User's Manual

CONFIDENTIAL

RH850/D1x Device Family

Renesas Graphics Library Video Output Warping Engine (VOWE) Driver

User's Manual: Software

Renesas Electronics
www.renesas.com

Rev. 2.0 May 2020

Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other claims involving patents, copyrights, or other intellectual property rights of third parties, by or arising from the use of Renesas Electronics products or technical information described in this document, including but not limited to, the product data, drawings, charts, programs, algorithms, and application examples.
3. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You shall not alter, modify, copy, or reverse engineer any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copying or reverse engineering.
5. Renesas Electronics products are classified according to the following two quality grades: “Standard” and “High Quality”. The intended applications for each Renesas Electronics product depends on the product’s quality grade, as indicated below.
   “Standard”: Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; industrial robots; etc.
   “High Quality”: Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.
   Unless expressly designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not intended or authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems; surgical implantations; etc.), or may cause serious property damage (space system; undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or any third parties arising from the use of any Renesas Electronics product that is inconsistent with any Renesas Electronics data sheet, user’s manual or other Renesas Electronics document.
6. When using Renesas Electronics products, refer to the latest product information (data sheets, user’s manuals, application notes, “General Notes for Handling and Using Semiconductor Devices” in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat dissipation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions, failure or accident arising out of the use of Renesas Electronics products outside of such specified ranges.
7. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics, such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Unless designated as a high reliability product or a product for harsh environments in a Renesas Electronics data sheet or other Renesas Electronics document, Renesas Electronics products are not subject to radiation resistance design. You are responsible for implementing safety measures to guard against the possibility of bodily injury, injury or damage caused by fire, and/or danger to the public in the event of a failure or malfunction of Renesas Electronics products, such as safety design for hardware and software, including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures. Because the evaluation of microcomputer software alone is very difficult and impractical, you are responsible for evaluating the safety of the final products or systems manufactured by you.
8. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. You are responsible for carefully and sufficiently investigating applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive, and using Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
9. Renesas Electronics products and technologies shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You shall comply with any applicable export control laws and regulations promulgated and administered by the governments of any countries asserting jurisdiction over the parties or transactions.
10. It is the responsibility of the buyer or distributor of Renesas Electronics products, or any other party who distributes, disposes of, or otherwise sells or transfers the product to a third party, to notify such third party in advance of the contents and conditions set forth in this document.
11. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note 1) “Renesas Electronics” as used in this document means Renesas Electronics Corporation and also includes its directly or indirectly controlled subsidiaries.

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

(Rev.4.0-1 November 2017)

How to Use This Manual

1. Purpose and Target Readers

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

> 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 VOWE 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
	Description of VOWE	Renesas Graphics Library Video Output Warping Engine (VOWE) Driver User's Manual: Software	LLWEB-10035993 (This manual)
	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	LLWEB-10063801

		User's Manual: Software	
	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
DL	Display List. It contains input data's shape & the output data's shape. In Addition, control commands are also inserted in DL.
H/W	Hardware
IER	Interrupt that shows VOWE H/W processed the invalid DL.
Layer	H/W concept of the stackable visual area on the display
OIR	Output Image Rendering. It is part of input/output of VOWE in VDCE.
Stride	Distance in pixels between two adjacent rows in the display.
TRA	Interrupt that shows the warping correction processing was complete for one frame.
VDCE	Video Data Controller E. This is H/W, which controls video input, image synthesis and video output.
VOWE	Video Output Warping Engine. This is H/W, which controls video warping.
Vsync	Vertical Synchronization. One of the reference signal for the display device.
Warping	Changing the shape of an input image as per the specification of the destination.
WM	Window Manager. This is a driver stack, which enables an abstract access to VDCE driver.

Table of Contents

1. Overview 3
1.1 Feature and Scope 3
1.2 Component Structure 3

2. Basic Specification 4
2.1 Summary Specification 4
2.2 Reserved Word 4
2.3 Interrupt Handler List 4
2.4 Error Handling 5
2.4.1 Return code 5
2.4.2 Callback 5
2.4.3 Other 5
2.5 State Transition 6

3. Function Description 8
3.1 Fundamental Concepts 8
3.1.1 VOWE unit 8
3.1.2 Warping engine 8
3.1.3 Display List 9
3.1.4 Processing timing 11
3.1.5 Buffering Mode 13
3.2 Using the API 14
3.2.1 VOWE initialization 14
3.2.2 Buffer setting 14
3.2.3 Delay setting 19
3.2.4 Start and Stop 20
3.2.5 VOWE completion 20
3.3 Device difference 21
3.4 Header File List 21

4. Functions 22
4.1 Function List 22
4.2 VOWE API Functions 23
4.2.1 R_VOWE_Init 23
4.2.2 R_VOWE_DeInit 25
4.2.3 R_VOWE_Open 26
4.2.4 R_VOWE_Close 28
4.2.5 R_VOWE_Start 29
4.2.6 R_VOWE_Stop 31
4.2.7 R_VOWE_DLChange 32
4.2.8 R_VOWE_VersionStringGet 33
4.2.9 R_VOWE_MacroVersionGet 34
4.3 Interrupt Functions 35
4.3.1 R_VOWE_Isr0 35

5. Types 36
5.1 Basic Types 36
5.2 VOWE API Types 37
5.2.1 r_vowe_CallbackFunction_t 37
5.3 Definition 38
5.3.1 API version 38
5.3.2 Callback factor 38
5.3.3 Pixel Size 38

5.4 Enumerated Types 39
5.4.1 r_vowe_Error_t 39
5.4.2 r_vowe_BufferMode_t 40
5.4.3 r_vowe_ColorFormat_t 41
5.4.4 r_vowe_DestMode_t 42
5.5 Structure 43
5.5.1 r_vowe_SourceConfig_t 43
5.5.2 r_vowe_DisplayList_t 44
5.5.3 r_vowe_DestConfig_t 45

1.Overview

1.1 Feature and Scope

The VOWE driver is a driver stack, which enables an abstract access to the device's Image Renderer 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 VOWE is shown in
Figure 1-1.

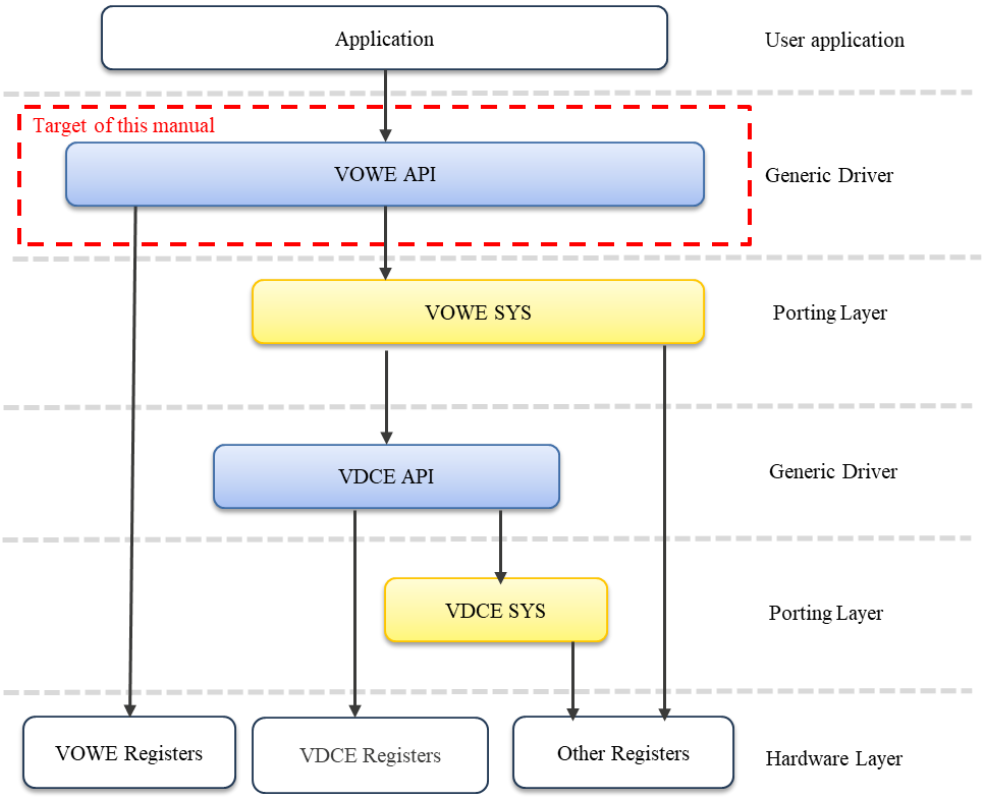

Figure 1-1 Component Structure

For 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 Specification

Items	Description
Target LSI	RH850/D1M1(H), RH850/D1M1-V2, RH850/D1M1A, RH850/D1M2(H).
Main Feature	• VOWE driver can change the shape of input image and output changed image data. • VOWE driver has 2 buffer modes: Frame-buffer mode & Ring-buffer mode. • Configure distortion with source and destination formats. • Start / stop warping. • Change the display list.
Semaphore / Mutex	N/A for VOWE. This can be implemented with porting layer.
Interrupts	Used in VOWE, for more details please see *Section 2.3*.

2.2 Reserved Word

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

Table 2-2 Prefixes

Prefix	Description
R_VOWE_*	Prefix for VOWE Module
r_vowe_*	

2.3 Interrupt Handler List

Table 2-3 Interrupt Handling List

No.	Interrupt Name	Interrupt Handling Name	Description
(1)	INTVOWE	R_VOWE_Isr0	Interrupt is triggered with following factor. TRA: warping correction is completed IER: Invalid display list is detected.

2.4 Error Handling

2.4.1 Return code

VOWE driver returns 3 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_VOWE_ERR_PARAM_INCORRECT
- R_VOWE_ERR_RANGE_UNIT
- R_VOWE_ERR_RANGE_PARAM

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_VOWE_ERR_NOT_ACCEPTABLE
- R_VOWE_ERR_SYS_VDCE

2.4.1.3 Hardware level

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

- R_VOWE_ERR_FATAL_HW
- R_VOWE_ERR_NG

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_VOWE_ERR_FATAL_OS

2.4.2 Callback

Following errors event occur by invalid display list. In this case, please set valid display list again.

- R_VOWE_FACTOR_IER

2.4.3 Other

When R_VOWE_FACTOR_STOP callback function isn't called with during the 2 VSYNC period after R_VOWE_Stop function execution, unexpected error occurs internally. In this case, please reset the RH850/D1x device.

2.5 State Transition

Table 2-4 State Details

No.	State Name	Description
(1)	Uninitialized	It shows the VOWE driver is not initialized.
(2)	Initialized	It shows the VOWE driver is initialized and all internal variables are set to default values.
(3)	Idle	It shows the configuration for warping correction has been set and no warping have occurred yet.
(4)	Executing	It shows the VOWE driver is performing a warping.

The image describes state transition.

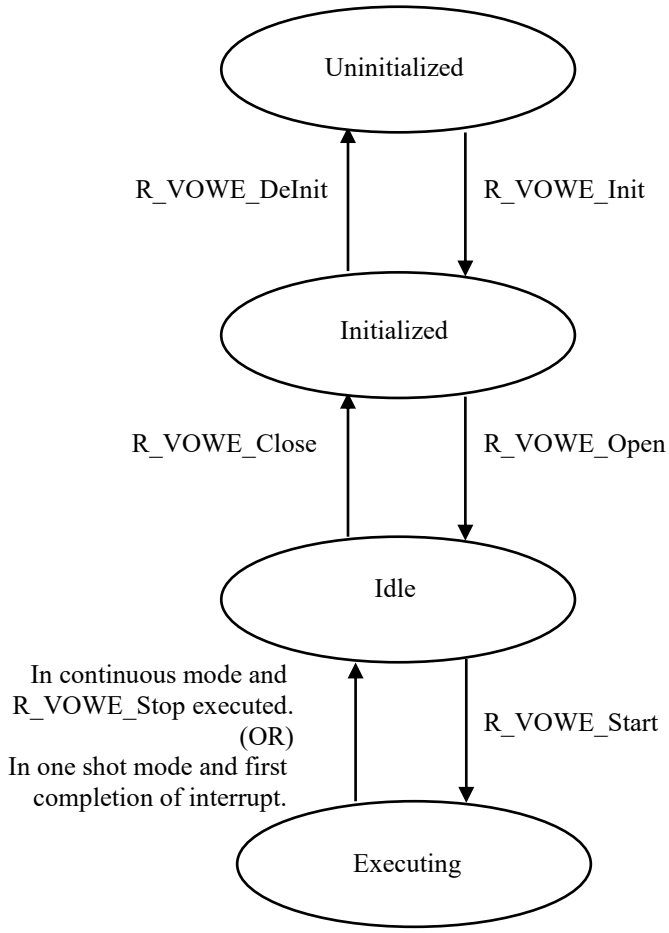

Figure 2-1 State Transition Diagram of VOWE driver

Table 2-5 State Transition Table of VOWE driver

unction name	State			
	Uninitialized	Initialized	Idle	Executing
R_VOWE_Init	OK	NG	NG	NG
R_VOWE_DeInit	NG	OK	NG	NG
R_VOWE_Open	NG	OK	NG	NG
R_VOWE_Close	NG	NG	OK	NG
R_VOWE_Start	NG	NG	OK	NG
R_VOWE_Stop	NG	NG	NG	OK
R_VOWE_DLChange	NG	NG	OK	OK
R_VOWE_MacroVersionGet	OK	OK	OK	OK
R_VOWE_VersionStringGet	OK	OK	OK	OK
R_VOWE_Isr0	OK	OK	OK	OK

3.Function Description

3.1 Fundamental Concepts

3.1.1 VOWE unit

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

Table 3-1 Number of units

Feature	RH850/D1x Device Name	
	D1L2(H)	D1M1(H) / D1M1-V2 / D1M1A / D1M2(H)
Number of units	0	1

Almost VOWE API functions have the argument “Unit”.
User specifies the VOWE H/W unit number to be controlled. The range is only 0.

3.1.2 Warping engine

The VOWE driver is API for Renesas special IP for image distortion. Relies on external generated command lists. The VOWE driver is a driver stack, which enables an abstract access to the device’s video output warping engine (VOWE) hardware macro. The abstraction shall simplify the usage by application developer and also make it possible to use the same API for different hardware. VOWE driver provides the function which controls the H/W.

The distortion correction hardware revises distortion, and by cooperation with VDCE, the distortion correction hardware automatically conducts the capturing image, revising distortion and outputting image. When the VOWE driver is used, the Unit0 of VDCE is controlled by the VOWE driver. VOWE driver needs to prepare “Display List” (DL). Basically, DL consists of two portions. One is input data’s shape, the other is output data’s shape. In addition, control commands are inserted in DL. The display list (DL) specifies desired source coordinates for the input image data in the input buffer and the output destination coordinates in the output buffer.

VDCE has 4 image-synthesizers (Scaler0, Scaler1, Image-synthesizer2, Image-synthesizer3).
Input of VOWE: Synthesized image data (max. 4 layers)
Output of VOWE: Corrected image data

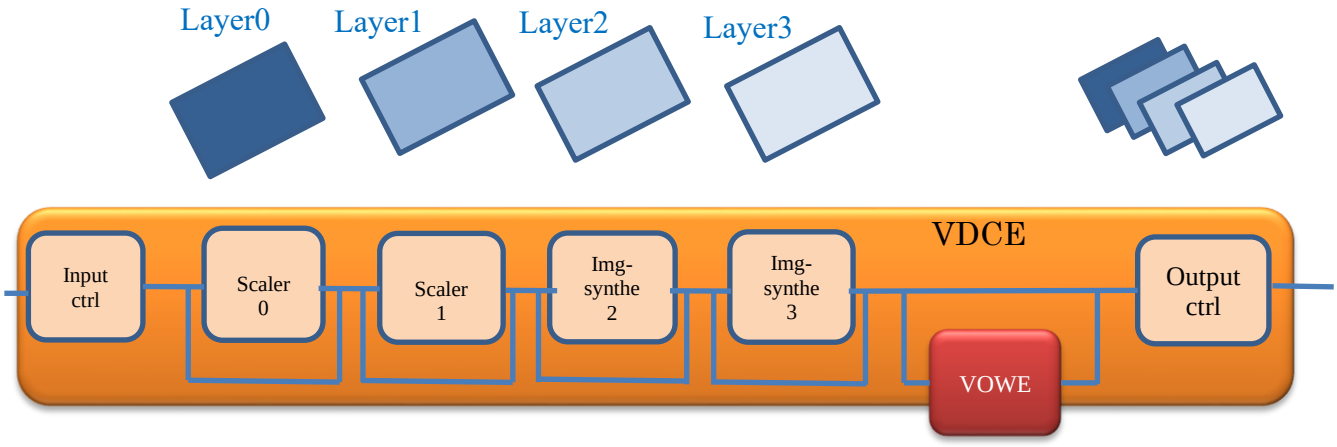

Figure 3-1 Video output module (VOWE & VDCE) overview

3.1.3 Display List

The display list contains detailed information on coordinates for use in correcting warping in (or reshaping) the input image data. When image data are input, the VOWE driver sequentially executes warping correction (or reshaping) by coordinate transformations based on the instructions in the display list. This continues until the data of an image is corrected.

Below figure shows a schematic view of coordinate transformation according to a display list. The display list specifies the desired source coordinates for the input image data in the input buffer and the output destination coordinates in the output buffer. The source coordinates and the destination coordinates are specified in units of meshes of triangles. The VOWE hardware reads the display list and handles the image transformation in these units.

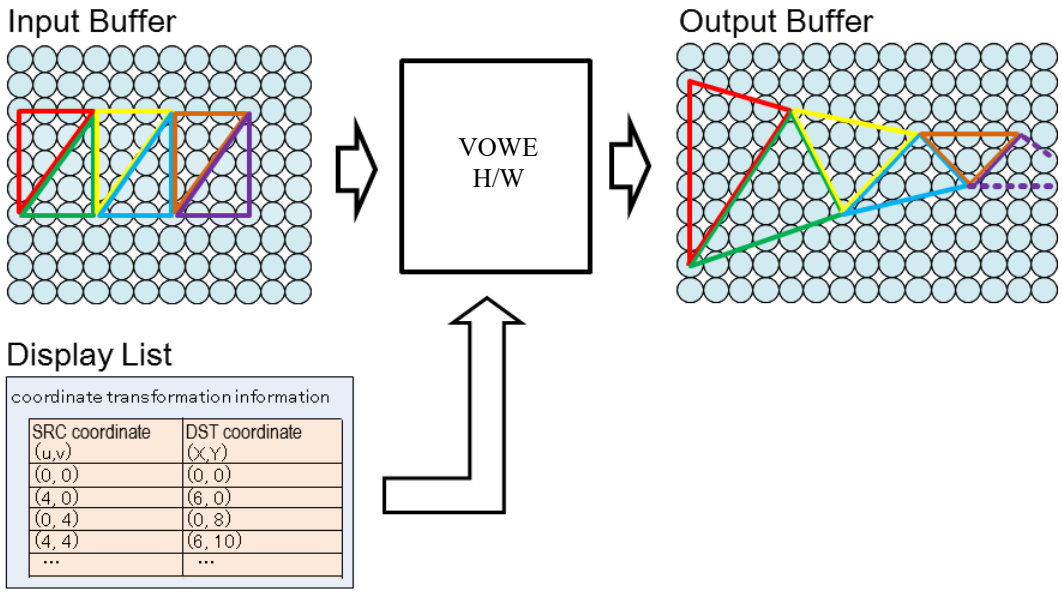

Figure 3-2 Concept of coordinate transformation

Following figures explain the Display List structure which VOWE driver is supporting.

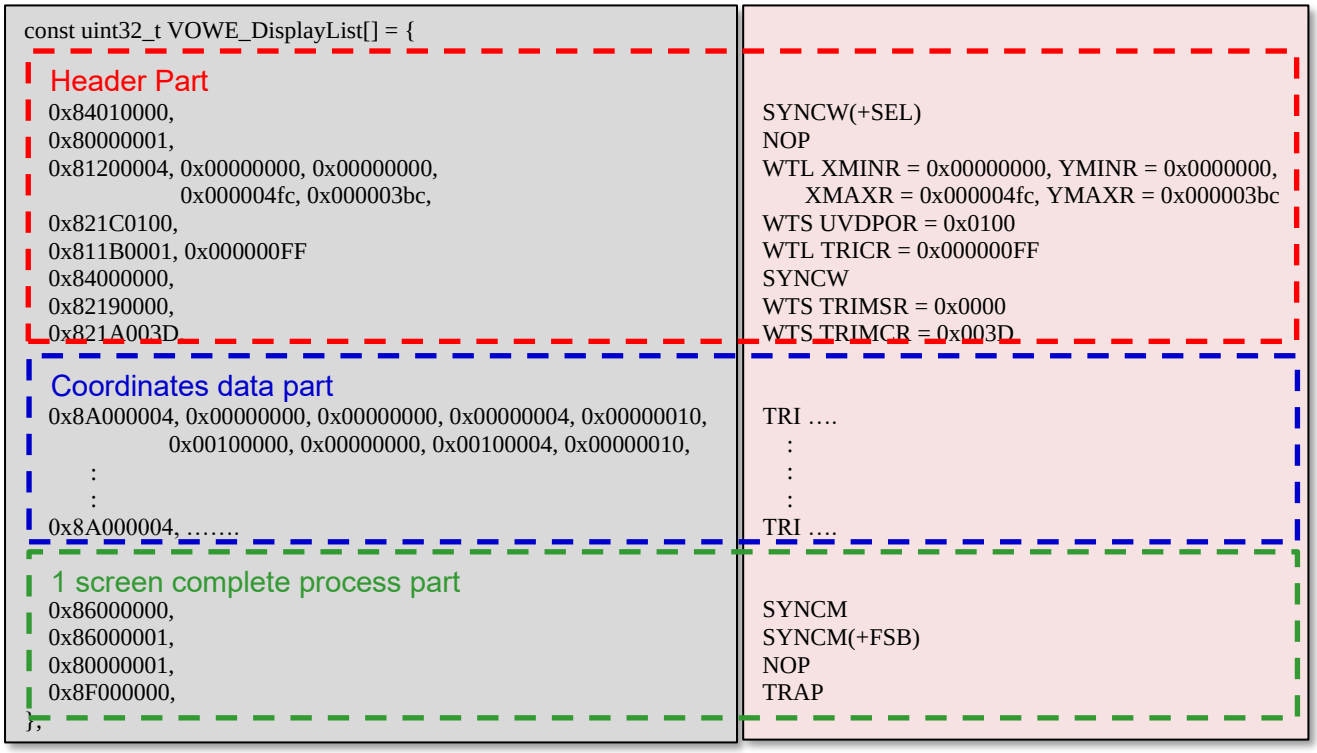

Figure 3-3 Display List structure (1)

Header part consist of the fixed syntax part and the changeable syntax part. If user generates the Display List originally, add the fixed syntax part to top of Display List.

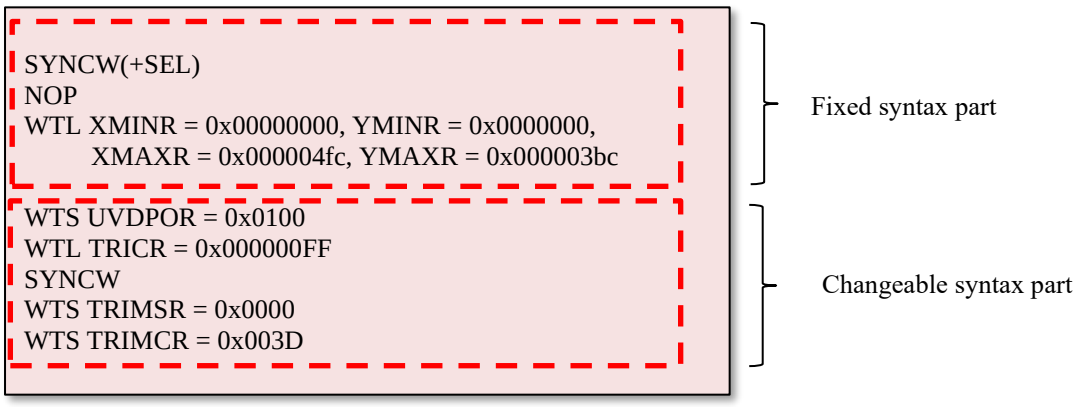

Figure 3-4 Display List structure (2)

3.1.4 Processing timing

The processing timing of the VOWE driver is in either of two modes: continuous mode or one-shot mode. Continuous mode is mainly for use in processing the video and one-shot mode is mainly used for use with still pictures. In continuous mode, the R_VOWE_Start function is called to start processing the warping correction, and this is repeated until the R_VOWE_Stop function is called. In one-shot mode, the R_VOWE_Start function is also called to start processing the warping correction, but this mode only corrects one image data, after which the processing is automatically stopped. The one-shot mode does not require the R_VOWE_Stop function.

When the R_VOWE_Start function is executed, the VOWE H/W macro starts the processing for warping correction in synchronization with the Vsync signal. On completion of the processing of one image data, the callback function which registered by the R_VOWE_Open function is called. The diagrams below show the timing with which each process starts and stops.

- The start timing of continuous mode

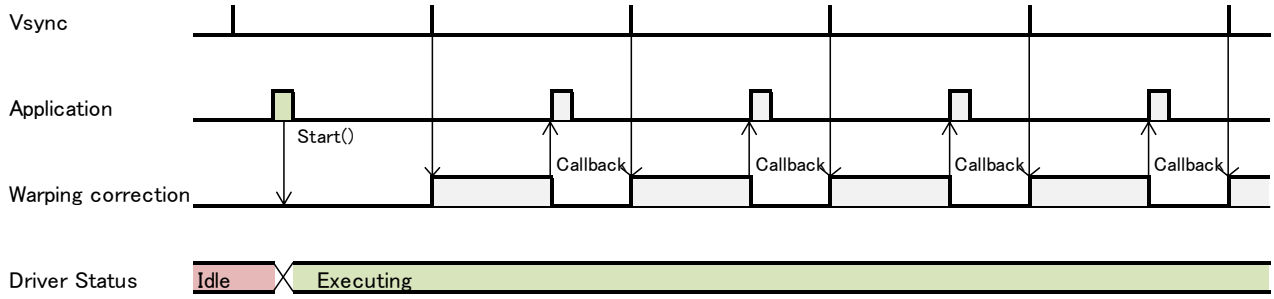

Figure 3-5 Timing chart of the start timing of continuous mode

- The stop timing of continuous mode

Figure 3-6 shows the timing when processing is stopped in the middle of a warping correction operation, *Figure 3-7* shows the timing in when processing is stopped immediately after the callback on completion of warping correction.

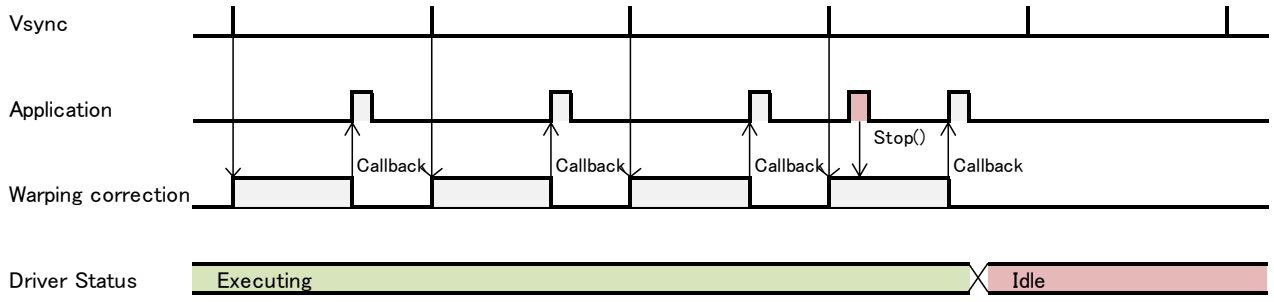

Figure 3-6 Timing chart of the stop timing of continuous mode (1)

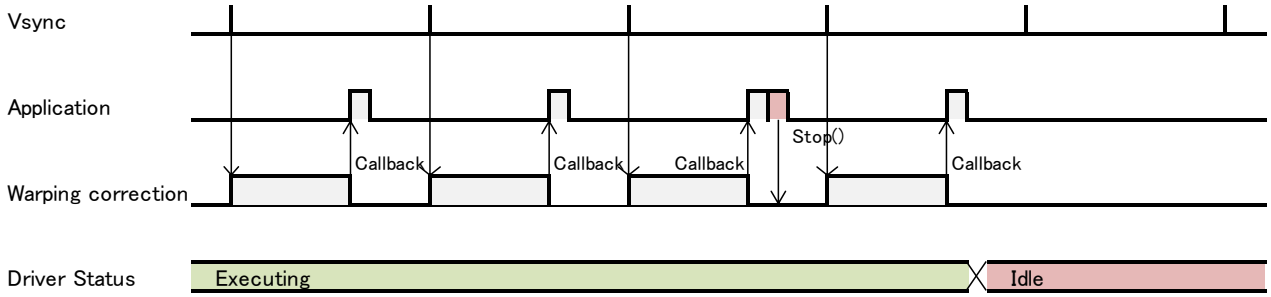

Figure 3-7 Timing chart of the stop timing of continuous mode (2)

- The start and stop timing of one-shot mode

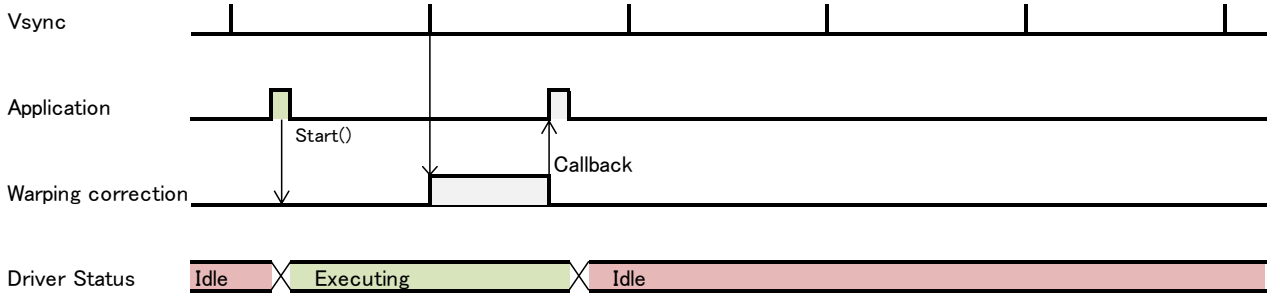

Figure 3-8 Timing chart of the start and stop timing of one-shot mode

3.1.5 Buffering Mode

The VOWE driver has two buffer modes: a frame-buffer mode and a ring-buffer mode.

3.1.5.1 Frame buffer mode

The frame-buffer mode uses two buffer areas, with one frame of data stored in each buffer area after the processing for warping correction. The data are stored alternately in each buffer after a frame is processed. The VDCE driver in the next stage reads-out data from the buffer to which the VOWE driver is not currently transferring data for storage.

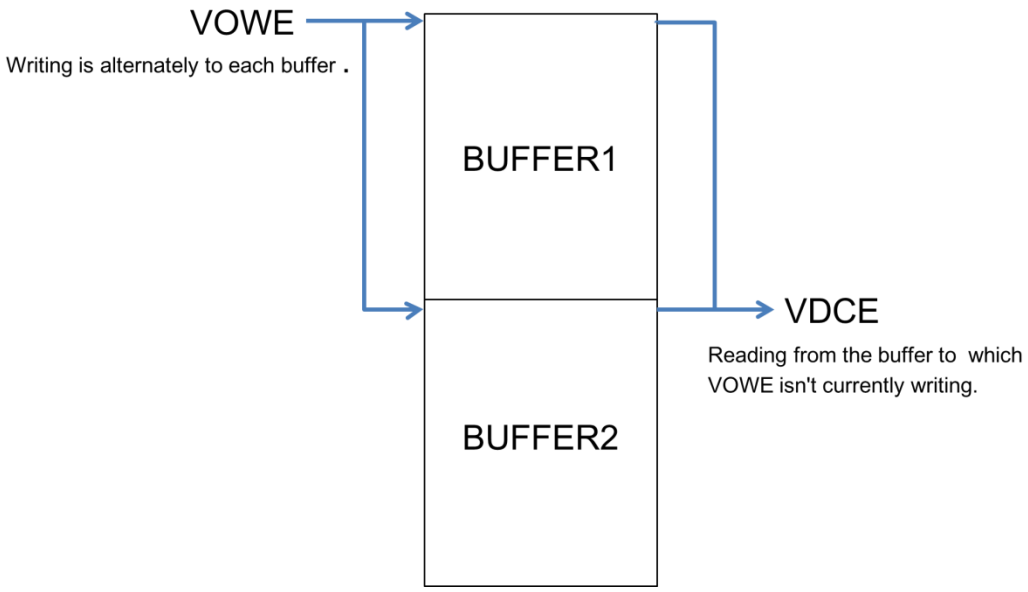

Figure 3-9 Frame-buffer mode

3.1.5.2 Ring buffer mode

The ring-buffer mode uses a buffer area with user-defined size as a single ring buffer. During processing by the VOWE driver, the data are written sequentially from the top of the buffer area. On the other hand, the VDCE driver in the next stage only reads-out the data following writing by the VOWE driver after a delay that is separately set as desired. The ring-buffer mode can only be used with continuous mode. Setting the ring-buffer mode and one-shot mode at the same time leads to an error.

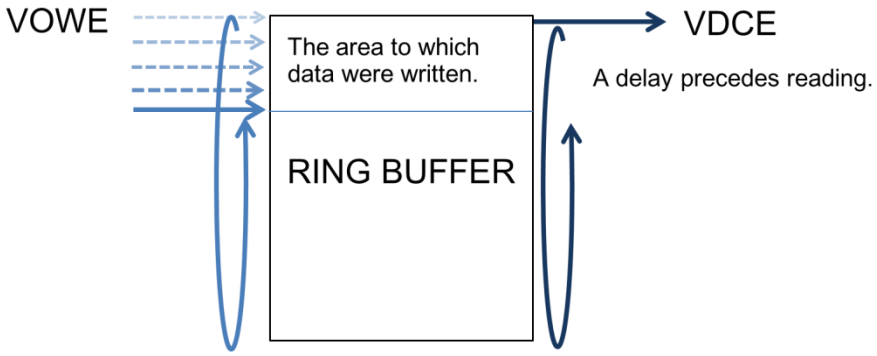

Figure 3-10 Ring-buffer mode

A Ring Buffer in the memory that holds enough VOWE data for the VDCE is sufficient to ensure continuous video data output. The VOWE Ring Buffer holds data for an integer number of lines. Thus, its size is calculated by
Ring Buffer size = (number of lines) x (line stride). The Ring Buffer size must be a power of 2, hence the number of lines and the line stride have to be a power of 2 as well.

3.2 Using the API

3.2.1 VOWE initialization

VOWE driver is initialized using R_VOWE_Init.
If using the VOWE driver with Window Manager (WM), initialize by the following procedure:

An example of the code for initializing the VOWE unit is shown below.

Example

```
/* Initialize Window Manager. */
R_WM_DevInit(LOC_WM_UNIT, msg_queue, WM_MSG_QUEUE_LEN, 0);
R_WM_ScreenTimingSetByName(LOC_WM_UNIT, LOC_DISPLAY_NAME);
R_WM_ScreenEnable(LOC_WM_UNIT);

R_WM_FrameEndMark(LOC_WM_UNIT, 0);
R_WM_FrameWait(LOC_WM_UNIT, 0);

/* Execute R_VOWE_Init after executing R_WM_DevInit */
/* Initialize VOWE driver. */
R_VOWE_Init(LOC_VOWE_UNIT);
```

3.2.2 Buffer setting

Work buffer of VOWE is set by R_VOWE_Open.

- SourceConfig->SourceWidth : Width of image
- SourceConfig->SourceHeight : Hight of image
- DestConfig->WorkBufferMode : Buffer mode （Frame buffer or Ring buffer)
- DestConfig->WorkBufferAddr : Start address of Work buffer
- DestConfig->WorkBufferSize : Work buffer size

Clip area is set by DL (Display List). The lower 2 bits come after the fixed point.

- XMINR : Minimum value of X coordinate of the clipping area.
- YMINR : Minimum value of Y coordinate of the clipping area.
- XMAXR : Maximum value of X coordinate of the clipping area.
- YMAXR : Maximum value of Y coordinate of the clipping area.

Width/Height of Source image and Width/Height of (Clipped) Destination image are same value.

Note: Specifying Local RAM area for the work buffer is prohibited.

3.2.2.1 Frame buffer mode

In case of frame buffer mode (w/o clip area) in below:

- Stride is calculated by VOWE driver with rounding up SourceWidth to 128 Pixel alignment.
- WorkBufferSize should be set as Stride * SouceHeight * 2.
- If SYNCM command (+FSB) is executed at the last of DL, address is changed DSAR ⇔ DSAR2 alternately.

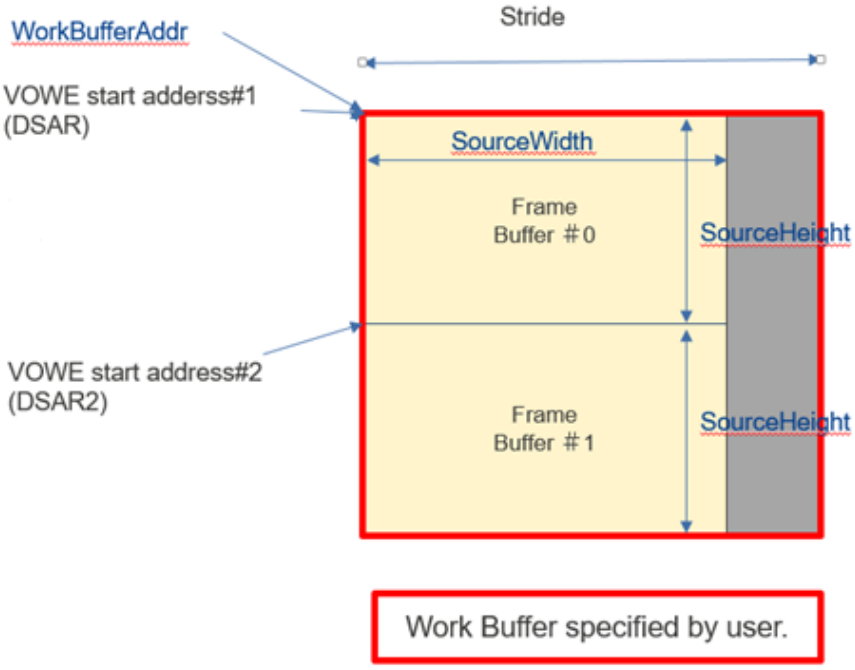

Figure 3-11 Frame buffer mode (w/o clip area)

In case of frame buffer mode (w/ clip area) in below:

- Stride is calculated by VOWE driver with rounding up (SourceWidth＋XMINR) to 128 Pixel alignment.
- YMINR area is set to VOWE H/W, the area is not written actually. User is not necessary to allocate this area.

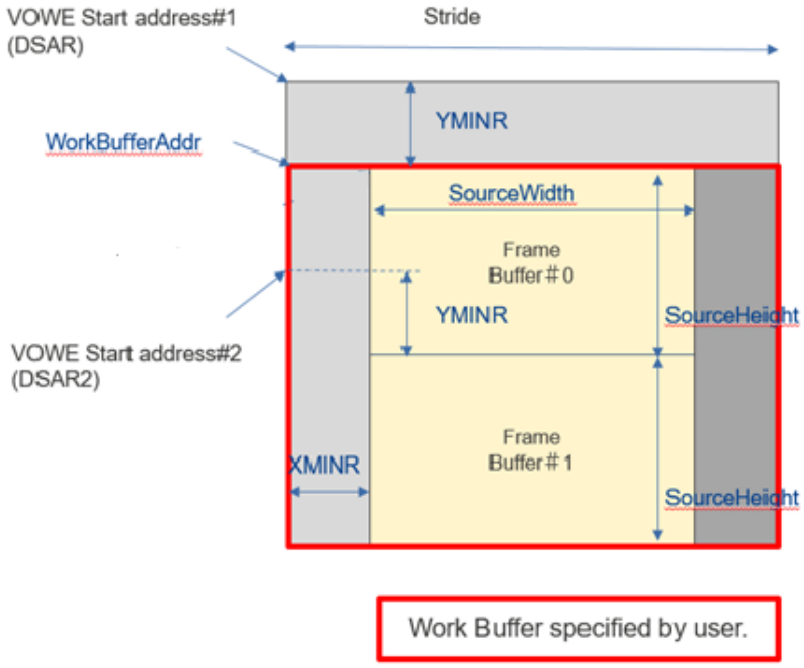

Figure 3-12 Frame buffer mode (w/ clip area)

An example of processing for frame-buffer mode w/o Clip area.

- Image size: 240 (width) × 320 (height)
- Color format: ARGB8888 (4 byte/pixel)
- Stride size: 256 = (((width + (128 - 1)) / 128) * 128
- Buffer size: 655360 = 256 (stride) * 320 (height) * 4 (byte/pixel) * 2 (buffer number)

Example

```
void VoweSettingToFrameBufferMode(void)
{
    r_vowe_Error_t        err;
    r_vowe_SourceConfig_t src_config;
    r_vowe_DisplayList_t  display_list;
    r_vowe_DestConfig_t   dest_config;

   /* Initialize VOWE driver. */
   err = R_VOWE_Init(LOC_VOWE_UNIT);

    /* Setup the source & destination config information. */
   src_config.LineStartPos    = 1;
   src_config.SourceWidth     = 240;
   src_config.SourceHeight    = 320;
   display_list.DspList       = loc_DisplayList1;
   display_list.Size          = sizeof(loc_DisplayList1);
   dest_config.WorkBufferMode = R_VOWE_FRAME_BUFFER_MODE;
   dest_config.WorkBufferAddr = loc_FrameBuffer;
   dest_config.WorkBufferSize = 655360;
   dest_config.ColorFormat    = R_VOWE_FORMAT_ARGB8888;
   dest_config.DitherMode     = R_FALSE;
   dest_config.RingBufferDelay = 1;
   dest_config.DestMode       = R_VOWE_DEST_MODE_NORMAL;

   /* Open VOWE driver. */
   err = R_VOWE_Open(LOC_VOWE_UNIT, &src_config, &display_list, &dest_config,
                     loc_VOWE_CallbackFunction, 0);

   /* Start VOWE driver. */
   err = R_VOWE_Start(LOC_VOWE_UNIT, R_TRUE);
 }
```

3.2.2.2 Ring buffer mode

In case of ring buffer mode in below:

- Stride is calculated by VOWE driver with rounding up SourceWidth to power-of two (2^n) and 128Pixel alignment.
- Ring buffer size should be set the power-of-two value.
- In case of Ring buffer mode, XMINR and YMINR must be set as 0.

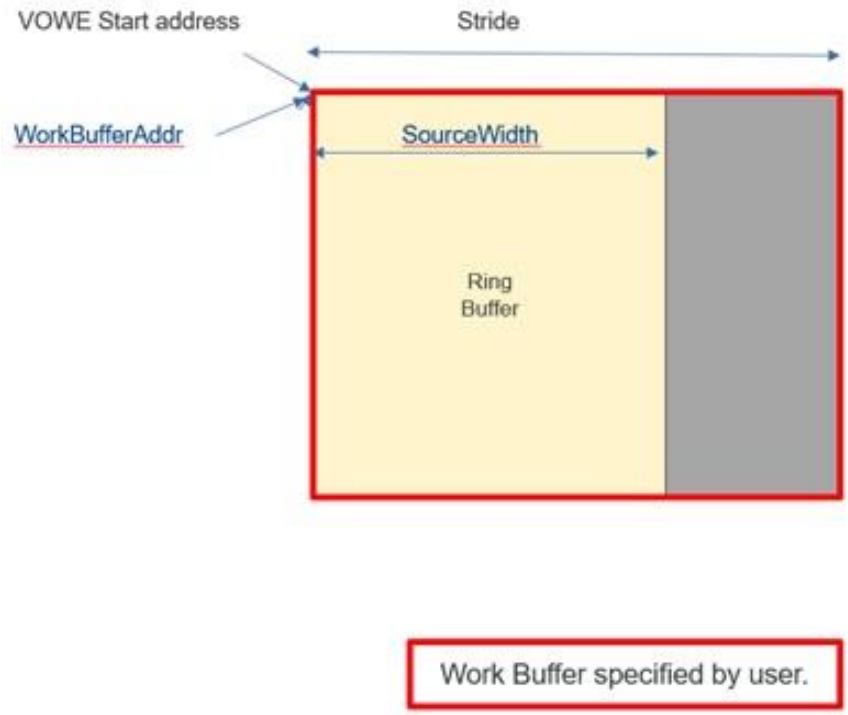

Figure 3-13 Ring buffer mode

Allocate the work buffer at the following size in case of ring buffer mode (R_VOWE_RING_BUFFER_MODE):
Work buffer size (byte) is 2 exponentiations.

An example of processing for the ring-buffer mode.

- Image size: 320(width) × 240(height)
- Color format: RGB565 (2byte/pixel)
- Stride size: 512 = Round up to power-of-2 from 320
- Buffer size: 128Kbyte = 512 (stride) × 128 (line) × 2 (byte/pixel)
- Delay line: 50 (line)

Example

```
void VoweSettingToRingBufferMode(void)
{
    r_vowe_Error_t        err;
    r_vowe_SourceConfig_t src_config;
    r_vowe_DisplayList_t  display_list;
    r_vowe_DestConfig_t   dest_config;

    /* Initialize VOWE driver. */
    err = R_VOWE_Init(LOC_VOWE_UNIT);

    /* Setup the source & destination config information. */
    src_config.LineStartPos     = 1;
    src_config.SourceWidth      = 320;
    src_config.SourceHeight     = 240;
    display_list.DspList        = loc_DisplayList2;
    dest_config.WorkBufferMode  = R_VOWE_RING_BUFFER_MODE;
    dest_config.WorkBufferAddr  = loc_FrameBuffer;
    dest_config.WorkBufferSize  = 131072;
    dest_config.ColorFormat     = R_VOWE_FORMAT_RGB565;
    dest_config.DitherMode      = R_TRUE;
    dest_config.RingBufferDelay = 50;

    /* Open VOWE driver. */
    err = R_VOWE_Open(LOC_VOWE_UNIT, &src_config, &display_list, &dest_config,
                      loc_VOWE_CallbackFunction, 0);

    /* Start VOWE driver. */
    err = R_VOWE_Start(LOC_VOWE_UNIT, R_TRUE);
}
```

3.2.3 Delay setting

Delay of VOWE is set by R_VOWE_Open.

- DestConfig->RingBufferDelay：Delay Line number

This delay works Between writing to Line memory by VDCE (input to VOWE) and reading from Work buffer by VDCE (output from VOWE).
In case of Ring buffer mode, RingBufferDelay is set to H/W.
In case of Frame buffer mode, RingBufferDelay is ignored, Delay is fixed 1Line.

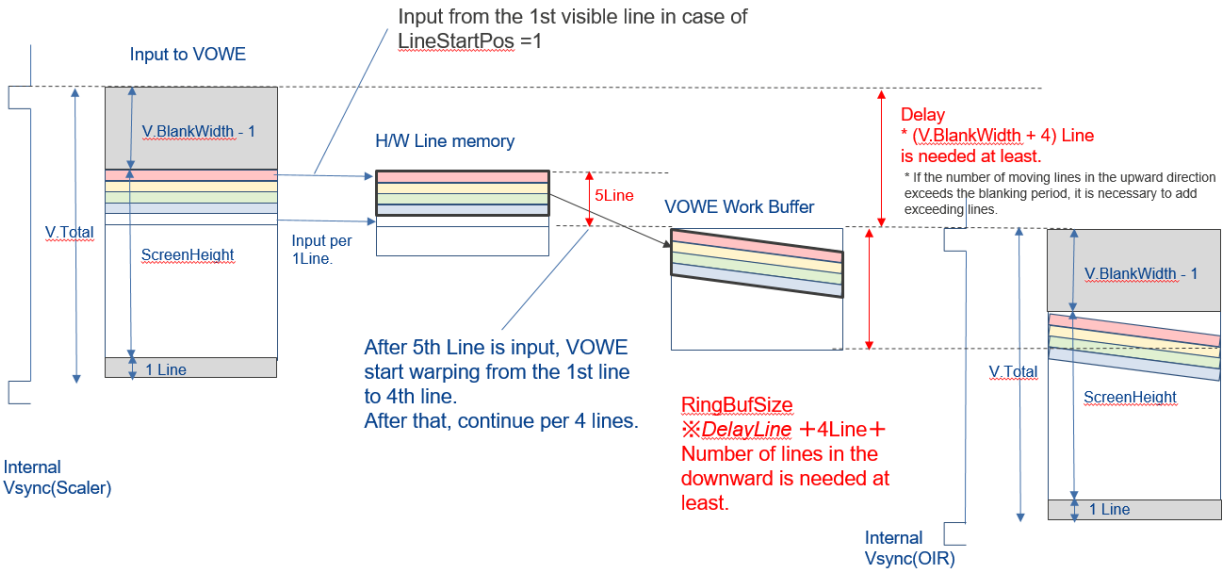

Figure 3-14 Delay image

3.2.4 Start and Stop

When user wants to start the warping correction, execute R_VOWE_Start.
In case if user specifies the R_TRUE to ContinuousMode of second parameter of R_VOWE_Start, VOWE driver executes the warping correction process every V-Sync signal.
When user want to stop the warping correction, execute R_VOWE_Stop.
User specifies the R_TRUE to ContinuousMode usually.

If user specifies the R_FALSE to ContinuousMode when R_VOWE_Start was called, VOWE driver executes the warping correction process one time at 1st V-Sync signal and VOWE driver stops automatically. So, user does not need to call the R_VOWE_Stop.

3.2.5 VOWE completion

If user want to complete the VOWE driver, user call R_VOWE_Close and R_VOWE_DeInit.

3.3 Device difference

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

Table 3-2 APIs supported by VOWE driver

Function	RH850/D1x Device Name	
	D1L2(H)	D1M1(H) / D1M1-V2 / D1M1A / D1M2(H)
All API of VOWE driver	No	Full

3.4 Header File List

Table 3-3 Header File List

No.	Header File Name	Description
(1)	r_vowe_api.h	The header file for VOWE API.
(2)	r_typedefs.h	Header file for predefined data types.

4.Functions

4.1 Function List

This section describes about the VOWE 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 VOWE API Functions

Function Name	Purpose
R_VOWE_Init	This function initializes the VOWE driver.
R_VOWE_DeInit	This function de-initializes the VOWE driver.
R_VOWE_Open	This function opens the VOWE driver.
R_VOWE_Close	This function closes the VOWE driver
R_VOWE_Start	This function starts the warping correction processing.
R_VOWE_Stop	This function stops the warping correction processing.
R_VOWE_DLChange	This function changes the display list (DL).
R_VOWE_VersionStringGet	This function returns the version string of this VOWE driver.
R_VOWE_MacroVersionGet	This function returns the major and minor version of the H/W macro.
R_VOWE_Isr0	This function processes the interrupt factor.

4.2 VOWE API Functions

This chapter describes the application interface functions, which are required for general use of the driver, but which are related to a specific functionality of the macro itself.

4.2.1 R_VOWE_Init

Function Prototype

```
r_vowe_Error_t R_VOWE_Init(const uint32_t Unit)
```

Input Parameter

Table 4-2 Input parameter of R_VOWE_Init

Parameter	Description
Unit	Specifies the VOWE unit number.

Input-Output Parameter

None

Output Parameter

None

Return Codes

R_VOWE_ERR_OK	- No error had occurred.
R_VOWE_ERR_RANGE_UNIT	- Unit-number was outside the range.
R_VOWE_ERR_NOT_ACCEPTABLE	- A function was called in an incorrect state.
R_VOWE_ERR_FATAL_OS	- Fatal error had occurred at OS interface.
R_VOWE_ERR_FATAL_HW	- Fatal Error had occurred at H/W.
R_VOWE_ERR_SYS_VDCE	- The error has occurred at the driver support function of VDCE driver.

Description

This function initializes the VOWE driver.
R_VOWE_Init function shall be the first call and should be executed only one time without an intervening R_VOWE_DeInit function.
If the function successfully executes, the return code will be R_VOWE_ERR_OK and the status will be in the Initialized state.
This function calls R_VOWE_Sys_Init and R_VOWE_Sys_Init will call R_VDCE_DisplayTimingAdjust to adjust display timing. So, this function should be called when VDCE unit is Initialized state or Idle state.

Reentrancy

Non-reentrant.
If user implements following functions to prevent multiple executions, this function will become re-entrant.

- R_VOWE_Sys_Lock
- R_VOWE_Sys_Unlock
- R_VDCE_Sys_Lock
- R_VDCE_Sys_Unlock

Sync/Async

Synchronous

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-5* about status conditions.
R_WM_DevInit or R_VDCE_Init should be executed before executing this function.

See also

r_vowe_Error_t

4.2.2 R_VOWE_DeInit

Function Prototype

```
r_vowe_Error_t R_VOWE_DeInit(const uint32_t Unit)
```

Input Parameter

Table 4-3 Input parameter of R_VOWE_DeInit

Parameter	Description
Unit	Specifies the VOWE unit number.

Input-Output Parameter

None

Output Parameter

None

Return Codes

R_VOWE_ERR_OK	- No error had occurred.
R_VOWE_ERR_RANGE_UNIT	- Unit-number was outside the range.
R_VOWE_ERR_NOT_ACCEPTABLE	- A function was called in an incorrect state.
R_VOWE_ERR_FATAL_OS	- Fatal error had occurred at OS interface.

Description

This function de-initializes the VOWE driver.
R_VOWE_DeInit function should be the last call after all VOWE driver related resources have been released.
If the function successfully executes, the return code will be R_VOWE_ERR_OK and the status will be in the Uninitialized state.

Reentrancy

Non-reentrant.
If user implements following functions to prevent multiple executions, this function will become re-entrant.

- R_VOWE_Sys_Lock
- R_VOWE_Sys_Unlock

Sync/Async

Synchronous

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-5* about status conditions.

See also

r_vowe_Error_t

4.2.3 R_VOWE_Open

Function Prototype

```
r_vowe_Error_t R_VOWE_Open(const uint32_t                       Unit,
                           const r_vowe_SourceConfig_t *const   SourceConfig,
                           const r_vowe_DisplayList_t *const    DisplayList,
                           const r_vowe_DestConfig_t *const     DestConfig,
                           const r_vowe_CallbackFunction_t      CallbackFunc,
                           const uint32_t                       CallbackParam)
```

Input Parameter

Table 4-4 Input parameter of R_VOWE_Open

Parameter	Description
Unit	Specifies the VOWE unit number.
SourceConfig	Specifies the source image information.
DisplayList	Specifies the display list (DL) information.
DestConfig	Specifies the destination buffer information.
CallbackFunc	Specifies the callback function that will be called when the warping correction processing is finished, or some error occurred. If the callback function is unnecessary, set NULL.
CallbackParam	Specifies the value which is passed the callback function.

Input-Output Parameter

None

Output Parameter

None

Return Codes

R_VOWE_ERR_OK	- No error had occurred.
R_VOWE_ERR_PARAM_INCORRECT	- A parameter provided to a function was incorrect.
R_VOWE_ERR_RANGE_UNIT	- Unit-number was outside the range.
R_VOWE_ERR_RANGE_PARAM	- A parameter was the outside the range.
R_VOWE_ERR_NOT_ACCEPTABLE	- A function was called in an incorrect state.
R_VOWE_ERR_FATAL_OS	- Fatal error had occurred at OS interface.
R_VOWE_ERR_SYS_VDCE	- The error has occurred at the driver support function of VDCE driver.

Description

This function sets the configuration and callback function.
If the function successfully executes, the return code will be R_VOWE_ERR_OK and the status will be in the Idle state.
This function calls R_VOWE_Sys_VDCEnable and R_VOWE_Sys_VDCEnable will call R_VDCE_OirEnable.
So, this function should be called when VDCE unit is Idle state or Executing state.

Reentrancy

Non-reentrant.
If user implements following functions to prevent multiple executions, this function will become re-entrant.

- R_VOWE_Sys_Lock
- R_VOWE_Sys_Unlock
- R_VDCE_Sys_Lock
- R_VDCE_Sys_Unlock

Sync/Async

Synchronous

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-5* about status conditions.
R_WM_ScreenEnable or R_VDCE_DisplayEnable should be executed before executing this function.

See also

r_vowe_SourceConfig_t
r_vowe_DisplayList_t
r_vowe_DestConfig_t
r_vowe_CallbackFunction_t
r_vowe_Error_t

4.2.4 R_VOWE_Close

Function Prototype

```
r_vowe_Error_t R_VOWE_Close(const uint32_t Unit)
```

Input Parameter

Table 4-5 Input parameter of R_VOWE_Close

Parameter	Description
Unit	Specifies the VOWE unit number.

Input-Output Parameter

None

Output Parameter

None

Return Codes

R_VOWE_ERR_OK	- No error had occurred.
R_VOWE_ERR_RANGE_UNIT	- The unit-number was outside of the range.
R_VOWE_ERR_NOT_ACCEPTABLE	- A function was called in an incorrect state.
R_VOWE_ERR_FATAL_OS	- Fatal error had occurred at OS interface.
R_VOWE_ERR_SYS_VDCE	- The error has occurred at the driver support function of VDCE driver.

Description

This function closes the VOWE driver.
If the function successfully executes, the return code will be R_VOWE_ERR_OK and the status will be in the Initialized state.

Reentrancy

Non-reentrant.
If user implements following functions to prevent multiple executions, this function will become re-entrant.

- R_VOWE_Sys_Lock
- R_VOWE_Sys_Unlock
- R_VDCE_Sys_Lock
- R_VDCE_Sys_Unlock

Sync/Async

Synchronous

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-5* about status conditions.

See also

r_vowe_Error_t

4.2.5 R_VOWE_Start

Function Prototypes

```
r_vowe_Error_t R_VOWE_Start(const uint32_t Unit,
                            const uint32_t ContinuousMode)
```

Input Parameter

Table 4-6 Input parameter of R_VOWE_Start

Parameter	Description
Unit	Specifies the VOWE unit number.
ContinuousMode	Specifies the continuous mode or one-shot mode. R_TRUE: continuous mode R_FALSE: one-shot mode

Input-Output Parameter

None

Output Parameter

None

Return Codes

R_VOWE_ERR_OK	- No error had occurred.
R_VOWE_ERR_PARAM_INCORRECT	- A parameter provided to a function was incorrect.
R_VOWE_ERR_RANGE_UNIT	- The unit-number was outside of the range.
R_VOWE_ERR_NOT_ACCEPTABLE	- A function was called in an incorrect state.
R_VOWE_ERR_FATAL_OS	- Fatal error had occurred at OS interface.
R_VOWE_ERR_SYS_VDCE	- The error has occurred at the driver support function of VDCE driver.

Description

This function starts the warping correction processing.
If the function successfully executes, the return code will be R_VOWE_ERR_OK and the status will be in the Executing state.

If ContinuousMode is R_TRUE, VOWE driver executes the warping correction process every V-Sync signal.
If ContinuousMode is R_FALSE, then warping correction is executed one time at 1st V-Sync and VOWE driver stop automatically. So, user does not need to call R_VOWE_Stop.

Reentrancy

Non-reentrant.
If user implements following functions to prevent multiple executions, this function will become re-entrant.

- R_VOWE_Sys_Lock
- R_VOWE_Sys_Unlock
- R_VDCE_Sys_Lock
- R_VDCE_Sys_Unlock

Sync/Async

Asynchronous

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-5* about status conditions.

See also

r_vowe_Error_t

4.2.6 R_VOWE_Stop

Function Prototypes

```
r_vowe_Error_t R_VOWE_Stop(const uint32_t Unit)
```

Input Parameter

Table 4-7 Input parameter of R_VOWE_Stop

Parameter	Description
Unit	Specifies the VOWE unit number.

Input-Output Parameter

None

Output Parameter

None

Return Codes

R_VOWE_ERR_OK	- No error had occurred.
R_VOWE_ERR_RANGE_UNIT	- The unit-number was outside of the range.
R_VOWE_ERR_NOT_ACCEPTABLE	- A function was called in an incorrect state.
R_VOWE_ERR_FATAL_OS	- Fatal error had occurred at OS interface.

Description

This function stops the warping correction processing.
If the function successfully executes, the return code will be R_VOWE_ERR_OK.
When the warping correction processing stops, the callback function is called with R_VOWE_FACTOR_STOP and the state is in the Idle state.

When R_VOWE_FACTOR_STOP callback function isn't called with during the 2VSYNC period after R_VOWE_Stop function execution, unexpected error occurs internally. It should be reset the RH850/D1x device.

Reentrancy

Non-reentrant.
If user implements following functions to prevent multiple executions, this function will become re-entrant.

- R_VOWE_Sys_Lock
- R_VOWE_Sys_Unlock

Sync/Async

Asynchronous

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-5* about status conditions.

See also

r_vowe_Error_t

4.2.7 R_VOWE_DLChange

Function Prototypes

```
r_vowe_Error_t R_VOWE_DLChange(const uint32_t                       Unit,
                               const r_vowe_DisplayList_t * const   DisplayList)
```

Input Parameter

Table 4-8 Input parameter of R_VOWE_DLChange

Parameter	Description
Unit	Specifies the VOWE unit number
DisplayList	Specifies new display list information.

Input-Output Parameter

None

Output Parameter

None

Return Codes

R_VOWE_ERR_OK	- No error had occurred.
R_VOWE_ERR_PARAM_INCORRECT	- A parameter provided to a function was incorrect.
R_VOWE_ERR_RANGE_UNIT	- The unit-number was outside of the range.
R_VOWE_ERR_NOT_ACCEPTABLE	- A function was called in an incorrect state.
R_VOWE_ERR_FATAL_OS	- Fatal error had occurred at OS interface.

Description

This function changes the display list (DL).
If the function successfully executes, the return code will be R_VOWE_ERR_OK.

Reentrancy

Non-reentrant.
If user implements following functions to prevent multiple executions, this function will become re-entrant.

- R_VOWE_Sys_Lock
- R_VOWE_Sys_Unlock

Sync/Async

Synchronous

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-5* about status conditions.

See also

r_vowe_Error_t
r_vowe_DisplayList_t

4.2.8 R_VOWE_VersionStringGet

Function Prototypes

```
const uint8_t *R_VOWE_VersionStringGet(void)
```

Input Parameter

None

Input-Output Parameter

None

Output Parameter

None

Return Codes

Pointer of string.

Description

This function returns the version string of this VOWE driver.

Reentrancy

Reentrant

Sync/Async

Synchronous

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-5* about status conditions.

See also

None

4.2.9 R_VOWE_MacroVersionGet

Function Prototypes

```
r_vowe_Error_t R_VOWE_MacroVersionGet(uint32_t *const  Major,
                                      uint32_t *const  Minor)
```

Input Parameter

None

Input-Output Parameter

None

Output Parameter

Table 4-9 Output parameter of R_VOWE_MacroVersionGet

Parameter Out	Description
Major	Specifies the major version.
Minor	Specifies the minor version.

Return Codes

R_VOWE_ERR_OK - No error has occurred.
R_VOWE_ERR_PARAM_INCORRECT - A parameter provided to a function was incorrect.

Description

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

Reentrancy

Reentrant.

Sync/Async

Synchronous.

Call from Interrupt

Prohibited.

Preconditions

See *Table 2-5* about status conditions.

See also

r_vowe_Error_t

4.3 Interrupt Functions

4.3.1 R_VOWE_Isr0

Function Prototypes

```
void R_VOWE_Isr0(const uint32_t Unit)
```

Input Parameter

Table 4-10 Input parameter of R_VOWE_Isr0

Parameter	Description
Unit	Specifies the VOWE unit number.

Input-Output Parameter

None.

Output Parameter

None.

Return Codes

None.

Description

This function is called from INTVOWE ISR (Interrupt Service Routine) and processes the interrupt factor.

Reentrancy

Non-Reentrant.

Sync/Async

Synchronous.

Call from Interrupt

Permitted.

Preconditions

See *Table 2-5* about status conditions.

See also

None.

5.Types

5.1 Basic Types

This section shows the basic types used in this library.

Table 5-1 Basic type

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 VOWE API Types

This section shows the VOWE API function types used in this library.

5.2.1 r_vowe_CallbackFunction_t

Description

This type describes to callback function.

Definition

```
typedef void (*r_vowe_CallbackFunction_t)(const uint32_t      Unit,
                                          const uint32_t     Factor,
                                          const uint32_t     CallbackParam)
```

Table 5-2 Member/Value of r_vowe_CallbackFunction_t type

Member/Value	Description
Unit	This parameter shows the VOWE unit number.
Factor	This parameter shows the interrupt factor. It could be possible that the multiple flags with OR operation are notified at one time.
CallbackParam	This parameter is the CallbackParam which is specified the R_VOWE_Open function.

See also

R_VOWE_Open

5.3 Definition

This section shows the definition values used in VOWE API.

5.3.1 API version

This section shows the API versions used in VOWE API.

Table 5-3 Definition of VOWE API version

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

5.3.2 Callback factor

Types to describe the callback factor definitions of VOWE API.

Table 5-4 Definitions of VOWE API callback factor

Name	values	Description
R_VOWE_FACTOR_NON	0x00000000uL	This definition shows the no callback factor.
R_VOWE_FACTOR_TRA	0x00000001uL	This definition shows the TRA event occurred. It means that the warping correction processing was complete for one frame.
R_VOWE_FACTOR_IER	0x00000002uL	This definition shows the IER occurred. It means that VOWE H/W processed the invalid display list.
R_VOWE_FACTOR_STOP	0x00000008uL	This definition shows the stop command is finished, and the state is in the Idle state.

5.3.3 Pixel Size

The following are the types to describes the pixel size.

Table 5-5 Definitions of VOWE API pixel size

Name	values	Description
R_VOWE_SIZE_OF_ARGB8888	4	This definition shows byte per pixel of ARGB8888 format
R_VOWE_SIZE_OF_RGB565	2	This definition shows byte per pixel of RGB565 format

5.4 Enumerated Types

This section shows the enumerated types used in VOWE API Functions.

5.4.1 r_vowe_Error_t

Description

This type describes the error code of the VOWE driver.

Definition

```
typedef enum
{
 R_VOWE_ERR_OK = 0,
 R_VOWE_ERR_NG,
 R_VOWE_ERR_PARAM_INCORRECT,
 R_VOWE_ERR_RANGE_UNIT,
 R_VOWE_ERR_RANGE_PARAM,
 R_VOWE_ERR_NOT_ACCEPTABLE,
 R_VOWE_ERR_FATAL_OS,
 R_VOWE_ERR_FATAL_HW,
 R_VOWE_ERR_SYS_VDCE,
 R_VOWE_ERR_SYS_INTC
} r_vowe_Error_t
```

Table 5-6 Enumerator of r_vowe_Error_t

Name	Description
R_VOWE_ERR_OK	No error has occurred.
R_VOWE_ERR_NG	An error has occurred, but no specific error code is defined for it.
R_VOWE_ERR_PARAM_INCORRECT	A parameter provided to a function was incorrect.
R_VOWE_ERR_RANGE_UNIT	Unit-number was outside the range.
R_VOWE_ERR_RANGE_PARAM	A parameter was the outside the range
R_VOWE_ERR_NOT_ACCEPTABLE	A function was called in an incorrect state.
R_VOWE_ERR_FATAL_OS	Fatal error has occurred at OS interface.
R_VOWE_ERR_FATAL_HW	Fatal error has occurred at H/W.
R_VOWE_ERR_SYS_VDCE	The error has occurred at the driver support function of VDCE driver.
R_VOWE_ERR_SYS_INTC	The error has occurred at the driver support function of INTC (interrupt controller)

See also

None

5.4.2 r_vowe_BufferMode_t

Description

This type describes the work buffer mode.

Definition

```
typedef enum
{
 R_VOWE_FRAME_BUFFER_MODE  = 0,
 R_VOWE_RING_BUFFER_MODE
} r_vowe_BufferMode_t
```

Table 5-7 Enumerator of r_vowe_BufferMode_t

Name	Description
R_VOWE_FRAME_BUFFER_MODE	This member specifies the frame buffer mode.
R_VOWE_RING_BUFFER_MODE	This member specifies the ring buffer mode.

See also

r_vowe_DestConfig_t

5.4.3 r_vowe_ColorFormat_t

Description

This type describes the color format.

Definition

```
typedef enum
{
 R_VOWE_FORMAT_ARGB8888 = 0,
 R_VOWE_FORMAT_RGB565
} r_vowe_ColorFormat_t
```

Table 5-8 Enumerator of r_vowe_ColorFormat_t

Name	Description
R_VOWE_FORMAT_ARGB8888	This member specifies the ARGB8888 format.
R_VOWE_FORMAT_RGB565	This member specifies the RGB565 format.

See also

r_vowe_DestConfig_t

5.4.4 r_vowe_DestMode_t

Description

This type describes the destination mode.

Definition

```
typedef enum
{
 R_VOWE_DEST_MODE_NORMAL = 0,
 R_VOWE_DEST_MODE_SCREEN_SHOTS
} r_vowe_DestMode_t
```

Table 5-9 Enumerator of r_vowe_DestMode_t

Name	Description
R_VOWE_DEST_MODE_NORMAL	This member specifies the NORMAL mode. A distortion image is output to display.
R_VOWE_DEST_MODE_SCREEN_SHOTS	This member specifies the SCREEN SHOTS mode. A distortion image is not output to display. A distortion image data is only extracted by the work buffer.

See also

r_vowe_DestConfig_t

5.5 Structure

This section shows the structure used in VOWE API function.

5.5.1 r_vowe_SourceConfig_t

Description

This type describes the structure of the source image information.

Definition

```
typedef struct
{
 Uint32_t LineStartPos;
 Uint32_t SourceWidth;
 Uint32_t SourceHeight;
} r_vowe_SourceConfig_t
```

Table 5-10 Member of r_vowe_SourceConfig_t structure

Member	Description
LineStartPos	This member specifies the processing start position (line) of the source image. The range is 1 to 1019.
SourceWidth	This member specifies the source image width (pixel). It should be multiples of 2. The range is 4 to 1280.
SourceHeight	This member specifies the source image height (pixel). The range is 5 to 1024.

See also

None

5.5.2 r_vowe_DisplayList_t

Description

This type describes the structure of the display list information.

Definition

```
typedef struct
{
 uint32_t * DspList;
 uint32_t   Size;
} r_vowe_DisplayList_t
```

Table 5-11 Member of r_vowe_DisplayList_t structure

Member	Description
DspList	This is a pointer to the start address of the display list. The address must be a multiple of 8 bytes.
Size	This member specifies the size of display list in bytes.

See also

None

5.5.3 r_vowe_DestConfig_t

Description

This type describes the structure of the destination buffer settings information.

Definition

```
Typedef struct
{
 r_vowe_BufferMode_t   WorkBufferMode;
 uint8_t*              WorkBufferAddr;
 uint32_t              WorkBufferSize;
 r_vowe_ColorFormat_t  ColorFormat;
 uint32_t              DitherMode;
 uint32_t              RingBufferDelay;
 r_vowe_DestMode_t     DestMode;
} r_vowe_DestConfig_t
```

Table 5-12 Member of r_vowe_DestConfig_t

Member	Description
WorkBufferMode	This member specifies the work buffer mode. R_VOWE_FRAME_BUFFER_MODE R_VOWE_RING_BUFFER_MODE
WorkBufferAddr	This is the start address of the work buffer. This work buffer is allocated by user. It must be a multiple of 128 bytes in case of frame buffer mode. It must be a multiple of 1024 bytes in case of ring buffer mode. Note: Specifying Local RAM area is prohibited.
WorkBufferSize	This member specifies the work buffer size.
ColorFormat	This member specifies the color format of output image. R_VOWE_FORMAT_ARGB8888 R_VOWE_FORMAT_RGB565
DitherMode	This member specifies the dither mode. R_FALSE: dither is disabled. R_TRUE: dither is enabled. This setting is valid when R_VOWE_FORMAT_RGB565 is specified.
RingBufferDelay	This member specifies the number of output delay lines of VDCE. The range is from 1 to 255. This setting is valid in case of ring buffer mode.
DestMode	This member specifies the destination mode. R_VOWE_DEST_MODE_NORMAL R_VOWE_DEST_MODE_SCREEN_SHOTS

See also

r_vowe_BufferMode_t
r_vowe_ColorFormat_t
r_vowe_DestMode_t

Revision History	Renesas Graphics Library Video Output Warping Engine (VOWE) Driver User's Manual: Software

Rev.	Date	Description	
		Page	Summary
0.1	Nov 29, 2018	-	First edition.
0.2	Mar 28, 2019	35	Added the "const" to argument. R_VOWE_Isr0
		38	Added "uL" prefix to values.
1.0	June 12, 2019	5	Improve the description of error handling.
		24 27, 28	Add the description of calling VDCE function and status condition.
2.0	May 13, 2020	14, 45	Add the restriction of Local RAM.

Renesas Graphics Library
Video Output Warping Engine (VOWE) Driver
User's Manual: Software
Publication Date: Rev.0.1 Nov 29, 2018
Rev.2.0 May 13, 2020

Published by: Renesas Electronics Corporation

SALES OFFICES Renesas Electronics Corporation http://www.renesas.com

Refer to "http://www.renesas.com/" for the latest and detailed information.

Renesas Electronics Corporation
TOYOSU FORESIA, 3-2-24 Toyosu, Koto-ku, Tokyo 135-0061, Japan

Renesas Electronics America Inc. Milpitas Campus
1001 Murphy Ranch Road, Milpitas, CA 95035, U.S.A.
Tel: +1-408-432-8888, Fax: +1-408-434-5351

Renesas Electronics America Inc. San Jose Campus
6024 Silver Creek Valley Road, San Jose, CA 95138, USA
Tel: +1-408-284-8200, Fax: +1-408-284-2775

Renesas Electronics Canada Limited
9251 Yonge Street, Suite 8309 Richmond Hill, Ontario Canada L4C 9T3
Tel: +1-905-237-2004

Renesas Electronics Europe GmbH
Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-6503-0, Fax: +49-211-6503-1327

Renesas Electronics (China) Co., Ltd.
Room 101-T01, Floor 1, Building 7, Yard No. 7, 8th Street, Shangdi, Haidian District, Beijing 100085, China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

Renesas Electronics (Shanghai) Co., Ltd.
Unit 301, Tower A, Central Towers, 555 Langao Road, Putuo District, Shanghai 200333, China
Tel: +86-21-2226-0888, Fax: +86-21-2226-0999

Renesas Electronics Hong Kong Limited
Unit 1601-1611, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2265-6688, Fax: +852 2886-9022

Renesas Electronics Taiwan Co., Ltd.
13F, No. 363, Fu Shing North Road, Taipei 10543, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

Renesas Electronics Singapore Pte. Ltd.
80 Bendemeer Road, #06-02 Singapore 339949
Tel: +65-6213-0200, Fax: +65-6213-0300

Renesas Electronics Malaysia Sdn.Bhd.
Unit No 3A-1 Level 3A Tower 8 UOA Business Park, No 1 Jalan Pengaturcara U1/51A, Seksyen U1, 40150 Shah Alam, Selangor, Malaysia
Tel: +60-3-5022-1288, Fax: +60-3-5022-1290

Renesas Electronics India Pvt. Ltd.
No.777C, 100 Feet Road, HAL 2nd Stage, Indiranagar, Bangalore 560 038, India
Tel: +91-80-67208700

Renesas Electronics Korea Co., Ltd.
17F, KAMCO Yangjae Tower, 262, Gangnam-daero, Gangnam-gu, Seoul, 06265 Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5338

Colophon 7.0

ルネサス エレクトロニクス株式会社

■営業お問合せ窓口

http://www.renesas.com

※営業お問合せ窓口の住所は変更になることがあります。最新情報につきましては、弊社ホームページをご覧ください。

ルネサス エレクトロニクス株式会社 〒135-0061 東京都江東区豊洲3-2-24（豊洲フォレシア）

■技術的なお問合せおよび資料のご請求は下記へどうぞ。
総合お問合せ窓口：https://www.renesas.com/contact/

Colophon 3.1

Renesas Graphics Library
Video Output Warping Engine (VOWE) Driver

Renesas Electronics Corporation