cannot be restored only in this library.
- R_VDCE_ERR_FATAL_OS

2.4.1.5 Device level

Following errors occur when the function is not supported with target device. In this case, please skip the function call.
- R_VDCE_ERR_NOT_SUPPORTED

2.4.2   Callback

VDCE driver notifies H/W error event.

2.4.2.1 Missing V-sync

These events occur when external input Vsync signal is missing.
- R_VDCE_INTC_NO_VI_VSYNC_SIGNAL
- R_VDCE_INTC_NO_VI_VSYNC_SIGNAL_1

These may happen occasionally if the input clock and the RH850/D1x device clock are not completely synchronized.
In this case, please ignore the event.

2.4.2.2 Overflow

These events occur when H/W processing is not in time (e.g. Input or Output pixel clock is too high.).
- R_VDCE_INTC_ERR_OIR_UNDERFLOW
- R_VDCE_INTC_ERR_LAYER0_UNDERFLOW
- R_VDCE_INTC_ERR_LAYER1_UNDERFLOW
- R_VDCE_INTC_ERR_LAYER2_UNDERFLOW
- R_VDCE_INTC_ERR_LAYER3_UNDERFLOW
- R_VDCE_INTC_ERR_CAP_WRITE_OVERFLOW

Even under normal conditions, these events may occur immediately after R_VDCE_LayerEnable,
R_VDCE_LayerDisable, R_VDCE_OirEnable, R_VDCE_OirDisable, R_VDCE_CapEnable and R_VDCE_CapDisable.
If it does not occur regularly during normal operation, there is a high possibility that there is no problem. In this case,
please ignore the event.
If it occurs regularly during normal operation, video output signal will be noisy. In this case, please reset the RH850/D1x
device.

2.5   State Transition

Each VDCE unit has following status.

Table 2-4 State Details

No.	State Name	Description
(1)	Uninitialized	Specifies that the VDCE driver is not initialized.
(2)	Initialized	Specifies that the VDCE driver is initialized.
(3)	Idle	Specifies that display enabled.
(4)	Executing	Specifies that layer, OIR or capturing is enabled.

The image describes state transition.
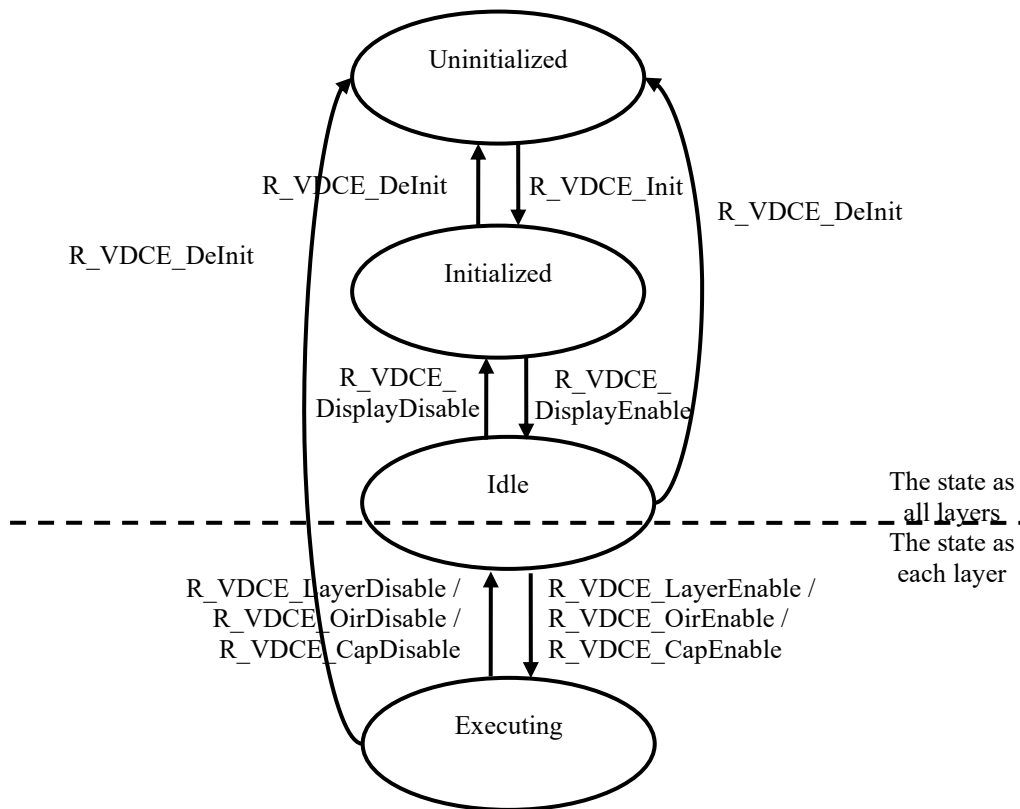
Figure 2-1 State Transition Diagram of VDCE driver

The state of Uninitialized and Initialized are in the common state with all layers. It is possible to transfer to Initialized when all layers are Idle.

Table 2-5 State Transition Table of VDCE driver

Function Name	State			
	Uninitialized	**Initialized**	**Idle**	**Executing**
R_VDCE_Init	OK	NG	NG	NG
R_VDCE_DeInit	OK *2	OK	OK	OK *1
R_VDCE_VersionStringGet	OK	OK	OK	OK
R_VDCE_MacroVersionGet	OK	OK	OK	OK
R_VDCE_ErrorCallbackSet	NG	OK	OK	OK
R_VDCE_IntcCallbackSet	NG	OK	OK	OK
R_VDCE_IntcCallbackGet	NG	OK	OK	OK
R_VDCE_IntcEnable	NG	OK	OK	OK
R_VDCE_IntcDisable	NG	OK	OK	OK
R_VDCE_IntcScanlineSet	NG	OK	OK	OK
R_VDCE_IntcScanlineGet	NG	OK	OK	OK
R_VDCE_IntcOirScanlineSet	NG	OK	OK	OK
R_VDCE_IntcOirScanlineGet	NG	OK	OK	OK
R_VDCE_IntcCapScanlineSet	NG *9	NG *9	NG *9	NG *9
R_VDCE_IntcCapScanlineGet	NG	OK	OK	OK
R_VDCE_DisplayTimingSet	NG	OK *3	NG	NG
R_VDCE_DisplayTimingAdjust	NG	OK *3	OK *5	NG
R_VDCE_DisplayOutEndianSet	NG	OK *3	NG	NG
R_VDCE_DisplayOutSwapBR	NG	OK *3	NG	NG
R_VDCE_DisplayColorSet	NG	OK *3	OK *5	OK *5
R_VDCE_DisplayHsyncSet	NG	OK *3	NG	NG
R_VDCE_DisplayHsyncGet	NG	OK	OK	OK
R_VDCE_DisplaySignalSet	NG	OK *3	NG	NG
R_VDCE_DisplaySignalGet	NG	OK	OK	OK
R_VDCE_DisplayTconPinSet	NG	OK *3	NG	NG

R_VDCE_DisplayTconPinGet	NG	OK	OK	OK
R_VDCE_DisplayPolaritySet	NG	OK *3	NG	NG
R_VDCE_DisplayCalibrationSet	NG	OK *5	OK *5	OK *5
R_VDCE_DisplayGammaCorrectSet	NG	OK *5	OK *5	OK *5
R_VDCE_DisplayOutFormatSet	NG	OK *3	NG	NG
R_VDCE_DisplaySerialRGBSet	NG	OK *3	NG	NG
R_VDCE_DisplayVsyncProtectionSet	NG	OK *3	NG	NG
R_VDCE_DisplayEnable	NG	OK	OK *2	OK *2
R_VDCE_DisplayDisable	OK *2	OK *2	OK	NG
R_VDCE_DisplayTimingGet	NG	OK	OK	OK
R_VDCE_LayerBaseSet	NG	OK *4	OK *4	OK *5
R_VDCE_OirBaseSet	NG	OK *4	OK *4	OK *5
R_VDCE_LayerMemGeometrySet	NG	OK *4	OK *4	OK *4
R_VDCE_OirMemGeometrySet	NG	OK *4	OK *4	OK *4
R_VDCE_LayerFormatSet	NG	OK *4	OK *4	OK *5
R_VDCE_OirFormatSet	NG	OK *4	OK *4	OK *5
R_VDCE_LayerViewPortSet	NG	OK *4	OK *4	OK *5
R_VDCE_OirViewPortSet	NG	OK *4	OK *4	OK *5
R_VDCE_LayerRingBufferEnable	NG	OK *4	OK *4	OK *4
R_VDCE_OirRingBufferDisable	NG	OK *4	OK *4	OK *4
R_VDCE_LayerRingBufferDisable	NG	OK *4	OK *4	OK *4
R_VDCE_OirRingBufferEnable	NG	OK *4	OK *4	OK *4
R_VDCE_LayerVSyncDelaySet	NG	OK *4	OK *4	OK *4
R_VDCE_OirVSyncDelaySet	NG	OK *4	OK *4	OK *4
R_VDCE_LayerModeSet	NG	OK *4	OK *4	OK *4
R_VDCE_OirModeSet	NG	OK *4	OK *4	NG
R_VDCE_LayerEnable	NG	NG	OK	OK *2
R_VDCE_OirEnable	NG	NG	OK	OK *2

R_VDCE_LayerDisable	OK *2	OK *2	NG	OK
R_VDCE_OirDisable	OK *2	OK *2	NG	OK
R_VDCE_LayerMatrixSet	NG	OK *5	OK *5	OK *5
R_VDCE_LayerMatrixBT601Set	NG	OK *5	OK *5	OK *5
R_VDCE_LayerMatrixJPEGSet	NG	OK *5	OK *5	OK *5
R_VDCE_LayerMatrixUnitySet	NG	OK *5	OK *5	OK *5
R_VDCE_LayerImgScaleX	NG	OK *4	OK *4	OK *5
R_VDCE_LayerImgScaleY	NG	OK *4	OK *4	OK *5
R_VDCE_LayerImgScaleModeSet	NG	OK *7	OK *7	OK *8
R_VDCE_LayerBufSet	NG	OK *4	OK *4	OK *5
R_VDCE_LayerAlphaChannelEnable	NG	OK *4	OK *4	OK *4
R_VDCE_LayerAlphaChannelDisable	NG	OK *4	OK *4	OK *4
R_VDCE_LayerPremultipliedAlphaEnable	NG	OK *4	OK *4	OK *4
R_VDCE_LayerPremultipliedAlphaDisable	NG	OK *4	OK *4	OK *4
R_VDCE_LayerAlphaConstEnable	NG	OK *4	OK *4	OK *5
R_VDCE_LayerAlphaConstDisable	NG	OK *4	OK *4	OK *4
R_VDCE_LayerClutSet	NG	OK *5	OK *5	OK *5
R_VDCE_LayerChromaKeyEnable	NG	OK *4	OK *4	OK *5
R_VDCE_LayerChromaKeyDisable	NG	OK *4	OK *4	OK *4
R_VDCE_CapBufGeometrySetup	NG	OK *6	OK *6	OK *6
R_VDCE_CapModeSet	NG	OK *6	OK *6	OK *6
R_VDCE_CapBufSet	NG	OK *6	OK *6	OK *5
R_VDCE_CapBufFieldSetup1	NG	OK *6	OK *6	OK *5
R_VDCE_CapBufFieldSetup2	NG	OK *6	OK *6	OK *5
R_VDCE_CapEnable	NG	NG *10	OK	OK *2
R_VDCE_CapDisable	NG	NG	OK *2	OK
R_VDCE_CapViewPortSet	NG	OK	OK	OK
R_VDCE_CapRateSet	NG	OK	OK	NG

R_VDCE_CapExtVsyncSet	NG	OK	OK	NG

*1 If each layer is Enable, it makes Disable.

*2 In this state, do nothing.

*3 The parameter is effective when executing R_VDCE_DisplayEnable.

*4 The parameter is effective when executing R_VDCE_LayerEnable / R_VDCE_OirEnable.

*5 The parameter is immediately set.

*6 The parameter is effective when executing R_VDCE_CapEnable.

*7 The parameter is effective when executing R_VDCE_LayerEnable (if scale is enlarged) / R_VDCE_CapEnable (if scale is reduced).

*8 The parameter is set when executing R_VDCE_LayerEnable (if scale is enlarged) / R_VDCE_CapEnable (if scale is reduced) / R_VDCE_LayerImgScaleX (if target layer is already enabled.)/ R_VDCE_LayerImgScaleY (if target layer is already enabled.).

*9 This function returns error regardless of the status.

*10 It is OK when R_VDCE_CAP_NO_DISPLAY is specified.

3.Function Description

3.1 Fundamental Concepts

3.1.1 VDCE unit

RH850/D1x device has the following number of units of the VDCE.

Table 3-1 Number of units

Feature	RH850/D1x Device Name				
	D1M2H	D1M2	D1M1A	D1M1(H) D1M1-V2	D1L2(H)
Number of Units	2 (Unit0,1)	2 (Unit0,1)	2 (Unit0,1)	1 (Unit0)	1 (Unit0)
Number of Capture units	2 (Unit0,1)	1 (Unit1)	1 (Unit0)	1 (Unit0)	0

Almost VDCE API functions have the argument "Unit".
User specifies the VDCE H/W unit number to be controlled. The range is 0 to 1.

3.1.2 Layer and Image Synthesizer

Image synthesizer acquire data from memory and generates a synthesized image to be displayed. There are four image synthesizers (Scaler0, Scaler1, Image Synthesizer2, Image Synthesizer3).

Layer is a hardware concept of the stackable visual area on the display. Layer can be either Video image layer (Layer consists of input video) or graphics layer (Layer consists of graphics elements like boxes, lines, texts etc.).

Table 3-2 Relationship of Layer and Image synthesizer

Layer	Image Synthesizer
Layer 0	Scaler 0
Layer 1	Scaler 1
Layer 2	Image synthesizer 2
Layer 3	Image synthesizer 3

The Layer 0 is the lowest layer. The Layer 3 is the highest layer.
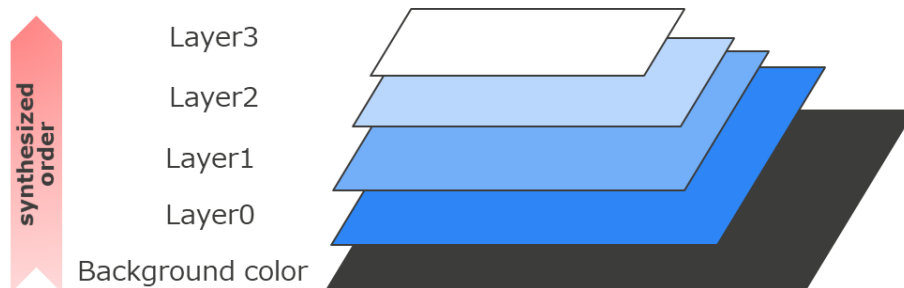
Figure 3-1 Layer order

The result by the image synthesis of each layer becomes like a figure below.
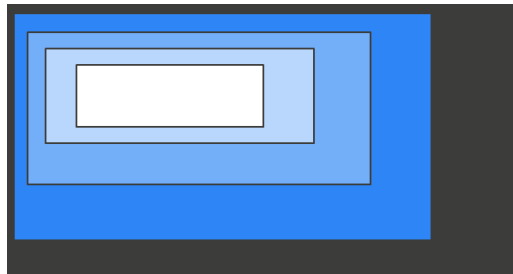
Figure 3-2 Synthesis result

3.1.3 Outline of operation

This section describes the data flow within the VDCE driver.

It can be selected for synthesis.
- Four graphics layers
- One video layer + three graphics layers
- Two video layers + two graphics layers

3.1.3.1 Graphics layer x 4

The following diagram shows a data flow diagram, in which graphics are input to each of Scaler0, Scaler1, ImageSynthesizer2 and ImageSynthesizer3 and then these 4 images are synthesized to output. The image format in the diagram indicates the data format that can be captured into each frame buffer, and application can specify the selection of data format via the VDCE Driver. Image quality improving block includes color matrix, and it can transform the image format.
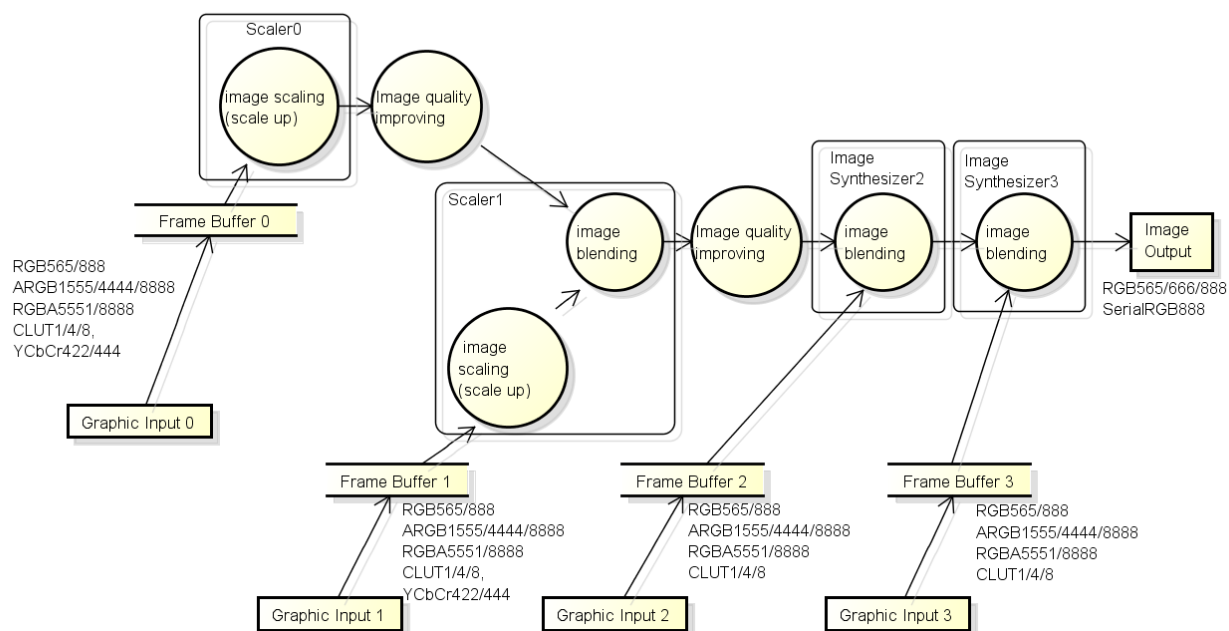
Figure 3-3 Data Flow Diagram (Graphics layer x 4)

3.1.3.2 Video layer x 1 + Graphics layer x 3

The following figure shows a data flow diagram, in which a video is input to Scaler0 and graphics are input to each of Scaler1, ImageSynthesizer2 and ImageSynthesizer3 and then these 4 images are synthesized to output. The image format in the diagram indicates the data format that can be captured into each frame buffer and user can specify the selection of data format via the VDCE API. Video Input block and image quality improving block include color matrix and they can transform the image format.
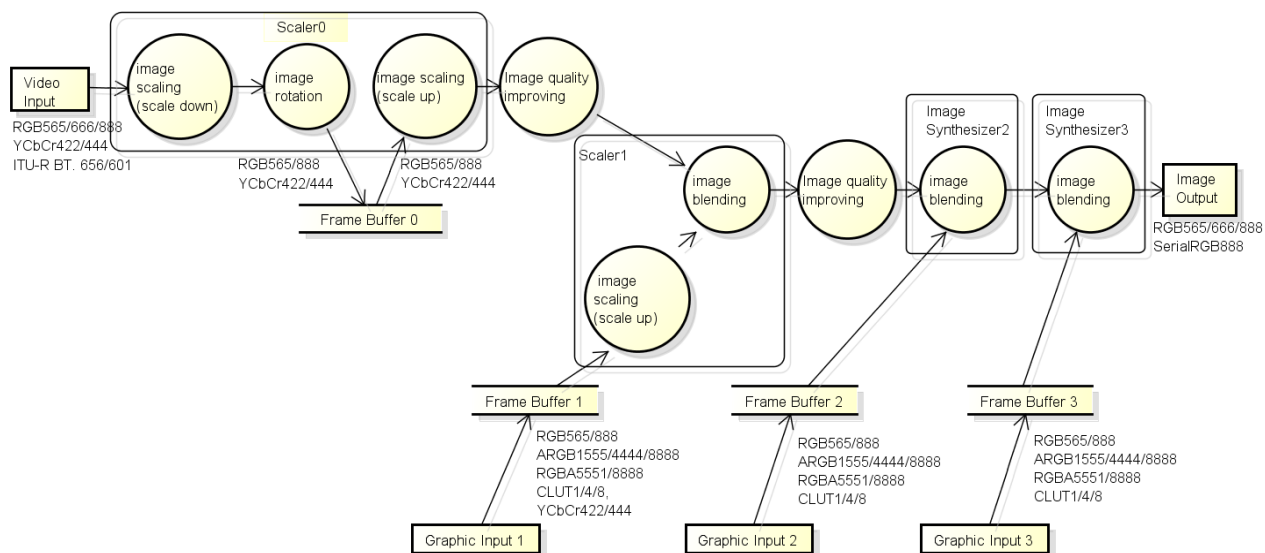
Figure 3-4 Data Flow Diagram (Video Image layer x 1 + Graphics layer x 3)

3.1.3.3 Video layer x 1 (OIR)

The following figure shows a data flow diagram, in which a video image is input to Scaler0 and a distortion correction of the output image is performed to output.
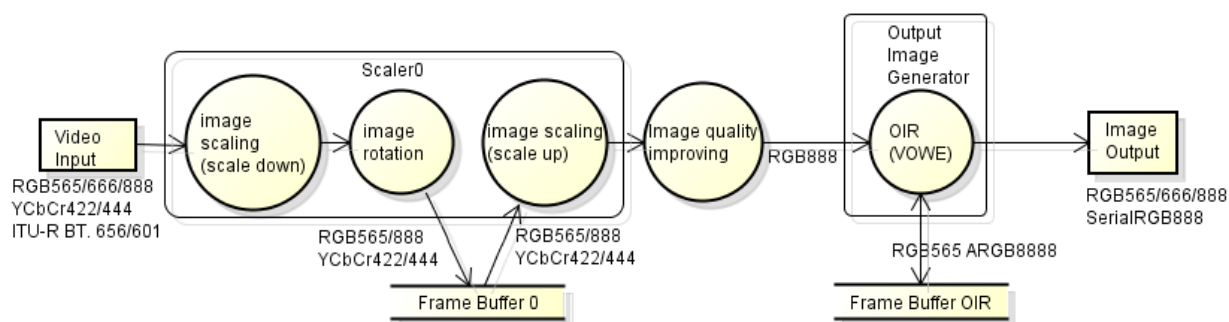
Figure 3-5 Data Flow Diagram (Video Image layer x 1 (OIR)

3.1.4   Processing timing

Many of the setting information items by VDCE API are reflected to H/W at a timing of rise of the vertical synchronizing signal (Vsync). At that time, note that it takes one cycle time of the vertical synchronizing signal at the maximum from the value setting of the H/W to the reflection of the setting. For detail, refer to VDCE H/W specification. The following diagram shows an example of timing chart in which the start address of Frame Buffer of Scaler0 is modified in the executing period.
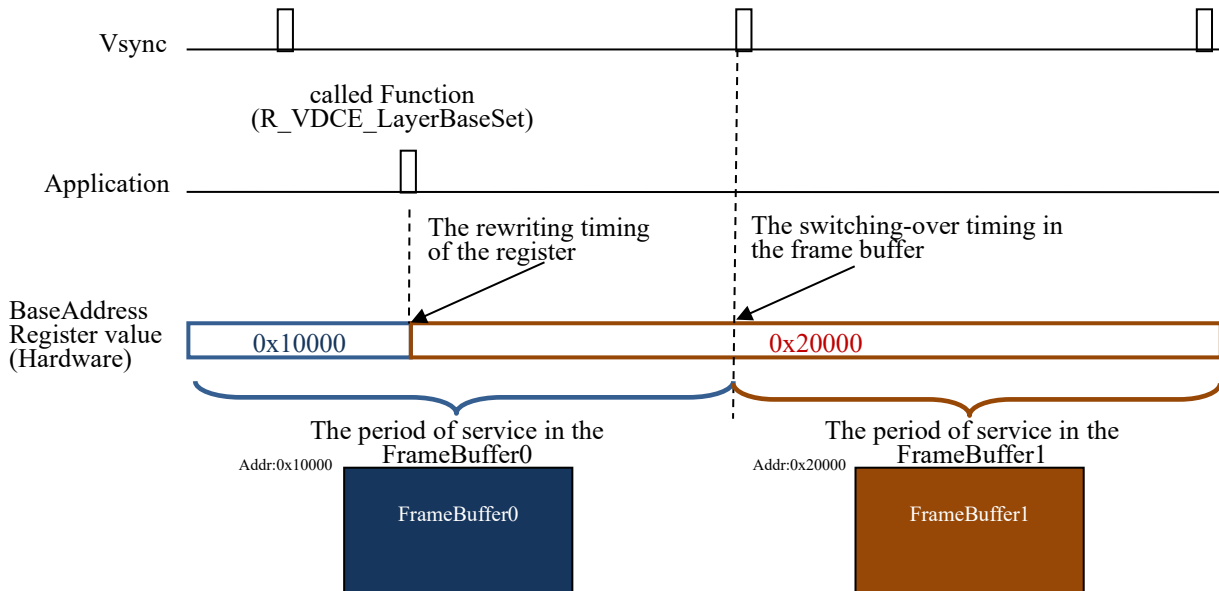
Figure 3-6 Timing chart

3.2 Using the API

3.2.1 Initialization / De-Initialization

R_VDCE_Init initializes the driver and the hardware as far as necessary. The Unit parameter holds a number that specifies the VDCE unit number being initialized. This driver ensures that macro is set to default configuration. R_VDCE_DeInit function de-initializes the driver and the hardware as far as necessary. The function disables all layers and the display.

3.2.2 Display

3.2.2.1 Display area

R_VDCE_DisplayTimingSet sets the display area and output signals.
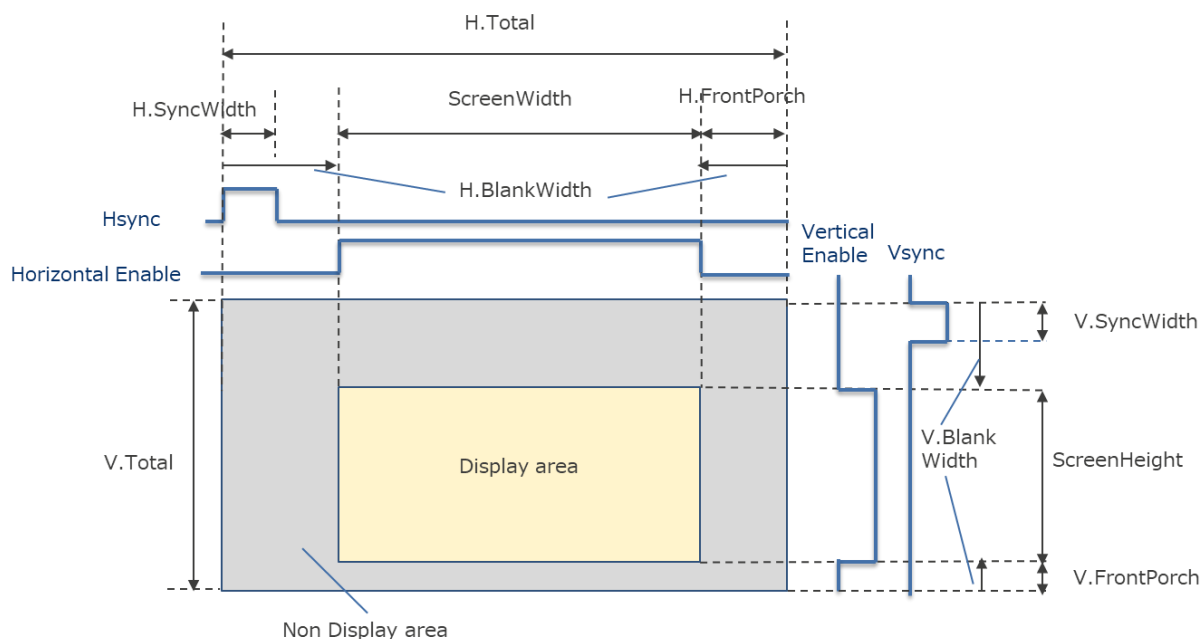
Figure 3-7 Display signal image

Table 3-3 Parameter range

Timing parameter	Range	
	Min	Max
Timing->ScreenWidth	16	1280 / 1024 / 480 (*2)
Timing->ScreenHeight	16	1024 / 320 (*2)
Timing->H.Total	32	2048
Timing->H.BlankWidth	16	2032
Timing->H.SyncWidth	1	Timing->H.BlankWidth-1.
Timing->H.FrontPorch	0	Timing->H.BlankWidth-2.
Horizontal back porch (*1)	1	Timing->H.BlankWidth-1
Timing->V.Total	21	2048
Timing->V.BlankWidth	5	2032
Timing->V.SyncWidth	1	Timing->V.BlankWidth-2
Timing->V.FrontPorch	1	Timing->V.BlankWidth-2
Vertical back porch (*1)	1	Timing->V.BlankWidth-2
Timing->PixelClock	1	- (*3)

(*1): back porch = BlankWidth – SyncWidth – FrontPorch
(*2): maximum value is depending on RH850/D1x device.
(*3): maximum pixel clock is depending on RH850/D1x device and output format (LVTTL, Serial RGB etc).

3.2.2.2 TCON signal

VDCE outputs 7 TCON signals. Following table shows the default assignment.

Table 3-4 TCON Pin assignment

VDCE output signal	Default assigned signal
LCD_TCON0	Vsync signal
LCD_TCON1	Vertical enable signal
LCD_TCON2	Hsync signal
LCD_TCON3	Data Enable signal
LCD_TCON4	-
LCD_TCON5	-
LCD_TCON6	-

Data Enable (DE) signal is active when both Horizontal enable signal and Vertical enable signal are active.
R_VDCE_DisplaySignalSet can change the signal configuration.
R_VDCE_DisplayTconPinSet can change the assignment of signal and output pin.

3.2.2.3 Output color data

VDCE outputs 24 color data signals (LCD_DATA23..00). Data signal assignment is depending on following settings.

- OutFormat which is set by R_VDCE_DisplayOutFormatSet
- OutEndian which is set by R_VDCE_DisplayOutEndianSet
- OutSwap which is set by R_VDCE_DisplayOutSwapBR

Table 3-5 Output color data

OutFormat	OutEndian	OutSwap	LCD_DATA23 <---> LCD_DATA00
RGB888	Little	Off	$R_7R_6R_5R_4R_3R_2R_1R_0$ $G_7G_6G_5G_4G_3G_2G_1G_0$ $B_7B_6B_5B_4B_3B_2B_1B_0$
		On	$B_7B_6B_5B_4B_3B_2B_1B_0$ $G_7G_6G_5G_4G_3G_2G_1G_0$ $R_7R_6R_5R_4R_3R_2R_1R_0$
	Big	Off	$R_0R_1R_2R_3R_4R_5R_6R_7$ $G_0G_1G_2G_3G_4G_5G_6G_7$ $B_0B_1B_2B_3B_4B_5B_6B_7$
		On	$B_0B_1B_2B_3B_4B_5B_6B_7$ $G_0G_1G_2G_3G_4G_5G_6G_7$ $R_0R_1R_2R_3R_4R_5R_6R_7$
RGB666	Little	Off	0 0 0 0 0 0 R_7R_6 $R_5R_4R_3R_2G_7G_6G_5G_4$ $G_3G_2B_7B_6B_5B_4B_3B_2$
		On	0 0 0 0 0 0 B_7B_6 $B_5B_4B_3B_2G_7G_6G_5G_4$ $G_3G_2R_7R_6R_5R_4R_3R_2$
	Big	Off	0 0 0 0 0 0 R_2R_3 $R_4R_5R_6R_7G_2G_3G_4G_5$ $G_6G_7B_2B_3B_4B_5B_6B_7$
		On	0 0 0 0 0 0 B_2B_3 $R_4R_5R_6R_7G_2G_3G_4G_5$ $G_6G_7B_2B_3B_4B_5B_6B_7$
RGB565	Little	Off	0 0 0 0 0 0 0 0 $R_7R_6R_5R_4R_3G_7G_6G_5$ $G_4G_3G_2B_7B_6B_5B_4B_3$
		On	0 0 0 0 0 0 0 0 $B_7B_6B_5B_4B_3G_7G_6G_5$ $G_4G_3G_2R_7R_6R_5R_4R_3$
	Big	Off	0 0 0 0 0 0 0 0 $R_3R_4R_5R_6R_7G_2G_3G_4$ $G_5G_6G_7B_3B_4B_5B_6B_7$
		On	0 0 0 0 0 0 0 0 $B_3B_4B_5B_6B_7G_2G_3G_4$ $G_5G_6G_7R_3R_4R_5R_6R_7$
Serial RGB	Little	-	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 $S_7S_6S_5S_4S_3S_2S_1S_0$
	Big	-	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 $S_0S_1S_2S_3S_4S_5S_6S_7$

R_n, G_n and B_n (n=0~7) are RGB888 data in VDCE internal. For example, R=128 (0x80): R_7=1b, R_6 = 0b, ... R_0 =0b.
In case of Serial RGB format, S_n consists of serialized R_n, G_n and B_n data. The order of serialization is depending on following settings.

- Freq, Scan and Swap which are set by R_VDCE_DisplaySerialRGBSet
- OutSwap which is set by R_VDCE_DisplayOutSwapBR

Table 3-6 Serial RGB output data

Freq	Scan	Swap	OutSwap	Odd Line	Even Line
3	Forward	On	Off	R -> G -> B	G -> B -> R
			On	B -> G -> R	G -> R -> B
		Off	Off	R -> G -> B	R -> G -> B
			On	B -> G -> R	B -> G -> R
	Reverse	On	Off	B -> G -> R	R -> B -> G
			On	R -> G -> B	B -> R -> G
		Off	Off	B -> G -> R	B -> G -> R
			On	R -> G -> B	R -> G -> B
4	Forward	-	Off	R -> G -> B -> X	R -> G -> B -> X
		-	On	B -> G -> R -> X	B -> G -> R -> X
	Reverse	-	Off	X -> B -> G -> R	X -> B -> G -> R
		-	On	X -> R -> G -> B	X -> R -> G -> B

X: undefined data.

3.2.2.4 Pixel Clock

VDCE output the LCD_CLK is generated based on PixelClock that is the argument of R_VDCE_DisplayTimingSet.
Pixel clock should be calculated by following formula.

Table 3-7 LCD_CLK calculation

Parameter		Formula
OutFormat	Freq	
RGB888 RGB666 RGB565	-	Timing->PixelClock = Timing->H.Total * Timing->V.Total * FPS
Serial RGB	3	Timing->PixelClock = Timing->H.Total * Timing->V.Total * FPS * 3
	4	Timing->PixelClock = Timing->H.Total * Timing->V.Total * FPS * 4

FPS is frames per second. FPS = 60 in case of 60fps, FPS = 30 in case of 30fps.
Maximum pixel clock is depending on RH850/D1x device. See Section 1 Overview of H/W user's manual.

3.2.2.5 Output data correction

Output color data can be corrected with following sequence.
- Brightness adjustment
- Contrast adjustment
- Gamma correction
- Dither process

These controls are set by R_VDCE_DisplayCalibrationSet and R_VDCE_DisplayGammaCorrectSet.
These functions show the H/W register specification as it is. Refer to H/W user's manual for the detail.

Table 3-8 Output adjustment

Function	Parameter	H/W register bit name
R_VDCE_Display CalibrationSet	Bright->B	PBRT_B
	Bright->G	PBRT_G
	Bright->R	PBRT_R
	Contrast->B	CONT_B
	Contrast->G	CONT_G
	Contrast->R	CONT_R
	Dither->Sel	PDTH_SEL
	Dither->Pa	PDTH_PA
	Dither->Pb	PDTH_PB
	Dither->Pc	PDTH_PC
	Dither->Pd	PDTH_PD
R_VDCE_Display GammaCorrectSet	Gamma->B->Area[n] (n = 1~31)	GAM_B_TH_01 to GAM_B_TH_31
	Gamma->B->Gain[n] (n = 0~31)	GAM_B_GAIN_00 to GAM_B_GAIN_31
	Gamma->G->Area[n] (n = 1~31)	GAM_G_TH_01 to GAM_G_TH_31
	Gamma->G->Gain[n] (n = 0~31)	GAM_G_GAIN_00 to GAM_G_GAIN_31
	Gamma->R->Area[n] (n = 1~31)	GAM_R_TH_01 to GAM_R_TH_31
	Gamma->R->Gain[n] (n = 0~31)	GAM_R_GAIN_00 to GAM_R_GAIN_31

3.2.3 Image synthesizer

3.2.3.1 Layer basic setting

The frame buffer start address is set by Address which is parameter of R_VDCE_LayerBaseSet. The start address should be 128 byte-aligned.

Frame buffer size and display position are set by R_VDCE_LayerMemGeometrySet and R_VDCE_LayerViewPortSet. The stride of the memory should be 128 byte-aligned. Thus, if color format is 32bpp, the parameter "Stride" should be 32 pixel-aligned. Following table shows the range information.

If the setting value is out of range, the function returns the error code.

Table 3-9 Parameter range

Function	Parameter	Setting unit	Range		Alignment
			Min	Max	
R_VDCE_Layer BaseSet	Address	Byte	-	-	128 Byte
R_VDCE_Layer MemGeometrySet	Stride	Pixel	1024 / bpp	261120 / bpp	128 Byte
	MemHeight	Pixel	1	4096	-
R_VDCE_Layer ViewPortSet	DispPosX	Pixel	-1280	1279	-
	DispPosY	Pixel	-1024	1023	-
	DispWidth	Pixel	3	1280	-
	DispHeight	Pixel	1	1024	-

* ScreenWidth and ScreenHeight are the parameter of R_VDCE_DisplayTimingSet.

* bpp is depending on color format set by R_VDCE_LayerFormatSet.
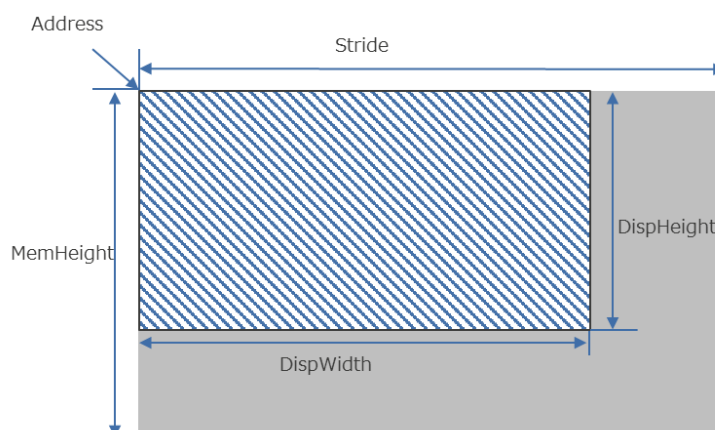
Figure 3-8 Memory image (without scale-up)
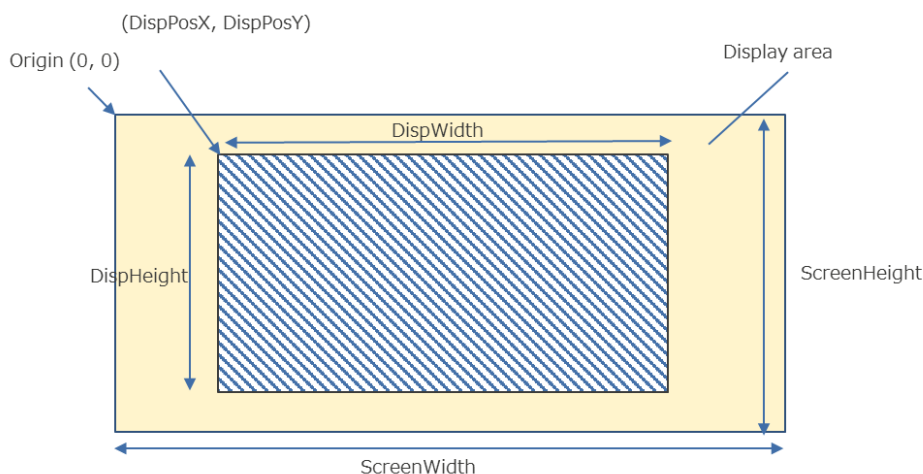
Figure 3-9 Display image of the layer

Renesas Graphics Library Video Data Controller E (VDCE) Driver

DispPosX and DispPosY also have the following range.
If the setting value is smaller than the minimum value, the VDCE driver operates as if the minimum value is set.
If the setting value is smaller than the maximum value, the VDCE driver operates as if the maximum value is set.

Table 3-10 Parameter range (2)

Parameter	Setting unit	Range	
		Min	Max
DispPosX	Pixel	3 - DispWidth	ScreenWidth - 3
DispPosY	Pixel	1 - DispHeight	ScreenHeight - 1

Following figures show the display image of the minimum and maximum values of DispPosX and DispPosY.
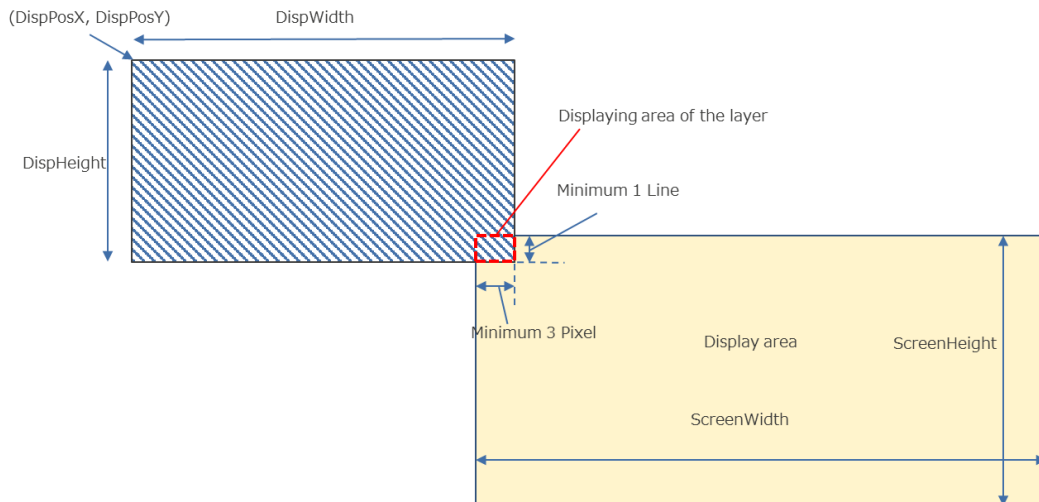
Figure 3-10 Viewport Image minimum position
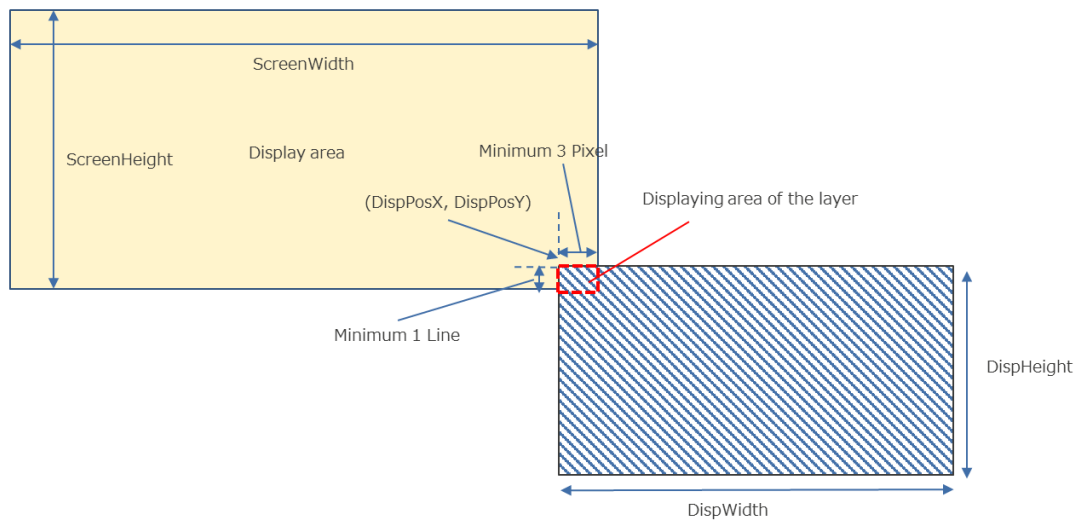
Figure 3-11 Viewport Image maximum position

3.2.3.2 Layer color format

Color format of the input data to the layer is specified by R_VDCE_LayerFormatSet. VDCE driver supports following color format.

Table 3-11 Color Format

Layer color format	bpp	Memory image (MSB <--> LSB)	remarks
R_VDCE_RGB565	16	RRRRRGGG GGGBBBBB	
R_VDCE_RGB0888	32	xxxxxxxx RRRRRRRR GGGGGGGG BBBBBBBB	
R_VDCE_ARGB1555	16	ARRRRRGG GGGBBBBB	
R_VDCE_ARGB4444	16	AAAARRRR GGGGBBBB	
R_VDCE_ARGB8888	32	AAAAAAAA RRRRRRRR GGGGGGGG BBBBBBBB	
R_VDCE_RGBA5551	16	RRRRRGGG GGBBBBBA	
R_VDCE_RGBA8888	32	RRRRRRRR GGGGGGGG BBBBBBBB AAAAAAAA	
R_VDCE_CLUT8	8	TTTTTTTT	
R_VDCE_CLUT4	4	TTTT	
R_VDCE_CLUT1	1	T	
R_VDCE_YCBCR_422	16	UUUUUUUU YYYYYYYY VVVVVVVV YYYYYYYY	32bits / 2Pixels
R_VDCE_YCBCR_444	32	xxxxxxxx VVVVVVVV YYYYYYYY UUUUUUUU	
R_VDCE_YUV_YUYV	16	VVVVVVVV YYYYYYYY UUUUUUUU YYYYYYYY	32bits / 2Pixels
R_VDCE_YUV_UYVY	16	YYYYYYYY VVVVVVVV YYYYYYYY UUUUUUUU	32bits / 2Pixels
R_VDCE_YUV_YVYU	16	UUUUUUUU YYYYYYYY VVVVVVVV YYYYYYYY	32bits / 2Pixels
R_VDCE_YUV_VYUY	16	YYYYYYYY UUUUUUUU YYYYYYYY VVVVVVVV	32bits / 2Pixels

A/R/G/B: each component of ARGB, x: unused, T: table index,
Y: Y component, U: Cb or U component, V: Cr or V component

YCbCr/YUV format is supported only Scaler 0 and Scaler 1.

Table 3-12 Color Format with Layer

Layer color format	Scaler 0	Scaler 1	Image Synthesizer 2	Image Synthesizer 3
R_VDCE_RGB565	✔	✔	✔	✔
R_VDCE_RGB0888	✔	✔	✔	✔
R_VDCE_ARGB1555	✔	✔	✔	✔
R_VDCE_ARGB4444	✔	✔	✔	✔
R_VDCE_ARGB8888	✔	✔	✔	✔
R_VDCE_RGBA5551	✔	✔	✔	✔
R_VDCE_RGBA8888	✔	✔	✔	✔
R_VDCE_CLUT8	✔	✔	✔	✔
R_VDCE_CLUT4	✔	✔	✔	✔
R_VDCE_CLUT1	✔	✔	✔	✔
R_VDCE_YCBCR_422	✔	✔	-	-
R_VDCE_YCBCR_444	✔	✔	-	-
R_VDCE_YUV_YUYV	✔	✔	-	-
R_VDCE_YUV_UYVY	✔	✔	-	-
R_VDCE_YUV_YVYU	✔	✔	-	-
R_VDCE_YUV_VYUY	✔	✔	-	-

✔: Supported   -: Not supported

3.2.3.3 Layer feature

Following table shows the features supported by each Image synthesizer.

Table 3-13 Supported Feature with Layer

Image Synthesizer	Features						
	Scaling-up	Vertical Rotation	Alpha blending per one pixel	Constant Alpha	Pre-multiplied alpha	Chroma Key	Color Look up table
Scaler 0	✔	✔	-	-	-	-	✔
Scaler 1	✔	✔	✔	✔	✔	✔	✔
Image synthesizer 2	-	✔	✔	✔	✔	✔	✔
Image synthesizer 3	-	✔	✔	✔	✔	✔	✔

✔ : Supported   -: Not supported

Following table shows the supported features depending on the layer color format.

Table 3-14 Supported Feature with Color Format

Layer color format	Features						
	Scaling-up	Vertical Rotation	Alpha blending per one pixel	Constant Alpha	Pre-multiplied alpha	Chroma Key	Color Look up table
R_VDCE_RGB565 R_VDCE_RGB0888	✔	✔	-	✔	-	✔	-
R_VDCE_ARGB1555 R_VDCE_ARGB4444 R_VDCE_ARGB8888 R_VDCE_RGBA5551 R_VDCE_RGBA8888	✔	✔	✔	✔	✔	✔	-
R_VDCE_CLUT8 R_VDCE_CLUT4 R_VDCE_CLUT1	✔	✔	✔	✔	✔	✔	✔
R_VDCE_YCBCR_422 R_VDCE_YCBCR_444 R_VDCE_YUV_YUYV R_VDCE_YUV_UYVY R_VDCE_YUV_YVYU R_VDCE_YUV_VYUY	✔	✔	-	-	-	-	-

✔ : Supported   -: Not supported

3.2.3.4 Scaling-Up

R_VDCE_LayerImgScaleX and R_VDCE_LayerImgScaleY sets the scaling parameters in X and Y direction.
When R_VDCE_SCALING_LARGER is selected, enlarged image can be displayed.
ScaledWidth and ScaledHeight are specified the target memory size to be enlarged.

Table 3-15 R_VDCE_SCALING_LARGER

Function	Parameter	Setting unit	Range		Alignment
			Min	**Max**	
R_VDCE_LayerImgScaleX	ScaledWidth	Pixel	4	DispWidth - 1	-
R_VDCE_LayerImgScaleY	ScaledHeight	Pixel	4	DispHeight - 1	-
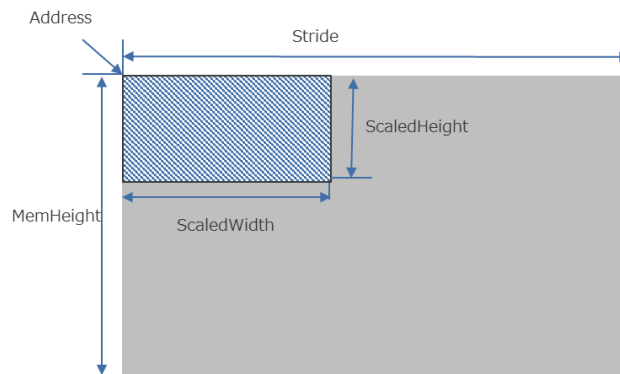
Figure 3-12 memory image with scaling-up
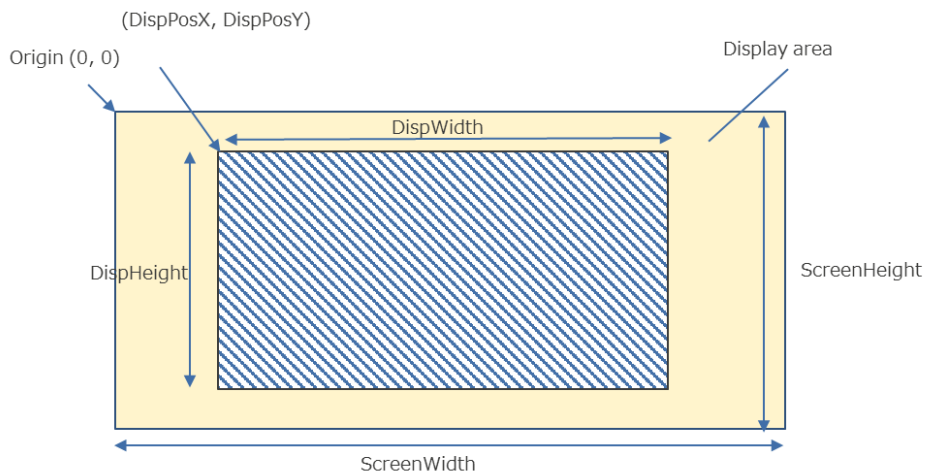
Figure 3-13 Display image with scale-up

When scaling-up (or scaling-down) feature is used, it is necessary to input the clock selected by the register below.
In particular, be careful when video capturing is not used.
See H/W user's manual about the register.

Table 3-16 Clock selection

VDCE	Scaler	Setup Clock	Remarks
VDCE0	Scaler 0	CKSC_IVDCE0VIS_CTL	-
VDCE0	Scaler 1	VDCECTL.VI1CTL	RH850/D1M2(H) Only
VDCE1	Scaler 0	VDCECTL.VI1CTL	RH850/D1M2(H) Only
VDCE1	Scaler 1	CKSC_IVDCE0VIS_CTL	-

3.2.3.5 Vertical Rotation

Vertical rotation is set by R_VDCE_LayerModeSet or R_VDCE_CapModeSet depending on input type.
If vertical rotation is enabled, rotation image is synthesized after the image is read from frame buffer.

Table 3-17 Vertical rotation

Input Type	Function	Flag
Graphic	R_VDCE_LayerModeSet	R_VDCE_LAYER_MODE_V_MIRORING
Video	R_VDCE_CapModeSet	R_VDCE_CAP_MODE_V_MIRRORING
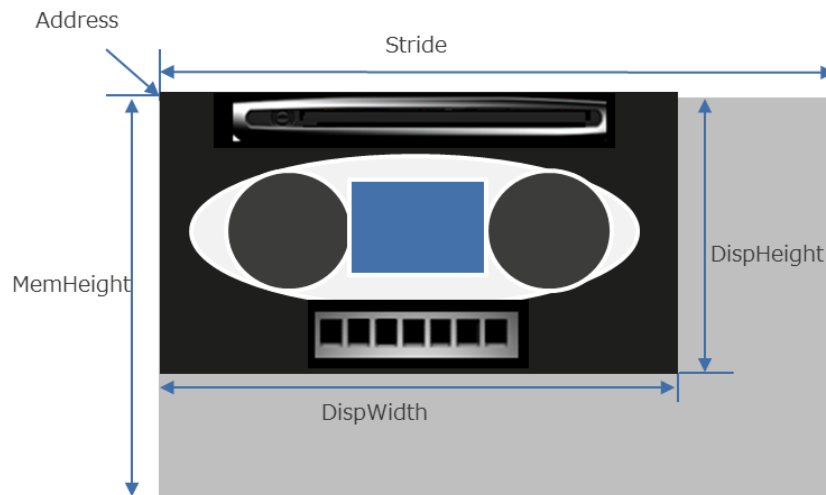
Figure 3-14 Memory image before vertical rotation
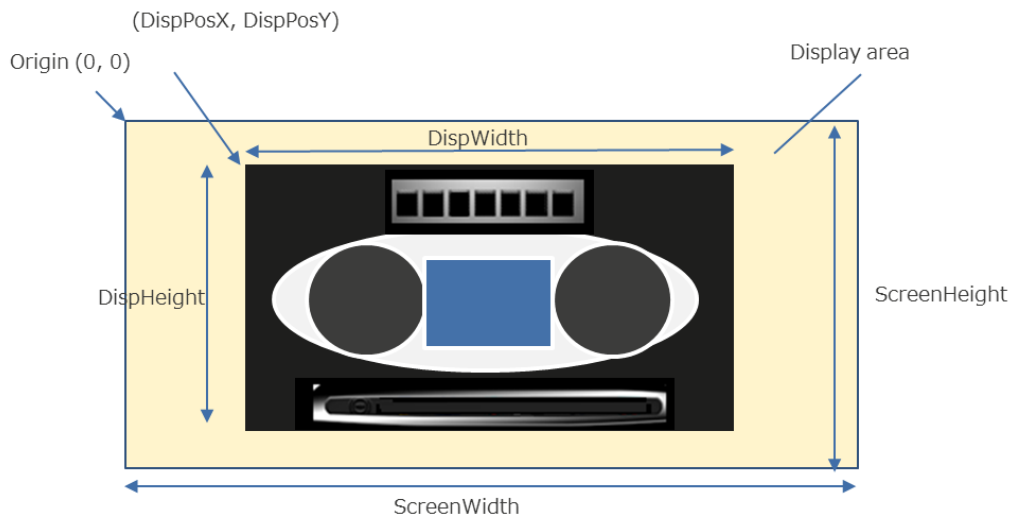
Figure 3-15 Display image after vertical rotation

3.2.3.6 Alpha blending

Scaler1, Image Synthesizer2 and Image synthesizer3 support alpha blending features.
Scaler0 doesn't support alpha blending feature.

- Alpha blending per one pixel

  When color format has alpha channel (i.e. ARGBxxxx, RGBAxxxx, CLUTx), alpha value of each pixel data is enabled as default.
  If user calls R_VDCE_LayerAlphaChannelDisable, these alpha channels can be disabled.
  R_VDCE_LayerAlphaChannelEnable is prepared for re-enabling the disabled alpha channels.

- Constant Alpha

  The constant alpha value can set by AlphaConst of R_VDCE_LayerAlphaConstEnable. If constant alpha is enabled, specified vale is multiplied to all pixel data of the layer. Constant alpha is disabled as default.

- Pre-multiplied alpha

  R_VDCE_LayerPremultipliedAlphaEnable can enable the pre-multiplied alpha feature. If Pre-multiplied alpha is enabled, VDCE skips the multiplication of input RGB data and alpha value. This feature should be enabled when input RGB data has already pre-multiplied. Pre-multiplied alpha is disabled as default.

Alpha value is calculated with following formula.

Table 3-18 Alpha value

Alpha blending per one pixel	Constant Alpha	calculation
Enable	Enable	alpha value = A[in] * AlphaConst / 255
Enable	Disable	alpha value = A[in]
Disable	Enable	alpha value = AlphaConst
Disable	Disable	alpha value = 255

A[in] is alpha value of the pixel data that is converted from 0 to 255.

Each pixel data of the layer is calculated with following formula.

Table 3-19 Layer color data

Pre-multiplied alpha	calculation
Disable	G[out] = ((G[in1] * alpha value) + (G[in0] * (255 − alpha value))) / 255 B[out] = ((B[in1] * alpha value) + (B[in0] * (255 − alpha value))) / 255 R[out] = ((R[in1] * alpha value) + (R[in0] * (255 − alpha value))) / 255
Enable	G[out] = G[in1] + (G[in0] * (255 − alpha value)) / 255 B[out] = B[in1] + (B[in0] * (255 − alpha value)) / 255 R[out] = R[in1] + (R[in0] * (255 − alpha value)) / 255

G[in0], B[in0], R[in0] are G/B/R value of the lower-layer.
G[in1], B[in1], R[in1] are G/B/R value of the input graphics data.

3.2.3.7 Chromakey

R_VDCE_LayerChromaKeyEnable can set one set of chromakey for a layer to convert color data.
The color before conversion is specified by CkTarget, and the color after conversion is specified by CkReplace.

In case of RGB format, R/G/B data is set to CkTarget. Regardless of alpha value, colors with specified RGB value is converted.
In case of CLUT format, table index is set to A of CkTarget.

CkReplace is set with RGB format with alpha value. R/G/B values are specified by values in the range that depends on the color format, but alpha value is specified by 8 bits without depending on the color format.

Following table shows the range of each member depending on color format.

Table 3-20 Range of argument

Layer color format	CkTarget				CkReplace			
	B	G	R	A	B	G	R	A
R_VDCE_RGB565	0-31	0-63	0-31	0	0-31	0-63	0-31	0-255
R_VDCE_RGB0888 R_VDCE_ARGB8888 R_VDCE_RGBA8888	0-255	0-255	0-255	0	0-255	0-255	0-255	0-255
R_VDCE_ARGB1555 R_VDCE_RGBA5551	0-31	0-31	0-31	0	0-31	0-31	0-31	0-255
R_VDCE_ARGB4444	0-15	0-15	0-15	0	0-15	0-15	0-15	0-255
R_VDCE_CLUT8	0	0	0	0-255	0-255	0-255	0-255	0-255
R_VDCE_CLUT4	0	0	0	0-15	0-255	0-255	0-255	0-255
R_VDCE_CLUT1	0	0	0	0-1	0-255	0-255	0-255	0-255

Example of ARGB8888 format:
Example 1: Convert (B, G, R) = (255, 0, 0) to (B, G, R, A) = (0, 255, 0, 255)
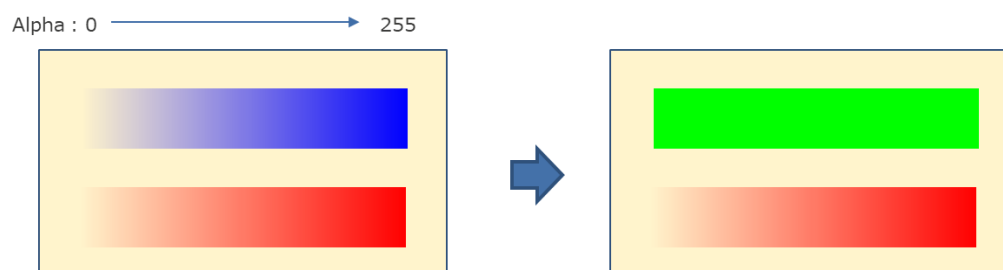
Figure 3-16 Chromakey example 1

Example 2: Convert (B, G, R) = (0, 0, 255) to (B, G, R, A) = (0, 255, 0, 63)
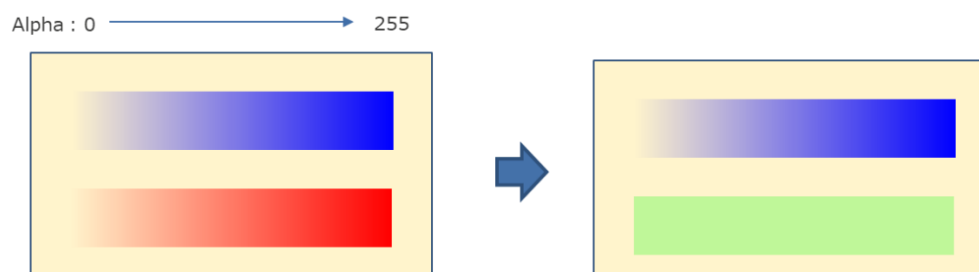
Figure 3-17 Chromakey example 2

3.2.3.8   Color Look up table

CLUT (Color look up table) mode can be selected by layer color format. Each pixel data of frame buffer should consist of the index of the table.

Table 3-21 Range of table index

Layer color format	Range of table index
R_VDCE_CLUT8	0 - 255
R_VDCE_CLUT4	0 - 15
R_VDCE_CLUT1	0 - 1

Actual color data is set to CLUT by R_VDCE_LayerClutSet. Set position is specified by Offset and ClutSize. Color data is specified by the array of Clut.
For example, if Offset = 2 and ClutSize = 3, then Clut[0] to Clut[2] are set to the table.

Table 3-22 Setting example

Table index	Set data
#0	-
#1	-
#2	Clut[0]
#3	Clut[1]
#4	Clut[2]
#5	-
:	:

There are two CLUTs for each layer to be able to update while displaying. VDCE driver uses CULT #0 and CLUT #1 alternately. R_VDCE_LayerClutSet should be called only once during 1 Vsync period.
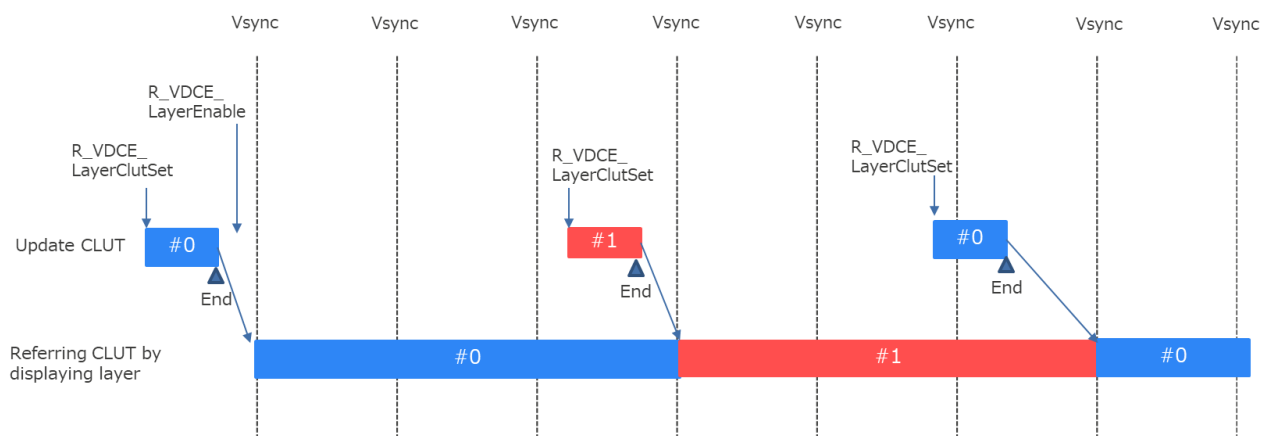
Figure 3-18 CULT update timing

By configuring the frame buffer as shown below, user can switch the display image only by updating CLUT.
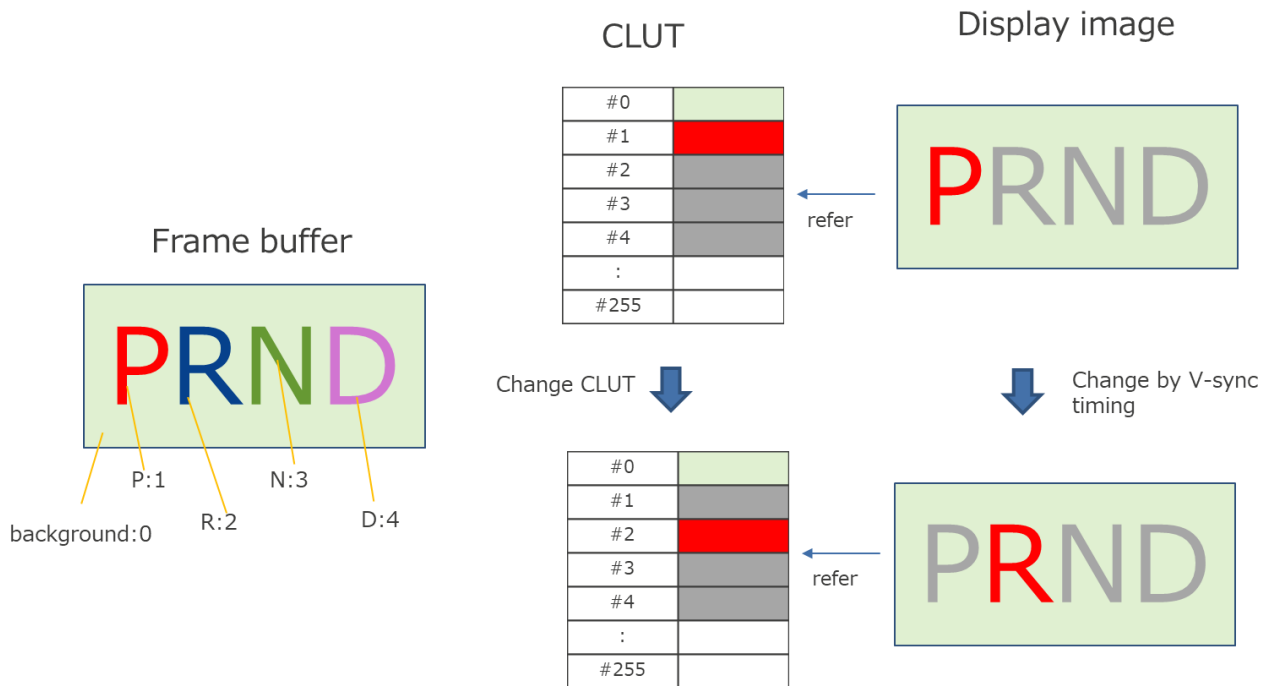
Figure 3-19 CULT use case

3.2.4 OIR layer

OIR layer is VDCE interface with VOWE hardware.
Following functions are prepared and VOWE porting layer controls these functions.
- R_VDCE_OirBaseSet
- R_VDCE_OirMemGeometrySet
- R_VDCE_OirFormatSet
- R_VDCE_OirViewPortSet
- R_VDCE_OirRingBufferEnable
- R_VDCE_OirRingBufferDisable
- R_VDCE_OirVSyncDelaySet
- R_VDCE_OirModeSet
- R_VDCE_OirEnable
- R_VDCE_OirDisable

3.2.5 Video capture

3.2.5.1 Capture Unit and Image Synthesizer

VDCE has Input controller for receiving external video input. Scaler 0 has the feature to write the received video data to the capture buffer. The capture buffer can be synthesized by inputting it to Scaler 0 or Scaler 1 as a frame buffer.
Video input captured data with VDCE0 can be synthesized by VDCE0 Scaler 0 or VDCE1 Scaler1.
Video input captured with VDCE1 can be synthesized by VDCE1 Scaler 0 or VDCE0 Scaler1.

R_VDCE_CapEnable selects the connection.
Following table shows supported connection on RH850/D1x device.

Table 3-23 Capture Unit and Image Synthesizer

Parameter setting		VDCE Unit/Layer		RH850/D1x Device			
Unit	OutputUnit	Input controller	Synthesizer	D1M2H	D1M2	D1M1A	D1M1(H), D1M1-V2
0	0	VDCE0	VDCE0 Scaler0	✔	-	✔	✔
0	1	VDCE0	VDCE1 Scaler1	✔	-	✔	-
1	1	VDCE1	VDCE1 Scaler0	✔	✔	-	-
1	0	VDCE1	VDCE0 Scaler1	✔	✔	-	-

✔: Supported   -: Not supported
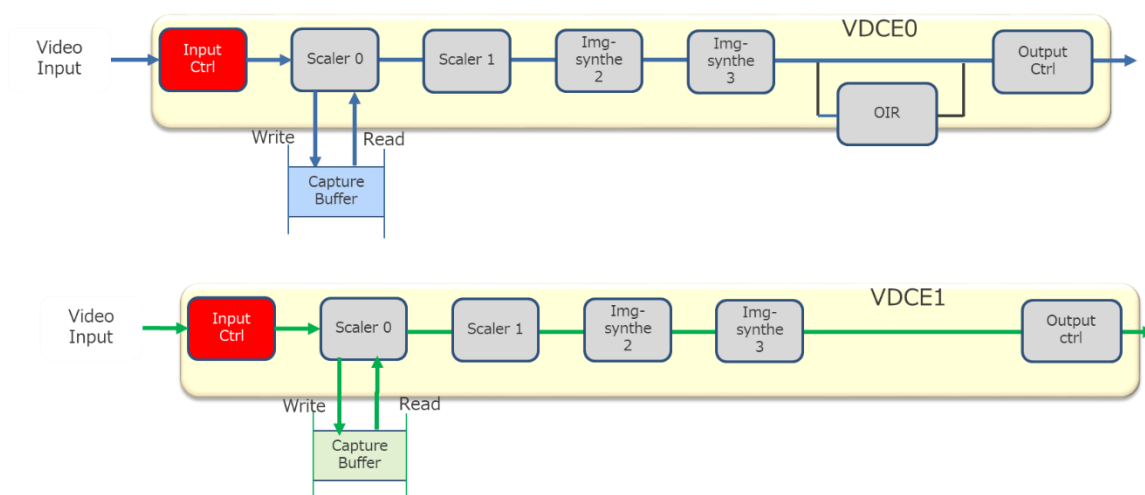
Figure 3-20 Capture data flow (1) for RH850/D1M2H
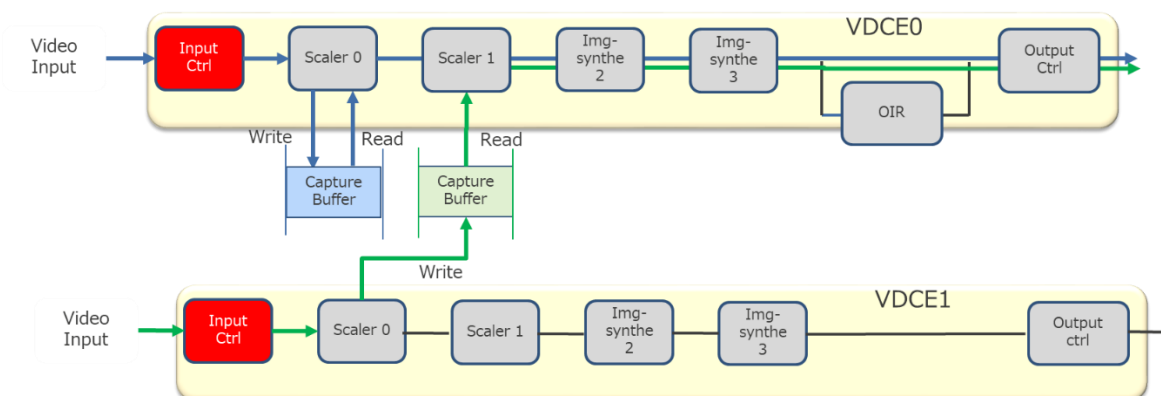
Figure 3-21 Capture data flow (2) for RH850/D1M2H

Renesas Graphics Library Video Data Controller E (VDCE) Driver
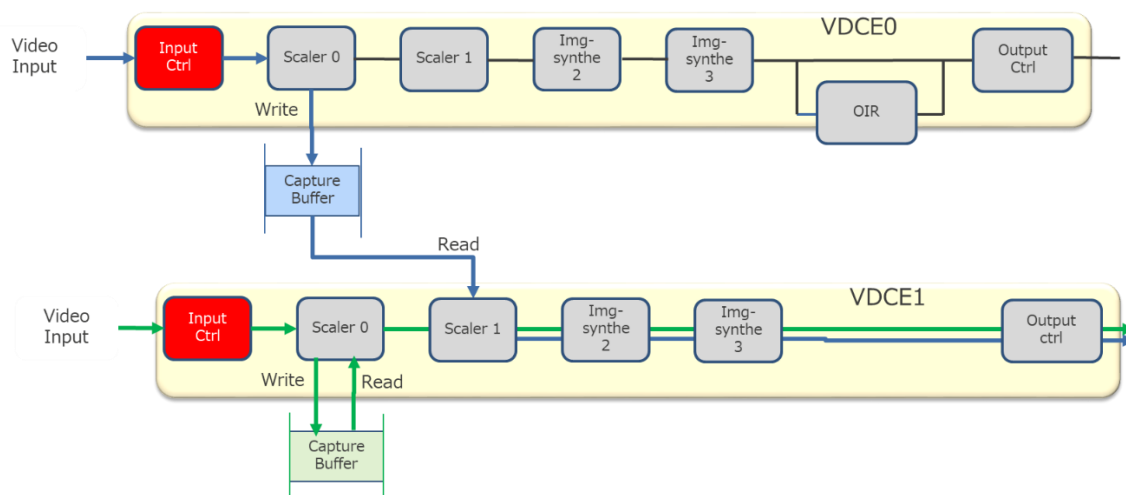
Figure 3-22 Capture data flow (3) for RH850/D1M2H

In the case of controlling over two units (e.g. Input from VDCE0 and output from VDCE1), both units must transition to Idle state before capturing is enabled. During setup of this case, "Unit" of VDCE API parameter is selected as follows.

Table 3-24 Unit selection

VDCE API		Unit selection	
Function	**Parameter**	**Input controller**	**Synthesizer**
Capture Functions - R_VDCE_CapBufGeometrySetup - R_VDCE_CapModeSet - R_VDCE_CapBufSet - R_VDCE_CapBufFieldSetup1 - R_VDCE_CapBufFieldSetup2 - R_VDCE_CapEnable - R_VDCE_CapDisable - R_VDCE_CapViewPortSet - R_VDCE_CapRateSet - R_VDCE_CapExtVsyncSet	-	✔	-
Capture Interrupt control - R_VDCE_IntcCallbackSet - R_VDCE_IntcCallbackGet - R_VDCE_IntcEnable - R_VDCE_IntcDisable	R_VDCE_INTC_NO_VI_VSYNC_SIGNAL R_VDCE_INTC_NO_VI_VSYNC_SIGNAL_1	-	✔
	R_VDCE_INTC_CAP_VBLANK R_VDCE_INTC_CAP_END_OF_FIELD R_VDCE_INTC_ERR_CAP_WRITE_OVERFLOW	✔	-
Capture Interrupt getting - R_VDCE_IntcCapScanlineGet	-	✔	-
External V-sync setting - R_VDCE_DisplayVsyncProtectionSet - R_VDCE_LayerVSyncDelaySet	-	-	✔
Scaling-Up/Down - R_VDCE_LayerImgScaleX - R_VDCE_LayerImgScaleY - R_VDCE_LayerImgScaleModeSet	-	-	✔
Color matrix setting - R_VDCE_LayerMatrixSet - R_VDCE_LayerMatrixBT601Set - R_VDCE_LayerMatrixJPEGSet - R_VDCE_LayerMatrixUnitySet	R_VDCE_LAYER_INPUT	✔	-
	R_VDCE_LAYER_SCALER0 R_VDCE_LAYER_SCALER1	-	✔

✔: Apply -: Not aplly

3.2.5.2 Input video format

The input video format is selected by R_VDCE_CapModeSet.

Table 3-25 Input video format

Input video format	Set flag
BT656	R_VDCE_CAP_MODE_YUV_ITU656
BT601	R_VDCE_CAP_MODE_YUV_8BIT
YCbCr422	R_VDCE_CAP_MODE_YUV_16BIT
RGB565	R_VDCE_CAP_MODE_RGB_16BPP
RGB666	R_VDCE_CAP_MODE_RGB_18BPP
RGB888	R_VDCE_CAP_MODE_RGB_24BPP
YCbCr444	R_VDCE_CAP_MODE_RGB_24BPP

24 color data signals (DV_DATA23..00) are input to VDCE. Data signal assignment is depending on following settings.
- Input video format which is set by R_VDCE_CapModeSet.
- Endian which is set by R_VDCE_CAP_MODE_BIG_ENDIAN flag of R_VDCE_CapModeSet
- Swap which is set by R_VDCE_FB_RB_SWAP flag of R_VDCE_CapBufGeometrySetup

Table 3-26 Input color data

Setting			DV_DATA23 <---> DV_DATA00
Input video format	Endian	Swap	
RGB888	Little	Off	$R_7R_6R_5R_4R_3R_2R_1R_0$ $G_7G_6G_5G_4G_3G_2G_1G_0$ $B_7B_6B_5B_4B_3B_2B_1B_0$
		On	$B_7B_6B_5B_4B_3B_2B_1B_0$ $G_7G_6G_5G_4G_3G_2G_1G_0$ $R_7R_6R_5R_4R_3R_2R_1R_0$
	Big	Off	$R_0R_1R_2R_3R_4R_5R_6R_7$ $G_0G_1G_2G_3G_4G_5G_6G_7$ $B_0B_1B_2B_3B_4B_5B_6B_7$
		On	$B_0B_1B_2B_3B_4B_5B_6B_7$ $G_0G_1G_2G_3G_4G_5G_6G_7$ $R_0R_1R_2R_3R_4R_5R_6R_7$
RGB666	Little	Off	0 0 0 0 0 0 R_7R_6 $R_5R_4R_3R_2G_7G_6G_5G_4$ $G_3G_2B_7B_6B_5B_4B_3B_2$
		On	0 0 0 0 0 0 B_7B_6 $B_5B_4B_3B_2G_7G_6G_5G_4$ $G_3G_2R_7R_6R_5R_4R_3R_2$
	Big	Off	0 0 0 0 0 0 R_2R_3 $R_4R_5R_6R_7G_2G_3G_4G_5$ $G_6G_7B_2B_3B_4B_5B_6B_7$
		On	0 0 0 0 0 0 B_2B_3 $R_4R_5R_6R_7G_2G_3G_4G_5$ $G_6G_7B_2B_3B_4B_5B_6B_7$
RGB565	Little	Off	0 0 0 0 0 0 0 0 $R_7R_6R_5R_4R_3G_7G_6G_5$ $G_4G_3G_2B_7B_6B_5B_4B_3$
		On	0 0 0 0 0 0 0 0 $B_7B_6B_5B_4B_3G_7G_6G_5$ $G_4G_3G_2R_7R_6R_5R_4R_3$
	Big	Off	0 0 0 0 0 0 0 0 $R_3R_4R_5R_6R_7G_2G_3G_4$ $G_5G_6G_7B_3B_4B_5B_6B_7$
		On	0 0 0 0 0 0 0 0 $B_3B_4B_5B_6B_7G_2G_3G_4$ $G_5G_6G_7R_3R_4R_5R_6R_7$
YCbCr444	Little	Off	$V_7V_6V_5V_4V_3V_2V_1V_0$ $Y_7Y_6Y_5Y_4Y_3Y_2Y_1Y_0$ $U_7U_6U_5U_4U_3U_2U_1U_0$
		On	$U_7U_6U_5U_4U_3U_2U_1U_0$ $Y_7Y_6Y_5Y_4Y_3Y_2Y_1Y_0$ $V_7V_6V_5V_4V_3V_2V_1V_0$
	Big	Off	$V_0V_1V_2V_3V_4V_5V_6V_7$ $Y_0Y_1Y_2Y_3Y_4Y_5Y_6Y_7$ $U_0U_1U_2U_3U_4U_5U_6U_7$
		On	$U_0U_1U_2U_3U_4U_5U_6U_7$ $Y_0Y_1Y_2Y_3Y_4Y_5Y_6Y_7$ $V_0V_1V_2V_3V_4V_5V_6V_7$
YCbCr422	Little	Off	0 0 0 0 0 0 0 0 $Y_7Y_6Y_5Y_4Y_3Y_2Y_1Y_0$ $C_7C_6C_5C_4C_3C_2C_1C_0$
		On	0 0 0 0 0 0 0 0 $C_7C_6C_5C_4C_3C_2C_1C_0$ $Y_7Y_6Y_5Y_4Y_3Y_2Y_1Y_0$
	Big	Off	0 0 0 0 0 0 0 0 $Y_0Y_1Y_2Y_3Y_4Y_5Y_6Y_7$ $C_0C_1C_2C_3C_4C_5C_6C_7$
		On	0 0 0 0 0 0 0 0 $C_0C_1C_2C_3C_4C_5C_6C_7$ $Y_0Y_1Y_2Y_3Y_4Y_5Y_6Y_7$
BT656 / BT601	Little	-	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 $T_7T_6T_5T_4T_3T_2T_1T_0$
	Big	-	0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 $T_0T_1T_2T_3T_4T_5T_6T_7$

R_n, G_n and B_n (n=0~7): RGB data.
Y_n: Y component, U_n: Cb component, V_n: Cr component,
C_n: Cb or Cr component is input alternately.
T_n: BT656 / BT601 input data.

3.2.5.3 Capture buffer color format

The color format of the capture buffer is set by FbFormat of R_VDCE_CapBufGeometrySetup. Because VDCE has color matrix for input data, any input video format can convert to specified capture buffer color format.

Table 3-27 Capture buffer color format

Capture buffer color format	bpp	Memory image (MSB <--> LSB)	remarks
R_VDCE_FB_FORMAT_YCBCR_422	16	UUUUUUUU YYYYYYYY VVVVVVVV YYYYYYYY	32bits / 2Pixels
R_VDCE_FB_FORMAT_YCBCR_444	32	xxxxxxxx VVVVVVVV YYYYYYYY UUUUUUUU	
R_VDCE_FB_FORMAT_RGB565	16	RRRRRGGG GGGBBBBB	Dither works
R_VDCE_FB_FORMAT_RGB0888	32	xxxxxxxx RRRRRRRR GGGGGGGG BBBBBBBB	

When R_VDCE_FB_FORMAT_RGB565 is selected, as bit reduction processing of RGB565, rounding off or 2x2 pattern dither can be selected with R_VDCE_CAP_MODE_DITHER flag of R_VDCE_CapModeSet.

The capture buffer can be input to the image synthesizer as a frame buffer. The layer color format should be same as capture buffer color format. Following table shows the correspondence.

Table 3-28 Layer color format

Capture buffer	Layer color format
R_VDCE_FB_FORMAT_YCBCR_422	R_VDCE_YCBCR_422
R_VDCE_FB_FORMAT_YCBCR_444	R_VDCE_YCBCR_444
R_VDCE_FB_FORMAT_RGB565	R_VDCE_RGB565
R_VDCE_FB_FORMAT_RGB0888	R_VDCE_RGB0888

3.2.5.4 Capture basic setting for progressive

The capture area and captured frame buffer is set by R_VDCE_CapBufGeometrySetup.
The start address of frame buffer should be 128 byte-aligned.
The stride of the memory should be 128 byte-aligned. Thus, if capture buffer color format is 16 bpp, the parameter "Stride" should be 64 pixel-aligned. Following table shows the range information.

Table 3-29 Parameter range

Parameter	Setting unit	Range		Alignment
		Min	**Max**	
Buffer1	Byte	-	-	128 Byte
Buffer2	Byte	-	-	128 Byte
Stride	Pixel	1024 / bpp	261120 / bpp	128 Byte
Width	Pixel	4	2008	4 Pixel
Height	Pixel	4	1024	4 Pixel
StartX	Pixel	16	2011	-
StartY	Pixel	4	2035	-
StartX + Width	Pixel	20	2015	-
StartY + Height	Pixel	8	2039	-

* bpp is depending on color format of capture buffer.
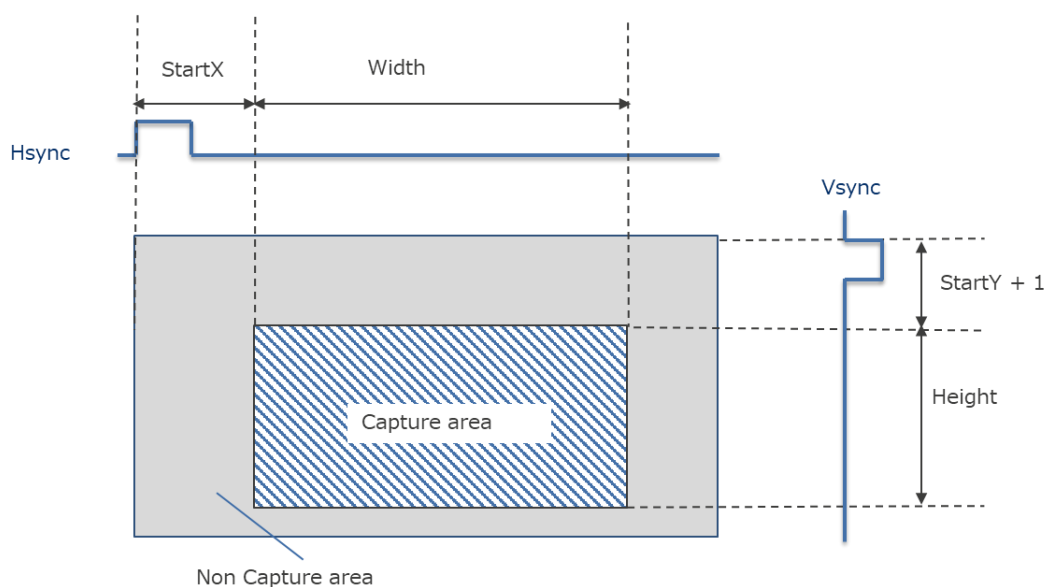
Figure 3-23 Capture area with Vsync / Hsync
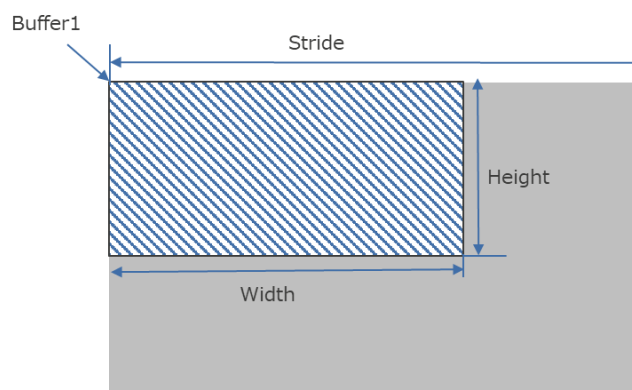
Figure 3-24 Capture buffer (without scale-down)

3.2.5.5 Capture by DE signal

If DE mode is enabled by R_VDCE_CAP_MODE_DE_MODE flag of R_VDCE_CapModeSet, VDCE capture the video signal by DE signal instead of Hsync signal.
In case of DE mode, StartX and StartY can be set 0 to capture all data enable area.
The range of other parameters are same as *Table 3-29*.

Table 3-30 Parameter range of DE mode

Parameter	Setting unit	Range		Alignment
		Min	Max	
StartX	Pixel	0	2011	-
StartY	Pixel	0	2035	-
StartX + Width	Pixel	4	2015	-
StartY + Height	Pixel	4	2039	-
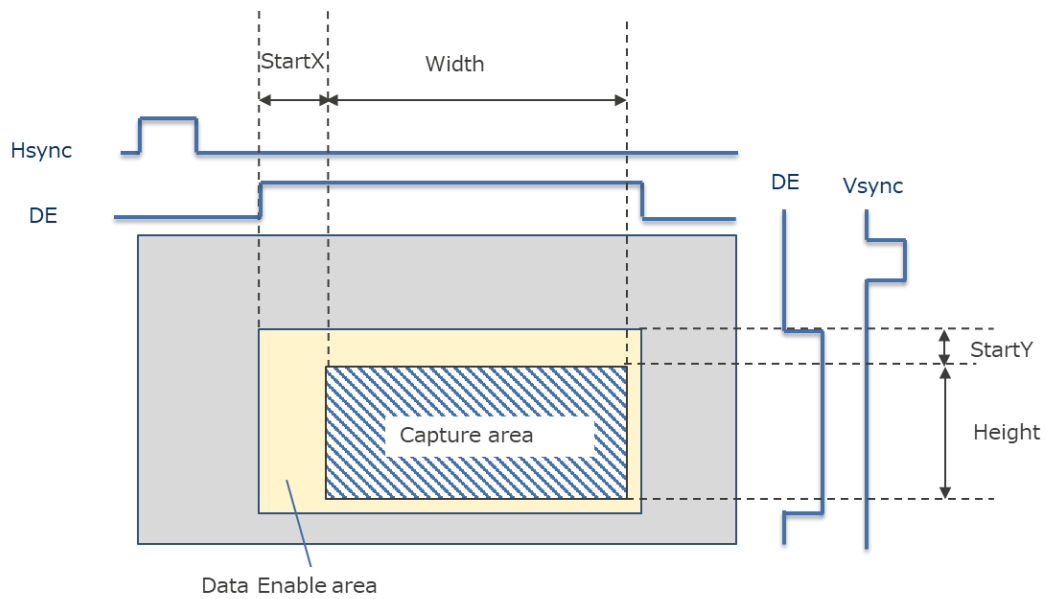
Figure 3-25 Capture area with DE signal

3.2.5.6 Scale-down

R_VDCE_LayerImgScaleX and R_VDCE_LayerImgScaleY sets the scaling parameters in X and Y direction.
When R_VDCE_SCALING_SMALLER is selected, reduced image will be captured.
ScaledWidth and ScaledHeight are specified the target memory size to be reduced.

Table 3-31 R_VDCE_SCALING_SMALLER

Function	Parameter	Setting unit	Range Min	Range Max	Alignment
R_VDCE_LayerImgScaleX	ScaledWidth	Pixel	4	1024 and (Width - 1)	4 Pixel
R_VDCE_LayerImgScaleY	ScaledHeight	Pixel	4	1024 and (Height - 1)	4 Pixel
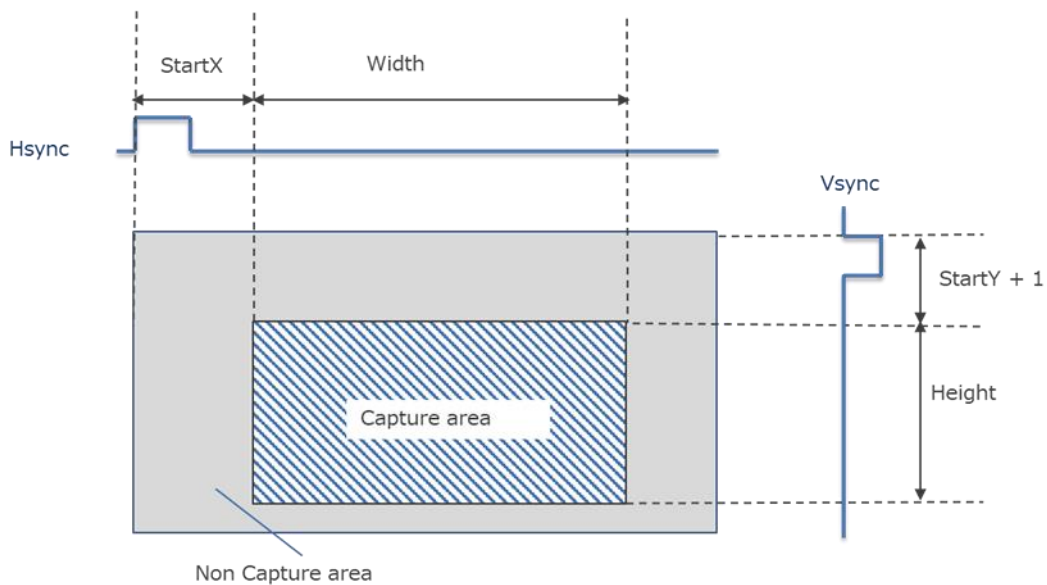
Figure 3-26 Capture area with Vsync / Hsync
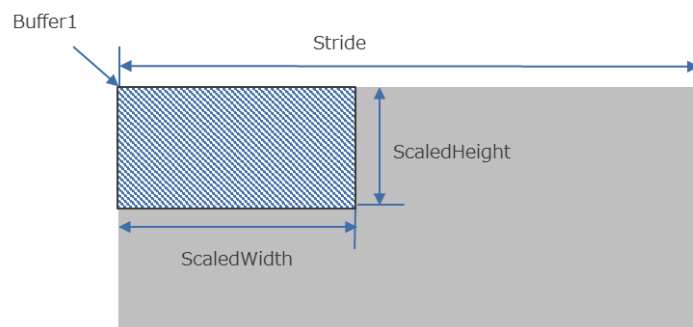
Figure 3-27 Capture buffer with Scale-down

3.2.5.7 Capture basic setting for interlace

If video input format is BT656 or BT601, VDCE can capture the interlace signal.
If video input format is BT656, VDCE generates Hsync signal from SAV/EAV code.
See H/W user's manual for the detail.
In case of BT656 or BT601 format, each pixel is captured twice. Therefore, Width setting should be doubled and
horizontal size should be scaled down by half in order to keep original input size.

This is the sample code to capture 720x480 interlace signal and display the image with 720x480.

```
/* Capture setting */
R_VDCE_CapBufGeometrySetup(
      Unit, Buffer1, Buffer2,
      736, /* Stride aligns from 720 pixels to 128 bytes */
      1440, /* Width = 720*2 */
      240, /* Height = 480/2 */
      StartX, StartY,
      R_VDCE_FB_FORMAT_RGB0888 /* 32bpp */
      );
R_VDCE_LayerImgScaleX(
      Unit, LayerNr,
      720, /* ScaledWidth = 720 */
      R_VDCE_SCALING_SMALLER /* horizontal size should be scaled down */
      );

/* Layer setting */
R_VDCE_LayerBaseSet(
      Unit, LayerNr,
      Buffer1 /* Select top field or bottom field */
      );
R_VDCE_LayerMemGeometrySet(
      Unit, LayerNr,
      736, /* should be same stride as capture buffer */
      240
      );
R_VDCE_LayerFormatSet(
      Unit, LayerNr,
      R_VDCE_RGB0888 /* should be same format as capture buffer */
      );
R_VDCE_LayerViewPortSet (
      Unit, LayerNr, 0, 0,
      DispPosX, DispPosY,
      720, /* DispWidth = 720 */
      480  /* DispHeight = 480 */
      );
R_VDCE_LayerImgScaleY(
      Unit, LayerNr,
      240, /* ScaledHeight = 240 */
      R_VDCE_SCALING_LARGER /* Vertical size is scaled-up */
      );
```

The following figures show the operation image of the above sample code.
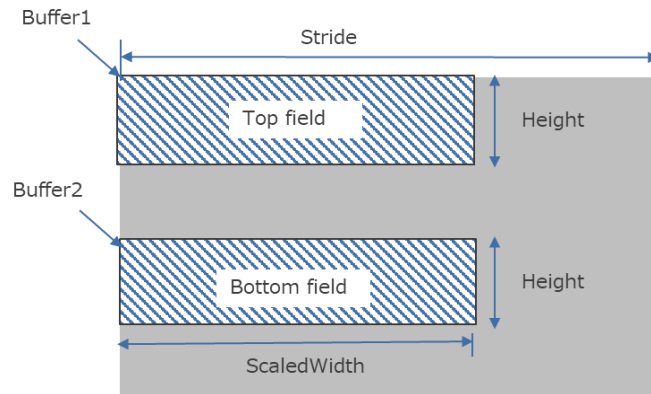
Figure 3-28 Capture buffer of interlace signal
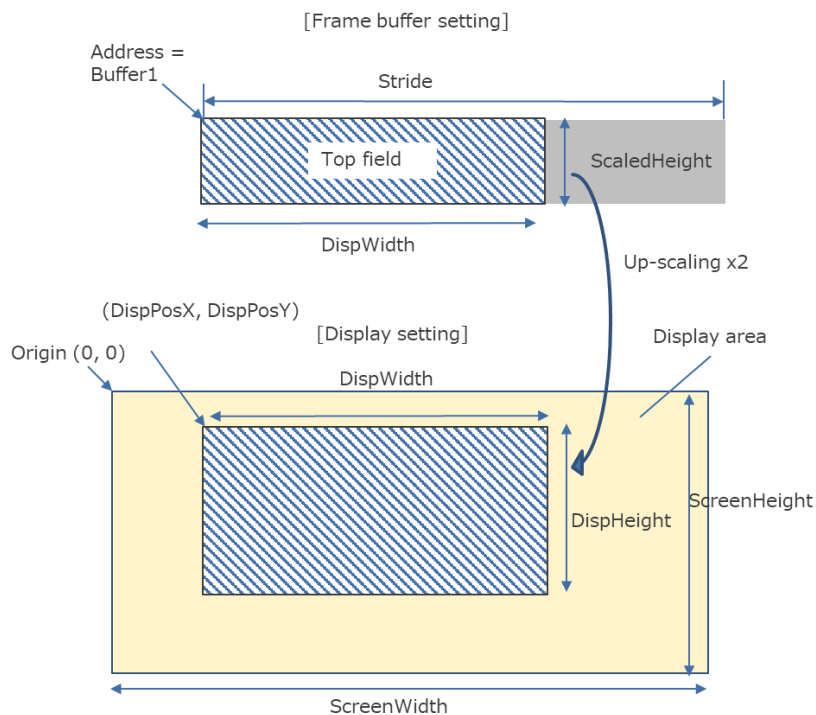
Figure 3-29 Frame buffer and display image

R_VDCE_CapRateSet can set writing rate. If 1/2 rate and top field is selected, only top field can be captured to the buffer and capture buffer for bottom field is not needed. If 1/2 rate and bottom field is selected, only bottom field can be captured to the buffer and capture buffer for top field is not needed.

This is example only top field.

```
R_VDCE_CapRateSet(
     Unit,
     R_VDCE_CAP_RATE_PER2, /* rate 1/2 */
     R_VDCE_CAP_FIELD_TOP
     );
```

3.2.5.8 Capture buffer

Two capture buffer address can be set by R_VDCE_CapBufGeometrySetup.
Buffer1 cannot be set to 0. Buffer2 can be set to 0 if not needed.

Table 3-32 Capture buffer

Setting		Input signal	Output buffer
Buffer1	Buffer2		
not 0	not 0	interlace Top field	Buffer1
		interlace Bottom field	Buffer2
		Progressive frame	Buffer1
not 0	0	interlace Top field	Buffer1
		interlace Bottom field	Buffer1
		Progressive frame	Buffer1

Buffer address can be changed by following functions during capturing.
- R_VDCE_CapBufSet
- R_VDCE_CapBufFieldSetup1
- R_VDCE_CapBufFieldSetup2

3.2.5.9 Horizontal rotation

Horizontal rotation cab be set by R_VDCE_CAP_MODE_H_MIRRORING option.
If horizontal rotation is enabled, rotation image is stored to capture buffer.

Table 3-33 Horizontal rotation

Input Type	Function	Flag
Video	R_VDCE_CapModeSet	R_VDCE_CAP_MODE_H_MIRRORING
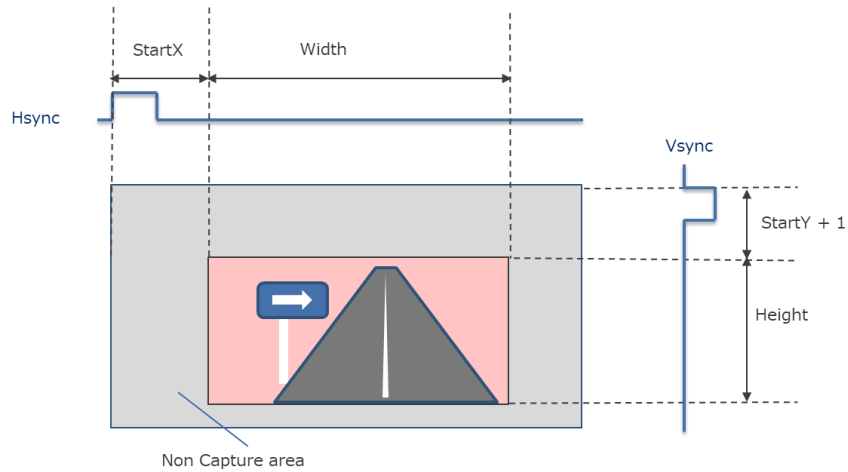
Figure 3-30 Capture image before horizontal rotation
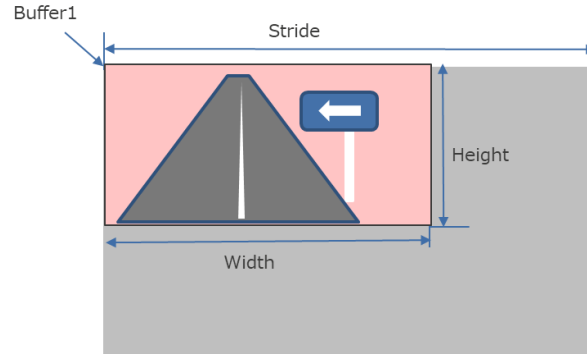
Figure 3-31 Capture buffer image after horizontal rotation

3.2.5.10 External Vsync

Synchronization Vsync of scaler 0 and scaler 1 is selected by R_VDCE_CAP_MODE_FIXED_VSYNC option.
Vsync of Scaler 1 also depends on the scaling-up and layer color format.
Following tables show the Vsync selection of each block.
External Vsync is externally input Vsync signal from the input controller.
Internal Vsync is internally generated free-running Vsync signal.

Table 3-34 Synchronization Vsync of Scaler 0

Condition		Result
Input	**R_VDCE_CAP_MODE _FIXED_VSYNC**	**Vsync of Scaler0**
Video	Off	External Vsync
	On	Internal Vsync
Graphic	Off	External Vsync (*1)
	On	Internal Vsync
None	-	Internal Vsync

(*1) If external V-sync is used in graphic input, set R_VDCE_CapEnable is needed with
R_VDCE_CAP_MODE_ONLY_SYNC option.

Table 3-35 Synchronization Vsync of Scaler 1

Condition				Result
Input	**R_VDCE_CAP_MODE _FIXED_VSYNC**	**Scale-up**	**Layer Color Format**	**Vsync of Scaler1**
Video	Off	On	-	External Vsync
		Off	YCbCr	External Vsync
			RGB	Vsync from Scaler 0
	On	On	-	Internal Vsync
		Off	YCbCr	Internal Vsync
			RGB	Vsync from Scaler 0
Graphic	-	On	-	Internal Vsync
		Off	YCbCr	Internal Vsync
			RGB, CLUT	Vsync from Scaler 0
None	-	-	-	Vsync from Scaler 0

Table 3-36 Synchronization Vsync for subsequent blocks

Block	Vsync of block
Image synthesizer 2	Vsync from Scaler 0
Image synthesizer 3	Vsync from Scaler 0
OIR	Vsync from Scaler 0
Output controller	Vsync from Scaler 0

3.2.6 Color Matrix

VDCE has 3 color matrixes to convert the color format.
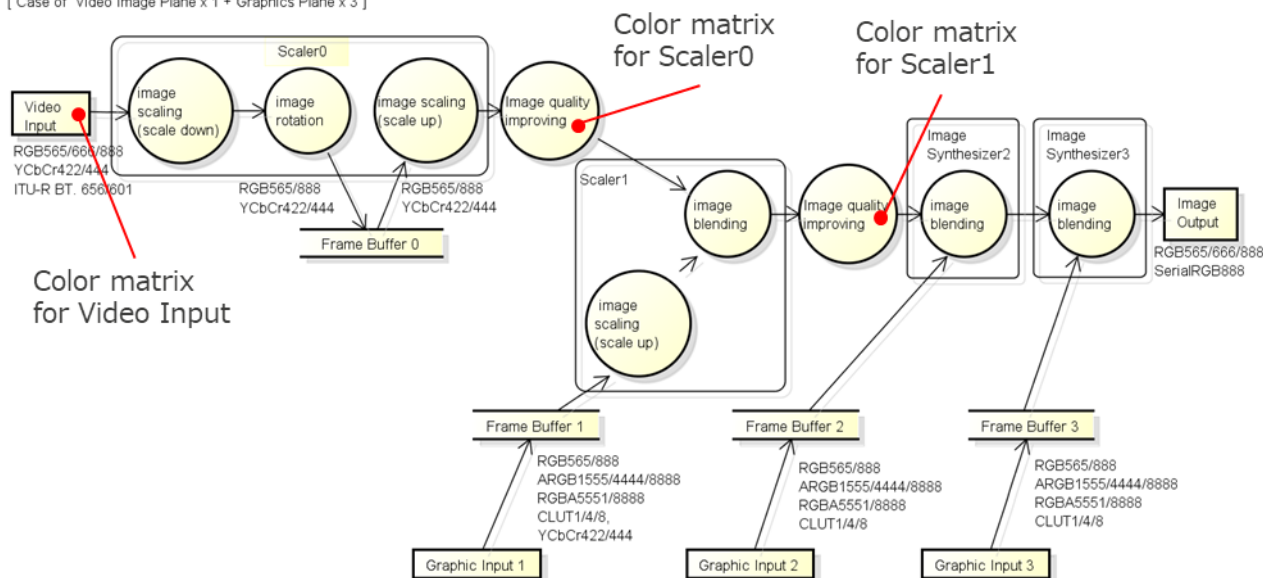
Figure 3-32 Color Matrix

Regarding to Color matrix for Video Input, input is capture data color format, and output is frame buffer color format.
VDCE driver decides the conversion type depending on input and output color format as follows.

Table 3-37 Conversion type for Video Input

Input color format	Output color format	Conversion type
R_VDCE_CAP_MODE_YUV_ITU656 R_VDCE_CAP_MODE_YUV_8BIT R_VDCE_CAP_MODE_YUV_16BIT	R_VDCE_FB_FORMAT_YCBCR_422 R_VDCE_FB_FORMAT_YCBCR_444	YCbCr to YCbCr
	R_VDCE_FB_FORMAT_RGB565 R_VDCE_FB_FORMAT_RGB0888	YCbCr to GBR
R_VDCE_CAP_MODE_RGB_16BPP R_VDCE_CAP_MODE_RGB_18BPP R_VDCE_CAP_MODE_RGB_24BPP	R_VDCE_FB_FORMAT_YCBCR_422 R_VDCE_FB_FORMAT_YCBCR_444	GBR to YCbCr
	R_VDCE_FB_FORMAT_RGB565 R_VDCE_FB_FORMAT_RGB0888	GBR to GBR

Regarding to Color matrix for Scaler0 and Scaler1, input is frame buffer data, output is ARGB data.
VDCE driver decides the conversion type depending on input color format as follows.

Table 3-38 Conversion type for Scaler0 and Scaler1

Input color format	Conversion type
R_VDCE_RGB565 R_VDCE_RGB0888 R_VDCE_ARGB1555 R_VDCE_ARGB4444 R_VDCE_ARGB8888 R_VDCE_RGBA5551 R_VDCE_RGBA8888 R_VDCE_CLUT8 R_VDCE_CLUT4 R_VDCE_CLUT1	GBR to GBR
R_VDCE_YCBCR_422 R_VDCE_YCBCR_444 R_VDCE_YUV_YUYV R_VDCE_YUV_UYVY R_VDCE_YUV_YVYU R_VDCE_YUV_VYUY	YCbCr to GBR

Following table shows the formula for each conversion type.
- G[in], B[in], R[in], Y[in], Cb[in] and Cr[in] are input RGB / YCbCr color data.
- G[out], B[out], R[out], Y[out], Cb[out] and Cr[out] are output RGB color data.
- Y_OFF, U_OFF, V_OFF, GY, BY, RY, GU, BU, RU, GV, BV and RV are offset / coefficient value.
  R_VDCE_LayerMatrixSet can set each value manually.

Table 3-39 Conversion formula

Conversion type	Formula	
GBR to GBR	G' = G[in] + Y_OFF B' = B[in] + U_OFF R' = R[in] + V_OFF	G[out] = G' * GY + B' * GU + R' * GV B[out] = G' * BY + B' * BU + R' * BV R[out] = G' * RY + B' * RU + R' * RV
GBR to YCbCr	G' = G[in] + Y_OFF B' = B[in] + U_OFF R' = R[in] + V_OFF	Y[out] = G' * GY + B' * GU + R' * GV Cb[out] = G' * BY + B' * BU + R' * BV + 128 Cr [out] = G' * RY + B' * RU + R' * RV + 128
YCbCr to GBR	Y' = Y[in] + Y_OFF U' = Cb[in] - 128 V' = Cr[in] - 128	G[out] = Y' * GY + U' * GU + V' * GV B[out] = Y' * BY + U' * BU + V' * BV R[out] = Y' * RY + U' * RU + V' * RV
YCbCr to YCbCr	Y' = Y[in] + Y_OFF U' = Cb[in] - 128 V' = Cr[in] - 128	Y[out] = Y' * GY + U' * GU + V' * GV Cb[out] = Y' * BY + U' * BU + V' * BV + 128 Cr [out] = Y' * RY + U' * RU + V' * RV + 128

Following table shows the constant values when user uses each matrix setting function or user doesn't use any matrix setting functions.

Table 3-40 Constant values

Matrix setting function	(Assumed) Conversion type	Values			
		Y_OFF U_OFF V_OFF	GY BY RY	GU BU RU	GV BV RV
Default (No matrix function is called.)	GBR to GBR	0 0 0	1.00000 0.00000 0.00000	0.00000 1.00000 0.00000	0.00000 0.00000 1.00000
	GBR to YCbCr	0 0 0	0.58594 -0.33203 -0.41797	0.11328 0.50000 -0.08203	0.30078 -0.16797 0.50000
	YCbCr to GBR	0 0 0	1.00000 1.00000 1.00000	-0.34375 1.77344 0.00000	-0.71484 0.00000 1.40234
	YCbCr to YCbCr	0 0 0	1.00000 0.00000 0.00000	0.00000 1.00000 0.00000	0.00000 0.00000 1.00000
R_VDCE_LayerMatrixBT601Set	YCbCr to GBR	-16 0 0	1.16406 1.16406 1.16406	-0.39199 2.01758 0.00000	-0.81250 0.00000 1.59570
R_VDCE_LayerMatrixJPEGSet	YCbCr to GBR	0 0 0	1.00000 1.00000 1.00000	-0.33398 1.77246 0.00000	-0.71387 0.00000 1.40234
R_VDCE_LayerMatrixUnitySet	GBR to GBR YCbCr to YCbCr	0 0 0	1.00000 0.00000 0.00000	0.00000 1.00000 0.00000	0.00000 0.00000 1.00000

3.3   Device difference

The following table shows maximum value difference depending on the device.

Table 3-41 Maximum value

Feature	RH850/D1x Device Name				
	D1M2H	D1M2	D1M1A	D1M1(H) D1M1-V2	D1L2(H)
Number of Units	2 (Unit0,1)	2 (Unit0,1)	2 (Unit0,1)	1 (Unit0)	1 (Unit0)
Number of Capture units	2 (Unit0,1)	1 (Unit1)	1 (Unit0)	1 (Unit0)	0
Display maximum screen width	1280	1280	1280	1024	480
Display maximum screen height	1024	1024	1024	1024	320

The following table shows the function differences depending on the device.

Table 3-42 Device difference

Feature / Function	RH850/D1x Device Name				
	D1M2H	D1M2	D1M1A D1M1-V2	D1M1(H)	D1L2(H)
OIR Interrupt control [Function] - R_VDCE_IntcCallbackSet - R_VDCE_IntcCallbackGet - R_VDCE_IntcEnable - R_VDCE_IntcDisable [Setting] - R_VDCE_INTC_OIR_SCANLINE - R_VDCE_INTC_OIR_VBLANK - R_VDCE_INTC_OIR_VSCYNC_WRITE - R_VDCE_INTC_ERR_OIR_UNDERFLOW	OK	OK	OK	OK	NG *3
OIR Interrupt setting [Function] - R_VDCE_IntcOirScanlineSet - R_VDCE_IntcOirScanlineGet	OK	OK	OK	OK	NG *3
Capture Interrupt control [Function] - R_VDCE_IntcCallbackSet - R_VDCE_IntcCallbackGet - R_VDCE_IntcEnable - R_VDCE_IntcDisable [Setting] - R_VDCE_INTC_NO_VI_VSYNC_SIGNAL - R_VDCE_INTC_NO_VI_VSYNC_SIGNAL_1 - R_VDCE_INTC_CAP_VBLANK - R_VDCE_INTC_CAP_END_OF_FIELD - R_VDCE_INTC_ERR_CAP_WRITE_OVERFLOW	OK	OK	OK	OK	NG *3
Capture Interrupt setting [Function] - R_VDCE_IntcCapScanlineGet	OK	OK	OK	OK	NG *3
Display Gamma Correction [Function] - R_VDCE_DisplayGammaCorrectSet	OK	OK	OK	OK	NG *1
Display Serial RGB output [Function] - R_VDCE_DisplayOutFormatSet [Setting] - R_VDCE_OUT_FORMAT_SERIAL_RGB	NG *2	NG *2	OK	NG *2	NG *2

Display Serial RGB output [Function] - R_VDCE_DisplaySerialRGBSet	NG *1	NG *1	OK	NG *1	NG *1
OIR layer [Function] - R_VDCE_OirBaseSet - R_VDCE_OirMemGeometrySet - R_VDCE_OirFormatSet - R_VDCE_OirViewPortSet - R_VDCE_OirRingBufferEnable - R_VDCE_OirRingBufferDisable - R_VDCE_OirVSyncDelaySet - R_VDCE_OirModeSet - R_VDCE_OirEnable - R_VDCE_OirDisable	OK	OK	OK	OK	NG *3
Scaling-Up/Down [Function] - R_VDCE_LayerImgScaleX - R_VDCE_LayerImgScaleY [Setting] - R_VDCE_SCALING_LARGER - R_VDCE_SCALING_SMALLER	OK	OK	OK	OK	NG *2
Scaling-Up/Down [Function] - R_VDCE_LayerImgScaleModeSet	OK	OK	OK	OK	NG *1
Input layer selection [Function] - R_VDCE_LayerMatrixSet - R_VDCE_LayerMatrixBT601Set - R_VDCE_LayerMatrixJPEGSet - R_VDCE_LayerMatrixUnitySet [Setting] - R_VDCE_LAYER_INPUT	OK	OK	OK	OK	NG *3
Capture Functions [Function] - R_VDCE_CapBufGeometrySetup - R_VDCE_CapModeSet - R_VDCE_CapBufSet - R_VDCE_CapBufFieldSetup1 - R_VDCE_CapBufFieldSetup2 - R_VDCE_CapEnable - R_VDCE_CapDisable - R_VDCE_CapViewPortSet - R_VDCE_CapRateSet - R_VDCE_CapExtVsyncSet	OK	OK	OK	OK	NG *1
Data Enable capturing [Function] - R_VDCE_CapModeSet [Setting] - R_VDCE_CAP_MODE_DE_MODE	OK	OK	OK	NG *2	NG *1
Capture start position with DE mode [Function] - R_VDCE_CapBufGeometrySetup - R_VDCE_CapBufFieldSetup1 - R_VDCE_CapBufFieldSetup2 - R_VDCE_CapViewPortSet [Setting] - StartX=0~15 - StartY=0~3	OK	OK	OK	NG *3	NG *1

*1: The function will fail and return error code.
*2: If the target mode is selected, the function will fail and return error code.
*3: The function will not fail, but the feature will not work.

3.4 Header File List

Table 3-43 Header File List

No.	Header File Name	Description
(1)	r_typedefs.h	Header file for predefined data types.
(2)	r_vdce_api.h	Header file for VDCE API.
(3)	r_ddb_api.h	Database for display timings (ddb).

4.Functions

4.1 Function List

This section describes about the VDCE API functions which are in *Table 4-1* and executable state of each function is described in the specification of each function.

Table 4-1 List of VDCE API Functions

Function Name	Purpose
R_VDCE_Init	Initializes the driver and the hardware as far as necessary.
R_VDCE_DeInit	This function de-initializes the driver and the hardware as far as necessary and frees the driver structure by deleting the used flag.
R_VDCE_VersionStringGet	This function returns the version string of this VDCE driver.
R_VDCE_MacroVersionGet	This function returns the major and minor version of the H/W macro.
R_VDCE_ErrorCallbackSet	This function sets a callback function that is called in case of an error.
R_VDCE_IntcCallbackSet	This function sets a callback for a special interrupt type.
R_VDCE_IntcCallbackGet	This function gets the pointer to the callback for the given interrupt type.
R_VDCE_IntcEnable	This function enables the activation of an interrupt of type IntType.
R_VDCE_IntcDisable	This function disables the activation of an interrupt of type IntType.
R_VDCE_IntcScanlineSet	This function sets the interrupt occurrence timing by the location of image line at Graphic 3 layer.
R_VDCE_IntcScanlineGet	This function gets the current location of image line being read on Graphics layer 3.
R_VDCE_IntcOirScanlineSet	This function sets the interrupt occurrence timing by the location of image line at OIR layer.
R_VDCE_IntcOirScanlineGet	This function gets the current location of image line output from OIR layer.
R_VDCE_IntcCapScanlineSet	This function is not supported.
R_VDCE_IntcCapScanlineGet	This function gets the current location of capturing line input to Scaler 0 (or Scaler 1) layer.
R_VDCE_Isr	This function is called from ISR (Interrupt Service Routines) and processes the interrupt factor which excludes the error of VDCE.
R_VDCE_IsrError	This function is called from ISR (Interrupt Service Routines) and processes the error interrupt factor of VDCE.
R_VDCE_DisplayTimingSet	This function sets the display timing, including display resolution, sync position blank widths and pixel clocks.
R_VDCE_DisplayTimingAdjust	This function adjusts the display timing.
R_VDCE_DisplayOutEndianSet	This function sets the video output data endian.
R_VDCE_DisplayOutSwapBR	This function swaps the video output data red and blue channel.
R_VDCE_DisplayColorSet	This function sets the display background color.
R_VDCE_DisplayHsyncSet	This function sets the TCON reference timing of Hsync.
R_VDCE_DisplayHsyncGet	This function gets the TCON reference timing of Hsync.
R_VDCE_DisplaySignalSet	This function sets the TCON signal configuration.
R_VDCE_DisplaySignalGet	This function gets the TCON signal configuration.
R_VDCE_DisplayTconPinSet	This function sets the TCON signal type which is output from the specified TCON pin.
R_VDCE_DisplayTconPinGet	This function gets the TCON signal type which is output from the specified TCON pin.
R_VDCE_DisplayPolaritySet	This function sets the polarity of the specified TCON signal type.
R_VDCE_DisplayCalibrationSet	This function sets the brightness, contrast and dithering mode.
R_VDCE_DisplayGammaCorrectSet	This function sets the RGB gamma correction parameters.
R_VDCE_DisplayOutFormatSet	This function sets the video output data format.

Function Name	Purpose
R_VDCE_DisplaySerialRGBSet	This function sets the Serial RGB output setting.
R_VDCE_DisplayVsyncProtectionSet	This function sets the Vsync protection.
R_VDCE_DisplayEnable	This function enables the display output.
R_VDCE_DisplayDisable	This function disables the display output.
R_VDCE_DisplayTimingGet	This function gets the display timing.
R_VDCE_LayerBaseSet	This function sets the layers base address.
R_VDCE_OirBaseSet	This function sets the output image renderer read layer base address.
R_VDCE_LayerMemGeometrySet	This function sets the layers memory geometry.
R_VDCE_OirMemGeometrySet	This function sets the output image renderers read layer memory geometry.
R_VDCE_LayerFormatSet	This function sets the color format for the layer.
R_VDCE_OirFormatSet	This function sets the color format for the OIR.
R_VDCE_LayerViewPortSet	This function sets the layers viewport parameters.
R_VDCE_OirViewPortSet	This function sets the OIR viewport parameters.
R_VDCE_LayerRingBufferEnable	This function switches from a full frame buffers for a layer to a ring buffer usage.
R_VDCE_OirRingBufferEnable	This function switches from a full frame buffers for a layer to a ring buffer usage.
R_VDCE_LayerRingBufferDisable	This function switches back from ring buffer a full frame buffers usage of the layer.
R_VDCE_OirRingBufferDisable	This function switches back from ring buffer a full frame buffers usage of the layer.
R_VDCE_LayerVSyncDelaySet	This function sets the Vsync signal delay between input Vsync to the scaler and output Vsync from the scaler.
R_VDCE_OirVSyncDelaySet	This function adjusts the read delay between output from GR3 and input of OIR layer.
R_VDCE_LayerModeSet	This function sets the optional mode for layer.
R_VDCE_OirModeSet	This function sets the OIR layer mode.
R_VDCE_LayerEnable	This function enables the layer display.
R_VDCE_OirEnable	This function enables the output image renderer layer display instead of the lower layer display.
R_VDCE_LayerDisable	This function disables the layer display.
R_VDCE_OirDisable	This function disables the output image renderer layer display and show regular layer setup.
R_VDCE_LayerMatrixSet	This function sets the color conversion matrix of layer 0, layer 1 and Input controller.
R_VDCE_LayerMatrixBT601Set	This function sets the color conversion matrix of layer 0 and layer 1 to BT601 spec.
R_VDCE_LayerMatrixJPEGSet	This function sets the color conversion matrix of layer 0 and layer 1 to JPEG spec.
R_VDCE_LayerMatrixUnitySet	This function sets the color conversion matrix of the video input to bypass mode.
R_VDCE_LayerImgScaleX	This function sets the scaling parameters in X direction.
R_VDCE_LayerImgScaleY	This function sets the scaling parameters in Y direction.
R_VDCE_LayerImgScaleModeSet	This function sets the optional mode for enlargement and reduction.
R_VDCE_LayerBufSet	This function sets the layers base address.
R_VDCE_LayerAlphaChannelEnable	This function re-enables the alpha channel per pixel data that is disabled by R_VDCE_LayerAlphaChannelDisable.
R_VDCE_LayerAlphaChannelDisable	This function disables the alpha channel per pixel data.
R_VDCE_LayerPremultipliedAlphaEnable	This function enables the layers pre-multiplied alpha channel.
R_VDCE_LayerPremultipliedAlphaDisable	This function disables the layers pre-multiplied alpha channel.
R_VDCE_LayerAlphaConstEnable	This function enables the layers alpha constant alpha function.
R_VDCE_LayerAlphaConstDisable	This function disables the Layers alpha constant alpha function.

Renesas Graphics Library Video Data Controller E (VDCE) Driver

Function Name	Purpose
R_VDCE_LayerClutSet	This function updates and switches between these two CLUTS.
R_VDCE_LayerChromaKeyEnable	This function enables the layers Chroma key.
R_VDCE_LayerChromaKeyDisable	This function disables the layers Chroma key.
R_VDCE_CapBufGeometrySetup	This function sets the parameters for the capture buffers and the size of the incoming video data.
R_VDCE_CapModeSet	This function sets the capturing mode.
R_VDCE_CapBufSet	This function sets address of double buffers for the capture.
R_VDCE_CapBufFieldSetup1	This function sets address of Buffer1 and value of StartY for the capture.
R_VDCE_CapBufFieldSetup2	This function sets address of Buffer2 and value of StartY for the capture.
R_VDCE_CapEnable	This function enables the video capturing.
R_VDCE_CapDisable	This function disables the video capturing.
R_VDCE_CapViewPortSet	This function sets the capture viewport parameters.
R_VDCE_CapRateSet	This function sets the writing rate of buffer for the capture.
R_VDCE_CapExtVsyncSet	This function sets the Hsync cycle of input signal.

4.2 VDCE API Functions

This chapter describes the application interface functions.

4.2.1 Basic functions

4.2.1.1 R_VDCE_Init

Function Prototypes

```
r_vdce_Error_t R_VDCE_Init(const uint32_t Unit)
```

Input Parameter

<p align="center">Table 4-2 Input parameter of R_VDCE_Init</p>

Parameter	Description
Unit	Specifies the VDCE unit number.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_RANGE_UNIT	- The unit number was outside the range.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_FATAL_HW	- Fatal error has occurred at H/W.
R_VDCE_ERR_UNIT_LOCKED	- VDCE was already initialized.
R_VDCE_ERR_PIXEL_CLOCK	- It failed in the pixel clock setting.

Description

This function initializes the VDCE driver and VDCE hardware.
If the function successfully executes, the return code will be R_VDCE_ERR_OK and the VDCE unit status will be in the initialized state.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Sync/Async

Synchronous

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.1.2   R_VDCE_DeInit

Function Prototypes

```
r_vdce_Error_t R_VDCE_DeInit(const uint32_t Unit)
```

Input Parameter

Table 4-3 Input parameter of R_VDCE_DeInit

Parameter	Description
Unit	Specifies the VDCE unit number.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_RANGE_UNIT	- The unit number was outside the range.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_PIXEL_CLOCK	- It failed in the pixel clock setting.

Description

This function de-initializes the driver and the hardware as far as necessary and frees the driver structure. This function disables all layers and the display.
If the function successfully executes, the return code will be R_VDCE_ERR_OK and the VDCE unit status will be in the uninitialized state.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.1.3　R_VDCE_VersionStringGet

Function Prototypes

```
const uint8_t *R_VDCE_VersionStringGet(void)
```

Parameter

None

Description

This function returns the version string of this VDCE driver.

Return Codes

Pointer of version string.

Reentrancy

Re-entrant.

Sync/Async

Synchronous

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

None

4.2.1.4 R_VDCE_MacroVersionGet

Function Prototypes

```
r_vdce_Error_t R_VDCE_MacroVersionGet(uint32_t *const Major,
                                      uint32_t *const Minor)
```

Input Parameter

None

Input -Output Parameter

None

Output Parameter

Table 4-4 Output parameter of R_VDCE_MacroVersionGet

Parameter	Description
Major	The major version.
Minor	The minor version.

Return Codes

R_VDCE_ERR_OK   - No error has occurred.
R_VDCE_ERR_PARAM_INCORRECT - Either parameter Major or parameter Minor was R_NULL

Description

This function returns the major and minor version of the H/W macro.

Reentrancy

Re-entrant.

Sync/Async

Synchronous

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.1.5 R_VDCE_ErrorCallbackSet

Function Prototypes

```
r_vdce_Error_t R_VDCE_ErrorCallbackSet(void  (* const       ErrorCallback )
                                      (const uint32_t       Unit,
                                       const r_vdce_Error_t Error))
```

Input Parameter

<p align="center">Table 4-5 Input parameter of R_VDCE_ErrorCallbackSet</p>

Parameter	Description
ErrorCallback	Specifies a function that is called in case an error occurred. Set R_NULL if callback is uninstalled.

<p align="center">Table 4-6 Input parameter of ErrorCallback</p>

Parameter	Description
Unit	VDCE unit number where the error occurred.
Error	Error type.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK   - No error has occurred.
R_VDCE_ERR_FATAL_OS   - Fatal error has occurred at OS interface.

Description

This function sets a callback function that is called in case of an error.
Error notified in this callback can be checked also by return value of each API function, so use of callback is not mandatory. The error callback is global for all VDCE units.
The error callback is notified during the VDCE unit is not Uninitialized state.
The installed error callback can be uninstalled by R_NULL setting in this function. And all VDCE units are de-initialized by R_VECE_DeInit, the callback is also uninstalled.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_AllLock and R_VDCE_Sys_AllUnlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.2 Interrupt functions

4.2.2.1 R_VDCE_IntcCallbackSet

Function Prototypes

```
r_vdce_Error_t R_VDCE_IntcCallbackSet(const uint32_t       Unit,
                                      const r_vdce_IntType_t IntType,
                                      void          (* const Isr)(void))
```

Input Parameter

Table 4-7 Parameter R_VDCE_IntcCallbackSet

Parameter	Description
Unit	Specifies the VDCE unit number.
IntType	Specifies the Interrupt event. 　R_VDCE_INTC_VBLANK 　R_VDCE_INTC_VBLANK_1 　R_VDCE_INTC_VBLANK_DELAY 　R_VDCE_INTC_OIR_VBLANK 　R_VDCE_INTC_SCANLINE 　R_VDCE_INTC_OIR_SCANLINE 　R_VDCE_INTC_OIR_VSCYNC_WRITE 　R_VDCE_INTC_NO_VI_VSYNC_SIGNAL 　R_VDCE_INTC_NO_VI_VSYNC_SIGNAL_1 　R_VDCE_INTC_CAP_VBLANK 　R_VDCE_INTC_CAP_END_OF_FIELD 　R_VDCE_INTC_ERR_OIR_UNDERFLOW 　R_VDCE_INTC_ERR_LAYER0_UNDERFLOW 　R_VDCE_INTC_ERR_LAYER1_UNDERFLOW 　R_VDCE_INTC_ERR_LAYER2_UNDERFLOW 　R_VDCE_INTC_ERR_LAYER3_UNDERFLOW 　R_VDCE_INTC_ERR_CAP_WRITE_OVERFLOW
Isr	Specifies a function that is called in case an interrupt occurred. Set R_NULL if callback is uninstalled.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_PARAM_INCORRECT	- Value of parameter IntType was not matching.
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED	- VDCE was not initialized.

Description

This function sets a callback for a specified interrupt type.
This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t
r_vdce_IntType_t

4.2.2.2 R_VDCE_IntcCallbackGet

Function Prototypes

```
r_vdce_Error_t R_VDCE_IntcCallbackGet(const uint32_t        Unit,
                                      const r_vdce_IntType_t  IntType,
                                       void          (** const Isr)(void))
```

Input Parameter

Table 4-8 Input parameter of R_VDCE_IntcCallbackGet

Parameter	Description
Unit	Specifies the VDCE unit number.
IntType	Specifies the Interrupt event. 　R_VDCE_INTC_VBLANK 　R_VDCE_INTC_VBLANK_1 　R_VDCE_INTC_VBLANK_DELAY 　R_VDCE_INTC_OIR_VBLANK 　R_VDCE_INTC_SCANLINE 　R_VDCE_INTC_OIR_SCANLINE 　R_VDCE_INTC_OIR_VSCYNC_WRITE 　R_VDCE_INTC_NO_VI_VSYNC_SIGNAL 　R_VDCE_INTC_NO_VI_VSYNC_SIGNAL_1 　R_VDCE_INTC_CAP_VBLANK 　R_VDCE_INTC_CAP_END_OF_FIELD 　R_VDCE_INTC_ERR_OIR_UNDERFLOW 　R_VDCE_INTC_ERR_LAYER0_UNDERFLOW 　R_VDCE_INTC_ERR_LAYER1_UNDERFLOW 　R_VDCE_INTC_ERR_LAYER2_UNDERFLOW 　R_VDCE_INTC_ERR_LAYER3_UNDERFLOW 　R_VDCE_INTC_ERR_CAP_WRITE_OVERFLOW

Input -Output Parameter

　None

Output Parameter

Table 4-9 Output parameter of R_VDCE_IntcCallbackGet

Parameter	Description
Isr	A function that is called in case an interrupt occurred.

Return Codes

```
R_VDCE_ERR_OK               - No error has occurred.
R_VDCE_ERR_PARAM_INCORRECT  - The parameter IntType value was not matching.
R_VDCE_ERR_RANGE_UNIT       - The unit-number was outside the range.
R_VDCE_ERR_UNIT_NOTLOCKED   - VDCE was not initialized.
```

Description

　This function gets the pointer to the callback for the given interrupt type.

Reentrancy

　Reentrant.

Sync/Async

　Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t
r_vdce_IntType_t

4.2.2.3 R_VDCE_IntcEnable

Function Prototypes

```
r_vdce_Error_t R_VDCE_IntcEnable(const uint32_t        Unit,
                                 const r_vdce_IntType_t  IntType )
```

Input Parameter

Table 4-10 Input parameter of R_VDCE_IntcEnable

Parameter	Description
Unit	Specifies the VDCE unit number.
IntType	Specifies the Interrupt event, which can be selected. R_VDCE_INTC_VBLANK R_VDCE_INTC_VBLANK_1 R_VDCE_INTC_VBLANK_DELAY R_VDCE_INTC_OIR_VBLANK R_VDCE_INTC_SCANLINE R_VDCE_INTC_OIR_SCANLINE R_VDCE_INTC_OIR_VSCYNC_WRITE R_VDCE_INTC_NO_VI_VSYNC_SIGNAL R_VDCE_INTC_NO_VI_VSYNC_SIGNAL_1 R_VDCE_INTC_CAP_VBLANK R_VDCE_INTC_CAP_END_OF_FIELD R_VDCE_INTC_ERR_OIR_UNDERFLOW R_VDCE_INTC_ERR_LAYER0_UNDERFLOW R_VDCE_INTC_ERR_LAYER1_UNDERFLOW R_VDCE_INTC_ERR_LAYER2_UNDERFLOW R_VDCE_INTC_ERR_LAYER3_UNDERFLOW R_VDCE_INTC_ERR_CAP_WRITE_OVERFLOW

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK   - No error has occurred.
R_VDCE_ERR_PARAM_INCORRECT - The parameter IntType value was not matching.
R_VDCE_ERR_RANGE_UNIT   - The unit-number was outside the range.
R_VDCE_ERR_FATAL_OS   - Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED   - VDCE was not initialized.

Description

This function enables the specified interrupt.
In case of VODDR output mode, both VDCE unit should be transit to Init state before this function is called. About VODDR mode, see Porting Layer Guide.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t
r_vdce_IntType_t

4.2.2.4 R_VDCE_IntcDisable

Function Prototypes

```
r_vdce_Error_t R_VDCE_IntcDisable(const uint32_t        Unit,
                                  const r_vdce_IntType_t  IntType)
```

Input Parameter

<p align="center">Table 4-11 Input parameter of R_VDCE_IntcDisable</p>

Parameter	Description
Unit	Specifies the VDCE unit number.

IntType	Specifies the Interrupt event, which can be selected.
	    R_VDCE_INTC_VBLANK
	    R_VDCE_INTC_VBLANK_1
	    R_VDCE_INTC_VBLANK_DELAY
	    R_VDCE_INTC_OIR_VBLANK
	    R_VDCE_INTC_SCANLINE
	    R_VDCE_INTC_OIR_SCANLINE
	    R_VDCE_INTC_OIR_VSCYNC_WRITE
	    R_VDCE_INTC_NO_VI_VSYNC_SIGNAL
	    R_VDCE_INTC_NO_VI_VSYNC_SIGNAL_1
	    R_VDCE_INTC_CAP_VBLANK
	    R_VDCE_INTC_CAP_END_OF_FIELD
	    R_VDCE_INTC_ERR_OIR_UNDERFLOW
	    R_VDCE_INTC_ERR_LAYER0_UNDERFLOW
	    R_VDCE_INTC_ERR_LAYER1_UNDERFLOW
	    R_VDCE_INTC_ERR_LAYER2_UNDERFLOW
	    R_VDCE_INTC_ERR_LAYER3_UNDERFLOW
	    R_VDCE_INTC_ERR_CAP_WRITE_OVERFLOW

Input -Output Parameter

None

Output Parameter

None

Return Codes

```
R_VDCE_ERR_OK                 - No error has occurred.
R_VDCE_ERR_PARAM_INCORRECT - The parameter IntType value was not matching.
R_VDCE_ERR_RANGE_UNIT         - The unit-number was outside the range.
R_VDCE_ERR_FATAL_OS           - Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED    - VDCE was not initialized.
```

Description

This function disables the specified interrupt.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t
r_vdce_IntType_t

4.2.2.5 R_VDCE_IntcScanlineSet

Function Prototypes

```
r_vdce_Error_t R_VDCE_IntcScanlineSet(const uint32_t Unit,
                                      const uint32_t Scanline)
```

Input Parameter

Table 4-12 Input parameter of R_VDCE_IntcScanlineSet

Parameter	Description
Unit	Specifies the VDCE unit number.
Scanline	Specifies the interrupt occurrence timing by the location of image line at the Graphics layer 3. The range is 0 to 2047.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK   - No error has occurred.
R_VDCE_ERR_RANGE_PARAM   - Parameter Scanline was the outside the range.
R_VDCE_ERR_RANGE_UNIT   - The unit-number was outside the range.
R_VDCE_ERR_FATAL_OS   - Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED   - VDCE was not initialized.

Description

This function sets the interrupt occurrence timing at Graphic layer 3 (Image synthesizer 3).
When the location of image line read by Graphic layer 3 matches this setting, R_VDCE_INTC_SCANLINE callback is notified. The interrupt occurs even if Graphic layer 3 is disabled.
This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.2.6 R_VDCE_IntcScanlineGet

Function Prototypes

```
r_vdce_Error_t R_VDCE_IntcScanlineGet(const uint32_t Unit,
                                      const uint32_t *Scanline)
```

Input Parameter

Table 4-13 Input parameter of R_VDCE_IntcScanlineGet

Parameter	Description
Unit	Specifies the VDCE unit number.

Input -Output Parameter

None

Output Parameter

Table 4-14 Output parameter of R_VDCE_IntcScanlineGet

Parameter	Description
Scanline	The current location of image line being read on Graphics layer 3.

Return Codes

R_VDCE_ERR_OK   - No error has occurred.
R_VDCE_ERR_PARAM_INCORRECT - Parameter Scanline was R_NULL.
R_VDCE_ERR_RANGE_UNIT   - The unit-number was outside the range.
R_VDCE_ERR_UNIT_NOTLOCKED   - VDCE was not initialized.

Description

This function gets the current location of image line being read on Graphics layer 3 (Image synthesizer 3).

Reentrancy

Reentrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.2.7   R_VDCE_IntcOirScanlineSet

Function Prototypes

```
r_vdce_Error_t R_VDCE_IntcOirScanlineSet(const uint32_t Unit,
                                    const uint32_t Scanline)
```

Input Parameter

Table 4-15 Input parameter of R_VDCE_IntcOirScanlineSet

Parameter	Description
Unit	Specifies the VDCE unit number.
Scanline	Specifies the interrupt occurrence timing by the location of image line at OIR layer. The range is 0 to 2047.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK   - No error has occurred.
R_VDCE_ERR_RANGE_PARAM   - Parameter Scanline was outside the range.
R_VDCE_ERR_RANGE_UNIT   - The unit-number is the outside of the range.
R_VDCE_ERR_FATAL_OS   - Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED - VDCE was not initialized.

Description

This function sets the interrupt occurrence timing by the location of image line at OIR layer.
When the location of image line output from OIR layer matches this setting, R_VDCE_INTC_OIR_SCANLINE callback is notified.
This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.2.8 R_VDCE_IntcOirScanlineGet

Function Prototypes

```
r_vdce_Error_t R_VDCE_IntcOirScanlineGet(const uint32_t Unit,
                                         const uint32_t *Scanline)
```

Input Parameter

Table 4-16 Input parameter of R_VDCE_IntcOirScanlineGet

Parameter	Description
Unit	Specifies the VDCE unit number.

Input -Output Parameter

None

Output Parameter

Table 4-17 Output parameter of R_VDCE_IntcOirScanlineGet

Parameter	Description
Scanline	The current location of image line being read on OIR layer.

Return Codes

R_VDCE_ERR_OK   - No error has occurred.
R_VDCE_ERR_PARAM_INCORRECT - Parameter Scanline was R_NULL.
R_VDCE_ERR_RANGE_UNIT   - The unit-number was outside the range.
R_VDCE_ERR_UNIT_NOTLOCKED   - VDCE was not initialized.

Description

This function gets the current location of image line output from OIR layer.

Reentrancy

Reentrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t.

4.2.2.9 R_VDCE_IntcCapScanlineSet

Function Prototypes

```
r_vdce_Error_t R_VDCE_IntcCapScanlineSet(const uint32_t Unit,
                                         const uint32_t Scanline)
```

Input Parameter

<div align="center">

Table 4-18 Input parameter of R_VDCE_IntcCapScanlineSet

</div>

Parameter	Description
Unit	Specifies the VDCE unit number.
Scanline	Specifies the interrupt occurrence timing by the location of capturing line.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_NOT_SUPPORTED   - The function is not supported.

Description

This function is not supported.

Reentrancy

Reentrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.2.10 R_VDCE_IntcCapScanlineGet

Function Prototypes

```
r_vdce_Error_t R_VDCE_IntcCapScanlineGet(const uint32_t Unit,
                                         const uint32_t *Scanline)
```

Input Parameter

Table 4-19 Input parameter of R_VDCE_IntcCapScanlineGet

Parameter	Description
Unit	Specifies the VDCE unit number.

Input -Output Parameter

None

Output Parameter

Table 4-20 Output parameter of R_VDCE_IntcCapScanlineGet

Parameter	Description
Scanline	The current location of capturing line input to Scaler 0 (or Scaler 1) layer.

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_PARAM_INCORRECT	- Parameter Scanline was R_NULL.
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED	- VDCE was not initialized.

Description

This function gets the current location of capturing line input to Scaler 0 (or Scaler 1) layer.

Reentrancy

Non-reentrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.2.11 R_VDCE_Isr

Function Prototypes

```
r_vdce_Error_t R_VDCE_Isr(const uint32_t         Unit,
                          const r_vdce_IntType_t IntType)
```

Input Parameter

Table 4-21 Input parameter of R_VDCE_Isr

Parameter	Description
Unit	Specifies the VDCE unit number.
IntType	Specifies the Interrupt event. 　R_VDCE_INTC_VBLANK 　R_VDCE_INTC_VBLANK_1 　R_VDCE_INTC_VBLANK_DELAY 　R_VDCE_INTC_OIR_VBLANK 　R_VDCE_INTC_SCANLINE 　R_VDCE_INTC_OIR_SCANLINE 　R_VDCE_INTC_OIR_VSCYNC_WRITE 　R_VDCE_INTC_NO_VI_VSYNC_SIGNAL 　R_VDCE_INTC_NO_VI_VSYNC_SIGNAL_1 　R_VDCE_INTC_CAP_VBLANK 　R_VDCE_INTC_CAP_END_OF_FIELD 　R_VDCE_INTC_ERR_OIR_UNDERFLOW 　R_VDCE_INTC_ERR_LAYER0_UNDERFLOW 　R_VDCE_INTC_ERR_LAYER1_UNDERFLOW 　R_VDCE_INTC_ERR_LAYER2_UNDERFLOW 　R_VDCE_INTC_ERR_LAYER3_UNDERFLOW 　R_VDCE_INTC_ERR_CAP_WRITE_OVERFLOW

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK   - No error has occurred.
R_VDCE_ERR_RANGE_PARAM   - The parameter IntType was outside the range.

Description

This function processes the interrupt factor.
This function is called from ISR (Interrupt Service Routines).

Reentrancy

Non-reentrant.

Sync/Async

Synchronous

Call from Interrupt

Permitted.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t
r_vdce_IntType_t

4.2.2.12 R_VDCE_IsrError

Function Prototypes

```
r_vdce_Error_t R_VDCE_IsrError(const uint32_t Unit)
```

Input Parameter

Table 4-22 Input parameter of R_VDCE_IsrError

Parameter	Description
Unit	Specifies the VDCE unit number.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK   - No error has occurred.

Description

This function processes the error interrupt factor of VDCE.
This function is called from ISR (Interrupt Service Routines).

Reentrancy

Non-reentrant.

Sync/Async

Synchronous

Call from Interrupt

Permitted.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.3 Display functions

4.2.3.1 R_VDCE_DisplayTimingSet

Function Prototypes

```
r_vdce_Error_t R_VDCE_DisplayTimingSet(const uint32_t          Unit,
                                       const r_ddb_Timing_t  *const  Timing)
```

Input Parameter

Table 4-23 Input parameter of R_VDCE_DisplayTimingSet

Parameter	Description
Unit	Specifies the VDCE unit number.
Timing	Specifies the timing parameter contains all the information to drive the display.

Input - Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK   - No error has occurred.
R_VDCE_ERR_PARAM_INCORRECT - Parameter Timing was R_NULL.
R_VDCE_ERR_RANGE_UNIT   - The unit-number was outside the range.
R_VDCE_ERR_RANGE_PARAM   - Parameter Timing was outside the range.
R_VDCE_ERR_NOT_ACCEPTABLE   - A function was called in an incorrect state.
R_VDCE_ERR_FATAL_OS   - Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED - VDCE was not initialized.

Description

This function sets the display timing, including display resolution, sync position blank widths and pixel clocks.
This setting is valid until R_VDCE_DeInit is executed.

See *Table 3-3* about the range.
Following equations must be satisfied. Otherwise this function will return error.
- (Timing->H.BlankWidth + Timing->ScreenWidth) = Timing->H.Total
- (Timing->V.BlankWidth + Timing->ScreenHeight) = Timing->V.Total

This function calls R_VDCE_Sys_PixelClockSet to set the pixel clock. Since the pixel clock is output by frequency division of the PLL, it rounds the set value to a close value.
Range of Timing->PixelClock is depending on RH850/D1x device and output format (LVTTL, Serial RGB etc). See H/W User's manual.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t
r_ddb_Timing_t

4.2.3.2 R_VDCE_DisplayTimingAdjust

Function Prototypes

```
r_vdce_Error_t R_VDCE_DisplayTimingAdjust(const uint32_t  Unit,
                                          const uint32_t VLines)
```

Input Parameter

Table 4-24 Input parameter of R_VDCE_DisplayTimingAdjust

Parameter	Description
Unit	Specifies the VDCE unit number.
VLines	Specifies the value of adjustment in vertical lines. The range is 0 ot 16.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_RANGE_PARAM	- A parameter VLines was outside the range.
R_VDCE_ERR_NOT_ACCEPTABLE	- A function was called in an incorrect state.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED	- VDCE was not initialized.
R_VDCE_ERR_PIXEL_CLOCK	- It failed in the pixel clock setting.

Description

This function adjusts the display timing using adjustment in vertical lines.
The set value is valid until R_VDCE_DeInit is executed.

This function is prepared for adjusting V-sync timing when VOWE is used.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.3.3 R_VDCE_DisplayOutEndianSet

Function Prototypes

```
r_vdce_Error_t R_VDCE_DisplayOutEndianSet(const uint32_t        Unit,
                                          const r_vdce_OutEndian_t OutEndian)
```

Input Parameter

Table 4-25 Input parameter of R_VDCE_DisplayOutEndianSet

Parameter	Description
Unit	Specifies the VDCE unit number.
OutEndian	Specifies video output data endian. 　R_VDCE_OUT_ENDIAN_LITTLE 　R_VDCE_OUT_ENDIAN_BIG

Input -Output Parameter

　　None

Output Parameter

　　None

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_PARAM_INCORRECT	- The parameter OutEndian value was not matching.
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_NOT_ACCEPTABLE	- A function was called in an incorrect state.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED	- VDCE was not initialized.

Description

　　This function sets the video output data endian. The default output data is little endian.
　　See *3.2.2.3* for the detail.
　　This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

　　Non-reentrant as default.
　　If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

　　Synchronous.

Call from Interrupt

　　Prohibited.

Preconditions

　　See *Table 2-4* about status conditions.

See also

r_vdce_Error_t
r_vdce_OutEndian_t

4.2.3.4   R_VDCE_DisplayOutSwapBR

Function Prototypes

```
r_vdce_Error_t R_VDCE_DisplayOutSwapBR (const uint32_t        Unit,
                                        const r_vdce_OutSwap_t OutSwap)
```

Input Parameter

Table 4-26 Input parameter of R_VDCE_DisplayOutSwapBR

Parameter	Description
Unit	Specifies the VDCE unit number.
OutSwap	Specifies the video output data swap. R_VDCE_OUT_SWAP_BR_OFF R_VDCE_OUT_SWAP_BR_ON

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK   - No error has occurred.
R_VDCE_ERR_PARAM_INCORRECT - The parameter OutSwap value was not matching.
R_VDCE_ERR_RANGE_UNIT   - The unit-number was outside the range.
R_VDCE_ERR_NOT_ACCEPTABLE   - A function was called in an incorrect state.
R_VDCE_ERR_FATAL_OS   - Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED   - VDCE was not initialized.

Description

This function swaps the video output data red and blue channel.
See *3.2.2.3* for the detail.
This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t
r_vdce_OutSwap_t

4.2.3.5 R_VDCE_DisplayColorSet

Function Prototypes

```
r_vdce_Error_t R_VDCE_DisplayColorSet(const uint32_t  Unit,
                                      const uint8_t   Red,
                                      const uint8_t   Green,
                                      const uint8_t   Blue)
```

Input Parameter

Table 4-27 Input parameter of R_VDCE_DisplayColorSet

Parameter	Description
Unit	Specifies the VDCE unit number.
Red	Specifies the Red color components of the background color.
Green	Specifies the Green color components of the background color.
Blue	Specifies the Blue color components of the background color.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK   - No error has occurred.
R_VDCE_ERR_RANGE_UNIT   - The unit-number was outside the range.
R_VDCE_ERR_FATAL_OS   - Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED   - VDCE was not initialized.

Description

This function sets the display background color. It is seen if no layer (or a transparent one) is on top of it.
The default background color is black (all value is 0).
This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.3.6 R_VDCE_DisplayHsyncSet

Function Prototypes

```
r_vdce_Error_t R_VDCE_DisplayHsyncSet(const uint32_t           Unit,
                                      const r_vdce_Hsync_t * const Hsync)
```

Input Parameter

Table 4-28 Input parameter of R_VDCE_DisplayHsyncSet

Parameter	Description
Unit	Specifies the VDCE unit number.
Hsync	Specifies the value of TCON reference timing of Hsync

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK   - No error has occurred.
R_VDCE_ERR_PARAM_INCORRECT   - Parameter Hsync was R_NULL.
R_VDCE_ERR_RANGE_UNIT   - The unit-number was outside the range.
R_VDCE_ERR_RANGE_PARAM   - Parameter Hsync was outside the range.
R_VDCE_ERR_NOT_ACCEPTABLE   - A function was called in an incorrect state.
R_VDCE_ERR_FATAL_OS   - Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED   - VDCE was not initialized.
R_VDCE_ERR_DISPLAY_NO_TIMING - R_VDCE_DisplayTimingSet is not done

Description

This function sets the TCON reference timing of Hsync.

Following functions initialize the TCON setting by display setting.
- R_VDCE_DisplayTimingSet
- R_VDCE_DisplayTimingAdjust
This function can modify the value after initial setting.

This setting is valid until one of the following functions is executed.
- R_VDCE_DisplayTimingSet
- R_VDCE_DisplayTimingAdjust
- R_VDCE_DeInit

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.
Execute R_VDCE_DisplayTimingSet before executing this function.

See also

r_vdce_Error_t
r_vdce_Hsync_t

4.2.3.7 R_VDCE_DisplayHsyncGet

Function Prototypes

```
r_vdce_Error_t R_VDCE_DisplayHsyncGet(const uint32_t        Unit,
                                      r_vdce_Hsync_t * const Hsync)
```

Input Parameter

<div align="center">Table 4-29 Input parameter of R_VDCE_DisplayHsyncGet</div>

Parameter	Description
Unit	Specifies the VDCE unit number.

Input -Output Parameter

None

Output Parameter

<div align="center">Table 4-30 Output parameter of R_VDCE_DisplayHsyncGet</div>

Parameter	Description
Hsync	The value of TCON reference timing of Hsync.

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_PARAM_INCORRECT	- Parameter Hsync was R_NULL.
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED	- VDCE was not initialized.
R_VDCE_ERR_DISPLAY_NO_TIMING	- R_VDCE_DisplayTimingSet is not done

Description

This function gets the TCON reference timing of Hsync.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.
Execute R_VDCE_DisplayTimingSet before executing this function.

See also

r_vdce_Error_t
r_vdce_Hsync_t

4.2.3.8 R_VDCE_DisplaySignalSet

Function Prototypes

```
r_vdce_Error_t R_VDCE_DisplaySignalSet(const uint32_t            Unit,
                                       const r_vdce_Pin_t         SigType,
                                       const r_vdce_Signal_t * const Signal)
```

Input Parameter

Table 4-31 Input parameter of R_VDCE_DisplaySignalSet

Parameter	Description
Unit	Specifies the VDCE unit number.
SigType	Specifies the TCON signal type. R_VDCE_PIN_VSYNC R_VDCE_PIN_HSYNC R_VDCE_PIN_VSYNC_E R_VDCE_PIN_HSYNC_E R_VDCE_PIN_CPV_GCK R_VDCE_PIN_POLA R_VDCE_PIN_POLB
Signal	Specifies the value of TCON signal configuration.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_PARAM_INCORRECT	- The parameter SigType value was not matching
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_RANGE_PARAM	- A parameter was outside the range.
R_VDCE_ERR_NOT_ACCEPTABLE	- A function was called in an incorrect state.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED	- VDCE was not initialized.
R_VDCE_ERR_DISPLAY_NO_TIMING	- R_VDCE_DisplayTimingSet is not done

Description

This function sets the TCON signal configuration.

Following functions initialize the TCON setting by display setting.
- R_VDCE_DisplayTimingSet
- R_VDCE_DisplayTimingAdjust

This function can modify the value after initial setting.

This setting is valid until one of the following functions is executed.
- R_VDCE_DisplayTimingSet
- R_VDCE_DisplayTimingAdjust
- R_VDCE_DeInit

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.
Execute R_VDCE_DisplayTimingSet before executing this function.

See also

r_vdce_Error_t
r_vdce_Signal_t
r_vdce_Pin_t

4.2.3.9 R_VDCE_DisplaySignalGet

Function Prototypes

```
r_vdce_Error_t R_VDCE_DisplaySignalGet(const uint32_t        Unit,
                                       const r_vdce_Pin_t     SigType,
                                       r_vdce_Signal_t  * const Signal)
```

Input Parameter

Table 4-32 Input parameter of R_VDCE_DisplaySignalGet

Parameter	Description
Unit	Specifies the VDCE unit number.
SigType	Specifies the TCON signal type. 　R_VDCE_PIN_VSYNC 　R_VDCE_PIN_HSYNC 　R_VDCE_PIN_VSYNC_E 　R_VDCE_PIN_HSYNC_E 　R_VDCE_PIN_CPV_GCK 　R_VDCE_PIN_POLA 　R_VDCE_PIN_POLB

Input -Output Parameter

None

Output Parameter

Table 4-33 Output parameter of R_VDCE_DisplaySignalGet

Parameter	Description
Signal	The value of TCON signal configuration.

Return Codes

R_VDCE_ERR_OK   - No error has occurred.
R_VDCE_ERR_PARAM_INCORRECT - The parameter was not matching.
R_VDCE_ERR_RANGE_UNIT   - The unit-number was outside the range.
R_VDCE_ERR_FATAL_OS   - Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED   - VDCE was not initialized.
R_VDCE_ERR_DISPLAY_NO_TIMING - R_VDCE_DisplayTimingSet is not done.

Description

This function gets the TCON signal configuration.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.
Execute R_VDCE_DisplayTimingSet before executing this function.

See also

r_vdce_Error_t
r_vdce_Signal_t
r_vdce_Pin_t

4.2.3.10 R_VDCE_DisplayTconPinSet

Function Prototypes

```
r_vdce_Error_t R_VDCE_DisplayTconPinSet(const uint32_t          Unit,
                                        const r_vdce_TconPin_t       Pin,
                                        const r_vdce_TconSig_t * const TconSig)
```

Input Parameter

Table 4-34 Input parameter of R_VDCE_DisplayTconPinSet

Parameter	Description
Unit	Specifies the VDCE unit number.

Pin	Specifies the TCON pin type.
	  R_VDCE_TCON_PIN_0
	  R_VDCE_TCON_PIN_1
	  R_VDCE_TCON_PIN_2
	  R_VDCE_TCON_PIN_3
	  R_VDCE_TCON_PIN_4
	  R_VDCE_TCON_PIN_5
	  R_VDCE_TCON_PIN_6

TconSig	Specifies the value of TCON signal type.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_PARAM_INCORRECT	- The parameter was incorrect.
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_NOT_ACCEPTABLE	- A function was called in an incorrect state.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED	- VDCE was not initialized.

Description

This function sets the connection of TCON pin and signal type.
The signal type of Pin0(Vsync) and Pin2(Hsync) cannot be changed. Only edge is changeable.
See *Table 3-4* about default connection.

This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t
r_vdce_TconPin_t
r_vdce_TconSig_t

4.2.3.11 R_VDCE_DisplayTconPinGet

Function Prototypes

```
r_vdce_Error_t R_VDCE_DisplayTconPinGet(const uint32_t          Unit,
                                        const r_vdce_TconPin_t   Pin,
                                        r_vdce_TconSig_t  * const TconSig)
```

Input Parameter

<div align="center">Table 4-35 Input parameter of R_VDCE_DisplayTconPinGet</div>

Parameter	Description
Unit	Specifies the VDCE unit number.

Pin	Specifies the TCON pin type.
	    R_VDCE_TCON_PIN_0
	    R_VDCE_TCON_PIN_1
	    R_VDCE_TCON_PIN_2
	    R_VDCE_TCON_PIN_3
	    R_VDCE_TCON_PIN_4
	    R_VDCE_TCON_PIN_5
	    R_VDCE_TCON_PIN_6

Input -Output Parameter

None

Output Parameter

<div align="center">Table 4-36 Output parameter of R_VDCE_DisplayTconPinGet</div>

Parameter	Description
TconSig	The value of TCON signal type.

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_PARAM_INCORRECT	- The parameter was incorrect.
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED	- VDCE was not initialized.

Description

This function gets the TCON signal type which is output from the specified TCON pin.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.
 Execute R_VDCE_DisplayTimingSet before executing this function.

See also

r_vdce_Error_t
 r_vdce_TconPin_t
 r_vdce_TconSig_t

4.2.3.12 R_VDCE_DisplayPolaritySet

Function Prototypes

```
r_vdce_Error_t R_VDCE_DisplayPolaritySet(const uint32_t        Unit,
                                         const r_vdce_Pin_t      Pin,
                                         const r_vdce_Polarity_t Polarity)
```

Input Parameter

Table 4-37 Input parameter of R_VDCE_DisplayPolaritySet

Parameter	Description
Unit	Specifies the VDCE unit number.
Pin	Specifies the TCON signal type. R_VDCE_PIN_VSYNC_E R_VDCE_PIN_HSYNC_E R_VDCE_PIN_CPV_GCK R_VDCE_PIN_POLA R_VDCE_PIN_POLB
Polarity	Specifies the polarity. R_VDCE_POLARITY_POSITIVE R_VDCE_POLARITY_NEGATIVE

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_PARAM_INCORRECT	- The parameter was not matching.
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_NOT_ACCEPTABLE	- A function was called in an incorrect state.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED	- VDCE was not initialized.

Description

This function sets the polarity of the specified TCON signal type.

About Following signals, the polarity should be set by R_VDCE_DisplayTimingSet with Timing->Flags.
- R_VDCE_PIN_ENABLE
- R_VDCE_PIN_VSYNC
- R_VDCE_PIN_HSYNC

Setting by this function is not effective.

This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t
r_vdce_Pin_t
r_vdce_Polarity_t

4.2.3.13 R_VDCE_DisplayCalibrationSet

Function Prototypes

```
r_vdce_Error_t R_VDCE_DisplayCalibrationSet(const uint32_t              Unit,
                                            const r_vdce_Bright_t * const    Bright,
                                            const r_vdce_Contrast_t * const  Contrast
                                            const r_vdce_Dither_t *const     Dither)
```

Input Parameter

Table 4-38 Input parameter of R_VDCE_DisplayCalibrationSet

Parameter	Description
Unit	Specifies the VDCE unit number.
Bright	Specifies the value of Brightness DC. When not changing, set NULL.
Contrast	Specifies the value of Contrast gain. When not changing, set NULL.
Dither	Specifies the value of panel dithering. When not changing, set NULL.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK   - No error has occurred.
R_VDCE_ERR_RANGE_UNIT   - The unit-number was outside the range.
R_VDCE_ERR_RANGE_PARAM   - Parameter was Bright or Dither outside the range.
R_VDCE_ERR_FATAL_OS   - Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED   - VDCE is not initialized.
R_VDCE_ERR_PARAM_INCORRECT   - Parameter Dither was incorrect.
R_VDCE_ERR_NG   - An error has occurred, but no specific error code is defined for it.

Description

This function sets the brightness, contrast and dithering mode.
The set value is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t
r_vdce_Bright_t
r_vdce_Contrast_t
r_vdce_Dither_t

4.2.3.14 R_VDCE_DisplayGammaCorrectSet

Function Prototypes

```
r_vdce_Error_t R_VDCE_DisplayGammaCorrectSet(const uint32_t           Unit,
                                             const r_vdce_Gamma_t * const Gamma)
```

Input Parameter

Table 4-39 Input parameter of R_VDCE_DisplayGammaCorrectSet

Parameter	Description
Unit	Specifies the VDCE unit number.
Gamma	Specifies the value of Gamma correction parameter. When the gamma correction is disabled, set to R_NULL

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK   - No error has occurred.
R_VDCE_ERR_RANGE_UNIT   - The unit-number was outside the range.
R_VDCE_ERR_RANGE_PARAM   - Parameter Gamma was outside the range.
R_VDCE_ERR_NOT_ACCEPTABLE - A function was called in an incorrect state.
R_VDCE_ERR_FATAL_OS   - Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED - VDCE was not initialized.

Description

This function sets the RGB gamma correction parameters.
The set value is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t
r_vdce_Gamma_t

4.2.3.15 R_VDCE_DisplayOutFormatSet

Function Prototypes

```
r_vdce_Error_t R_VDCE_DisplayOutFormatSet(const uint32_t          Unit,
                                          const r_vdce_OutFormat_t OutFormat)
```

Input Parameter

Table 4-40 Input parameter of R_VDCE_DisplayOutFormatSet

Parameter	Description
Unit	Specifies the VDCE unit number.
OutFormat	Specifies the output signal format R_VDCE_OUT_FORMAT_RGB888 R_VDCE_OUT_FORMAT_RGB666 R_VDCE_OUT_FORMAT_RGB565 R_VDCE_OUT_FORMAT_SERIAL_RGB The default output signal format is RGB888.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK   - No error has occurred.
R_VDCE_ERR_PARAM_INCORRECT   - Parameter OutFormat provided to the function was incorrect.
R_VDCE_ERR_RANGE_UNIT   - The unit-number was outside the range.
R_VDCE_ERR_NOT_ACCEPTABLE   - A function was called in an incorrect state.
R_VDCE_ERR_FATAL_OS   - Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED   - VDCE is not initialized.

Description

This function sets the video output signal format.
See *3.2.2.3* for the detail about output signal.

Serial RGB format is selectable depending on the RH850/D1x device. See *Table 3-42* for the detail.
This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t
r_vdce_OutFormat_t

4.2.3.16 R_VDCE_DisplaySerialRGBSet

Function Prototypes

```
r_vdce_Error_t R_VDCE_DisplaySerialRGBSet(const uint32_t             Unit,
                                          const r_vdce_SerialClkfreqsel_t Freq,
                                          const r_vdce_SerialClkphase_t  Phase,
                                          const r_vdce_SerialScan_t      Scan,
                                          const r_vdce_SerialSwap_t      Swap)
```

Input Parameter

Table 4-41 Input parameter of R_VDCE_DisplaySerialRGBSet

Parameter	Description
Unit	Specifies the VDCE unit number.

Freq	Specifies clock frequency control.
	  R_VDCE_SERIAL_CLKFRQ_3
	  R_VDCE_SERIAL_CLKFRQ_4

Phase	Specifies clock phase adjustment.
	  R_VDCE_SERIAL_CLKPHASE_0
	  R_VDCE_SERIAL_CLKPHASE_1
	  R_VDCE_SERIAL_CLKPHASE_2
	  R_VDCE_SERIAL_CLKPHASE_3
	Clock phase can be set 0 - 2clk in case of Triple speed mode (R_VDCE_SERIAL_CLKFRQ_3).
	Clock phase can be set 0 - 3clk in case of Quadruple speed mode (R_VDCE_SERIAL_CLKFRQ_4).

Scan	Specifies scan direction select.
	  R_VDCE_SERIAL_SCAN_FORWARD
	  R_VDCE_SERIAL_SCAN_REVERSE

Swap	Specifies swap on/off.
	  R_VDCE_SERIAL_SWAP_ON
	  R_VDCE_SERIAL_SWAP_OFF

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK   - No error has occurred.
R_VDCE_ERR_PARAM_INCORRECT - Parameter was incorrect.
R_VDCE_ERR_RANGE_UNIT   - The unit-number was outside the range.
R_VDCE_ERR_NOT_ACCEPTABLE   - A function was called in an incorrect state.
R_VDCE_ERR_FATAL_OS   - Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED   - VDCE was not initialized.

Description

This function sets the Serial RGB output setting.
This setting is valid when Serial RGB output is selected by R_VDCE_DisplayOutFormatSet.

This function is valid depending on the RH850/D1x device. See *Table 3-42* for the detail.
This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t
r_vdce_SerialClkfreqsel_t
r_vdce_SerialClkphase_t
r_vdce_SerialScan_t
r_vdce_SerialSwap_t

4.2.3.17 R_VDCE_DisplayVsyncProtectionSet

Function Prototypes

```
r_vdce_Error_t R_VDCE_DisplayVsyncProtectionSet(const uint32_t  Unit,
                                                const uint16_t  MaskMs,
                                                const uint16_t  LackMs)
```

Input Parameter

Table 4-42 Input parameter of R_VDCE_DisplayVsyncProtectionSet

Parameter	Description
Unit	Specifies the VDCE unit number.
MaskMs	Prevent V-sync coming faster than MaskMs msec. If non-zero value is set, vsync masking is enabled. If 0 is set, vsync masking is disabled.
LackMs	Compensate V-sync coming slower than LackMs msec. If non-zero value is set, vsync compensation is enabled. If 0 is set, vsync compensation is disabled.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK   - No error has occurred.
R_VDCE_ERR_RANGE_UNIT   - The unit-number was outside the range.
R_VDCE_ERR_NOT_ACCEPTABLE   - A function was called in an incorrect state.
R_VDCE_ERR_FATAL_OS   - Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED   - VDCE was not initialized.

Description

This function sets the V-sync protection timing.
This function is valid when selecting external V-sync synchronous mode with R_VDCE_CapModeSet.

V-sync protection timing is calculated by PixelClock set by R_VDCE_DisplayTimingSet.
The maximum value of MaskMs and LackMs is about (8,388,480,000 / PixelClock [Hz]) [msec].
If setting value is over the range, this function assumes that the maximum value is set.

If Vsync lack is detected by compensation protection, R_VDCE_INTC_NO_VI_VSYNC_SIGNAL or
R_VDCE_INTC_NO_VI_VSYNC_SIGNAL_1 interrupt occurs.

Vsync protections are disabled by default.
R_VDCE_CapExtVsyncSet provides the same feature and is even more accurate.

This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will
become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.3.18 R_VDCE_DisplayEnable

Function Prototypes

```
r_vdce_Error_t R_VDCE_DisplayEnable(const uint32_t  Unit)
```

Input Parameter

Table 4-43 Input parameter of R_VDCE_DisplayEnable

Parameter	Description
Unit	Specifies the VDCE unit number.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED	- VDCE is not initialized.
R_VDCE_ERR_DISPLAY_NO_TIMING	- R_VDCE_DisplayTimingSet is not done.

Description

This function enables the display output.
If the function successfully executes, the return code will be R_VDCE_ERR_OK and the VDCE unit status will be in the Idle state.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.
Execute R_VDCE_DisplayTimingSet before executing this function.

See also

r_vdce_Error_t

4.2.3.19 R_VDCE_DisplayDisable

Function Prototypes

```
r_vdce_Error_t R_VDCE_DisplayDisable(const uint32_t  Unit)
```

Input Parameter

Table 4-44 Input parameter of R_VDCE_DisplayDisable

Parameter	Description
Unit	Specifies the VDCE unit number.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK   - No error has occurred.
R_VDCE_ERR_RANGE_UNIT   - The unit-number was outside the range.
R_VDCE_ERR_NOT_ACCEPTABLE   - A function was called in an incorrect state.
R_VDCE_ERR_FATAL_OS   - Fatal error has occurred at OS interface.
R_VDCE_ERR_INTERRUPT_ENABLED   - Enabled interrupt is remained.

Description

This function disables the display output.
All interrupt must be disabled by R_VDCE_IntcDisable before executing this function.
If the function successfully executes, the return code will be R_VDCE_ERR_OK and the VDCE unit status will be in the initialized state.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.
Disable all the interrupts by R_VDCE_IntcDisable before executing this function.

See also

r_vdce_Error_t

4.2.3.20 R_VDCE_DisplayTimingGet

Function Prototypes

```
r_vdce_Error_t R_VDCE_DisplayTimingGet(const uint32_t        Unit,
                                       r_ddb_Timing_t  *const  Timing)
```

Input Parameter

Table 4-45 Input parameter of R_VDCE_DisplayTimingGet

Parameter	Description
Unit	Specifies the VDCE unit number.

Input - Output Parameter

None

Output Parameter

Table 4-46 Output parameter of R_VDCE_DisplayTimingGet

Parameter	Description
Timing	The timing parameter contains all the information to drive the display.

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_PARAM_INCORRECT	- Parameter Timing was R_NULL.
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_DISPLAY_NO_TIMING	- R_VDCE_DisplayTimingSet is not done.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED	- VDCE was not initialized.

Description

This function gets the display timing set by R_VDCE_DisplayTimingSet.
The display timing here is not affected by R_VDCE_DisplayTimingAdjust.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.
Execute R_VDCE_DisplayTimingSet before executing this function.

See also

  r_vdce_Error_t
  r_ddb_Timing_t

4.2.4 Layer functions

4.2.4.1 R_VDCE_LayerBaseSet

Function Prototypes

```
r_vdce_Error_t R_VDCE_LayerBaseSet(const uint32_t  Unit,
                                   const uint32_t  LayerNr,
                                   const uint32_t  Address)
```

Input Parameter

Table 4-47 Input parameter of R_VDCE_LayerBaseSet

Parameter	Description
Unit	Specifies the VDCE unit number.
LayerNr	Specifies the number of the corresponding layer. R_VDCE_LAYER_SCALER0 R_VDCE_LAYER_SCALER1 R_VDCE_LAYER_IMAGE2 R_VDCE_LAYER_IMAGE3
Address	Specifies the start address of the frame buffer. It should be 128 Byte aligned.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_PARAM_INCORRECT	- Parameter Address provided to the function was incorrect.
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_RANGE_LAYER	- The layer-number was outside the range.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED	- VDCE was not initialized.

Description

This function sets the frame buffer address.
This setting is valid until R_VDCE_DeInit is executed.
This function has same feature as R_VDCE_LayerBufSet except Lock/Unlock call.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.4.2 R_VDCE_OirBaseSet

Function Prototypes

```
r_vdce_Error_t R_VDCE_OirBaseSet(const uint32_t  Unit,
                                 const uint32_t  Address)
```

Input Parameter

Table 4-48 Input parameter of R_VDCE_OirBaseSet

Parameter	Description
Unit	Specifies the VDCE unit number.
Address	Specifies the start address of the OIR frame buffer. It should be 128 Byte aligned.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK   - No error has occurred.
R_VDCE_ERR_PARAM_INCORRECT - Parameter Address provided to the function was incorrect.
R_VDCE_ERR_RANGE_UNIT   - The unit-number was outside the range.
R_VDCE_ERR_FATAL_OS   - Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED - VDCE was not initialized.

Description

This function sets the frame buffer address for OIR.
When VOWE is used, warping result will be stored in this buffer.
This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.4.3 R_VDCE_LayerMemGeometrySet

Function Prototypes

```
r_vdce_Error_t R_VDCE_LayerMemGeometrySet(const uint32_t  Unit,
                                          const uint32_t  LayerNr,
                                          const uint32_t  Stride,
                                          const uint32_t  MemHeight)
```

Input Parameter

<p align="center">Table 4-49 Input parameter of R_VDCE_LayerMemGeometrySet</p>

Parameter	Description
Unit	Specifies the VDCE unit number.
LayerNr	Specifies the number of the corresponding layer. R_VDCE_LAYER_SCALER0 R_VDCE_LAYER_SCALER1 R_VDCE_LAYER_IMAGE2 R_VDCE_LAYER_IMAGE3
Stride	Specifies the width of the memory area in pixel. Range is (1024 / bpp) to (261120 / bpp). Stride should be 128 bytes aligned.
MemHeight	Specifies the height of the memory area in pixel. Range is 1 to 4096.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_RANGE_PARAM	- Parameter Stride or MemHeight was outside the range.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_RANGE_LAYER	- The layer-number is the outside of the range.
R_VDCE_ERR_UNIT_NOTLOCKED	- VDCE was not initialized.

Description

This function sets the memory geometry of the frame buffer for specified layer.
See *3.2.3.1* for the detail.
This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.4.4   R_VDCE_OirMemGeometrySet

Function Prototypes

```
r_vdce_Error_t R_VDCE_OirMemGeometrySet(const uint32_t  Unit,
                                        const uint32_t  Stride,
                                        const uint32_t  MemHeight)
```

Input Parameter

<div align="center">

Table 4-50 Input parameter of R_VDCE_OirMemGeometrySet

</div>

Parameter	Description
Unit	Specifies the VDCE unit number.
Stride	Specifies the width of the memory area in pixel. Range is (1024 / bpp) to (261120 / bpp). Stride should be 128 bytes aligned.
MemHeight	Specifies the height of the memory area in pixel. Range is 1 to 4096.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_RANGE_PARAM	- Parameter Stride or MemHeight was outside the range.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED	- VDCE was not initialized.

Description

This function sets the memory geometry of the frame buffer for OIR.
This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.4.5 R_VDCE_LayerFormatSet

Function Prototypes

```
r_vdce_Error_t R_VDCE_LayerFormatSet(const uint32_t        Unit,
                                     const uint32_t        LayerNr,
                                     const r_vdce_Format_t  Format)
```

Input Parameter

Table 4-51 Input parameter of R_VDCE_LayerFormatSet

Parameter	Description
Unit	Specifies the VDCE unit number.

LayerNr	Specifies the number of the corresponding layer.
	  R_VDCE_LAYER_SCALER0
	  R_VDCE_LAYER_SCALER1
	  R_VDCE_LAYER_IMAGE2
	  R_VDCE_LAYER_IMAGE3

Format	Specifies the color format.
	  R_VDCE_RGB565
	  R_VDCE_RGB0888
	  R_VDCE_ARGB1555
	  R_VDCE_ARGB4444
	  R_VDCE_ARGB8888
	  R_VDCE_CLUT8
	  R_VDCE_CLUT4
	  R_VDCE_CLUT1
	  R_VDCE_YCBCR_422
	  R_VDCE_YCBCR_444
	  R_VDCE_RGBA5551
	  R_VDCE_RGBA8888
	  R_VDCE_YUV_YUYV
	  R_VDCE_YUV_UYVY
	  R_VDCE_YUV_YVYU
	  R_VDCE_YUV_VYUY

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK   - No error has occurred.
R_VDCE_ERR_PARAM_INCORRECT   - Parameter Format provided to the function was incorrect.
R_VDCE_ERR_RANGE_UNIT   - The unit-number was outside the range.
R_VDCE_ERR_FATAL_OS   - Fatal error has occurred at OS interface.
R_VDCE_ERR_RANGE_LAYER   - The layer-number was outside the range.
R_VDCE_ERR_UNIT_NOTLOCKED   - VDCE was not initialized.

Description

This function sets the color format of frame buffer for specified layer.
See *3.2.3.2* for the detail.
This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t
r_vdce_Format_t

4.2.4.6 R_VDCE_OirFormatSet

Function Prototypes

```
r_vdce_Error_t R_VDCE_OirFormatSet(const uint32_t        Unit,
                                   const r_vdce_Format_t  Format)
```

Input Parameter

<div align="center">Table 4-52 Input parameter of R_VDCE_OirFormatSet</div>

Parameter	Description
Unit	Specifies the VDCE unit number.

Format	Specifies the color format.
	  R_VDCE_RGB565
	  R_VDCE_ARGB8888

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_PARAM_INCORRECT	- Parameter Format provided to the function was incorrect.
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED	- VDCE was not initialized.

Description

This function sets the color format of frame buffer for the OIR.
This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t
r_vdce_Format_t

4.2.4.7   R_VDCE_LayerViewPortSet

Function Prototypes

```
r_vdce_Error_t R_VDCE_LayerViewPortSet(const uint32_t  Unit,
                                       const uint32_t  LayerNr,
                                       const int32_t   MemPosX,
                                       const int32_t   MemPosY,
                                       const int32_t   DispPosX,
                                       const int32_t   DispPosY,
                                       const int32_t   DispWidth,
                                       const int32_t   DispHeight)
```

Input Parameter

Table 4-53 Input parameter of R_VDCE_LayerViewPortSet

Parameter	Description
Unit	Specifies the VDCE unit number.

LayerNr	Specifies the number of the corresponding layer.
	    R_VDCE_LAYER_SCALER0
	    R_VDCE_LAYER_SCALER1
	    R_VDCE_LAYER_IMAGE2
	    R_VDCE_LAYER_IMAGE3

MemPosX	Specifies the memory position x coordinate. Range is 0 only.
MemPosY	Specifies the memory position y coordinate. Range is 0 only.
DispPosX	Specifies the display position x coordinate. Origin is at the left of visual screen area. The range is (3 - DispWidth) to (ScreenWidth - 3).
DispPosY	Specifies the display position y coordinate. Origin is at the top of visual screen area. The range is (1 – DispHeight) to (ScreenHeight – 1).
DispWidth	Specifies the displayed width. Range is 3 to 1280.
DispHeight	Specifies the displayed height. Range is 1 to 1024.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_RANGE_PARAM	- Parameter was outside the range.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_RANGE_LAYER	- The layer-number was outside the range.
R_VDCE_ERR_UNIT_NOTLOCKED	- VDCE was not initialized.

Description

This function sets viewport parameters for specified layer.
See *3.2.3.1* for the detail.

If DispPosX or DispPosY is less than minimum value, this function will round up to the minimum value.
If DispPosX or DispPosY is more than maximum value, this function will round down to the maximum value.

This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.4.8 R_VDCE_OirViewPortSet

Function Prototypes

```
r_vdce_Error_t R_VDCE_OirViewPortSet(const uint32_t  Unit,
                                     const int32_t   MemPosX,
                                     const int32_t   MemPosY,
                                     const int32_t   DispPosX,
                                     const int32_t   DispPosY,
                                     const int32_t   DispWidth,
                                     const int32_t   DispHeight)
```

Input Parameter

Table 4-54 Input parameter of R_VDCE_OirViewPortSet

Parameter	Description
Unit	Specifies the VDCE unit number.
MemPosX	Specifies the memory position x coordinate. Range is 0 only.
MemPosY	Specifies the memory position y coordinate. Range is 0 only.
DispPosX	Specifies the display position x coordinate. Origin is at the left of visual screen area. The range is (3 - DispWidth) to (ScreenWidth - 3).
DispPosY	Specifies the display position y coordinate. Origin is at the top of visual screen area. The range is (1 – DispHeight) to (ScreenHeight – 1).
DispWidth	Specifies the displayed width. Range is 3 to 1280.
DispHeight	Specifies the displayed height. Range is 1 to 1024.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK   - No error has occurred.
R_VDCE_ERR_RANGE_UNIT   - The unit-number was outside the range.
R_VDCE_ERR_RANGE_PARAM   - Parameter was outside the range.
R_VDCE_ERR_FATAL_OS   - Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED   - VDCE was not initialized.

Description

This function sets viewport parameters for OIR.
This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.4.9 R_VDCE_LayerRingBufferEnable

Function Prototypes

```
r_vdce_Error_t R_VDCE_LayerRingBufferEnable(const uint32_t  Unit,
                                            const uint32_t  LayerNr,
                                            const uint32_t  Size)
```

Input Parameter

Table 4-55 Input parameter of R_VDCE_LayerRingBufferEnable

Parameter	Description
Unit	Specifies the VDCE unit number.
LayerNr	Specifies the number of the corresponding layer. 　R_VDCE_LAYER_SCALER0 　R_VDCE_LAYER_SCALER1
Size	Specifies size of the ring buffer in lines. e.g. 1 -> one-line ring buffer. 　　2 -> two-line ring buffer etc. Range is 1 to 2048.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK   - No error has occurred.
R_VDCE_ERR_RANGE_UNIT   - The unit-number was outside the range.
R_VDCE_ERR_RANGE_PARAM   - Parameter Size was outside the range.
R_VDCE_ERR_FATAL_OS   - Fatal error has occurred at OS interface.
R_VDCE_ERR_RANGE_LAYER   - The layer-number is the outside of the range.
R_VDCE_ERR_UNIT_NOTLOCKED - VDCE was not initialized.

Description

This function switches from a full frame buffers to a ring buffer usage for specified layer.
The size of the ring buffer in lines is given in Size parameter.
This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.4.10 R_VDCE_OirRingBufferEnable

Function Prototypes

```
r_vdce_Error_t R_VDCE_OirRingBufferEnable(const uint32_t  Unit,
                                          const uint32_t  Size)
```

Input Parameter

Table 4-56 Input parameter of R_VDCE_OirRingBufferEnable

Parameter	Description
Unit	Specifies the VDCE unit number.
Size	Specifies size of the ring buffer in lines. e.g. 1 -> one-line ring buffer 2 -> two-line ring buffer etc. Range is 1 to 2048.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_RANGE_PARAM	- Parameter Size was outside the range.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED	- VDCE was not initialized.

Description

This function switches from a full frame buffers to a ring buffer usage for OIR.
The size of the ring buffer in lines is given in Size parameter.
This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.4.11 R_VDCE_LayerRingBufferDisable

Function Prototypes

```
r_vdce_Error_t R_VDCE_LayerRingBufferDisable(const uint32_t  Unit,
                                             const uint32_t  LayerNr)
```

Input Parameter

<div align="center">

Table 4-57 Input parameter of R_VDCE_LayerRingBufferDisable

</div>

Parameter	Description
Unit	Specifies the VDCE unit number.
LayerNr	Specifies the number of the corresponding layer. R_VDCE_LAYER_SCALER0 R_VDCE_LAYER_SCALER1

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_RANGE_LAYER	- The layer-number was outside the range.
R_VDCE_ERR_UNIT_NOTLOCKED	- VDCE was not initialized.

Description

This function switches back from ring buffer to a full frame buffers usage of specified layer.
This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.4.12 R_VDCE_OirRingBufferDisable

Function Prototypes

```
r_vdce_Error_t R_VDCE_OirRingBufferDisable(const uint32_t  Unit)
```

Input Parameter

<div align="center">

Table 4-58 Input parameter of R_VDCE_OirRingBufferDisable

</div>

Parameter	Description
Unit	Specifies the VDCE unit number.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED	- VDCE was not initialized.

Description

This function switches back from ring buffer to a full frame buffers usage of OIR.
R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.4.13 R_VDCE_LayerVSyncDelaySet

Function Prototypes

```
r_vdce_Error_t R_VDCE_LayerVSyncDelaySet(const uint32_t  Unit,
                                         const uint32_t  LayerNr,
                                         const uint32_t  Delay)
```

Input Parameter

Table 4-59 Input parameter of R_VDCE_LayerVSyncDelaySet

Parameter	Description
Unit	Specifies the VDCE unit number.

LayerNr	Specifies the number of the corresponding layer.
	  R_VDCE_LAYER_SCALER0
	  R_VDCE_LAYER_SCALER1

Delay	Specifies the delays in the line units. e.g. 1 -> 1-line delay, 2 -> 2-line delay etc. Default value is 0. Range is 0 to 255.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_RANGE_PARAM	- Parameter Delay was outside the range.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED	- VDCE was not initialized.
R_VDCE_ERR_RANGE_LAYER	- The layer-number was outside the range.

Description

This function sets the Vsync signal delay between input V-sync to the scaler and output V-sync from the scaler.
This function is valid when selecting external V-sync synchronous mode with R_VDCE_CapModeSet.
This delay can be adjusted the frame buffer reading timing of sclaer. It is useful when the video input data is captured with single frame buffer.
This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.4.14 R_VDCE_OirVSyncDelaySet

Function Prototypes

```
r_vdce_Error_t R_VDCE_OirVSyncDelaySet(const uint32_t  Unit,
                                       const uint32_t  Delay)
```

Input Parameter

<div align="center">

Table 4-60 Input parameter of R_VDCE_OirVSyncDelaySet

</div>

Parameter	Description
Unit	Specifies the VDCE unit number.
Delay	Specifies the delays in line units. e.g. 1 -> 1-line delay 2 -> 2-line delay etc. Default value is 1. Range is 0 to 255.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_RANGE_PARAM	- Parameter Delay was outside the range.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED	- VDCE was not initialized.

Description

This function sets the Vsync signal delay between input V-sync to the OIR and output V-sync from the OIR.
This delay can be adjusted the frame buffer reading timing of VDCE. It is useful when the OIR is processed with ring buffer mode.
This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.4.15 R_VDCE_LayerModeSet

Function Prototypes

```
r_vdce_Error_t R_VDCE_LayerModeSet(const uint32_t         Unit,
                                   const uint32_t         LayerNr,
                                   const r_vdce_LayerMode_t  LayerMode)
```

Input Parameter

Table 4-61 Input parameter of R_VDCE_LayerModeSet

Parameter	Description
Unit	Specifies the VDCE unit number.

LayerNr	Specifies the number of the corresponding layer.
	    R_VDCE_LAYER_SCALER0
	    R_VDCE_LAYER_SCALER1
	    R_VDCE_LAYER_IMAGE2
	    R_VDCE_LAYER_IMAGE3

LayerMode	Specifies the layer mode.
	The flags of r_vdce_LayerMode_t can be combined with '|'.
	    R_VDCE_LAYER_MODE_NONE
	    R_VDCE_LAYER_MODE_V_MIRORING

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_RANGE_LAYER	- The layer-number was outside the range.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED	- VDCE was not initialized.

Description

This function sets the optional mode for layer.
This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t
r_vdce_LayerMode_t

4.2.4.16 R_VDCE_OirModeSet

Function Prototypes

```
r_vdce_Error_t R_VDCE_OirModeSet(const uint32_t       Unit,
                                 const r_vdce_OirMode_t OirMode)
```

Input Parameter

<div align="center">

Table 4-62 Input parameter of R_VDCE_OirModeSet

Parameter	Description
Unit	Specifies the VDCE unit number.

OirMode	Specifies OIR layer mode.
	  R_VDCE_OIR_MODE_NORMAL
	  R_VDCE_OIR_MODE_SCREEN_SHOTS
	The default OIR layer mode is NORMAL.

</div>

Input -Output Parameter

None

Output Parameter

None

Return Codes

```
R_VDCE_ERR_OK                 - No error has occurred.
R_VDCE_ERR_PARAM_INCORRECT    - Parameter OirMode provided to the function was incorrect.
R_VDCE_ERR_RANGE_UNIT         - The unit-number was outside the range.
R_VDCE_ERR_NOT_ACCEPTABLE     - A function was called in an incorrect state.
R_VDCE_ERR_FATAL_OS           - Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED     - VDCE was not initialized.
```

Description

This function sets the operation mode for OIR layer.
This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t
r_vdce_OirMode_t

4.2.4.17 R_VDCE_LayerEnable

Function Prototypes

```
r_vdce_Error_t R_VDCE_LayerEnable(const uint32_t  Unit,
                                  const uint32_t  LayerNr)
```

Input Parameter

Table 4-63 Parameter R_VDCE_LayerEnable

Parameter	Description
Unit	Specifies the VDCE unit number.
LayerNr	Specifies the number of the corresponding layer. R_VDCE_LAYER_SCALER0 R_VDCE_LAYER_SCALER1 R_VDCE_LAYER_IMAGE2 R_VDCE_LAYER_IMAGE3

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_NOT_ACCEPTABLE	- A function was called in an incorrect state.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_RANGE_LAYER	- The layer-number was outside the range.
R_VDCE_ERR_UNIT_NOTLOCKED	- VDCE was not initialized.
R_VDCE_ERR_LAYER_NO_FORMAT	- R_VDCE_LayerFormatSet was not done.
R_VDCE_ERR_LAYER_NO_MEM_GEOMETRY	- R_VDCE_LayerMemGeometrySet was not done.
R_VDCE_ERR_LAYER_NO_VIEW_PORT	- R_VDCE_LayerViewPortSet was not done.
R_VDCE_ERR_LAYER_NO_BASE	- R_VDCE_LayerBaseSet was not done.
R_VDCE_ERR_STRIDE_INCORRECT	- The setting of Stride was incorrect.
R_VDCE_ERR_CHROMAKEY_INCORRECT	- The setting of Chromakey was incorrect.
R_VDCE_ERR_SCALED_SIZE_INCORRECT	- The setting of Scaled size was incorrect.

Description

This function enables the layer and the layer will be visualized.
If the function successfully executes, the return code will be R_VDCE_ERR_OK and the layer status will be in the Executing state.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.
Execute following functions before executing this function.
- R_VDCE_LayerBaseSet
- R_VDCE_LayerMemGeometrySet
- R_VDCE_LayerFormatSet
- R_VDCE_LayerViewPortSet

See also

r_vdce_Error_t

4.2.4.18 R_VDCE_OirEnable

Function Prototypes

```
r_vdce_Error_t R_VDCE_OirEnable(const uint32_t  Unit)
```

Input Parameter

Table 4-64 Input parameter of R_VDCE_OirEnable

Parameter	Description
Unit	Specifies the VDCE unit number.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_NOT_ACCEPTABLE	- A function was called in an incorrect state.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED	- VDCE was not initialized.
R_VDCE_ERR_LAYER_NO_FORMAT	- R_VDCE_OirFormatSet is not done.
R_VDCE_ERR_LAYER_NO_MEM_GEOMETRY	- R_VDCE_OirMemGeometrySet is not done.
R_VDCE_ERR_LAYER_NO_VIEW_PORT	- R_VDCE_OirViewPortSet is not done.
R_VDCE_ERR_LAYER_NO_BASE	- R_VDCE_OirBaseSet is not done.
R_VDCE_ERR_STRIDE_INCORRECT	- The setting of Stride is incorrect.

Description

This function enables the OIR.
If the function successfully executes, the return code will be R_VDCE_ERR_OK and the OIR status will be in the Executing state.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.
Execute following functions before executing this function.
- R_VDCE_OirBaseSet
- R_VDCE_OirMemGeometrySet
- R_VDCE_OirFormatSet
- R_VDCE_OirViewPortSet

See also

r_vdce_Error_t

4.2.4.19 R_VDCE_LayerDisable

Function Prototypes

```
r_vdce_Error_t R_VDCE_LayerDisable(const uint32_t  Unit,
                                   const uint32_t  LayerNr)
```

Input Parameter

Table 4-65 Input parameter of R_VDCE_LayerDisable

Parameter	Description
Unit	Specifies the VDCE unit number.

LayerNr	Specifies the number of the corresponding layer.
	  R_VDCE_LAYER_SCALER0
	  R_VDCE_LAYER_SCALER1
	  R_VDCE_LAYER_IMAGE2
	  R_VDCE_LAYER_IMAGE3

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_NOT_ACCEPTABLE	- A function was called in an incorrect state.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_RANGE_LAYER	- The layer-number is the outside of the range.

Description

This function disables the layer.
If the function successfully executes, the return code will be R_VDCE_ERR_OK and the layer status will be in the Idle state.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.4.20 R_VDCE_OirDisable

Function Prototypes

```
r_vdce_Error_t R_VDCE_OirDisable(const uint32_t  Unit)
```

Input Parameter

<div align="center">

Table 4-66 Input parameter of t R_VDCE_OirDisable

</div>

Parameter	Description
Unit	Specifies the VDCE unit number.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_NOT_ACCEPTABLE	- A function was called in an incorrect state.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.

Description

This function disables the OIR.
If the function successfully executes, the return code will be R_VDCE_ERR_OK and the OIR status will be in the Idle state.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.4.21 R_VDCE_LayerMatrixSet

Function Prototypes

```
r_vdce_Error_t R_VDCE_LayerMatrixSet(const uint32_t          Unit,
                                     const uint32_t          LayerNr,
                                     const r_vdce_Matrix_t  * const Matrix)
```

Input Parameter

<p align="center">Table 4-67 Input parameter of R_VDCE_LayerMatrixSet</p>

Parameter	Description
Unit	Specifies the VDCE unit number.

LayerNr	Specifies the number of the corresponding layer.
	    R_VDCE_LAYER_SCALER0
	    R_VDCE_LAYER_SCALER1
	    R_VDCE_LAYER_INPUT

Matrix	Specifies a structure which contains the conversion matrix including offsets.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_PARAM_INCORRECT	- Parameter Matrix provided to the function was incorrect.
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_RANGE_PARAM	- Parameter Matrix was outside the range.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_RANGE_LAYER	- The layer-number is the outside of the range.
R_VDCE_ERR_UNIT_NOTLOCKED	- VDCE was not initialized.

Description

This function sets the color conversion matrix for specified layer.
See *3.2.6* for the detail.
When one of following functions isn't executed, the color matrixes automatically set up by the VDCE driver according to the color format.
- R_VDCE_LayerMatrixSet
- R_VDCE_LayerMatrixBT601Set
- R_VDCE_LayerMatrixJPEGSet
- R_VDCE_LayerMatrixUnitySet

This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t
r_vdce_Matrix_t

4.2.4.22 R_VDCE_LayerMatrixBT601Set

Function Prototypes

```
r_vdce_Error_t R_VDCE_LayerMatrixBT601Set(const uint32_t  Unit,
                                          const uint32_t  LayerNr)
```

Input Parameter

Table 4-68 Input parameter of R_VDCE_LayerMatrixBT601Set

Parameter	Description
Unit	Specifies the VDCE unit number.
LayerNr	Specifies the number of the corresponding layer. R_VDCE_LAYER_SCALER0 R_VDCE_LAYER_SCALER1 R_VDCE_LAYER_INPUT

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_RANGE_LAYER	- The layer-number was outside the range.
R_VDCE_ERR_UNIT_NOTLOCKED	- VDCE was not initialized.

Description

This function sets the color conversion matrix to BT601 spec for the specified layer .
See *3.2.6* for the detail.
This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.4.23 R_VDCE_LayerMatrixJPEGSet

Function Prototypes

```
r_vdce_Error_t R_VDCE_LayerMatrixJPEGSet(const uint32_t  Unit,
                                         const uint32_t  LayerNr)
```

Input Parameter

Table 4-69 Input parameter of R_VDCE_LayerMatrixJPEGSet

Parameter	Description
Unit	Specifies the VDCE unit number.
LayerNr	Specifies the number of the corresponding layer. R_VDCE_LAYER_SCALER0 R_VDCE_LAYER_SCALER1 R_VDCE_LAYER_INPUT

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK   - No error has occurred.
R_VDCE_ERR_RANGE_UNIT   - The unit-number was outside the range.
R_VDCE_ERR_FATAL_OS   - Fatal error has occurred at OS interface.
R_VDCE_ERR_RANGE_LAYER   - The layer-number was outside the range.
R_VDCE_ERR_UNIT_NOTLOCKED - VDCE was not initialized.

Description

This function sets the color conversion matrix to JPEG spec for the specified layer.
See *3.2.6* for the detail.
This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.4.24 R_VDCE_LayerMatrixUnitySet

Function Prototypes

```
r_vdce_Error_t R_VDCE_LayerMatrixUnitySet(const uint32_t   Unit,
                                          const uint32_t   LayerNr)
```

Input Parameter

Table 4-70 Input parameter of R_VDCE_LayerMatrixUnitySet

Parameter	Description
Unit	Specifies the VDCE unit number.
LayerNr	Specifies the number of the corresponding layer. R_VDCE_LAYER_SCALER0 R_VDCE_LAYER_SCALER1 R_VDCE_LAYER_INPUT

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK   - No error has occurred.
R_VDCE_ERR_RANGE_UNIT   - The unit-number was outside the range.
R_VDCE_ERR_FATAL_OS   - Fatal error has occurred at OS interface.
R_VDCE_ERR_RANGE_LAYER   - The layer-number was outside the range.
R_VDCE_ERR_UNIT_NOTLOCKED   - VDCE was not initialized.

Description

This function sets the color conversion matrix to unity mode for the specified layer.
See *3.2.6* for the detail.
This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.4.25 R_VDCE_LayerImgScaleX

Function Prototypes

```
uint32_t R_VDCE_LayerImgScaleX(const uint32_t      Unit,
                               const uint32_t      LayerNr,
                               const uint32_t      ScaledWidth,
                               const r_vdce_Scale_t  Scaling)
```

Input Parameter

Table 4-71 Input parameter of R_VDCE_LayerImgScaleX

Parameter	Description
Unit	Specifies the VDCE unit number.

LayerNr	Specifies the layer from the following:
	    R_VDCE_LAYER_SCALER0
	    R_VDCE_LAYER_SCALER1

ScaledWidth	Specifies the scaled / scaling width of frame buffer. The range is 4 or more. The maximum range and alignment information is described in *3.2.3.4* and *3.2.5.5*.

Scaling	Specifies the behavior.
	    R_VDCE_SCALING_LARGER
	    R_VDCE_SCALING_SMALLER
	    R_VDCE_SCALING_DISABLE

Input -Output Parameter

None

Output Parameter

None

Return Codes

Same value as ScaledWidth   - No error occurs.
0   - Error occurs.

Description

This function sets the scaling parameters in X direction.
This setting is valid until R_VDCE_DeInit is executed.
This scaling feature is valid depending on the RH850/D1x device. See *Table 3-42* for the detail.

When R_VDCE_SCALING_LARGER is selected, scaling-up image will be displayed.
ScaledWidth should be specified the original size in frame buffer.
See *3.2.3.4* for the detail.

When R_VDCE_SCALING_SMALLER is selected, scaling-down image will be captured.
ScaledWidth should be specified the scaled size in frame buffer.
See *3.2.5.5* for the detail.

If R_VDCE_SCALING_LARGER is selected, following function is not effective in the layer.
- R_VDCE_LayerAlphaChannelEnable
- R_VDCE_LayerPremultipliedAlphaEnable
- R_VDCE_LayerAlphaConstEnable
- R_VDCE_LayerChromaKeyEnable

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Scale_t

4.2.4.26 R_VDCE_LayerImgScaleY

Function Prototypes

```
uint32_t R_VDCE_LayerImgScaleY(const uint32_t       Unit,
                               const uint32_t       LayerNr,
                               const uint32_t       ScaledHeight,
                               const r_vdce_Scale_t Scaling)
```

Input Parameter

Table 4-72 Input parameter of R_VDCE_LayerImgScaleY

Parameter	Description
Unit	Specifies the VDCE unit number.

LayerNr	Specifies the layer from the following:
	  R_VDCE_LAYER_SCALER0
	  R_VDCE_LAYER_SCALER1

ScaledHeight	Specifies the scaled / scaling height of frame buffer. The range is 4 or more. The maximum range and alignment information is described in *3.2.3.4* and *3.2.5.5*.

Scaling	Specifies the behavior.
	  R_VDCE_SCALING_LARGER
	  R_VDCE_SCALING_SMALLER
	  R_VDCE_SCALING_DISABLE

Input -Output Parameter

None

Output Parameter

None

Return Codes

Same value as ScaledHeight   - No error occurs.
0   - Error occurs.

Description

This function sets the scaling parameters in Y direction.
This setting is valid until R_VDCE_DeInit is executed.
This scaling feature is valid depending on the RH850/D1x device. See *Table 3-42* for the detail.

When R_VDCE_SCALING_LARGER is selected, scaling-up image will be displayed.
ScaledHeight should be specified the original size in frame buffer.
See *3.2.3.4* for the detail.

When R_VDCE_SCALING_SMALLER is selected, scaling-down image will be captured.
ScaledHeight should be specified the scaled size in frame buffer.
See *3.2.5.5* for the detail.

If R_VDCE_SCALING_LARGER is selected, following function is not effective in the layer.
- R_VDCE_LayerAlphaChannelEnable
- R_VDCE_LayerPremultipliedAlphaEnable
- R_VDCE_LayerAlphaConstEnable
- R_VDCE_LayerChromaKeyEnable

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Scale_t

4.2.4.27 R_VDCE_LayerImgScaleModeSet

Function Prototypes

```
r_vdce_Error_t R_VDCE_LayerImgScaleModeSet(const uint32_t        Unit,
                                           const uint32_t        LayerNr,
                                           const r_vdce_ScaleMode_t  Mode)
```

Input Parameter

Table 4-73 Input parameter of R_VDCE_LayerImgScaleModeSet

Parameter	Description
Unit	Specifies the VDCE unit number.
LayerNr	Specifies the number of the corresponding layer. R_VDCE_LAYER_SCALER0 R_VDCE_LAYER_SCALER1
Mode	Specifies the optional mode for enlargement and reduction. It can be set multiple flags with OR operation. R_VDCE_SCALE_H_PREFILTER R_VDCE_SCALE_H_HOLD_INTERPOL R_VDCE_SCALE_V_HOLD_INTERPOL

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_RANGE_LAYER	- The layer-number was outside the range.
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED	- VDCE was not initialized.

Description

This function sets the optional mode for enlargement and reduction.
This function is valid depending on the RH850/D1x device. See *Table 3-42* for the detail.
This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t
r_vdce_ScaleMode_t

4.2.4.28 R_VDCE_LayerBufSet

Function Prototypes

```
r_vdce_Error_t R_VDCE_LayerBufSet(const uint32_t  Unit,
                                  const uint32_t  LayerNr,
                                  const uint32_t  Address)
```

Input Parameter

<div align="center">

Table 4-74 Input parameter of R_VDCE_LayerBufSet

</div>

Parameter	Description
Unit	Specifies the VDCE unit number.
LayerNr	Specifies the number of the corresponding layer. 　R_VDCE_LAYER_SCALER0 　R_VDCE_LAYER_SCALER1 　R_VDCE_LAYER_IMAGE2 　R_VDCE_LAYER_IMAGE3
Address	Specifies the start address of the frame buffer. It should be 128 Byte aligned.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_PARAM_INCORRECT	- Parameter Address provided to the function was incorrect.
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_RANGE_LAYER	- The layer-number was outside the range.
R_VDCE_ERR_UNIT_NOTLOCKED	- VDCE was not initialized.

Description

This function sets the frame buffer address.
This setting is valid until R_VDCE_DeInit is executed.
This function has same feature as R_VDCE_LayerBaseSet except Lock/Unlock call.

Reentrancy

Non-reentrant.
This function doesn't call R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock.
User should control not to re-enter the same VDCE unit. And, user should not execute this function while other layer functions and extended layer functions of same VDCE unit are being executed.

Sync/Async

Synchronous.

Call from Interrupt

Permitted.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.5 Extended Layer functions

4.2.5.1 R_VDCE_LayerAlphaChannelEnable

Function Prototypes

```
r_vdce_Error_t R_VDCE_LayerAlphaChannelEnable(const uint32_t  Unit,
                                              const uint32_t  LayerNr)
```

Input Parameter

Table 4-75 Input parameter of R_VDCE_LayerAlphaChannelEnable

Parameter	Description
Unit	Specifies the VDCE unit number.
LayerNr	Specifies the number of the corresponding layer. R_VDCE_LAYER_SCALER1 R_VDCE_LAYER_IMAGE2 R_VDCE_LAYER_IMAGE3

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_RANGE_LAYER	- The layer-number was outside the range.
R_VDCE_ERR_UNIT_NOTLOCKED	- VDCE was not initialized.

Description

This function re-enables the alpha channel per pixel data that is disabled by R_VDCE_LayerAlphaChannelDisable.
See *3.2.3.6* for the detail.
This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.5.2 R_VDCE_LayerAlphaChannelDisable

Function Prototypes

```
r_vdce_Error_t R_VDCE_LayerAlphaChannelDisable(const uint32_t  Unit,
                                               const uint32_t  LayerNr)
```

Input Parameter

<div align="center">Table 4-76 Input parameter of R_VDCE_LayerAlphaChannelDisable</div>

Parameter	Description
Unit	Specifies the VDCE unit number.

LayerNr	Specifies the number of the corresponding layer.
	  R_VDCE_LAYER_SCALER1
	  R_VDCE_LAYER_IMAGE2
	  R_VDCE_LAYER_IMAGE3

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_RANGE_LAYER	- The layer-number was outside the range.
R_VDCE_ERR_UNIT_NOTLOCKED	- VDCE was not initialized.

Description

This function disables the alpha channel per pixel data.
If frame buffer color format has alpha channel (i.e. ARGBxxxx, RGBAxxxx, CLUTx), the alpha channel per pixel data is enabled as default. This function can disable this alpha channel.
See *3.2.3.6* for the detail.

This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.5.3 R_VDCE_LayerPremultipliedAlphaEnable

Function Prototypes

```
r_vdce_Error_t R_VDCE_LayerPremultipliedAlphaEnable(const uint32_t  Unit,
                                                    const uint32_t  LayerNr)
```

Input Parameter

<p align="center">Table 4-77 Input parameter of R_VDCE_LayerPremultipliedAlphaEnable</p>

Parameter	Description
Unit	Specifies the VDCE unit number.
LayerNr	Specifies the number of the corresponding layer. R_VDCE_LAYER_SCALER1 R_VDCE_LAYER_IMAGE2 R_VDCE_LAYER_IMAGE3

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK   - No error has occurred.
R_VDCE_ERR_RANGE_UNIT   - The unit-number was outside the range.
R_VDCE_ERR_FATAL_OS   - Fatal error has occurred at OS interface.
R_VDCE_ERR_RANGE_LAYER   - The layer-number was outside the range.
R_VDCE_ERR_UNIT_NOTLOCKED - VDCE was not initialized.

Description

This function enables pre-multiplied alpha channel of the layer.
See *3.2.3.6* for the detail.
This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.5.4 R_VDCE_LayerPremultipliedAlphaDisable

Function Prototypes

```
r_vdce_Error_t R_VDCE_LayerPremultipliedAlphaDisable(const uint32_t  Unit,
                                                     const uint32_t  LayerNr)
```

Input Parameter

<div align="center">Table 4-78 Input parameter of R_VDCE_LayerPremultipliedAlphaDisable</div>

Parameter	Description
Unit	Specifies the VDCE unit number.
LayerNr	Specifies the number of the corresponding layer. R_VDCE_LAYER_SCALER1 R_VDCE_LAYER_IMAGE2 R_VDCE_LAYER_IMAGE3

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK   - No error has occurred.
R_VDCE_ERR_RANGE_UNIT   - The unit-number was outside the range.
R_VDCE_ERR_FATAL_OS   - Fatal error has occurred at OS interface.
R_VDCE_ERR_RANGE_LAYER   - The layer-number was outside the range.
R_VDCE_ERR_UNIT_NOTLOCKED - VDCE was not initialized.

Description

This function disables pre-multiplied alpha channel of the layer.
See *3.2.3.6* for the detail.
This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.5.5 R_VDCE_LayerAlphaConstEnable

Function Prototypes

```
r_vdce_Error_t R_VDCE_LayerAlphaConstEnable(const uint32_t  Unit,
                                            const uint32_t  LayerNr,
                                            const uint8_t   AlphaConst)
```

Input Parameter

Table 4-79 Input parameter of R_VDCE_LayerAlphaConstEnable

Parameter	Description
Unit	Specifies the VDCE unit number.

LayerNr	Specifies the number of the corresponding layer.
	  R_VDCE_LAYER_SCALER1
	  R_VDCE_LAYER_IMAGE2
	  R_VDCE_LAYER_IMAGE3

AlphaConst	Specifies a constant alpha value from 0-255 that is used for every pixel in the layer.

Input -Output Parameter

    None

Output Parameter

    None

Return Codes

    R_VDCE_ERR_OK   - No error has occurred.
    R_VDCE_ERR_RANGE_UNIT   - The unit-number was outside the range.
    R_VDCE_ERR_FATAL_OS   - Fatal error has occurred at OS interface.
    R_VDCE_ERR_RANGE_LAYER   - The layer-number was outside the range.
    R_VDCE_ERR_UNIT_NOTLOCKED - VDCE was not initialized.

Description

    This function enables the constant alpha of the layers.
    See *3.2.3.6* for the detail of constant alpha.
    If constant alpha channel is enabled, function R_VDCE_LayerChromaKeyEnable is not effective in the layer.
    This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

    Non-reentrant as default.
    If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

    Synchronous.

Call from Interrupt

    Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.5.6 R_VDCE_LayerAlphaConstDisable

Function Prototypes

```
r_vdce_Error_t R_VDCE_LayerAlphaConstDisable(const uint32_t  Unit,
                                             const uint32_t  LayerNr)
```

Input Parameter

Table 4-80 Input parameter of R_VDCE_LayerAlphaConstDisable

Parameter	Description
Unit	Specifies the VDCE unit number.

LayerNr	Specifies the number of the corresponding layer.
	    R_VDCE_LAYER_SCALER1
	    R_VDCE_LAYER_IMAGE2
	    R_VDCE_LAYER_IMAGE3

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK   - No error has occurred.
R_VDCE_ERR_RANGE_UNIT   - The unit-number was outside the range.
R_VDCE_ERR_FATAL_OS   - Fatal error has occurred at OS interface.
R_VDCE_ERR_RANGE_LAYER   - The layer-number was outside the range.
R_VDCE_ERR_UNIT_NOTLOCKED   - VDCE was not initialized.

Description

This function disables the constant alpha of the layer.
See *3.2.3.6* for the detail of constant alpha.
This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.5.7   R_VDCE_LayerClutSet

Function Prototypes

```
r_vdce_Error_t R_VDCE_LayerClutSet(const uint32_t            Unit,
                                   const uint32_t            LayerNr,
                                   const uint32_t            Offset,
                                   const uint32_t            ClutSize,
                                   const r_vdce_ClutARGB32_t * const Clut)
```

Input Parameter

Table 4-81 Input parameter of R_VDCE_LayerClutSet

Parameter	Description
Unit	Specifies the VDCE unit number.

LayerNr	Specifies the number of the corresponding layer.
	  R_VDCE_LAYER_SCALER0
	  R_VDCE_LAYER_SCALER1
	  R_VDCE_LAYER_IMAGE2
	  R_VDCE_LAYER_IMAGE3

Offset	Specifies index of the first CLUT entry to be written. Range is 0 to (ClutSize -1).
ClutSize	Specifies number of CLUT entry to be written. When format is R_VDCE_CLUT1, range is 1 to 2. When format is R_VDCE_CLUT4, range is 1 to 16. When format is R_VDCE_CLUT8, range is 1 to 256.
Clut	Specifies the CLUT data. It should be array of the number specified by ClutSize.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_PARAM_INCORRECT	- Parameter Clut provided to function was R_NULL.
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_RANGE_PARAM	- A parameter was outside the range.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_RANGE_LAYER	- The layer-number was outside the range.
R_VDCE_ERR_UNIT_NOTLOCKED	- VDCE was not initialized.

Description

This function updates the Color look up table (CLUT) and switches between two tables.
The CLUT in VDCE is double buffered to be able to update while displaying.
This function can be executed only once in Vsync period.
See *3.2.3.8* for the detail of constant alpha.
This function is used when frame buffer color format is CLUT format.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t
r_vdce_ClutARGB32_t

4.2.5.8   R_VDCE_LayerChromaKeyEnable

Function Prototypes

```
r_vdce_Error_t R_VDCE_LayerChromaKeyEnable(const uint32_t                Unit,
                                           const uint32_t                LayerNr,
                                           const r_vdce_CkARGB32_t * const CkTarget,
                                           const r_vdce_CkARGB32_t * const CkReplace)
```

Input Parameter

Table 4-82 Input parameter of R_VDCE_LayerChromaKeyEnable

Parameter	Description
Unit	Specifies the VDCE unit number.
LayerNr	Specifies the number of the corresponding layer. R_VDCE_LAYER_SCALER1 R_VDCE_LAYER_IMAGE2 R_VDCE_LAYER_IMAGE3
CkTarget	Specifies the target color (before replacing). Range is changed by the color format. See *Table 3-20*.
CkReplace	Specifies the color to replace (after replacing). Range is changed by the color format. See *Table 3-20*.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_PARAM_INCORRECT	- Parameter was incorrect.
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_RANGE_LAYER	- The layer-number was outside the range.
R_VDCE_ERR_UNIT_NOTLOCKED	- VDCE was not initialized.
R_VDCE_ERR_CHROMAKEY_INCORRECT	- The setting of Chromakey is incorrect.

Description

This function enables the Chroma key of the layer.
If Chroma key is enabled, pre-multiplied alpha is not effective in the layer.
If color format is YCbCr, this setting is invalid.
This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t
r_vdce_CkARGB32_t

4.2.5.9 R_VDCE_LayerChromaKeyDisable

Function Prototypes

```
r_vdce_Error_t R_VDCE_LayerChromaKeyDisable(const uint32_t Unit,
                                            const uint32_t LayerNr)
```

Input Parameter

Table 4-83 Input parameter of R_VDCE_LayerChromaKeyDisable

Parameter	Description
Unit	Specifies the VDCE unit number.

LayerNr	Specifies the number of the corresponding layer.
	   R_VDCE_LAYER_SCALER1
	   R_VDCE_LAYER_IMAGE2
	   R_VDCE_LAYER_IMAGE3

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_RANGE_LAYER	- The layer-number was outside the range.
R_VDCE_ERR_UNIT_NOTLOCKED	- VDCE was not initialized.

Description

This function disables Chroma key of the layer.
This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.6 Capture functions

4.2.6.1 R_VDCE_CapBufGeometrySetup

Function Prototypes

```
r_vdce_Error_t R_VDCE_CapBufGeometrySetup (const uint32_t        Unit,
                                           const uint32_t        Buffer1,
                                           const uint32_t        Buffer2,
                                           const uint32_t        Stride,
                                           const uint32_t        Width,
                                           const uint32_t        Height,
                                           const uint32_t        StartX,
                                           const uint32_t        StartY,
                                           const r_vdce_Fbformat_t FbFormat)
```

Input Parameter

Table 4-84 Input parameter of R_VDCE_CapBufGeometrySetup

Parameter	Description
Unit	Specifies the VDCE unit number.
Buffer1	Specifies address of the first capture buffer. 0 setting is prohibited.
Buffer2	Specifies address of the second capture buffer. 0 can be set if not needed.
Stride	Specifies distance in pixels between two adjacent pixel rows of the capture buffer in the capture buffer.
Width	Specifies the width of the input video in pixels.
Height	Specifies the height of the input video in pixels.
StartX	Specifies the x position to capture start in pixels. When DE mode is disabled, reference point is Hsync signal. StartX pixels are skipped after Hsync signal. The range is 16 to 2011. If setting value is 0 to 15, this function will round up to 16. When DE mode is enabled (see R_VDCE_CAP_MODE_DE_MODE), the reference point is DE signal. StartX pixels are skipped after DE signal. The range of 0 to 2011.
StartY	Specifies the y position to capture start in pixels. When DE mode is disabled, reference point is Vsync signal. (StartY + 1) lines are skipped after Vsync signal. The range is 4 to 2035. If setting value is 0 to 3, this function will round up to 4. When DE mode is enabled (see R_VDCE_CAP_MODE_DE_MODE), the reference point is DE signal after Vsync signal. StartY pixels are skipped after 1st DE signal. The range of 0 to 2035.

FbFormat	Specifies the frame buffer format.
	  R_VDCE_FB_FORMAT_YCBCR_422
	  R_VDCE_FB_FORMAT_RGB565
	  R_VDCE_FB_FORMAT_RGB0888
	  R_VDCE_FB_FORMAT_YCBCR_444
	Following optional flag can be set with OR operation.
	  R_VDCE_FB_RB_SWAP

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_PARAM_INCORRECT	- Parameter was incorrect.
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_RANGE_PARAM	- A parameter was outside the range.
R_VDCE_ERR_NOT_ACCEPTABLE	- A function was called in an incorrect state.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED	- VDCE was not initialized.
R_VDCE_ERR_STRIDE_INCORRECT	- The setting of Stride is incorrect.

Description

This function sets the parameters for the capture buffers and the size of the incoming video data.
About range and alignment information, see *Table 3-29* and *Table 3-30* for the detail.
About capture buffer address, see *Table 3-32* for the detail.
When Width is greater than 1024 pixels, horizontal image size must be reduced to 1024 pixels or less.

This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t
r_vdce_Fbformat_t

4.2.6.2 R_VDCE_CapModeSet

Function Prototypes

```
r_vdce_Error_t R_VDCE_CapModeSet(const uint32_t        Unit,
                                 const r_vdce_CapMode_t  Mode)
```

Input Parameter

<div align="center">Table 4-85 Input parameter of R_VDCE_CapModeSet</div>

Parameter	Description
Unit	Specifies the VDCE unit number.
Mode	Specifies input video format and capturing mode. It can be set several flags with OR operation.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_PARAM_INCORRECT	- Parameter was incorrect.
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_NOT_ACCEPTABLE	- A function was called in an incorrect state.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED	- VDCE was not initialized.

Description

This function sets the capturing mode.
One of these six flags must be selected to specify the input video format, or the function call will fail.
- R_VDCE_CAP_MODE_YUV_ITU656
- R_VDCE_CAP_MODE_YUV_8BIT
- R_VDCE_CAP_MODE_YUV_16BIT
- R_VDCE_CAP_MODE_RGB_16BPP
- R_VDCE_CAP_MODE_RGB_18BPP
- R_VDCE_CAP_MODE_RGB_24BPP

Other flags are optional flag.

This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t
r_vdce_CapMode_t

4.2.6.3 R_VDCE_CapBufSet

Function Prototypes

```
r_vdce_Error_t R_VDCE_CapBufSet(const uint32_t Unit,
                                const uint32_t Buffer1,
                                const uint32_t Buffer2)
```

Input Parameter

Table 4-86 Input parameter of R_VDCE_CapBufSet

Parameter	Description
Unit	Specifies the VDCE unit number.
Buffer1	Specifies address of the first capture buffer. This parameter is the same as Buffer1 of R_VDCE_CapBufGeometrySetup. If Buffer1 is 0, the buffer address is not updated.
Buffer2	Specifies address of the second capture buffer. This parameter is the same as Buffer2 of R_VDCE_CapBufGeometrySetup. If Buffer2 is 0, the buffer address is not updated.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_PARAM_INCORRECT	- Parameter was incorrect.
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_NOT_ACCEPTABLE	- A function was called in an incorrect state.
R_VDCE_ERR_UNIT_NOTLOCKED	- VDCE was not initialized.

Description

This function sets address of capture buffers.
About alignment information, see *Table 3-29* and *Table 3-30* for the detail.
About capture buffer address, see *Table 3-32* for the detail.
If both Buffer1 and Buffer2 are set to 0, this function will return error.
This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant.
This function doesn't call R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock.
User should control not to re-enter the same VDCE unit. And, user should not execute this function while other capture functions of same VDCE unit are being executed.

Sync/Async

Synchronous.

Call from Interrupt

Permitted.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.6.4 R_VDCE_CapBufFieldSetup1

Function Prototypes

```
r_vdce_Error_t R_VDCE_CapBufFieldSetup1(const uint32_t    Unit,
                                        const uint32_t    Buffer1,
                                        const uint32_t    StartY)
```

Input Parameter

Table 4-87 Input parameter of R_VDCE_CapBufFieldSetup1

Parameter	Description
Unit	Specifies the VDCE unit number.
Buffer1	Specifies address of the first capture buffer. This parameter is the same as Buffer1 of R_VDCE_CapBufGeometrySetup. 0 setting is prohibited.
StartY	Specifies the y position to capture start in pixels. This parameter is the same as StartY of R_VDCE_CapBufGeometrySetup.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK   - No error has occurred.
R_VDCE_ERR_PARAM_INCORRECT - Parameter was incorrect.
R_VDCE_ERR_RANGE_UNIT   - The unit-number was outside the range.
R_VDCE_ERR_RANGE_PARAM   - Parameter StartY was outside the range.
R_VDCE_ERR_NOT_ACCEPTABLE   - A function was called in an incorrect state.
R_VDCE_ERR_FATAL_OS   - Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED   - VDCE was not initialized.

Description

This function sets capture buffer address and capture start position.
About alignment information, see *Table 3-29* and *Table 3-30* for the detail.
About capture buffer address, see *Table 3-32* for the detail.
This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.6.5 R_VDCE_CapBufFieldSetup2

Function Prototypes

```
r_vdce_Error_t R_VDCE_CapBufFieldSetup2(const uint32_t    Unit,
                                        const uint32_t    Buffer2,
                                        const uint32_t    StartY)
```

Input Parameter

Table 4-88 Input parameter of R_VDCE_CapBufFieldSetup2

Parameter	Description
Unit	Specifies the VDCE unit number.
Buffer2	Specifies address of the second capture buffer. This parameter is the same as Buffer2 of R_VDCE_CapBufGeometrySetup. 0 can be set if not needed.
StartY	Specifies the y position to capture start in pixels. This parameter is the same as StartY of R_VDCE_CapBufGeometrySetup.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_PARAM_INCORRECT	- Parameter Buffer2 provided to the function was incorrect.
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_RANGE_PARAM	- Parameter StartY was outside the range.
R_VDCE_ERR_NOT_ACCEPTABLE	- A function was called in an incorrect state.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED	- VDCE was not initialized.

Description

This function sets capture buffer address for bottom field and capture start position.
About alignment information, see *Table 3-29* and *Table 3-30* for the detail.
About capture buffer address, see *Table 3-32* for the detail.
This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.6.6 R_VDCE_CapEnable

Function Prototypes

```
r_vdce_Error_t R_VDCE_CapEnable(const uint32_t  Unit,
                                const uint32_t  OutputUnit)
```

Input Parameter

Table 4-89 Input parameter of R_VDCE_CapEnable

Parameter	Description
Unit	Specifies the VDCE unit number.
OutputUnit	Specifies the VDCE unit number which outputs the capturing data. R_VDCE_CAP_NO_DISPLAY is also selectable when capture is independently of displaying.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_RANGE_PARAM	- A parameter was outside the range.
R_VDCE_ERR_NOT_ACCEPTABLE	- A function was called in an incorrect state.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED	- VDCE was not initialized.
R_VDCE_ERR_LAYER_NO_FORMAT	- R_VDCE_LayerFormatSet was not done.
R_VDCE_ERR_LAYER_NO_MEM_GEOMETRY	- R_VDCE_LayerMemGeometrySet was not done.
R_VDCE_ERR_LAYER_NO_VIEW_PORT	- R_VDCE_LayerViewPortSet was not done.
R_VDCE_ERR_LAYER_NOR_BASE	- R_VDCE_LayerBaseSet was not done.
R_VDCE_ERR_CAP_NO_BUF_GEOMETRY	- R_VDCE_CapBufGeometrySetup was not done.
R_VDCE_ERR_CAP_NO_MODE	- R_VDCE_CapModeSet was not done.
R_VDCE_ERR_STRIDE_INCORRECT	- The setting of Stride was incorrect.
R_VDCE_ERR_CHROMAKEY_INCORRECT	- The setting of Chroma key was incorrect.
R_VDCE_ERR_SCALED_SIZE_INCORRECT	- The setting of Scaled size was incorrect.

Description

This function enables the video capturing.
See *Table 3-23* about the connection depending on the parameter.

If OutputUnit is set as 0 or 1, this function enables the connecting layer with following sequence.
1. R_VDCE_LayerDisable (if the layer is already enabled.)
2. Enables capturing
3. R_VDCE_LayerEnable
Note the precondition of R_VDCE_LayerEnable and parameters reflected at the next R_VDCE_LayerEnable timing.

If OutputUnit is set as R_VDCE_CAP_NO_DISPLAY, the capturing starts independently of the layer control and displaying. So, layer control like a Scaling-down is not available.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.
Execute following functions before executing this function.
- R_VDCE_CapBufGeometrySetup
- R_VDCE_CapModeSet

Execute following functions before executing this function if OutputUnit is set as 0 or 1.
- R_VDCE_LayerBaseSet
- R_VDCE_LayerMemGeometrySet
- R_VDCE_LayerFormatSet
- R_VDCE_LayerViewPortSet

See also

r_vdce_Error_t

4.2.6.7   R_VDCE_CapDisable

Function Prototypes

```
r_vdce_Error_t R_VDCE_CapDisable(const uint32_t  Unit)
```

Input Parameter

Table 4-90 Input parameter of R_VDCE_CapDisable

Parameter	Description
Unit	Specifies the VDCE unit number.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_NOT_ACCEPTABLE	- A function was called in an incorrect state.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED	- VDCE was not initialized.
R_VDCE_ERR_STRIDE_INCORRECT	- The setting of Stride was incorrect.
R_VDCE_ERR_CHROMAKEY_INCORRECT	- The setting of Chromakey was incorrect.
R_VDCE_ERR_SCALED_SIZE_INCORRECT	- The setting of Scaled size was incorrect.

Description

This function disables the video capturing.

This function disables the connecting layer with following sequence.
1. R_VDCE_LayerDisable
2. Disables capturing
When user executes R_VDCE_LayerEnable before R_VDCE_CapEnable or during capturing, this function enables the layer.
3. R_VDCE_LayerEnable (if the layer is enabled individually)

Note the precondition of R_VDCE_LayerEnable and parameters reflected at the next R_VDCE_LayerEnable timing.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

4.2.6.8 R_VDCE_CapViewPortSet

Function Prototypes

```
r_vdce_Error_t R_VDCE_CapViewPortSet(const uint32_t  Unit,
                                     const uint32_t  Width,
                                     const uint32_t  Height,
                                     const uint32_t  StartX,
                                     const uint32_t  StartY)
```

Input Parameter

Table 4-91 Input parameter of R_VDCE_CapViewPortSet

Parameter	Description
Unit	Specifies the VDCE unit number.
Width	Specifies the width of the incoming video in pixels.
Height	Specifies the height of the incoming video in pixels.
StartX	Specifies the x position to capture start in pixels.
StartY	Specifies the y position to capture start in pixels.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_RANGE_UNIT	- The unit-number was outside the range.
R_VDCE_ERR_RANGE_PARAM	- A parameter was outside the range.
R_VDCE_ERR_NOT_ACCEPTABLE	- A function was called in an incorrect state.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED	- VDCE was not initialized.
R_VDCE_ERR_SCALED_SIZE_INCORRECT	- The setting of Scaled size was incorrect.

Description

This function sets the capture viewport parameters.
This function can be changed the capturing start position and size from first setting by
R_VDCE_CapBufGeometrySetup.
Dynamic changes can cause momentary screen disruptions. To avoid this, change to IDLE state by
R_VDCE_CapDisable and then call this function.

About range information, see *Table 3-29* and *Table 3-30* for the detail.
This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will
become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

 Prohibited.

Preconditions

 See *Table 2-4* about status conditions.

See also

 r_vdce_Error_t

4.2.6.9 R_VDCE_CapRateSet

Function Prototypes

```
r_vdce_Error_t R_VDCE_CapRateSet(const uint32_t        Unit,
                                 const r_vdce_CapRate_t   Rate,
                                 const r_vdce_CapField_t  Field)
```

Input Parameter

Table 4-92 Input parameter of R_VDCE_CapRateSet

Parameter	Description
Unit	Specifies the VDCE unit number.
Rate	Specifies the writing rate. R_VDCE_CAP_RATE_PER1 R_VDCE_CAP_RATE_PER2 R_VDCE_CAP_RATE_PER4 R_VDCE_CAP_RATE_PER8
Field	Specifies the writing field. R_VDCE_CAP_FIELD_TOP R_VDCE_CAP_FIELD_BOTTOM R_VDCE_CAP_FIELD_FRAME

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK	- No error has occurred.
R_VDCE_ERR_PARAM_INCORRECT	- Parameter was incorrect.
R_VDCE_ERR_RANGE_PARAM	- A parameter was outside the range.
R_VDCE_ERR_NOT_ACCEPTABLE	- A function was called in an incorrect state.
R_VDCE_ERR_FATAL_OS	- Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED	- VDCE was not initialized.

Description

This function sets the writing rate of buffer for the capture.
If this function is not called, all captured data is written to capture buffer.

This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t
r_vdce_CapRate_t
r_vdce_CapField_t

4.2.6.10 R_VDCE_CapExtVsyncSet

Function Prototypes

```
r_vdce_Error_t R_VDCE_CapExtVsyncSet(const uint32_t        Unit,
                                     const uint16_t        HsyncCycle,
                                     const uint32_t        VsyncMaskUs,
                                     const uint32_t        VsyncLackUs)
```

Input Parameter

Table 4-93 Input parameter of R_VDCE_CapExtVsyncSet

Parameter	Description
Unit	Specifies the VDCE unit number.
HsyncCycle	Specifies the horizontal cycle of input signal. The range is 4 to 2046. If 0 is set, VDCE driver set default value automatically.
VsyncMaskUs	Prevent Vsync coming faster than VsyncMaskUs [usec]. If non-0 value is set, vsync masking is enabled. If 0 is set, vsync masking is disabled.
VsyncLackUs	Compensate Vsync coming slower than VsyncLackUs [usec]. If non-0 value is set, vsync compensation is enabled. If 0 is set, vsync compensation is disabled.

Input -Output Parameter

None

Output Parameter

None

Return Codes

R_VDCE_ERR_OK   - No error has occurred.
R_VDCE_ERR_RANGE_UNIT   - The unit-number was outside the range.
R_VDCE_ERR_RANGE_PARAM   - A parameter was outside the range.
R_VDCE_ERR_NOT_ACCEPTABLE   - A function was called in an incorrect state.
R_VDCE_ERR_FATAL_OS   - Fatal error has occurred at OS interface.
R_VDCE_ERR_UNIT_NOTLOCKED   - VDCE was not initialized.

Description

This function sets the Hsync cycle of input signal and Vsync protection.

Hsync cycle is used to determine the field of the interlace signal.
In case of NTSC signal input, the default value is 1716.
In case of PAL signal input, the default value is 1728.
If HsyncCycle is set as 0 or this function is not called, the default value is selected automatically.
NTSC or PAL is selected by R_VDCE_CAP_MODE_PAL flag with R_VDCE_CapModeSet.

Vsync protection timing is made by PixelClock set by R_VDCE_DisplayTimingSet.
The maximum value of VsyncMaskUs and VsyncLackUs is (8,388,480 / PixelClock[MHz])[usec].
If setting value is over the range, this function assumes that the maximum value is set.
If Vsync lack is detected by compensation protection, R_VDCE_INTC_NO_VI_VSYNC_SIGNAL or
R_VDCE_INTC_NO_VI_VSYNC_SIGNAL_1 interrupt occurs.
Vsync protections are disabled by default.
R_VDCE_DisplayVsyncProtectionSet provides the same feature, but with lower accurate.

This setting is valid until R_VDCE_DeInit is executed.

Reentrancy

Non-reentrant as default.
If user implements R_VDCE_Sys_Lock and R_VDCE_Sys_Unlock to prevent multiple executions, this function will become re-entrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-4* about status conditions.

See also

r_vdce_Error_t

5. Types

5.1 Basic Types

This section shows the basic types used in this library.

Table 5-1 Basic Types

Types	Definition		Basic types
char_t	typedef char	char_t	signed char
int8_t	typedef signed char	int8_t	signed char
int16_t	typedef signed short	int16_t	signed short
int32_t	typedef signed int	int32_t	signed int
int64_t	typedef signed long long	int64_t	signed long long
uint8_t	typedef unsigned char	uint8_t	unsigned char
uint16_t	typedef unsigned short	uint16_t	unsigned short
uint32_t	typedef unsigned int	uint32_t	unsigned int
uint64_t	typedef unsigned long long	uint64_t	unsigned long long
float32_t	typedef float	float32_t	float
float64_t	typedef double	float64_t	double

5.2 Definition

This section shows the definition value used in VDCE API.

5.2.1 API Version

This constant is the value which shows the version information of the VDCE driver.

Table 5-2 Definition of VDCE API Version

Name	Description
R_VDCE_VERSION_HI	MSB byte of the version information. It is major version information. This value is changed with release version.
R_VDCE_VERSION_LO	LSB byte of the version information. It is miner version information. This value is changed with release version.

5.2.2 Layer number

The definition of the layer to specify a layer.

Table 5-3 Layer number

Name	Values	Description
R_VDCE_LAYER_SCALER0	0u	This constant is Scaler 0
R_VDCE_LAYER_SCALER1	R_VDCE_LAYER_SCALER0 + 1u	This constant is Scaler 1.
R_VDCE_LAYER_IMAGE2	R_VDCE_LAYER_SCALER1 + 1u	This constant is Image Synthesizer 2.
R_VDCE_LAYER_IMAGE3	R_VDCE_LAYER_IMAGE2 + 1u	This constant is Image Synthesizer 3.
R_VDCE_LAYER_INPUT	R_VDCE_LAYER_LAST + 1u	This constant is Input Controller.

5.2.3 Capture without display

This is used when specifying capturing without the display by R_VDCE_CapEnable.

Table 5-4 Capture without display

Name	Values	Description
R_VDCE_CAP_NO_DISPLAY	0xFFFFFFFFu	Capture without display.

5.2.4 Number of Gamma correction area

The definition of the number of area to gamma correction.

Table 5-5 Number of Gamma correction area

Name	Values	Description
R_VDCE_GAMMA_AREA_NUM	32u	Number of Gamma correction area.

5.3 Enumerated Type

This section shows the enumerated value used in VDCE API Function.

5.3.1 r_vdce_Error_t

Description

Return codes used in almost all API functions. The type describes the error code of VDCE driver functions.

Definition

```
typedef enum
{
 R_VDCE_ERR_OK = 0,
 R_VDCE_ERR_NG,
 R_VDCE_ERR_PARAM_INCORRECT,
 R_VDCE_ERR_RANGE_UNIT,
 R_VDCE_ERR_RANGE_PARAM,
 R_VDCE_ERR_NOT_ACCEPTABLE,
 R_VDCE_ERR_FATAL_OS,
 R_VDCE_ERR_FATAL_HW,
 R_VDCE_ERR_RANGE_LAYER,
 R_VDCE_ERR_UNIT_LOCKED,
 R_VDCE_ERR_UNIT_NOTLOCKED,
 R_VDCE_ERR_DISPLAY_NO_TIMING,
 R_VDCE_ERR_LAYER_NO_FORMAT,
 R_VDCE_ERR_LAYER_NO_MEM_GEOMETRY,
 R_VDCE_ERR_LAYER_NO_VIEW_PORT,
 R_VDCE_ERR_LAYER_NO_BASE,
 R_VDCE_ERR_CAP_NO_BUF_GEOMETRY,
 R_VDCE_ERR_CAP_NO_MODE,
 R_VDCE_ERR_PIXEL_CLOCK,
 R_VDCE_ERR_STRIDE_INCORRECT,
 R_VDCE_ERR_CHROMAKEY_INCORRECT,
 R_VDCE_ERR_SCALED_SIZE_INCORRECT,
 R_VDCE_ERR_INTERRUPT_ENABLED,
 R_VDCE_ERR_NOT_SUPPORTED,
} r_vdce_Error_t
```

Table 5-6 Enumerator of r_vdce_Error_t

Name	Description
R_VDCE_ERR_OK	No error has occurred.
R_VDCE_ERR_NG	An error has occurred, but no specific error code is defined for it.
R_VDCE_ERR_PARAM_INCORRECT	A parameter provided to a function was incorrect. (It excludes the outside of the range)
R_VDCE_ERR_RANGE_UNIT	The unit-number was outside the range.
R_VDCE_ERR_RANGE_PARAM	A parameter was outside the range.
R_VDCE_ERR_NOT_ACCEPTABLE	A function was called in an incorrect state.
R_VDCE_ERR_FATAL_OS	Fatal error has occurred at OS interface.
R_VDCE_ERR_FATAL_HW	Fatal error has occurred at H/W.
R_VDCE_ERR_RANGE_LAYER	The layer-number was outside the range.
R_VDCE_ERR_UNIT_LOCKED	VDCE was already initialized.
R_VDCE_ERR_UNIT_NOTLOCKED	VDCE was not initialized.
R_VDCE_ERR_DISPLAY_NO_TIMING	R_VDCE_DisplayTimingSet was not done.
R_VDCE_ERR_LAYER_NO_FORMAT	R_VDCE_LayerFormatSet or R_VDCE_OirFormatSet is not done.
R_VDCE_ERR_LAYER_NO_MEM_GEOMETRY	R_VDCE_LayerMemGeometrySet or R_VDCE_OirMemGeometrySet is not done.
R_VDCE_ERR_LAYER_NO_VIEW_PORT	R_VDCE_LayerViewPortSet or R_VDCE_OirViewPortSet is not done.
R_VDCE_ERR_LAYER_NO_BASE	R_VDCE_LayerBaseSet or R_VDCE_OirBaseSet is not done.
R_VDCE_ERR_CAP_NO_BUF_GEOMETRY	R_VDCE_CapBufGeometrySetup is not done.
R_VDCE_ERR_CAP_NO_MODE	R_VDCE_CapModeSet is not done.
R_VDCE_ERR_PIXEL_CLOCK	Function failed in the pixel clock setting.
R_VDCE_ERR_STRIDE_INCORRECT	The setting of Stride was incorrect.
R_VDCE_ERR_CHROMAKEY_INCORRECT	The setting of Chromakey was incorrect.
R_VDCE_ERR_SCALED_SIZE_INCORRECT	The setting of Scaled size was incorrect.
R_VDCE_ERR_INTERRUPT_ENABLED	Enabled interrupt was remained.
R_VDCE_ERR_NOT_SUPPORTED	The function was not supported with target device.

See also

None

5.3.2   r_vdce_IntType_t

Description

This type shows the interrupt types.

Definition

```
typedef enum
{
 R_VDCE_INTC_NONE                  = 0,
 R_VDCE_INTC_VBLANK                = 1,
 R_VDCE_INTC_VBLANK_1              = 2,
 R_VDCE_INTC_VBLANK_DELAY          = 3,
 R_VDCE_INTC_SCANLINE              = 4,
 R_VDCE_INTC_OIR_SCANLINE          = 5,
 R_VDCE_INTC_OIR_VBLANK            = 6,
 R_VDCE_INTC_OIR_VSCYNC_WRITE      = 7,
 R_VDCE_INTC_ERROR                 = 8,
 R_VDCE_INTC_NO_VI_VSYNC_SIGNAL    = 9,
 R_VDCE_INTC_NO_VI_VSYNC_SIGNAL_1  = 10,
 R_VDCE_INTC_ERR_OIR_UNDERFLOW     = 11,
 R_VDCE_INTC_ERR_LAYER0_UNDERFLOW  = 12,
 R_VDCE_INTC_ERR_LAYER1_UNDERFLOW  = 13,
 R_VDCE_INTC_ERR_LAYER2_UNDERFLOW  = 14,
 R_VDCE_INTC_ERR_LAYER3_UNDERFLOW  = 15,
 R_VDCE_INTC_CAP_VBLANK            = 16,
 R_VDCE_INTC_CAP_END_OF_FIELD      = 17,
 R_VDCE_INTC_ERR_CAP_WRITE_OVERFLOW = 18,
 R_VDCE_INTC_MAX                   = 19
} r_vdce_IntType_t
```

Table 5-7 Enumerator of r_vdce_IntType_t

Name	Description
R_VDCE_INTC_NONE	No interrupt events.
R_VDCE_INTC_VBLANK	VSYNC output at Scaler 0 interrupt.
R_VDCE_INTC_VBLANK_1	VSYNC output at Scaler 1 interrupt.
R_VDCE_INTC_VBLANK_DELAY	VBLANK detection at Graphics 3.
R_VDCE_INTC_SCANLINE	Scan Line detection at Graphics3 interrupt.
R_VDCE_INTC_OIR_SCANLINE	Scan Line detection at OIR Interrupt.
R_VDCE_INTC_OIR_VBLANK	VSYNC output at OIR interrupt.
R_VDCE_INTC_OIR_VSCYNC_WRITE	VSYNC input at OIR interrupt.
R_VDCE_INTC_ERROR	This type is not supported.
R_VDCE_INTC_NO_VI_VSYNC_SIGNAL	Missing Vsync signal for scaler 0.
R_VDCE_INTC_NO_VI_VSYNC_SIGNAL_1	Missing Vsync signal for scaler 1.
R_VDCE_INTC_ERR_OIR_UNDERFLOW	Frame buffer read underflow signal for OIR.
R_VDCE_INTC_ERR_LAYER0_UNDERFLOW	Frame buffer read underflow signal for graphics 0.
R_VDCE_INTC_ERR_LAYER1_UNDERFLOW	Frame buffer read underflow signal for graphics 1.
R_VDCE_INTC_ERR_LAYER2_UNDERFLOW	Frame buffer read underflow signal for graphics 2.
R_VDCE_INTC_ERR_LAYER3_UNDERFLOW	Frame buffer read underflow signal for graphics 3.
R_VDCE_INTC_CAP_VBLANK	VSYNC input at Scaler 0 interrupt.
R_VDCE_INTC_CAP_END_OF_FIELD	End of field for record function at Scaler 0 interrupt.
R_VDCE_INTC_ERR_CAP_WRITE_OVERFLOW	Frame buffer write overflow signal for scaler 0.
R_VDCE_INTC_MAX	Maximum number of interrupts

See also

R_VDCE_IntcCallbackSet
R_VDCE_IntcCallbackGet
R_VDCE_IntcEnable
R_VDCE_IntcDisable
R_VDCE_Isr

5.3.3 r_vdce_Pin_t

Description

This type is used to select a signal type.

Definition

```
typedef enum
{
  R_VDCE_PIN_ENABLE = 0,
  R_VDCE_PIN_VSYNC,
  R_VDCE_PIN_HSYNC,
  R_VDCE_PIN_VSYNC_E,
  R_VDCE_PIN_HSYNC_E,
  R_VDCE_PIN_CPV_GCK,
  R_VDCE_PIN_POLA,
  R_VDCE_PIN_POLB
} r_vdce_Pin_t
```

Table 5-8 Enumerator of r_vdce_Pin_t

Name	Description
R_VDCE_PIN_ENABLE	Data enable signal. (DE)
R_VDCE_PIN_VSYNC	Vsync signal. (STVA/VS)
R_VDCE_PIN_HSYNC	Hsync signal. (STH/SP/HS)
R_VDCE_PIN_VSYNC_E	Gate start signal / Vertical enable signal. (STVB/VE)
R_VDCE_PIN_HSYNC_E	Source strobe signal / Horizontal enable signal. (STB/LP/HE)
R_VDCE_PIN_CPV_GCK	Gate clock signal. (CPV/GCK)
R_VDCE_PIN_POLA	VCOM voltage polarity control signal A. (POLA)
R_VDCE_PIN_POLB	VCOM voltage polarity control signal B. (POLB)

See also

R_VDCE_DisplayPolaritySet
R_VDCE_DisplaySignalSet
R_VDCE_DisplaySignalGet
r_vdce_TconSig_t

5.3.4 r_vdce_Polarity_t

Description

This type is used to select the polarity in the function R_VDCE_DisplayPolaritySet.

Definition

```
typedef enum
{
    R_VDCE_POLARITY_POSITIVE = 0,
    R_VDCE_POLARITY_NEGATIVE
} r_vdce_Polarity_t
```

Table 5-9 Enumerator of r_vdce_Polarity_t

Name	Description
R_VDCE_POLARITY_POSITIVE	Polarity is positive.
R_VDCE_POLARITY_NEGATIVE	Polarity is negative.

See also

R_VDCE_DisplayPolaritySet

5.3.5 r_vdce_OutEndian_t

Description

This type is used to select the output data endian in function R_VDCE_DisplayOutEndianSet.

Definition

```
typedef enum
{
 R_VDCE_OUT_ENDIAN_LITTLE = 0,
 R_VDCE_OUT_ENDIAN_BIG
} r_vdce_OutEndian_t
```

Table 5-10 Enumerator of r_vdce_OutEndian_t

Name	Description
R_VDCE_OUT_ENDIAN_LITTLE	Output data is little endian.
R_VDCE_OUT_ENDIAN_BIG	Output data is big endian.

See also

R_VDCE_DisplayOutEndianSet

5.3.6   r_vdce_Format_t

Description

The type is used to specify the Frame buffer format in the function R_VDCE_LayerFormatSet.

Definition

```
typedef enum
{
 R_VDCE_RGB565 = 0,
 R_VDCE_RGB0888,
 R_VDCE_ARGB1555,
 R_VDCE_ARGB4444,
 R_VDCE_ARGB8888,
 R_VDCE_CLUT8,
 R_VDCE_CLUT4,
 R_VDCE_CLUT1,
 R_VDCE_YCBCR_422,
 R_VDCE_YCBCR_444,
 R_VDCE_RGBA5551,
 R_VDCE_RGBA8888,
 R_VDCE_YUV_YUYV,
 R_VDCE_YUV_UYVY,
 R_VDCE_YUV_YVYU,
 R_VDCE_YUV_VYUY
} r_vdce_Format_t
```

Table 5-11 Enumerator of r_vdce_Format_t

Name	Description(format)
R_VDCE_RGB565	RGB565
R_VDCE_RGB0888	RGB0888
R_VDCE_ARGB1555	ARGB1555
R_VDCE_ARGB4444	ARGB4444
R_VDCE_ARGB8888	ARGB8888
R_VDCE_CLUT8	CLUT8
R_VDCE_CLUT4	CLUT4
R_VDCE_CLUT1	CLUT1
R_VDCE_YCBCR_422	YCbCr422
R_VDCE_YCBCR_444	YCbCr444
R_VDCE_RGBA5551	RGBA5551
R_VDCE_RGBA8888	RGBA8888
R_VDCE_YUV_YUYV	YUV422 (Y->U->Y->V)
R_VDCE_YUV_UYVY	YUV422 (U->Y->V->Y)
R_VDCE_YUV_YVYU	YUV422 (Y->V->Y->U)
R_VDCE_YUV_VYUY	YUV422 (V->Y->U->Y)

See also

R_VDCE_LayerFormatSet
R_VDCE_OirFormatSet

5.3.7 r_vdce_Scale_t

Description

The type is used to specify the scaling behavior in functions R_VDCE_LayerImgScaleX and R_VDCE_LayerImgScaleY.

Definition

```
typedef enum
{
 R_VDCE_SCALING_LARGER = 0,
 R_VDCE_SCALING_SMALLER,
 R_VDCE_SCALING_DISABLE
} r_vdce_Scale_t
```

Table 5-12 Enumerator of r_vdce_Scale_t

Name	Description
R_VDCE_SCALING_LARGER	Select scale-up.
R_VDCE_SCALING_SMALLER	Select scale-down.
R_VDCE_SCALING_DISABLE	Scaling is disabled.

See also

R_VDCE_LayerImgScaleX
R_VDCE_LayerImgScaleY

5.3.8 r_vdce_CapMode_t

Description

The type is used to specify the mode in the function R_VDCE_CapModeSet.

Definition

```
typedef enum
{
  R_VDCE_CAP_MODE_NONE            = 0,
  R_VDCE_CAP_MODE_YUV_ITU656      = (int32_t)(1uL << 0),
  R_VDCE_CAP_MODE_YUV_8BIT        = (int32_t)(1uL << 1),
  R_VDCE_CAP_MODE_YUV_16BIT       = (int32_t)(1uL << 2),
  R_VDCE_CAP_MODE_RGB_16BPP       = (int32_t)(1uL << 3),
  R_VDCE_CAP_MODE_RGB_18BPP       = (int32_t)(1uL << 4),
  R_VDCE_CAP_MODE_RGB_24BPP       = (int32_t)(1uL << 5),
  R_VDCE_CAP_MODE_DITHER          = (int32_t)(1uL << 6),
  R_VDCE_CAP_MODE_YUV_Y_UV_INVERT = (int32_t)(1uL << 10),
  R_VDCE_CAP_MODE_VSYNC_INVERT    = (int32_t)(1uL << 11),
  R_VDCE_CAP_MODE_HSYNC_INVERT    = (int32_t)(1uL << 12),
  R_VDCE_CAP_MODE_DATA_CLK_INVERT = (int32_t)(1uL << 15),
  R_VDCE_CAP_MODE_VSYNC_CLK_INVERT = (int32_t)(1uL << 16),
  R_VDCE_CAP_MODE_HSYNC_CLK_INVERT = (int32_t)(1uL << 17),
  R_VDCE_CAP_MODE_H_MIRRORING     = (int32_t)(1uL << 18),
  R_VDCE_CAP_MODE_V_MIRRORING     = (int32_t)(1uL << 19),
  R_VDCE_CAP_MODE_FIXED_VSYNC     = (int32_t)(1uL << 20),
  R_VDCE_CAP_MODE_BIG_ENDIAN      = (int32_t)(1uL << 21),
  R_VDCE_CAP_MODE_DE_MODE         = (int32_t)(1uL << 22),
  R_VDCE_CAP_MODE_PAL             = (int32_t)(1uL << 23),
  R_VDCE_CAP_MODE_EAV             = (int32_t)(1uL << 24),
  R_VDCE_CAP_MODE_SYNC_ONLY       = (int32_t)(1uL << 25),
} r_vdce_CapMode_t
```

Table 5-13 Enumerator of r_vdce_CapMode_t

Name	Description
R_VDCE_CAP_MODE_NONE	None of the capture mode selected.
R_VDCE_CAP_MODE_YUV_ITU656	Select ITU-R BT.656 format.
R_VDCE_CAP_MODE_YUV_8BIT	Select ITU-R BT.601 format.
R_VDCE_CAP_MODE_YUV_16BIT	Select YCbCr422 format.
R_VDCE_CAP_MODE_RGB_16BPP	Select RGB565 format.
R_VDCE_CAP_MODE_RGB_18BPP	Select RGB666 format.
R_VDCE_CAP_MODE_RGB_24BPP	Select RGB888 / YCbCr444 format.
R_VDCE_CAP_MODE_DITHER	If this flag is off, dithering mode is Round off. If this flag is on, dithering mode is 2x2 pattern dithering. Dithering is effective in case of frame buffer format is RGB565.
R_VDCE_CAP_MODE_YUV_Y_UV_INVERT	If this flag is off, capturing order is Y->Cb->Y->Cr If this flag is on, capturing order is Cb->Y->Cr->Y This flag is effective in case of ITU-R BT.656 or ITU-R BT.601 input. See INP_H_POS description in H/W User's manual.
R_VDCE_CAP_MODE_VSYNC_INVERT	If this flag is off, V-sync is positive polarity. If this flag is on, V-sync is negative polarity.
R_VDCE_CAP_MODE_HSYNC_INVERT	If this flag is off, H-sync is positive polarity. If this flag is on, H-sync is negative polarity.
R_VDCE_CAP_MODE_DATA_CLK_INVERT	If this flag is off, DV_DATA23..00 capturing timing is falling edge. If this flag is on, DV_DATA23..00 capturing timing is rising edge.
R_VDCE_CAP_MODE_VSYNC_CLK_INVERT	If this flag is off, V-sync signal capturing timing is falling edge. If this flag is on, V-sync signal capturing timing is rising edge.
R_VDCE_CAP_MODE_HSYNC_CLK_INVERT	If this flag is off, H-sync signal capturing timing is falling edge. If this flag is on, H-sync signal capturing timing is rising edge.
R_VDCE_CAP_MODE_H_MIRRORING	If this flag is off, horizontal mirroring is disabled. If this flag is on, horizontal mirroring is enabled.
R_VDCE_CAP_MODE_V_MIRRORING	If this flag is off, vertical mirroring is disabled. If this flag is on, vertical mirroring is enabled.
R_VDCE_CAP_MODE_FIXED_VSYNC	If this flag is off, synchronous V-sync of connecting layer is external input V-sync. If this flag is on, synchronous V-sync of connecting layer is fixed as internally generated free-running Vsync signal.
R_VDCE_CAP_MODE_BIG_ENDIAN	If this flag is off, capturing with little endian. If this flag is on, capturing with big endian.
R_VDCE_CAP_MODE_DE_MODE	If this flag is off, DE mode is disabled. Hsync signal is used to capture. If this flag is on, DE mode is enabled. DE signal is used instead of Hsync signal. DE signal should be input to VDCE0_VI_HSYNC/VDCE1_VI_HSYNC port. R_VDCE_CAP_MODE_HSYNC_INVERT or R_VDCE_CAP_MODE_HSYNC_CLK_INVERT flag is valid for DE signal. DE mode can be selected in case of YCbCr422/RGB565 /RGB666/RGB888/YCbCr444 input mode.
R_VDCE_CAP_MODE_PAL	If this flag is off, input signal is assumed as NTSC (525 Lines/59.94 Hz) system. If this flag is on, input signal is assumed as PAL (625 Lines/50.00 Hz) system. This flag is valid in case of BT.656 or BT.601 input mode.
R_VDCE_CAP_MODE_EAV	If this flag is off, SAV code is converted to Hsync signal. If this flag is on, EAV code is converted to Hsync signal. This flag is valid in case of BT.656 input mode.
R_VDCE_CAP_MODE_SYNC_ONLY	If this flag is off, input data is written to capture buffer. If this flag is on, input data is not written to capture buffer. Only clock, Vsync and Hsync are input from external signal. This mode is used when video data is not captured but the video is output in synchronization with the external Vsync input.

See also

R_VDCE_CapModeSet

5.3.9 r_vdce_Fbformat_t

Description

The type is used to specify the Frame buffer format in the function R_VDCE_CapBufGeometrySetup.

Definition

```
typedef enum
{
 R_VDCE_FB_FORMAT_YCBCR_422 = 0,
 R_VDCE_FB_FORMAT_RGB565    = 1,
 R_VDCE_FB_FORMAT_RGB0888   = 2,
 R_VDCE_FB_FORMAT_YCBCR_444 = 3,
 R_VDCE_FB_RB_SWAP          = (int32_t)(1u << 16)
} r_vdce_Fbformat_t
```

Table 5-14 Enumerator of r_vdce_Fbformat_t

Name	Description
R_VDCE_FB_FORMAT_YCBCR_422	YCbCr422
R_VDCE_FB_FORMAT_RGB565	RGB565
R_VDCE_FB_FORMAT_RGB0888	RGB0888
R_VDCE_FB_FORMAT_YCBCR_444	YCbCr444
R_VDCE_FB_RB_SWAP	Swap R and B signal.

See also

R_VDCE_CapBufGeometrySetup

5.3.10   r_vdce_OirMode_t

Description

This type is used to select the OIR layer mode in function R_VDCE_OirModeSet.

Definition

```
typedef enum
{
 R_VDCE_OIR_MODE_NORMAL = 0,
 R_VDCE_OIR_MODE_SCREEN_SHOTS
} r_vdce_OirMode_t
```

Table 5-15 Enumerator of r_vdce_OirMode_t

Name	Description
R_VDCE_OIR_MODE_NORMAL	OIR layer mode is normal. A distortion image by VOWE is output to display.
R_VDCE_OIR_MODE_SCREEN_SHOTS	OIR layer mode is screen shots. A distortion image by VOWE is not output to display. A distortion image data is only extracted by the frame buffer of OIR.

See also

R_VDCE_OirModeSet

5.3.11 r_vdce_OutFormat_t

Description

This type is used to select the output data format in the function R_VDCE_DisplayOutFormatSet.

Definition

```
typedef enum
{
 R_VDCE_OUT_FORMAT_RGB888 = 0,
 R_VDCE_OUT_FORMAT_RGB666,
 R_VDCE_OUT_FORMAT_RGB565,
 R_VDCE_OUT_FORMAT_SERIAL_RGB
} r_vdce_OutFormat_t
```

Table 5-16 Enumerator of r_vdce_OutFormat_t

Name	Description
R_VDCE_OUT_FORMAT_RGB888	Output data format is RGB888.
R_VDCE_OUT_FORMAT_RGB666	Output data format is RGB666.
R_VDCE_OUT_FORMAT_RGB565	Output data format is RGB565.
R_VDCE_OUT_FORMAT_SERIAL_RGB	Output data format is Serial RGB.

See also

R_VDCE_DisplayOutFormatSet

5.3.12 r_vdce_DitherMd_t

Description

This type is used to select the dither mode in function R_VDCE_DisplayCalibrationSet.

Definition

```
typedef enum
{
 R_VDCE_DTH_MD_TRU = 0,
 R_VDCE_DTH_MD_RDOF,
 R_VDCE_DTH_MD_2X2,
 R_VDCE_DTH_MD_RAND,
 R_VDCE_DTH_MD_NUM
} r_vdce_DitherMd_t
```

Table 5-17 Enumerator of r_vdce_DitherMd_t

Name	Description
R_VDCE_DTH_MD_TRU	Truncate.
R_VDCE_DTH_MD_RDOF	Round-off.
R_VDCE_DTH_MD_2X2	2x2 pattern dither.
R_VDCE_DTH_MD_RAND	Random pattern dither.

See also

r_vdce_Dither_t
R_VDCE_DisplayCalibrationSet

5.3.13   r_vdce_TconPolmode_t

Description

This type is used to select the signal generation mode in the functions R_VDCE_DisplaySignalSet and
R_VDCE_DisplaySignalGet.

Definition

```
typedef enum
{
 R_VDCE_TCON_POLMD_NORMAL = 0,
 R_VDCE_TCON_POLMD_1X1REV,
 R_VDCE_TCON_POLMD_1X2REV,
 R_VDCE_TCON_POLMD_2X2REV
} r_vdce_TconPolmode_t
```

Table 5-18 Enumerator of r_vdce_TconPolmode_t

Name	Description
R_VDCE_TCON_POLMD_NORMAL	Normal mode. Generates the signal that changes twice a horizontal period.
R_VDCE_TCON_POLMD_1X1REV	1x1 reverse mode. Generates the signal whose polarity is inverted every horizontal period.
R_VDCE_TCON_POLMD_1X2REV	1x2 reverse mode. Generates the signal whose polarity is inverted in the first horizontal period and is subsequently inverted every two horizontal periods.
R_VDCE_TCON_POLMD_2X2REV	2x2 reverse mode. Generates the signal whose polarity is inverted every two horizontal periods.

See also

R_VDCE_DisplaySignalSet
R_VDCE_DisplaySignalGet

5.3.14   r_vdce_TconRefsel_t

Description

This type is used to select the signal operating reference in the function R_VDCE_DisplaySignalSet and
R_VDCE_DisplaySignalGet.

Definition

```
typedef enum
{
 R_VDCE_TCON_REFSEL_HSYNC      = 0,
 R_VDCE_TCON_REFSEL_OFFSET_H   = 1
} r_vdce_TconRefsel_t
```

Table 5-19 Enumerator of r_vdce_TconRefsel_t

Name	Description
R_VDCE_TCON_REFSEL_HSYNC	Hsync signal reference.
R_VDCE_TCON_REFSEL_OFFSET_H	Offset Hsync signal reference.

See also

R_VDCE_DisplaySignalSet
R_VDCE_DisplaySignalGet

5.3.15 r_vdce_TconPin_t

Description

This type is used to select the output pin for LCD driving signal in the function R_VDCE_DisplayTconPinSet and R_VDCE_DisplayTconPinGet.

Definition

```
typedef enum
{
 R_VDCE_TCON_PIN_0 = 0,
 R_VDCE_TCON_PIN_1,
 R_VDCE_TCON_PIN_2,
 R_VDCE_TCON_PIN_3,
 R_VDCE_TCON_PIN_4,
 R_VDCE_TCON_PIN_5,
 R_VDCE_TCON_PIN_6,
 R_VDCE_TCON_PIN_LAST
} r_vdce_TconPin_t
```

Table 5-20 Enumerator of r_vdce_TconPin_t

Name	Description
R_VDCE_TCON_PIN_0	Pin0
R_VDCE_TCON_PIN_1	Pin1
R_VDCE_TCON_PIN_2	Pin2
R_VDCE_TCON_PIN_3	Pin3
R_VDCE_TCON_PIN_4	Pin4
R_VDCE_TCON_PIN_5	Pin5
R_VDCE_TCON_PIN_6	Pin6

See also

R_VDCE_DisplayTconPinSet
R_VDCE_DisplayTconPinGet

5.3.16 r_vdce_SigEdge_t

Description

This type is used to select the signal edge in the function R_VDCE_DisplayTconPinSet and
R_VDCE_DisplayTconPinGet.

Definition

```
typedef enum
{
 R_VDCE_SIG_EDGE_RISING = 0,
 R_VDCE_SIG_EDGE_FALLING
} r_vdce_SigEdge_t
```

<div align="center">Table 5-21 Enumerator of r_vdce_SigEdge_t</div>

Name	Description
R_VDCE_SIG_EDGE_RISING	Rising edge
R_VDCE_SIG_EDGE_FALLING	Falling edge

See also

R_VDCE_DisplayTconPinSet
R_VDCE_DisplayTconPinGet

5.3.17 r_vdce_ScaleMode_t

Description

The type is used to specify the optional mode for enlargement and reduction.

Definition

```
typedef enum
{
  R_VDCE_SCALE_MODE_NONE       = 0,
  R_VDCE_SCALE_H_PREFILTER      = (int32_t)(1uL<< 0),
  R_VDCE_SCALE_H_HOLD_INTERPOL  = (int32_t)(1uL<< 1),
  R_VDCE_SCALE_V_HOLD_INTERPOL  = (int32_t)(1uL<< 2)
} r_vdce_ScaleMode_t
```

Table 5-22 Enumerator of r_vdce_ScaleMode_t

Name	Description
R_VDCE_SCALE_MODE_NONE	All flags are off.
R_VDCE_SCALE_H_PREFILTER	If this flag is off, horizontal prefilter is disabled. If this flag is on, horizontal prefilter is enabled. The frequency band for Brightness (Y) and RGB signals are suppressed. Horizontal prefilter can be used when the horizontal scale is reduced.

R_VDCE_SCALE_H_HOLD_INTERPOL	Horizontal interpolation mode.
	If this flag is off, linear interpolation mode is used.
	If this flag is on, hold interpolation mode is used.
	This mode can be selected when the horizontal scale is enlarged or reduced.
	When the interpolation position is between input pixels X_n and X_{n+1}, the Xinterpo interpolation value is defined as follows.
	phase depends on interruption position.
	  hold mode : Xinterpo = X_n
	  linear mode: Xinterpo = (X_n * (4096 – phase) + X_{n+1} * phase) / 4096

R_VDCE_SCALE_V_HOLD_INTERPOL	Use hold interpolation mode as vertical interpolation. If this flag is off, linear interpolation mode is used. If this flag is on, hold interpolation mode is used. Vertical interpolation mode can be selected when the vertical scale is enlarged or reduced.

See also

R_VDCE_LayerImgScaleModeSet

5.3.18 r_vdce_OutSwap_t

Description

This type is used to select whether the output data' blue and red is swapped.

Definition

```
typedef enum
{
 R_VDCE_OUT_SWAP_BR_OFF = 0,
 R_VDCE_OUT_SWAP_BR_ON
} r_vdce_OutSwap_t
```

Table 5-23 Enumerator of r_vdce_OutSwap_t

Name	Description
R_VDCE_OUT_SWAP_BR_OFF	Output data red & blue is not changed.
R_VDCE_OUT_SWAP_BR_ON	Output data' blue channel is swapped with red.

See also

R_VDCE_DisplayOutSwapBR

5.3.19 r_vdce_LayerMode_t

Description

The type is used to specify the optional mode for layer.

Definition

```
typedef enum
{
  R_VDCE_LAYER_MODE_NONE        = 0,
  R_VDCE_LAYER_MODE_V_MIRORING  = (int32_t)(1uL<< 0),
} r_vdce_LayerMode_t
```

Table 5-24 Enumerator of r_vdce_LayerMode_t

Name	Description
R_VDCE_LAYER_MODE_NONE	All flags are off.
R_VDCE_LAYER_MODE_V_MIRORING	The flag is off, vertical mirroring is disabled. The flag is on, vertical mirroring is enabled. This flag can be selected when graphic data is input. When capture data is input, vertical mirroring should be set by R_VDCE_CapModeSet.

See also

R_VDCE_LayerModeSet

5.3.20   r_vdce_SerialClkfreqsel_t

Description

Clock frequency control for serial RGB.
This data type is using in function R_VDCE_DisplaySerialRGBSet.

Definition

```
typedef enum
{
 R_VDCE_SERIAL_CLKFRQ_3 = 0,
 R_VDCE_SERIAL_CLKFRQ_4
} r_vdce_SerialClkfreqsel_t
```

Table 5-25 Enumerator of r_vdce_SerialClkfreqsel_t

Name	Description
R_VDCE_SERIAL_CLKFRQ_3	Triple (x3) speed.
R_VDCE_SERIAL_CLKFRQ_4	Quadruple (x4) speed.

See also

R_VDCE_DisplaySerialRGBSet

5.3.21 r_vdce_SerialClkphase_t

Description

Clock phase adjustment for serial RGB.
Clock phase can be set 0 - 2clk in case of Triple speed mode (R_VDCE_SERIAL_CLKFRQ_3).
Clock phase can be set 0 - 3clk in case of Quadruple speed mode (R_VDCE_SERIAL_CLKFRQ_4).
This data type using in the function R_VDCE_DisplaySerialRGBSet.

Definition

```
typedef enum
{
 R_VDCE_SERIAL_CLKPHASE_0 = 0,
 R_VDCE_SERIAL_CLKPHASE_1,
 R_VDCE_SERIAL_CLKPHASE_2,
 R_VDCE_SERIAL_CLKPHASE_3
} r_vdce_SerialClkphase_t
```

Table 5-26 Enumerator of r_vdce_SerialClkphase_t

Name	Description
R_VDCE_SERIAL_CLKPHASE_0	0 clk.
R_VDCE_SERIAL_CLKPHASE_1	1 clk.
R_VDCE_SERIAL_CLKPHASE_2	2 clk.
R_VDCE_SERIAL_CLKPHASE_3	3 clk.

See also

R_VDCE_DisplaySerialRGBSet

5.3.22   r_vdce_SerialScan_t

Description

Scan direction select for serial RGB.
This data type using in the function R_VDCE_DisplaySerialRGBSet.

Definition

```
typedef enum
{
 R_VDCE_SERIAL_SCAN_FORWARD   = 0,
 R_VDCE_SERIAL_SCAN_REVERSE
} r_vdce_SerialScan_t
```

Table 5-27 Enumerator of r_vdce_SerialScan_t

Name	Description
R_VDCE_SERIAL_SCAN_FORWARD	Forward scan.
R_VDCE_SERIAL_SCAN_REVERSE	Reverse scan.

See also

R_VDCE_DisplaySerialRGBSet

5.3.23　r_vdce_SerialSwap_t

Description

Swap On/Off for serial RGB.
This data type using in the function R_VDCE_DisplaySerialRGBSet.

Definition

```
typedef enum
{
 R_VDCE_SERIAL_SWAP_ON   = 0,
 R_VDCE_SERIAL_SWAP_OFF
} r_vdce_SerialSwap_t
```

Table 5-28 Enumerator of r_vdce_SerialSwap_t

Name	Description
R_VDCE_SERIAL_SWAP_ON	Swap On.
R_VDCE_SERIAL_SWAP_OFF	Swap Off.

See also

R_VDCE_DisplaySerialRGBSet

5.3.24 r_ddb_TimingFlags_t

Description

Timing flags for display signal.

Definition

```
typedef enum
{
 R_DDB_DISP_FLAG_NONE        = (0uL << 0u),
 R_DDB_DISP_FLAG_VOEN        = (1uL << 0u),
 R_DDB_DISP_FLAG_CSYNC       = (1uL << 1u),
 R_DDB_DISP_FLAG_NEGCLK      = (1uL << 2u),
 R_DDB_DISP_FLAG_HSYNC_ACTHI  = (1uL << 3u),
 R_DDB_DISP_FLAG_VSYNC_ACTHI  = (1uL << 4u),
 R_DDB_DISP_FLAG_DESYNC_ACTHI = (1uL << 5u),
 R_DDB_DISP_FLAG_RSDS_RBSW     = (1uL << (6u + 0u)),
 R_DDB_DISP_FLAG_RSDS_PHSEL90  = (0uL << (6u + 1u)),
 R_DDB_DISP_FLAG_RSDS_PHSEL180 = (1uL << (6u + 1u)),
 R_DDB_DISP_FLAG_RSDS_PHSEL270 = (2uL << (6u + 1u)),
 R_DDB_DISP_FLAG_RSDS_PHSEL360 = (3uL << (6u + 1u)),
 R_DDB_DISP_FLAG_RSDS_ENABLE   = (1uL << (6u + 3u))
} r_ddb_TimingFlags_t
```

Table 5-29 Enumerator of r_ddb_TimingFlags_t

Flags	Description
R_DDB_DISP_FLAG_NONE	Empty flag.
R_DDB_DISP_FLAG_VOEN	This flag is not supported.
R_DDB_DISP_FLAG_CSYNC	This flag is not supported.
R_DDB_DISP_FLAG_NEGCLK	If this flag is off, LCD_DATA23..00 outputs with rising edge. If this flag is on, LCD_DATA23..00 outputs with falling edge.
R_DDB_DISP_FLAG_HSYNC_ACTHI	If this flag is off, H-sync is positive polarity. If this flag is on, H-sync is negative polarity.
R_DDB_DISP_FLAG_VSYNC_ACTHI	If this flag is off, V-sync is positive polarity. If this flag is on, V-sync is negative polarity.
R_DDB_DISP_FLAG_DESYNC_ACTHI	If this flag is off, DE signal is positive polarity. If this flag is on, DE signal is negative polarity.
R_DDB_DISP_FLAG_RSDS_RBSW	If this flag is off, RSDS output with even bit first. If this flag is on, RSDS output with odd bit first.
R_DDB_DISP_FLAG_RSDS_PHSEL90	RSDS output with phase shift of 90-degree (default).
R_DDB_DISP_FLAG_RSDS_PHSEL180	RSDS output with phase shift of 180-degree.
R_DDB_DISP_FLAG_RSDS_PHSEL270	RSDS output with phase shift of 270-degree.
R_DDB_DISP_FLAG_RSDS_PHSEL360	RSDS output with phase shift of 360/0-degree.
R_DDB_DISP_FLAG_RSDS_ENABLE	If this flag is off, RSDS output is disabled. If this flag is on, RSDS output is enabled.

Note:
Following options are valid when R_DDB_DISP_FLAG_RSDS_ENABLE flag is on.
- R_DDB_DISP_FLAG_RSDS_RBSW
- R_DDB_DISP_FLAG_RSDS_PHSEL90
- R_DDB_DISP_FLAG_RSDS_PHSEL180
- R_DDB_DISP_FLAG_RSDS_PHSEL270
- R_DDB_DISP_FLAG_RSDS_PHSEL360

Phase shift should be selected one from four options (90/180/270/360).
See H/W User's manual 37.9.6 about RSDS output.

See also

r_ddb_Timing_t

5.3.25 r_vdce_CapRate_t

Description

This type is used to select the capture buffer writing rate.
This data type using in the function R_VDCE_CapRateSet.

Definition

```
typedef enum
{
 R_VDCE_CAP_RATE_PER1   = 0,
 R_VDCE_CAP_RATE_PER2,
 R_VDCE_CAP_RATE_PER4,
 R_VDCE_CAP_RATE_PER8
} r_vdce_CapRate_t
```

Table 5-30 Enumerator of r_vdce_CapRate_t

Name	Description
R_VDCE_CAP_RATE_PER1	1/1 an input signal.
R_VDCE_CAP_RATE_PER2	1/2 an input signal.
R_VDCE_CAP_RATE_PER4	1/4 an input signal.
R_VDCE_CAP_RATE_PER8	1/8 an input signal.

See also

R_VDCE_CapRateSet

5.3.26   r_vdce_CapField_t

Description

This type is used to select the capture buffer writing field select.
This data type using in the function R_VDCE_CapRateSet.
When writing rate 1/2, 1/4, or 1/8 is selected, either the top or bottom field can be selected for writing.
When writing rate 1/1 is selected, top or bottom setting is ignored.

Definition

```
typedef enum
{
 R_VDCE_CAP_FIELD_TOP    = 0,
 R_VDCE_CAP_FIELD_BOTTOM,
 R_VDCE_CAP_FIELD_FRAME
} r_vdce_CapField_t
```

Table 5-31 Enumerator of r_vdce_CapField_t

Name	Description
R_VDCE_CAP_FIELD_TOP	Input is interlace. Write both field (1/1) or write only top field (1/2, 1/4, 1/8),
R_VDCE_CAP_FIELD_BOTTOM	Input is interlace. Write both field (1/1) or write only bottom field (1/2, 1/4, 1/8).
R_VDCE_CAP_FIELD_FRAME	Input is progressive.

See also

R_VDCE_CapRateSet

5.4 Structure Type

This section shows the enumerated types used in VDCE API Function.

5.4.1 r_vdce_Bright_t

Description

The structure holding the value of Brightness DC in the function R_VDCE_DisplayCalibrationSet.

Definition

```
typedef struct
{
 uint16_t    B;
 uint16_t    G;
 uint16_t    R;
} r_vdce_Bright_t
```

Table 5-32 Member of r_vdce_Bright_t structure

Name	Description
B	Brightness (DC) adjustment of B signal. Range is 0x0000 (-512) to 0x03FF (+511). The default value is 0x200 (+0).
G	Brightness (DC) adjustment of G signal. Range is 0x0000 (-512) to 0x03FF (+511). The default value is 0x200 (+0).
R	Brightness (DC) adjustment of R signal. Range is 0x0000 (-512) to 0x03FF (+511). The default value is 0x200 (+0).

See also

R_VDCE_DisplayCalibrationSet

5.4.2 r_vdce_Contrast_t

Description

The structure holding the value of Contrast gain in the function R_VDCE_DisplayCalibrationSet.

Definition

```
typedef struct
{
 uint8_t     B;
 uint8_t     G;
 uint8_t     R;
} r_vdce_Contrast_t
```

Table 5-33 Member of r_vdce_Contrast_t structure

Name	Description
B	Contrast (gain) adjustment of B signal. Range is 0x00 (x0) to 0xFF (x 255/128 = approx. x2). The default value is 0x80 (x 128/128 = x1).
G	Contrast (gain) adjustment of G signal. Range is 0x00 (x0) to 0xFF (x 255/128 = approx. x2). The default value is 0x80 (x 128/128 = x1).
R	Contrast (gain) adjustment of R signal. Range is 0x00 (x0) to 0xFF (x 255/128 = approx. x2). The default value is 0x80 (x 128/128 = x1).

See also

R_VDCE_DisplayCalibrationSet

5.4.3 r_vdce_Matrix_t

Description

The structure holding the conversion coefficients for the conversion matrix.
This structure used in the function R_VDCE_LayerMatrixSet.

Definition

```
typedef struct
{
 float64_t RY;
 float64_t RU;
 float64_t RV;
 float64_t GY;
 float64_t GU;
 float64_t GV;
 float64_t BY;
 float64_t BU;
 float64_t BV;
 int32_t Y_OFF;
 int32_t U_OFF;
 int32_t V_OFF;
 uint32_t UV_SHIFT_ZERO;
} r_vdce_Matrix_t
```

Table 5-34 Member of r_vdce_Matrix_t structure

Name	Description
RY	Value of RY. Range is -4.0 to +4.0.
RU	Value of RU. Range is -4.0 to +4.0.
RV	Value of RV. Range is -4.0 to +4.0.
GY	Value of GY. Range is -4.0 to +4.0.
GU	Value of GU. Range is -4.0 to +4.0.
GV	Value of GV. Range is -4.0 to +4.0.
BY	Value of BY. Range is -4.0 to +4.0.
BU	Value of BU. Range is -4.0 to +4.0.
BV	Value of BV. Range is -4.0 to +4.0.
Y_OFF	Value of offset of Y. Range is -128 to +127.
U_OFF	Value of offset of U. Range is -128 to +127.
V_OFF	Value of offset of V. Range is -128 to +127.
UV_SHIFT_ZERO	Convert type 0: GBR to GBR 1: GBR to YCbCr (valid only R_VDCE_LAYER_INPUT) 2: YCbCr to GBR 3: YCbCr to YCbCr (valid only R_VDCE_LAYER_INPUT)

See also

R_VDCE_LayerMatrixSet

5.4.4   r_vdce_ClutARGB32_t

Description

The structure holding the value of ARGB for CLUT in the function R_VDCE_LayerClutSet.

Definition

```
typedef struct
{
 uint8_t B;
 uint8_t G;
 uint8_t R;
 uint8_t A;
} r_vdce_ClutARGB32_t
```

Table 5-35 Member of r_vdce_ClutARGB32_t structure

Name	Description
B	Value of Blue.
G	Value of Green.
R	Value of Red.
A	Value of Alpha.

See also

R_VDCE_LayerClutSet

5.4.5 r_vdce_CkARGB32_t

Description

The structure holding the value of ARGB for Chromakey in the function R_VDCE_LayerChromaKeyEnable.

Definition

```
typedef struct
{
 uint8_t B;
 uint8_t G;
 uint8_t R;
 uint8_t A;
} r_vdce_CkARGB32_t
```

Table 5-36 Member of r_vdce_CkARGB32_t structure

Name	Description
B	Value of Blue.
G	Value of Green.
R	Value of Red.
A	Value of Alpha.

See also

R_VDCE_LayerChromaKeyEnable

5.4.6   r_ddb_Timing_t

Description

Sub-Timing information of horizontal / vertical display line. This is using in the function R_VDCE_DisplayTimingSet.

Definition

```
typedef struct
{
 const char          *Name;
 int32_t             ScreenWidth;
 int32_t             ScreenHeight;
 r_ddb_SubTiming_t   H;
 r_ddb_SubTiming_t   V;
 uint32_t            Flags;
 uint32_t            PixelClock;
} r_ddb_Timing_t
```

Table 5-37 Member of r_ddb_Timing_t structure

Name	Description
Name	Display name as unique identifier of a specific display. VDCE driver doesn't use this parameter.
ScreenWidth	Width of screen (in pixel).
ScreenHeight	Height of screen (in pixel).
H	Horizontal sub-timing.
V	Vertical sub-timing.
Flags	Timing Flags. Several flags can be set with OR operation. Available flag is described in r_ddb_TimingFlags_t
PixelClock	Pixel clock frequency (in Hz)

See also

r_ddb_SubTiming_t
r_ddb_TimingFlags_t
R_VDCE_DisplayTimingSet

5.4.7   r_ddb_SubTiming_t

Description

The structure holding the value of output timing for LCD panel.

Definition

```
typedef struct
{
 int32_t Total;
 int32_t BlankWidth;
 int32_t FrontPorch;
 int32_t SyncWidth;
} r_ddb_SubTiming_t
```

Table 5-38 Member of r_ddb_SubTiming_t structure

Name	Description
Total	Length of total line. (In pixel)
BlankWidth	Length of blanking period. (In pixel)
FrontPorch	Length of front porch. (In pixel)
SyncWidth	Length of active sync pulse. (In pixel)

See also

r_ddb_Timing_t

5.4.8   r_vdce_Dither_t

Description

The structure holding the value of dithering mode in the function R_VDCE_DisplayCalibrationSet.

Definition

```
typedef struct
{
 r_vdce_DitherMd_t  Sel;
 uint8_t            Pa;
 uint8_t            Pb;
 uint8_t            Pc;
 uint8_t            Pd;
} r_vdce_Dither_t
```

Table 5-39 Member of r_vdce_Dither_t structure

Name	Description
Sel	Panel dither operation mode.
Pa	Pattern value (A) of 2x2 pattern dither. Range is 0 to 3. The default value is 3.
Pb	Pattern value (B) of 2x2 pattern dither. Range is 0 to 3. The default value is 0.
Pc	Pattern value (C) of 2x2 pattern dither. Range is 0 to 3. The default value is 2.
Pd	Pattern value (D) of 2x2 pattern dither. Range is 0 to 3. The default value is 1.

See also

r_vdce_DitherMd_t
R_VDCE_DisplayCalibrationSet

5.4.9   r_vdce_Hsync_t

Description

The structure holding the value of TCON reference timing of Hsync in the function R_VDCE_DisplayHsyncSet and R_VDCE_DisplayHsyncGet.

Definition

```
typedef struct
{
 uint16_t          Half;
 uint16_t          Offset;
} r_vdce_Hsync_t
```

Table 5-40 Member of r_vdce_Hsync_t structure

Name	Description
Half	Clock count from the rising edge of the Hsync signal as the counting timing of horizontal counter. 1/2fH Timing. Range is 0 to 0x7FF.
Offset	Offset Hsync Signal Timing. The clock cycle count from the rising edge of the Hsync signal. Range is 0 to 0x7FF.

See also

R_VDCE_DisplayHsyncSet
R_VDCE_DisplayHsyncGet

5.4.10 r_vdce_Signal_t

Description

The structure holding the value of TCON signal configuration in the function R_VDCE_DisplaySignalSet and R_VDCE_DisplaySignalGet.

Definition

```
typedef struct
{
  uint16_t              TconHsvs;
  uint16_t              TconHwvw;
  r_vdce_TconPolmode_t  TconMd;
  r_vdce_TconRefsel_t   TconHsSel;
} r_vdce_Signal_t
```

Table 5-41 Member of r_vdce_Signal_t structure

Name	Description
TconHsvs	Signal Pulse Start Position (First Changing Timing). If signal type is R_VDCE_PIN_VSYNC or R_VDCE_PIN_VSYNC_E, it starts pulse output after the time specified by this value from the rising edge of the Vsync signal (1/2fH cycles). Range is 0 to 0x7FF. If signal type is R_VDCE_PIN_HSYNC, R_VDCE_PIN_HSYNC_E or R_VDCE_PIN_CPV_GCK, starts pulse output after the time specified by this value + 1 from the rising edge of the Hsync signal (clock cycles). Range is 0 to 0x7FF. If signal type is R_VDCE_PIN_POLA or R_VDCE_PIN_POLB, it starts pulse output after the time specified by this value from the rising edge of the Hsync signal (clock cycles). If signal generation mode is R_VDCE_TCON_POLMD_NORMAL, range is 0 to 0x7FF. If signal generation mode is not R_VDCE_TCON_POLMD_NORMAL, range is 1 to 0x7FF.
TconHwvw	Pulse width (Second Changing Timing). Range is 0 to 0x7FF. Outputs a pulse of the duration of this value. If signal type is R_VDCE_PIN_VSYNC or R_VDCE_PIN_VSYNC_E, the value is by the 1/2fH cycles. Otherwise, the value is by the clock cycles.
TconMd	Signal generation mode select. This is valid when signal type is R_VDCE_PIN_POLA or R_VDCE_PIN_POLB.
TconHsSel	Signal operating reference select. This is valid when signal type is not R_VDCE_PIN_VSYNC or R_VDCE_PIN_VSYNC_E.

See also

r_vdce_TconPolmode_t
r_vdce_TconRefsel_t
R_VDCE_DisplaySignalSet
R_VDCE_DisplayHsyncGet

5.4.11 r_vdce_TconSig_t

Description

The structure holding the value of TCON pin setting in the function R_VDCE_DisplayTconPinSet and R_VDCE_DisplayTconPinGet.

Definition
```
typedef struct
{
 r_vdce_Pin_t          SigType;
 r_vdce_SigEdge_t      Edge;
} r_vdce_TconSig_t
```

Table 5-42 Member of r_vdce_TconSig_t structure

Name	Description
SigType	Signal type to output.
Edge	Signal edge.

See also

r_vdce_Pin_t
r_vdce_SigEdge_t
R_VDCE_DisplayTconPinSet
R_VDCE_DisplayTconPinGet

5.4.12   r_vdce_Gamma_t

Description

This structure is used to set the Gamma correction in the function R_VDCE_DisplayGammaCorrectSet.

Definition

```
typedef struct
{
 r_vdce_GammaCorrect_t*   B;
 r_vdce_GammaCorrect_t*   G;
 r_vdce_GammaCorrect_t*   R;
} r_vdce_Gamma_t
```

Table 5-43 Member of r_vdce_Gamma_t structure

Name	Description
B	Pointer to structure of Gamma adjustment of B signal. If this is set to NULL, Gamma adjustment of B signal is not changed.
G	Pointer to structure of Gamma adjustment of G signal. If this is set to NULL, Gamma adjustment of G signal is not changed.
R	Pointer to structure of Gamma adjustment of R signal. If this is set to NULL, Gamma adjustment of R signal is not changed.

See also

r_vdce_GammaCorrect_t
R_VDCE_DisplayGammaCorrectSet

5.4.13   r_vdce_GammaCorrect_t

Description

Gamma correction parameters. This is used by r_vdce_Gamma_t.
Gamma correction is carried out by dividing an input signal having 256 gradation levels into 32 and controlling the gain of each area. Gain coefficient of each area can be set as 0 to approx. 2.0 [times].
Refer to H/W UM 38.8.1.6 for more details.

Definition

```
typedef struct
{
 uint8_t   Area[R_VDCE_GAMMA_AREA_NUM];
 uint16_t  Gain[R_VDCE_GAMMA_AREA_NUM];
} r_vdce_GammaCorrect_t
```

Table 5-44 Member of r_vdce_GammaCorrect_t structure

Name	Description
Area[n]	Threshold of area #n which Gain #n is applied. The range of area #n is as followed. Area#0 is from Area[0] to Area[1]. Area#1 is from Area[1] to Area[2]. : Area#30 is from Area[30] to Area[31]. Area#31 is from Area[31] to 255. Each value should be set as following conditions. Area[0] = 0. Area[n-1] < Area[n] < Area[n+1] (1 <= n <=30). Area[31] <= 255. Initial value of Area[n] = n*8.
Gain[n]	Gain coefficient of area #n. The value of gain coefficient is Gain[n]/1024 [times]. The range of Gain[n] is from 0 to 2047. Initial value of Gain[n] = 1024 (coefficient = x1).

See also

r_vdce_Gamma_t

		Renesas Graphics Library
Revision History		Video Data Controller E (VDCE) Driver
		User's Manual: Software

Rev.	Date	Description	
		Page	Summary
0.1	Nov 29, 2018	-	First edition.
0.2	Mar 28, 2019	12	Changed the status error condition. R_VDCE_ErrorCallbackSet
		31	Added the description.
		38, 178	Changed the argument name from FBformat to FbFormat. R_VDCE_CapBufGeometrySetup
		58	Changed the error callback specification.

		68, 70	Added the rage information of Scanline.
			  R_VDCE_IntcScanlineSet
			  R_VDCE_IntcOirScanlineSet

		101	Added the "const" to argument. R_VDCE_DisplayCalibrationSet
		203, 205	Added the typecast to enum value.
		208, 211	Added the enum member for delimiter.
1.0	June 12, 2019	9, 10	Improved the description of Error handling.
		14, 190	Changed the status error condition of R_VDCE_CapDisable.
		20	Added the range of PixelClock
		24, 25	Added the range of DispPosX, DispPosY
		24,39, 115, 117	Fixed the range of Stride.
		53	Added the return code.
		64	Added the description of VODDR mode.

1.1	Oct 10, 2019	13, 14,
		37, 53,
		54, 111,
		115, 116,
		197~202,
			Added new API functions.
			  R_VDCE_DisplayTimingGet
			  R_VDCE_CapViewPortSet
			  R_VDCE_CapRateSet
			  R_VDCE_CapExtVsyncSet

		41	Added the description for DE mode capturing.
		43	Fixed the description for SAV/EAV code.
		44	Added the description for capturing one field.
		46	Added the description for external Vsync mode with graphic input.
		47	Fixed table title.

		50, 183,
		184, 189,
		191,
			Changed the specification to how to specify DE mode.
			  R_VDCE_CapBufGeometrySetup
			  R_VDCE_CapModeSet
			  R_VDCE_CapBufFieldSetup1
			  R_VDCE_CapBufFieldSetup2

		117, 187	Fixed the description for reentrancy.
			  R_VDCE_LayerBaseSet
			  R_VDCE_CapBufSet

		183, 187,
		189, 191,
			Changed the specification to specify capture buffer address.
			  R_VDCE_CapBufGeometrySetup
			  R_VDCE_CapBufSet
			  R_VDCE_CapBufFieldSetup1
			  R_VDCE_CapBufFieldSetup2

		214, 215	Added new enumerator to r_vdce_CapMode_t.

			R_VDCE_CAP_MODE_DE_MODE R_VDCE_CAP_MODE_PAL R_VDCE_CAP_MODE_EAV R_VDCE_CAP_MODE_SYNC_ONLY

		234, 235	Added new types.
			  r_vdce_CapRate_t
			  r_vdce_CapField_t

1.2	Nov 29, 2019	115, 117, 184, 185, 187, 189, 191,	Fixed the return code.
			  R_VDCE_DisplayTimingGet
			  R_VDCE_LayerBaseSet
			  R_VDCE_CapBufGeometrySetup
			  R_VDCE_CapModeSet
			  R_VDCE_CapBufSet
			  R_VDCE_CapBufFieldSetup1
			  R_VDCE_CapBufFieldSetup2

1.3	Dec 25, 2019	44, 45, 184, 185, 188, 190, 192	Fixed the capture buffer address specification. Add the chapter for capture buffer.
		51	Fixed the missing capture functions.
2.0	May 10, 2020	14, 54,	Add the new function. R_VDCE_LayerBufSet
		116	Add the description. R_VDCE_DisplayTimingGet
		118	Fixed the description for reentrancy. Restored to Ver1.0 specifications R_VDCE_LayerBaseSet
		200	Add the restriction of dynamic change. R_VDCE_CapViewPortSet

RENESAS

Renesas Electronics Corporation

http://www.renesas.com

Colophon 7.0

RENESAS

ルネサス エレクトロニクス株式会社

Renesas Graphics Library
Video Data Controller E (VDCE) Driver